

AR-R.1135

ISSUE A

(1)

OPTIMAL STATIONKEEP AND ATTITUDE  
CONTROL OF FLEXIBLE SPACECRAFT

PART 2: DESIGN VERIFICATION

P  
91  
C655  
058  
1983  
pt.2

**SPAR**

SPAR LIBRARY - TORONTO  
COPY NUMBER 6

P  
91  
C655  
058  
1983  
pt.2

SPAR-R.1135

ISSUE A

①  
OPTIMAL STATIONKEEP AND ATTITUDE  
CONTROL OF FLEXIBLE SPACECRAFT

PART 2: DESIGN VERIFICATION

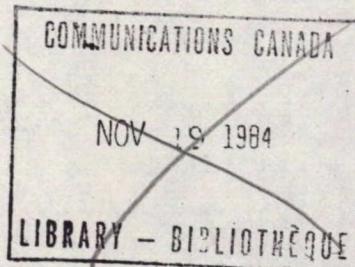
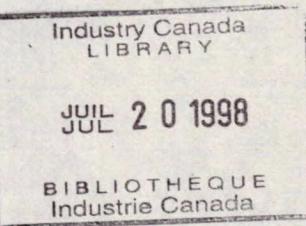
Prepared For:

DEPARTMENT OF COMMUNICATIONS

Reference:

DSS CONTRACT 15ST.36100-1-0102

FEBRUARY, 1983



**Spar Aerospace Limited**  
Remote Manipulator Systems Division  
1700 Ormont Drive, Weston, Ontario, Canada M9L 2W7

**SPAR** Space &  
Electronics Group

P  
91  
C655  
058  
1983  
pto 2

DD 493258/  
DL 4932645

SPAR LIBRARY - TORONTO

**COPY NUMBER** \_\_\_\_\_

SPAR-R.1135

ISSUE A

## **APPROVALS**

R. Ravindran *Rav*  
Section Chief,  
Controls and Analysis

H.F. Barker  
Manager,  
Controls and Analysis

S.S. Sachdev Manager  
Systems, Controls and  
Analysis, Engineering

P.A. McIntyre  
Director of Engineering

N. Roger

PREPARED BY

**REASON FOR RE-ISSUE**

February, 1983

DATE

## **AMENDMENTS**

NUMBER	REASON FOR AMENDMENT	SIGNATURE	DATE

**Spar Aerospace Limited**  
Remote Manipulator Systems Division  
1700 Ormont Drive, Weston, Ontario, Canada M9L 2W7





Government  
of Canada      Gouvernement  
du Canada

Department of Communications

DOC CONTRACTOR REPORT

DOC-CR-SP -82-064

DEPARTMENT OF COMMUNICATIONS - OTTAWA - CANADA

SPACE PROGRAM

TITLE:      OPTIMAL STATION KEEP AND ATTITUDE CONTROL OF FLEXIBLE  
-Flexible spacecraft -- PART II: DESIGN VERIFICATION

AUTHOR(S): N.A.C. Roger

ISSUED BY CONTRACTOR AS REPORT NO: SPAR-R.1135

PREPARED BY:      Spar Aerospace Limited  
1700 Ormont Drive  
Weston, Ontario  
Canada M9L 2W7

DEPARTMENT OF SUPPLY AND SERVICES CONTRACT NO: 15ST.36100-1-0102

DOC SCIENTIFIC AUTHORITY: A.H. Reynaud

CLASSIFICATION: Unclassified

This report presents the views of the author(s). Publication  
of this report does not constitute DOC approval of the reports  
findings or conclusions. This report is available outside the  
department by special arrangement.

DATE: February 1983

SPAR-R.1135  
ISSUE A

PREFACE

This document constitutes the second of two parts of the final report on the work performed by Spar Aerospace Limited under DSS Contract No. 15ST.36100-1-0102, Serial No. OST81-00137

TABLE OF CONTENTS

<u>Section</u>	<u>Title</u>	<u>Page</u>
1.0	INTRODUCTION	1-1
2.0	THE SPACECRAFT MODAL COORDINATES	2-1
3.0	DESCRIPTION OF COMPUTER PROGRAMS	3-1
3.1	The Flexible Dynamics and Control Program FDCSIM	3-1
3.1.1	Introduction	3-1
3.1.2	Program Design	3-1
3.1.3	Spacecraft Equations	3-2
3.1.4	Observer Equation	3-4
3.1.5	Integration of Equations	3-4
3.1.6	The Controller	3-6
3.1.7	Disturbance Forces and Torques	3-8
3.1.8	The Cost Function	3-10
3.2	The Controller Design Program RICQ	3-11
3.2.1	Introduction	3-11
3.2.2	The Continuous Spacecraft Equations	3-11
3.2.3	The Discrete Spacecraft Equations	3-13
3.2.4	The Calculation of Matrix Exponentials	3-14
3.2.5	Controller Design Equations	3-16
3.2.6	Stability	3-16
3.2.7	Convergence of Control Matrix	3-17
3.3	The Controller Design Program CON52	3-18
3.3.1	Introduction	3-18
3.3.2	Equations	3-18
3.3.3	Solution for $\bar{Q}$	3-20
3.3.4	Solution for $\bar{B}_u(t)$	3-21
3.3.5	Solution for $\bar{P}$	3-21
3.3.6	Solution for $\bar{R}$	3-22
3.3.7	Solution for $\bar{A}_c$ and $\bar{B}_c$	3-24
3.3.8	Comments	3-25

<u>TABLE OF CONTENTS - (Continued)</u>		<u>Page</u>
3.4	The Observer Design Program DECOBR	3-25
3.4.1	Introduction	3-25
3.4.2	The Observer Equation	3-26
3.4.3	Assignment of Decoupled Observer Eigenvalues	3-26
3.4.4	The Eigenvalue Assignment Routine	3-27
3.4.5	The Polynomial Coefficients Routine	3-27
3.4.6	The Kernel Routine	3-30
3.4.7	Calculation of the Rank of a Matrix	3-31
3.5	The Controller Stability Program STABF	3-32
3.5.1	Introduction	3-32
3.5.2	Equations	3-32
3.5.3	Comments	3-35
3.6	The Controller Plus Observer Stability Program STABFO	3-35
3.6.1	Introduction	3-35
3.6.2	Equations	3-36
3.6.3	Comments	3-38
3.7	Common Routines	3-38
3.7.1	Eigenvalue Module	3-39
3.7.2	Matrix Inversion Modules	3-39
3.7.3	Matrix Exponentials	3-40
4.0	RESULTS	4-1
4.1	Comparison of Numerical Simulation and Analytic Stability Programs	4-1
4.2	Effect of Changing the Control Period	4-2
4.3	Compensator with Fewer Controlled Modes	4-3
4.4	Stability Analysis of Controller Plus Observer	4-5
5.0	CONCLUSION	5-1

8/4/mcl702/5

SPAR-R.1135  
ISSUE A

TABLE OF CONTENTS - (Continued)

	<u>Page</u>
APPENDIX A: DESIGN DATA AND RESULTS	A-1
APPENDIX B: GRAPHS OF MODAL COORDINATES VERSUS TIME	B-1
APPENDIX C: PROGRAM FLOWCHARTS	C-1

LIST OF FIGURES

<u>Figure</u>	<u>Title</u>	<u>Page</u>
3-1	SPACECRAFT-CONTROLLER-OBSERVER SYSTEM	3-42
3-2	THE CONTROLLER	3-43

LIST OF TABLES

<u>Table</u>	<u>Title</u>	<u>Page</u>
2-1	FLEXIBLE MODE FREQUENCIES	2-2
3-1	PROGRAM PARAMETERS SET VIA ARRAY VC	3-44
4-1	COMPARISON OF RESULTS OF FDCSIM AND STABFO PROGRAMS	4-6
4-2	EFFECT OF CHANGING THE CONTROL PERIOD	4-7
4-3	RESULTS WITH ONE RIGID MODE, ONE CONTROLLED FLEXIBLE MODE AND ONE RESIDUAL MODE	4-8
4-4	RESULTS WITH ONE RIGID MODE, ONE CONTROLLED FLEXIBLE MODE AND ALL RESIDUAL MODES	4-9

8/4/mcl702/6

SPAR-R.1135  
ISSUE A

LIST OF TABLES - (Continued)

	<u>Page</u>
4-5 STABILITY ANALYSIS OF RUN 2	4-10
4-6 STABILITY ANALYSIS OF RUN 3	4-11
4-7 STABILITY ANALYSIS OF RUN 8	4-12
4-8 STABILITY ANALYSIS OF RUN 9	4-13
A-1 THE MODAL WEIGHTING MATRIX Qc	A-5
A-2 THE RICMATF CONTROL MATRIX	A-8
A-3 STABILITY ANALYSIS OF CONTROL MATRIX (RICMATF)	A-9
A-4 $\bar{Q}$ MATRIX PRODUCED BY CON52 PROGRAM	A-10
A-5 THE RICQF CONTROL MATRIX	A-12
A-6 STABILITY ANALYSIS OF CONTROL MATRIX (RICQF)	A-13
A-7 THE DECMATS OBSERVER	A-14

8/4/mcl702/7

SPAR-R.1135  
ISSUE A

1.0

INTRODUCTION

This report describes the computer programs developed to design and test the M-SAT compensator. The spacecraft model and the theory behind the compensator (controller plus observer) is given in the report SPAR-R.1134. The performance of the compensator is also evaluated, via computer simulation and stability analysis.

8/4/mcl702/8

SPAR-R.1135  
ISSUE A

2.0

THE SPACECRAFT MODAL COORDINATES

The spacecraft modal coordinates are described in the report SPAR-R.1134, but they are given here as they will often be referred to.

The spacecraft dynamic equations are:

$$\ddot{\eta}_r = B_r u$$

$$\ddot{\eta}_e + \tilde{C} \dot{\eta}_e + \tilde{\Omega}^2 \eta_e = B_e u$$

where the  $\eta_r$  vector refers to the rigid modal coordinates and the  $\eta_e$  vector refers to the flexible modal coordinates.

Since the third row of the  $B_r$  matrix is zero, the third rigid mode is uncontrollable and is simply deleted from the compensator design calculations. No attempt is made to control or observe this mode.

The diagonal matrix  $\tilde{\Omega}^2$  specifies the natural frequencies of the flexible modes (squared). These frequencies are listed in Table 2-1, together with the corresponding mode number. Modes 1, 3, 4, 6 are to be controlled and observed. The selection of these 'critical' modes is discussed in SPAR-R.1134. The remaining modes are 'residual' modes and are not controlled or observed.

8/4/mcla702/1

SPAR-R.1135  
ISSUE A

TABLE 2-1  
FLEXIBLE MODE FREQUENCIES

Mode Number	Frequency rad/sec.	Period seconds
*	0.12435	50.53
1	0.15118	41.56
2	0.23952	26.23
*	0.55630	11.29
3	0.69020	9.10
*	0.77964	8.06
4	1.55328	4.05
5	3.13688	2.00
6	3.95721	1.59
7	9.94893	0.63
8	14.00832	0.45
9		
10		
11		

Those marked \* are controlled and observed, remainder are residual modes.

3.0 DESCRIPTION OF COMPUTER PROGRAMS3.1 The Flexible Dynamics and Control Program FD-CSIM3.1.1 Introduction

The program FD-CSIM provides a digital simulation of M-SAT, by computing the behaviour of the spacecraft versus time. The program models the effects of the spacecraft dynamics, a control system and an observer.

The overall system is shown in Figure 3-1. The spacecraft model is excited by the disturbance input  $u_D$  (time dependent) and the control input  $u$ . The modal coordinates  $\eta$ , computed by the spacecraft model, are converted to physical coordinates  $q$  (actual spacecraft motions) by the matrix  $E$ . The measured outputs  $Y$  (sensor outputs) are computed from the physical coordinates by matrix  $P$ . The observer requires the control inputs  $u$  and measured outputs  $Y$  to compute an estimate of the spacecraft modal coordinates  $\hat{\eta}$ . The controller is driven by either the true spacecraft modal coordinates  $\eta$  or the estimate  $\hat{\eta}$ . In real-life, it would be driven by the observer, of course, but for test purposes the true modal coordinates may be used.

The program numerically integrates the equations in the spacecraft model and the observer, and periodically outputs the system status.

3.1.2 Program Design

The program design follows some previous work done at Spar, on a general purpose simulation. The main program provides a sequence of calls to the configuration routine CONFIG, to read in data, perform initialization, simulation output and system calculations. A new configuration routine can easily be 'plugged in' without needing to change the main program.

The equations to be integrated are written as a first-order differential equation, with the state vector V (array V in the program) containing the current values of all the variables. The configuration routine has to compute the first derivative of the state vector, DV, when called by the integration routine INTEG.

The main data is read in from one file, which contains the spacecraft dynamic data, controller and observer matrices. A separate file is used to specify a small number of changes which may be made to the data read in from the main file, so that different runs may be done without needing to alter the main data file. The array VC contains the initial values of various program variables, and is set by a DATA statement in the configuration routine. These default values may also be altered via the small changes data file.

The program can run either the Design Model (just the controlled modes) or the Evaluation Model (controlled modes plus residual modes). However, the data arrays are large and hard to manipulate and it would be difficult to set up models containing just certain selected modes. To overcome this problem, the program contains a facility to artificially hold any mode to its initial value, normally zero. The program can therefore be run with the full Evaluation Model data each time, with selected modes turned off, e.g. turn off all residual modes to produce the Design Model.

### 3.1.3 Spacecraft Equations

The spacecraft dynamics are represented by the equation:

$$M\ddot{q} + C\dot{q} + Kq = Bu + Bd u_d$$

where M is the mass matrix, C is the damping matrix, K is the stiffness matrix, B is the control input matrix and  $B_d$  is the disturbance input matrix.

$q$  is the vector of physical coordinates, which may be divided into rigid coordinates  $q_r$  and flexible coordinates  $q_e$ ,  $q^T = (q_r^T, q_e^T)$ .

$u$  is the control input vector,  $u^T = (u_g^T, u_t^T)$  where  $u_g$  is the gimbal torque and  $u_t$  is the thruster torque or force.

$u_d$  is the disturbance torque or force vector, and is assumed to depend only on orbit position, thus  $u_d$  is a function of time.

The three reaction wheels have not been included in the model, since the program is only a stationkeeping simulation, and the reaction wheels will not be used in this mode.

The spacecraft dynamics equation is transformed by:

$$\dot{q} = E \dot{\eta}$$

to give the modal dynamics equation:

$$\ddot{\eta} + \hat{C} \dot{\eta} + \hat{K} \eta = \hat{B} u + \hat{B}_d u_d \quad (3.1-1)$$

where  $\hat{K}$  is a diagonal matrix containing the squares of the modal frequencies.

The modal coordinate vector  $\eta$  contains a set of rigid coordinates  $\eta_r$  and flexible coordinates  $\eta_e$ ,  $\eta^T = (\eta_r^T, \eta_e^T)$ .

The modal dynamics equation can be rewritten as the first-order differential equation:

$$\frac{d}{dt} \begin{pmatrix} \eta \\ \dot{\eta} \end{pmatrix} = \begin{pmatrix} 0 & I \\ -\hat{K} & -\hat{C} \end{pmatrix} \begin{pmatrix} \eta \\ \dot{\eta} \end{pmatrix} + \begin{pmatrix} 0 \\ \hat{B} \end{pmatrix} u + \begin{pmatrix} 0 \\ \hat{B}_d \end{pmatrix} u_d \quad (3.1-2)$$

The spacecraft sensors are represented by pure gains, sensor dynamics are not modelled. The sensor measurements are specified by the vector  $Y$ , where:

$$Y = P_q$$

and  $P$  is a constant matrix. For simplicity,  $Y$  is assumed to be simply a subset of the physical coordinates  $q$ .

The control actuators are also assumed to be perfect, for example the thrusters may be switched on or off instantaneously. The minimum thruster on-time is limited by the simulation step size. However, the impulse averaging technique may be used to represent pulses with smaller on-times.

The characteristics of the on-board computer, such as numerical accuracy and cycle time, are also not being simulated by this program.

### 3.1.4 Observer Equation

The observer equation is:

$$\dot{\hat{x}} = A_o \hat{x} + K_o Y + B_o u \quad (3.1-3)$$

where  $\hat{x}$  is the observer output,  $Y$  is the sensor measurements and  $u$  is the control input. The matrices  $A_o$ ,  $K_o$  and  $B_o$  are read in as data.

Since the equation Eq (3.1-3) is already a first order differential equation,  $\hat{x}$  may be simply appended to the spacecraft dynamics state vector in Eq (3.1-2).

### 3.1.5 Integration of Equations

The subroutine INTEG computes the state-vector  $V$  at time  $t+\delta t$ , given the state-vector at time  $t$ , where  $\delta t$  is the step size. There are two methods of updating the time variable  $t$ ,

- (a) simply add  $\delta t$  to  $t$ , or,
- (b) multiply  $\delta t$  by an integer specifying the loop count.

Method (a) allows a variable step size, but suffers from a loss of accuracy as  $t$  becomes large relative to  $\delta t$ . Method (b) does not have the accuracy problem, but does not allow a variable step size. The second method was used in the simulation.

Subroutine INTEG contains two integration methods, the Euler integration and the 4th order Runge-Kutta integration. Only one of these methods is used during a run, which is determined by the flag INT.

During initial testing, the Euler method was used as this was thought to be cheaper to run, but it appeared to be unstable as shown below:

The modal dynamics equation for the spacecraft, with zero input, is:

$$\ddot{\eta} + \hat{C}\dot{\eta} + \hat{k}\eta = 0$$

Take  $\hat{C}$  to be diagonal, and let  $x$  be an element of vector  $\eta_e$ , and  $c$  and  $w^2$  be the corresponding elements of  $\hat{C}$  and  $\hat{k}$ , so that  $\ddot{x} + c\dot{x} + w^2x = 0$ .

Let  $p_k = x$  and  $q_k = \dot{x}$  at time  $k\delta t$ .

The Euler integration method computes  $p_{k+1}$  and  $q_{k+1}$  according to:

$$p_{k+1} = p_k + \delta t q_k$$

$$q_{k+1} = q_k + \delta t (-w^2 p_k - c q_k)$$

Rewrite this in matrix form:

$$\begin{pmatrix} p_{k+1} \\ q_{k+1} \end{pmatrix} = \begin{pmatrix} 1 & \delta t \\ -w^2 \delta t & 1 - \delta t c \end{pmatrix} \begin{pmatrix} p_k \\ q_k \end{pmatrix}$$

$$\text{or } V_{k+1} = A V_k$$

This system is stable only if the eigenvalues of  $A$  lie within the unit circle.

The eigenvalues are given by  $\lambda = 1 + \omega \delta t (-\zeta \pm \sqrt{\zeta^2 - 1})$   
where  $\zeta = c/(2\omega)$

Since  $\zeta < 1$ ,  $\lambda$  is complex and the magnitude of  $\lambda$  is given by:

$$|\lambda|^2 = 1 - c \zeta \delta t + (\omega \delta t)^2$$

If  $\zeta \delta t$  is chosen to be greater than  $c/w^2$ , the system will be unstable. The worst case is the 8th flexible mode in the evaluation model, which would require  $\zeta \delta t$  to be less than 0.0057 secs.

The 4th order Runge-Kutta method does not appear to suffer from this type of instability problem, but it is more expensive to run if a similar step size is used. It was found that the step size could be increased by a factor of 5 (from 0.01 sec. to 0.05 sec.) without causing significant errors, and the cost of a run was similar to what it was originally (about \$1 per second of simulation time).

## 3.1.6

The Controller

The control actuators consist of 8 thrusters and 2 gimbal torquing devices. The control routine is only activated at discrete times  $0, \gamma, 2\gamma, \dots$  etc. and causes the thrusters to fire an impulse at these times only. The gimbal control torque remains constant during each period  $\gamma$ , and is reset to a new level when the control routine is activated.

The input to the controller is the state vector  $x_c$ , consisting of the rigid and flexible modes that are to be controlled, plus their derivatives,  $x_c^T = (\eta_r^T, \dot{\eta}_r^T, \eta_e^T, \dot{\eta}_e^T)$ . These may be taken directly from the spacecraft model, for test purposes, but in real-life, the observer estimates must be used.

A bias value may be subtracted from each of the rigid modes in  $x_c$ , in order to be able to command a change in spacecraft attitude or antenna direction. The array EREF contains these bias values, which are applied when the simulation time exceeds the value of REFTIM.

The resulting rigid modes are then individually checked to see if they exceed a deadband level, contained in array DEAD. If all rigid modes are within their deadbands, then no control needs to be applied at this instant, i.e. the thrusters are not fired and the gimbal control torque is set to zero.

If one or more deadbands are exceeded, the thruster impulses  $u_t$  and gimbal torques  $u_g$  are calculated from:

$$\begin{pmatrix} u_g \\ u_t \end{pmatrix} = F x_c$$

where  $F$  is the control matrix which was read in as data. Since the thrusters cannot fire in the negative direction, a procedure is used to modify the  $u_t$  to ensure all values are positive. The modified  $u_t$  will produce the same effect on the spacecraft structure, but more fuel may be used.

To apply the impulse specified by  $u_t$ , the program fires the thruster over one step size  $\delta t$ , at a level given by  $u_t/\delta t$ . In real-life, the thruster level would be fixed, and the on-time would be varied to give the required impulse. Some modification may be needed if the required on-time is less than the minimum on-time. The gimbal torque is simply set equal to the value of  $u_g$ .

A diagram of the controller is shown in Figure 3-2 and program flowchart in the Appendix C.

3.1.7 Disturbance Torques and Forces

Solar pressure is the only disturbance effect simulated, in order to demonstrate the control system. The disturbance input  $u_d$  is specified as

$$u_d^T = (f^T, g^T, g_r^T, f_r^T, g_r^T) \quad \text{where } f$$

is the total force acting on the vehicle CM,  $g$  is the total torque about the bus frame origin,  $f_r$  is the disturbance force acting at the reflector CM,  $g_r$  is the torque about the reflector frame origin, and  $g_r'$  is the first two components of  $g_r$ . The disturbance input matrix  $B_d = I$ , the identity, therefore  $\hat{B}_d = E^T$ .

The values of the vectors  $f$ ,  $g$ ,  $f_r$ ,  $g_r$  are given below:

$$f = \begin{pmatrix} -A_5 \sin \gamma & -A_2 \sin 2\gamma \\ 0 \\ A_5 \cos \gamma + A_4 (\cos 2\gamma + 1) \end{pmatrix}$$

$$g = \begin{pmatrix} A_6 \cos \gamma + A_7 (\cos 2\gamma + 1) \\ A_8 \sin \gamma + A_9 \sin 2\gamma \\ A_6 \sin \gamma + A_{10} \sin 2\gamma \end{pmatrix}$$

$$f_r = \begin{pmatrix} A_3 \sin \gamma \\ 0 \\ A_3 \cos \gamma \end{pmatrix}$$

$$g_r = \begin{pmatrix} A_{11} \cos \gamma \\ A_{12} \sin \gamma \\ -A_{11} \sin \gamma \end{pmatrix}$$

where  $\gamma$  is the orbital position relative to the sun and the coefficients are given by:

$$A_1 = A_a(\Omega/C)(1 + \nu_a)$$

$$A_2 = A_b(\Omega/C)(1 - \nu_b)/2$$

$$A_3 = A_r(\Omega/C)(1 + \nu_r)/20$$

$$A_4 = A_b(\Omega/C)(1 + \nu_b)/2$$

8/4/mcl702/17

SPAR-R.1135  
ISSUE A

$$A_5 = A_1 + A_3$$

$$A_6 = A_1(\rho_a + \rho_{bay}) + A_3 (\rho_{bry} - \rho_{ry})$$

$$A_7 = A_4 \rho_{by}$$

$$A_8 = -A_1 \rho_{baz} - A_3 (\rho_{rz} + \rho_{brz})$$

$$A_9 = -A_2 \rho_{bz}$$

$$A_{10} = A_2 \rho_{by}$$

$$A_{11} = A_3 \rho_{ry}$$

$$A_{12} = A_3 \rho_{rz}$$

where:

$\rho_a$ ,  $\rho_r$ ,  $\rho_b$  are the surface areas of the array, reflector and bus respectively

$\rho_a$ ,  $\rho_r$ ,  $\rho_b$  are the reflectivities of the array, reflector and bus respectively

$Q/C$  is the solar energy constant

$\rho_{by}$ ,  $\rho_{bz}$  are the coordinates of the bus CP in the bus frame

$\rho_{ry}$ ,  $\rho_{rz}$  are the coordinates of reflector CP in the reflector frame

$\rho_a$  is the coordinate of the array CP in the array frame

$\rho_{bay}$ ,  $\rho_{baz}$  are the coordinates of the array CP in the bus frame

$\rho_{bry}$ ,  $\rho_{brz}$  are the coordinates of the reflector CP in the bus frame

3.1.8 The Cost Function

The cost function is given by

$$J_k = \sum_{h=0}^{k-1} \left( x_{c,h+1}^T Q_c x_{c,h+1} + u_h^T \bar{R} u_h \right) + x_{c,k}^T Q_{ck} x_{c,k}$$

where  $x_{c,h}$  is the value of the control state vector  $x_c$  at time  $t = h\gamma$ ,  $u_h$  is the value of the control vector  $\begin{pmatrix} u_g \\ u_b \end{pmatrix}$  at time  $t = h\gamma$

and  $Q_c$ ,  $\bar{R}$  and  $Q_{ck}$  are constant matrices.

The terminal condition specified by  $Q_{ck}$  is expected to be small compared with the rest of the cost function and has not been included in the simulation (mainly because of the large amount of space occupied by array  $Q_{ck}$ ).

Since the value of  $K$  is large, the summation is not terminated but continues for the whole duration of the simulation. The current value of the cost function is output periodically from the program.

The matrices  $Q_c$  and  $\bar{R}$  are read in as data and are given on the same data file as the control matrix  $F$ .

The cost function can be used as a check, to see if matrix  $F$  does indeed produce a minimum value of  $J_k$  (the control vector  $u$  can be artificially modified by the UMULT array). Also the total cost, predicted by the program which computes  $F$  can be checked.

3.2 The Controller Design Program RICO3.2.1 Introduction

The program RICO calculates the control feedback matrix  $F$  which relates the control input vector  $u_b$  to the control state vector  $x_{cb}$ ,  $u_b = Fx_{cb}$ . The control law is applied only at discrete times  $b\gamma$ , where  $\gamma$  is the control period, and  $b$  is an integer.

The program stores the  $F$  matrix in a separate file on the computer, in a form which can be easily read by another program. The  $F$  matrix may be used by the numerical simulation program FDGSIM, or one of the stability programs STABF or STABFO.

The control matrix is designed to minimize a particular cost function, which is composed of a weighted sum of the controlled modes, the control input and the control spillover.

The controller produced by this program is an earlier version of that given in the report SPAR-R.1134.

3.2.2 The Continuous Spacecraft Equations

The controlled state vector is assumed to be in the form  $x_c^T = (\eta_r^T, \dot{\eta}_r^T, \eta_{ec}^T, \dot{\eta}_{ec}^T)$  where  $\eta_r$  is the rigid mode vector and  $\eta_{ec}$  is the controlled elastic mode vector. The state vector for the residual modes is in the form  $x_R^T = (\eta_{er}^T, \dot{\eta}_{er}^T)$  where  $\eta_{er}$  represents the residual elastic modes.

Since all of the rigid modes  $\eta_r$  are assumed to be controlled, the third rigid mode must be deleted from the equations since it is uncontrollable. To accomplish this, the third row (containing only zeros) is deleted from the control input matrix  $B_r$ .

The spacecraft equations are:

$$\ddot{\eta}_r = B_r u$$

$$\ddot{\eta}_e + \tilde{C} \dot{\eta}_e + \tilde{\Omega}^2 \eta_e = B_e u$$

The elastic modes in the second equation must be rewritten in terms of  $\eta_{elT}^1 = (\eta_{ec}^T, \eta_{er}^T)$  where  $\eta_{ec}$  are the controlled elastic modes and  $\eta_{er}$  are the residual elastic modes:

$$\ddot{\eta}_e^1 + \hat{C}' \dot{\eta}_e^1 + \hat{\Omega}'^2 \eta_e^1 = B'_e u$$

where  $\hat{C}^1$ ,  $\hat{\Omega}^1$  and  $B_e^1$  are simply rearranged form of  $\hat{C}$ ,  $\hat{\Omega}$  and  $B_e$  respectively.

$$\text{Let } \hat{C}' = \begin{pmatrix} \hat{C}_c & \hat{C}_{12} \\ \hat{C}_{21} & \hat{C}_R \end{pmatrix}, \quad \hat{\Omega}' = \begin{pmatrix} \hat{\Omega}_c & 0 \\ 0 & \hat{\Omega}_R \end{pmatrix} \text{ and } B'_e = \begin{pmatrix} B_{ec} \\ B_{er} \end{pmatrix}.$$

If we make the assumption that  $\hat{C}_{12} = \hat{C}_{21} = 0$ , then the controlled and residual equations can be decoupled:

$$\ddot{\eta}_{ec} + \hat{C}_c \dot{\eta}_{ec} + \hat{\Omega}_c^2 \eta_{ec} = B_{ec} u$$

$$\text{and } \ddot{\eta}_{er} + \hat{C}_R \dot{\eta}_{er} + \hat{\Omega}_R^2 \eta_{er} = B_{er} u$$

The spacecraft equations can now be written as:

$$\dot{x}_c = A_c x_c + B_c u \quad (3.2-1)$$

$$\dot{x}_R = A_R x_R + B_R u$$

$$A_c = \begin{pmatrix} 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I \\ 0 & 0 & -\hat{\Omega}_c^2 & -\hat{C}_c \end{pmatrix} \quad A_R = \begin{pmatrix} 0 & I \\ -\hat{\Omega}_R^2 & -\hat{C}_R \end{pmatrix}$$

$$x_c = \begin{pmatrix} \eta_r \\ \dot{\eta}_r \\ \eta_{ec} \\ \dot{\eta}_{ec} \end{pmatrix}, \quad B_c = \begin{pmatrix} 0 \\ B_r \\ 0 \\ B_{ec} \end{pmatrix}, \quad x_R = \begin{pmatrix} \eta_{er} \\ \dot{\eta}_{er} \end{pmatrix}, \quad B_R = \begin{pmatrix} 0 \\ B_{er} \end{pmatrix}$$

3.2.3 The Discrete Spacecraft Equations

The control input  $u = \begin{pmatrix} u_g \\ u_t \end{pmatrix}$ , where  $u_g$  refers to the gimbal input and  $u_t$  is the thruster input. The control input matrices can be partitioned to match  $U_g$  and  $U_t$ :

$$B_c = (B_{cg} \quad B_{ct}) \text{ and } B_R = (B_{Rg} \quad B_{Rt})$$

Referring to Eq (3.2-1), the spacecraft equations become:

$$\dot{x}_c = A_c x_c + B_{cg} u_g + B_{ct} u_t$$

$$\dot{x}_R = A_R x_R + B_{Rg} u_g + B_{Rt} u_t$$

where  $x_c$  contains the controlled modes and their first-derivatives, and  $x_R$  the residual modes (and their first-derivatives).

The control law is applied at discrete times  $b\tau$ , where  $b$  is an integer and  $\tau$  is the control period. The thrusters fire an impulse specified by  $u_t$  at these discrete times and the gimbal torque is held constant at  $u_g$  over the control period.

The discretized spacecraft equations are:

$$x_{c,b+1} = \bar{A}_c x_{c,b} + \bar{B}_c u_{b\tau}$$

$$x_{R,b+1} = \bar{A}_R x_{R,b} + \bar{B}_R u_{b\tau}$$

where  $x_{c,b} = x_c(b\tau)$ ,  $x_{R,b} = x_R(b\tau)$ ,  $u^T = (u_g^T, u_t^T)$

$$\bar{A}_c = \exp(A_c \tau), \quad \bar{A}_R = \exp(A_R \tau), \quad \bar{B}_c = (\bar{B}_{cg} \quad \bar{B}_{ct}), \quad \bar{B}_R = (\bar{B}_{Rg} \quad \bar{B}_{Rt})$$

$$\bar{B}_{cg} = \exp(A_c \tau) \int_0^\tau \exp(-A_c s) ds B_{cg}, \quad \bar{B}_{ct} = \exp(A_c \tau) B_{ct}$$

$$\bar{B}_{Rg} = \exp(A_R \tau) \int_0^\tau \exp(-A_R s) ds B_{Rg}, \quad \bar{B}_{Rt} = \exp(A_R \tau) B_{Rt}$$

## 3.2.4

The Calculation of Matrix Exponentials

This section covers the calculation of the matrix exponentials for the discretized spacecraft matrices  $\tilde{A}_C$ ,  $\tilde{B}_C$ ,  $\tilde{A}_R$ ,  $\tilde{B}_R$ .

If the damping matrix  $\hat{C}$  is assumed to be diagonal, the matrix exponentials can be calculated directly, otherwise the eigenvector method is used. If the program flag IDIAG is zero,  $\hat{C}$  is taken to be diagonal, otherwise the full  $\hat{C}$  is used.

(a) Diagonal damping matrix  $\hat{C}$ 

$$\text{Let } \tilde{A}_C = \begin{pmatrix} A_1 & 0 \\ 0 & A_2 \end{pmatrix} \text{ where } A_1 = \begin{pmatrix} 0 & I \\ 0 & 0 \end{pmatrix} \text{ and } A_2 = \begin{pmatrix} 0 & I \\ -\Omega_c^2 & -2G_c\Omega_c \end{pmatrix}$$

$$\text{and } 2G_c\Omega_c = \hat{C}_c, \quad \Omega_c = \text{diag}\{\omega_{ci}\}, \quad G_c = \text{diag}\{g_{ci}\}$$

$$\text{Then } \exp(\tilde{A}_C\gamma) = \begin{pmatrix} \exp(A_1\gamma) & 0 \\ 0 & \exp(A_2\gamma) \end{pmatrix}$$

The series method can be used for  $\exp(A_1\gamma)$ :

$$\exp(A_1\gamma) = \begin{pmatrix} I & \gamma I \\ 0 & I \end{pmatrix}$$

The resolvent matrix method gives the solution for  $\exp(A_2\gamma)$ :

$$\exp(A_2\gamma) = \begin{pmatrix} A(\gamma) & B(\gamma) \\ \dot{A}(\gamma) & \dot{B}(\gamma) \end{pmatrix}$$

$$\text{where } A(t) = \text{diag}\{\alpha_i(t)\}, \quad B(t) = \text{diag}\{\beta_i(t)\},$$

$$\dot{A}(t) = \text{diag}\{\dot{\alpha}_i(t)\}, \quad \dot{B}(t) = \text{diag}\{\dot{\beta}_i(t)\}$$

$$\text{and } \alpha_i(t) = e^{-\lambda_i t} (\cos \omega_{ni} t + \frac{\lambda_i}{\omega_{ni}} \sin \omega_{ni} t)$$

$$\beta_i(t) = \frac{1}{\omega_{ni}} e^{-\lambda_i t} \sin \omega_{ni} t$$

$$\dot{\alpha}_i(t) = -\omega_{ni}^2 \beta_i(t), \quad \dot{\beta}_i(t) = \alpha_i(t) - 2G_c \omega_{ci} \beta_i(t)$$

$$\omega_{ni} = \omega_{ci} \sqrt{1 - g_{ci}^2}, \quad \lambda_i = G_c \omega_{ci}$$

The matrix integral  $\exp(A_C \tau) \int_0^\tau \exp(-A_C s) ds$  is given by:

$$\exp(A_C \tau) \int_0^\tau \exp(-A_C s) ds = \begin{pmatrix} L_1 & 0 \\ 0 & L_2 \end{pmatrix}$$

$$\text{where } L_1 = \begin{pmatrix} \gamma I & \frac{1}{4}\tau^2 I \\ 0 & \gamma I \end{pmatrix}$$

$$\text{and } L_2 = \begin{pmatrix} 2g_c \Omega_c^{-1} (I - A(\tau)) + B\gamma & \Omega_c^{-2} (I - A(\tau)) \\ -2g_c \Omega_c^{-1} \dot{A}(\tau) + B(\tau) - I & -\Omega_c^{-2} \dot{A}(\tau) \end{pmatrix}$$

The values of  $\alpha_i$ ,  $\beta_i$ ,  $\dot{\alpha}_i$  and  $\dot{\beta}_i$  are calculated by the SETUP subroutine. Subroutine EXPACT assembles the matrix  $\exp(A_C \tau)$  and subroutine EXPINT assembles  $\exp(A_C \tau) \int_0^\tau \exp(-A_C s) ds$ .

The matrix  $A_R$  has the same form as  $A_2$  and so the same subroutines can be used.

(b) Damping matrix not diagonal.

For this case, the eigenvector method is used to calculate the matrix exponential. This method is described under a separate chapter (3.7.3), since it is used by a number of other programs. The exponential of the full  $A_C$  matrix cannot be computed directly because  $A_C$  contains a block row and block column of zeros.

$$A_C \text{ can again be partitioned as } A_C = \begin{pmatrix} A_1 & 0 \\ 0 & A_2 \end{pmatrix}$$

then  $\exp(A_1 \tau) = \begin{pmatrix} I & \gamma I \\ 0 & I \end{pmatrix}$  and  $\exp(A_2 \tau)$  can be computed by the eigenvector method.

For the matrix integral

$$\exp(A_C \tau) \int_0^\tau \exp(-A_C s) ds = \begin{pmatrix} L_1 & 0 \\ 0 & L_2 \end{pmatrix},$$

$L_1$  is given as above, and  $L_2$  is given by:

$$L_2 = (\exp(A_c \gamma) - I) A_c^{-1}$$

The discretized spacecraft matrices can now be calculated.

### 3.2.5 Controller Design Equations

The cost function to be minimized by the control matrix  $F$  is:

$$J_K = \sum_{b=0}^{K-1} \left( x_{cb+1}^T Q_c x_{cb+1} + u_b^T R u_b + u_b^T \bar{B}_c^T R_R \bar{B}_R u_b \right) + x_{ck}^T Q_{ck} x_{ck}$$

The solution is given by:

$$F = -F_b$$

$$\text{where } F_b = \left\{ \bar{R} + \bar{B}_c^T Q_{b+1} \bar{B}_c \right\}^{-1} \bar{B}_c^T Q_{b+1} \bar{A}_c$$

$$Q_b = Q_c + P_b, \quad \bar{R} = R + \bar{B}_R^T R_R \bar{B}_R$$

$$P_b = \bar{A}_c^T Q_{b+1} [\bar{A}_c - \bar{B}_c F_b], \quad P_K = Q_{ck}$$

$$\text{and } b = 0, 1, \dots, K-1$$

The solution can be computed backwards, starting from  $P_K$  and finishing with  $F_0$  and  $P_0$ . The value of  $K$  should be large enough so that  $F_b$  and  $P_b$  converge to almost constant matrices as  $b \rightarrow 0$ .

The total cost over the time period 0 to  $K \gamma$  is given by  $x_{co}^T P_0 x_{co}$ .

### 3.2.6 Stability

The stability of the spacecraft plus controller system can be calculated, assuming the spacecraft state-vector is fed back directly to the controller (no observer in the loop).

The discretized and decoupled spacecraft equations are:

$$x_{C,h+1} = \bar{A}_C x_{C,h} + \bar{B}_C u_h$$

$$x_{R,h+1} = \bar{A}_R x_{R,h} + \bar{B}_R u_h$$

and control feedback:

$$u_h = F x_{C,h}$$

Therefore

$$x_{C,h+1} = (\bar{A}_C + \bar{B}_C F) x_{C,h}$$

The controlled modes are stable if and only if the eigenvalues of  $(\bar{A}_C + \bar{B}_C F)$  all have magnitude less than unity. The residual modes are stable if and only if the eigenvalues of  $\bar{A}_R$  all have magnitude less than unity.

The residual modes should always be stable since they only depend on the spacecraft structure, but the controlled modes are affected by the control matrix F.

### 3.2.7 Convergence of Control Matrix

The value of K should be large enough for the matrices  $F_h$  and  $P_h$  to converge to some constant values as  $h \rightarrow 0$ . The program calculates the sum of the absolute difference between the elements of  $F_{h+1}$  and  $F_h$ , and  $P_{h+1}$  and  $P_h$ , and prints the result for each value of  $h$ . This allows the user to see if the matrices are in fact converging.

The convergence appears to be affected by an imbalance between the weighting matrices. For example,  $Q_C$  and  $Q_{CK}$  were set to the 'modal output weighting matrix' Q specified in the DYNACON memorandum (August 25, 1982) with a typical element of the order of  $10^{-6}$ , and R and  $R_R$  were set to the identity matrix. A value of K = 50 produced no discernable convergence. When R and  $R_R$  were reset to  $10^{-6}I$ , convergence was obtained. Normally K = 25 is used, producing typically a final difference in  $F_h$  of 0.1 and  $P_h$  of  $10^{-5}$ .

3.3 The Controller Design Program CON523.3.1 Introduction

The program CON52 was originally intended to calculate the feedback control matrix  $F$ , as defined in the document SPAR-R.1134. However, the program has not been finished and it only calculates the  $\bar{Q}$  matrix given by equation (5-4) in SPAR-R.1134. This  $\bar{Q}$  matrix can be used as the modal weighting matrix in the controller design program RICQ.

3.3.2 Equations

The solution to all the equations in (5-4) of document SPAR-R.1134 is given here, although only  $\bar{Q}$  is implemented at present.

The equations are (writing  $\Upsilon$  instead of  $A\Upsilon$  for convenience):

$$\bar{Q} = \int_0^{\tau} (e^{A_c t})^T Q_c (e^{A_c t}) dt$$

$$B_u(t) = \left[ \int_0^t e^{-A_c s} ds B_{cg} : B_{ct} \right]$$

$$\bar{P} = \int_0^{\tau} (e^{A_c t})^T Q_c (e^{A_c t}) B_u(t) dt$$

$$\bar{R} = \text{diag}[R_g \tau, R_t] + \int_0^{\tau} (e^{A_c t} B_u(t))^T Q_c (e^{A_c t} B_u(t)) dt$$

$$\bar{A}_c = e^{A_c \tau}, \quad \bar{B}_c = \bar{A}_c B_u(\tau)$$

where  $A_c = \begin{pmatrix} G & 0 \\ 0 & H \end{pmatrix}$

$$G = \begin{pmatrix} 0 & I \\ 0 & 0 \end{pmatrix}, \quad H = \begin{pmatrix} 0 & I \\ -\Omega_c^2 & -C_c \end{pmatrix}$$

with dimensions  $G: 2n_{rc} \times 2n_{rc}$ ,  $H: 2n_{ec} \times 2n_{ec}$   
 where  $n_{rc}$  is the number of rigid controlled modes  
 (7) and  $n_{ec}$  is the number of flexible controlled  
 modes (normally 4).

The modal weighting matrix  $Q_C$  must be partitioned  
 as follows:

$$Q_C = \begin{pmatrix} Q_{C11} & Q_{C12} \\ Q_{C21} & Q_{C22} \end{pmatrix} \quad Q_{C11}: 2n_{rc} \times 2n_{rc}, \quad Q_{C12}: 2n_{rc} \times 2n_{ec}$$

$$Q_{C21}: 2n_{ec} \times 2n_{rc}, \quad Q_{C22}: 2n_{ec} \times 2n_{ec}$$

and  $Q_{C11} = \begin{pmatrix} Q_{C11A} & Q_{C11B} \\ Q_{C11C} & Q_{C11D} \end{pmatrix} \quad Q_{C11A}, Q_{C11B}, Q_{C11C}, Q_{C11D}: n_{rc} \times n_{rc}$

$$Q_{C12} = \begin{pmatrix} Q_{C12A} \\ Q_{C12B} \end{pmatrix} \quad Q_{C12A}, Q_{C12B}: n_{rc} \times 2n_{ec}$$

$$Q_{C21} = \begin{pmatrix} Q_{C21A} & Q_{C21B} \end{pmatrix} \quad Q_{C21A}, Q_{C21B}: 2n_{ec} \times n_{rc}$$

The control distribution matrices are partitioned  
 as:

$$B_{CG} = \begin{pmatrix} 0 \\ Br_g \\ 0 \\ Be_{cg} \end{pmatrix} \quad B_{CT} = \begin{pmatrix} 0 \\ Br_t \\ 0 \\ Be_{ct} \end{pmatrix}$$

$Br_g: n_{rc} \times n_g$ ,  $Be_{cg}: n_{ec} \times n_g$ ,  $Br_t: n_{rc} \times n_t$ ,  
 $Be_{ct}: n_{ec} \times n_t$  where  $n_g$  is the number of gimbal  
 control inputs (2) and  $n_t$  is the number of  
 thrusters (8).

The exponential of  $A_{CT}$  is given by:

$$\exp(A_{CT}) = \begin{pmatrix} \exp(Gt) & 0 \\ 0 & \exp(Ht) \end{pmatrix}$$

$\exp(Gt)$  can be simply calculated by the series  
 method:

$$\exp(Gt) = I + Gt + G^2 t^2 + \dots$$

$$= \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix} + \begin{pmatrix} 0 & I \\ 0 & 0 \end{pmatrix} t + \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} t^2 + \dots$$

$$\therefore \exp(Gt) = \begin{pmatrix} I & tI \\ 0 & I \end{pmatrix} \quad (1)$$

$\exp(Ht)$  must be calculated by the eigenvalue method:

$$\exp(Ht) = V \exp(\Delta t) V^{-1} \quad (2)$$

where  $\Delta$  is the diagonal matrix of (complex) eigenvalues of  $H$   $\{\lambda_i, i=1 \dots 2n_{ec}\}$  and  $V$  is the corresponding matrix of (complex) eigenvectors.

Note that the exponential of  $A_C t$  cannot be given directly by the eigenvalue method, since  $A_C$  has repeated eigenvalues of zero. Jordan blocks could be introduced to allow for this, but there does not seem to be any advantage over the method used here.

The element in the  $i$ th row and  $j$ th column of some matrix  $A$  may be written as  $(A)_{i,j}$  where required.

## 3.3.3

Solution for  $\bar{Q}$

$$\bar{Q} = \begin{pmatrix} \bar{Q}_{11} & \bar{Q}_{12} \\ \bar{Q}_{21} & \bar{Q}_{22} \end{pmatrix} \quad \begin{aligned} \bar{Q}_{11} &: 2n_{rc} \times 2n_{rc}, & \bar{Q}_{12} &: 2n_{rc} \times 2n_{ec} \\ \bar{Q}_{21} &: 2n_{ec} \times 2n_{rc}, & \bar{Q}_{22} &: 2n_{ec} \times 2n_{ec} \end{aligned} \quad (3)$$

$$\text{where } \bar{Q}_{11} = \begin{pmatrix} T Q_{C11A} & \gamma_2 T^2 Q_{C11A} + \gamma Q_{C11B} \\ \gamma_2 T^2 Q_{C11A} + \gamma Q_{C11C}, & \gamma_3 T^3 Q_{C11A} + \frac{1}{2} T^2 (Q_{C11B} + Q_{C11C}) + T Q_{C11D} \end{pmatrix} \quad (4)$$

$$\bar{Q}_{12} = \begin{pmatrix} Q_{C12A} H_1 \\ Q_{C12A} H_2 + Q_{C12B} H_1 \end{pmatrix} \quad (7)$$

$$\bar{Q}_{21} = \begin{pmatrix} H_1^T Q_{C21A} \\ H_2^T Q_{C21A} + H_1^T Q_{C21B} \end{pmatrix} \quad (8)$$

$$\bar{Q}_{22} = V^{-1} {}^T Q'_H V^{-1} \quad (9)$$

$$\text{and } (Q'_H)_{i,j} = \frac{(Q'_{22})_{i,j}}{(\lambda_i + \lambda_j)} \left[ \exp(\{\lambda_i + \lambda_j\} \gamma) - 1 \right] \quad (10)$$

for  $i = 1 \dots 2n_{rc}$ ,  $j = 1 \dots 2n_{ec}$

$$\bar{R}_{11A} = B_{rg}^T \left[ \frac{1}{20} \gamma^5 Q_{c11A} + \frac{1}{8} \gamma^4 (Q_{c11B} + Q_{c11C}) + \frac{1}{3} \gamma^3 Q_{c11D} \right] B_{rg} \quad (29)$$

$$\bar{R}_{11B} = B_{rg}^T \left[ \frac{1}{2} Q_{c12A} H_7 + Q_{c12B} H_4 \right] \begin{pmatrix} 0 \\ B_{ecg} \end{pmatrix} \quad (30)$$

$$\bar{R}_{11C} = (0 : B_{ecg}^T) \left[ \frac{1}{2} H_7^T Q_{c21A} + H_4^T Q_{c21B} \right] B_{rg} \quad (31)$$

$$\bar{R}_{11D} = (0 : B_{ecg}^T) V^{-1T} Q_L' V^{-1} \begin{pmatrix} 0 \\ B_{ecg} \end{pmatrix} \quad (32)$$

$$\bar{R}_{12A} = B_{rg}^T \left[ \frac{1}{8} \gamma^4 Q_{c11A} + \frac{1}{3} \gamma^3 \left( \frac{1}{2} Q_{c11B} + Q_{c11C} \right) + \frac{1}{2} \gamma^2 Q_{c11D} \right] B_{rt} \quad (33)$$

$$\bar{R}_{12B} = B_{rg}^T \left[ \frac{1}{2} Q_{c12A} H_5 + Q_{c12B} H_2 \right] \begin{pmatrix} 0 \\ B_{ect} \end{pmatrix} \quad (34)$$

$$\bar{R}_{12C} = (0 : B_{ecg}^T) \left[ H_4^T Q_{c12A} + H_3^T Q_{c12B} \right] B_{rt} \quad (35)$$

$$\bar{R}_{12D} = (0 : B_{ecg}^T) V^{-1T} Q_N' V^{-1} \begin{pmatrix} 0 \\ B_{ect} \end{pmatrix} \quad (36)$$

$$\bar{R}_{21A} = B_{rt}^T \left[ \frac{1}{8} \gamma^4 Q_{c11A} + \frac{1}{3} \gamma^3 (Q_{c11B} + \frac{1}{2} Q_{c11C}) + \frac{1}{2} \gamma^2 Q_{c11D} \right] B_{rg} \quad (37)$$

$$\bar{R}_{21B} = B_{rt}^T \left[ Q_{c12A} H_4 + Q_{c12B} H_3 \right] \begin{pmatrix} 0 \\ B_{ecg} \end{pmatrix} \quad (38)$$

$$\bar{R}_{21C} = (0 : B_{ect}^T) \left[ \frac{1}{2} H_5^T Q_{c21A} + H_2^T Q_{c21B} \right] B_{rg} \quad (39)$$

$$\bar{R}_{21D} = (0 : B_{ect}^T) V^{-1T} Q_M' V^{-1} \begin{pmatrix} 0 \\ B_{ecg} \end{pmatrix} \quad (40)$$

$$\bar{R}_{22A} = B_{rt}^T \left[ \frac{1}{3} \gamma^3 Q_{c11A} + \frac{1}{2} \gamma^2 (Q_{c11B} + Q_{c11C}) + \gamma Q_{c11D} \right] B_{rt} \quad (41)$$

$$\bar{R}_{22B} = B_{rt}^T \left[ Q_{c12A} H_2 + Q_{c12B} H_1 \right] \begin{pmatrix} 0 \\ B_{ect} \end{pmatrix} \quad (42)$$

$$\bar{R}_{22C} = (0 : B_{ect}^T) \left[ H_2^T Q_{c21A} + H_1^T Q_{c21B} \right] B_{rt} \quad (43)$$

$$\bar{R}_{22D} = (0 : B_{ect}^T) \bar{Q}_{22} \begin{pmatrix} 0 \\ B_{ect} \end{pmatrix} \quad (44)$$

where  $H_7 = V \left\{ \exp(\Delta\gamma) \left[ \gamma^2 \Delta^{-2} - 2\gamma \Delta^{-3} + 2\Delta^{-4} \right] - 2\Delta^{-4} - \frac{1}{3}\gamma^3 \Delta^{-1} \right\} V^{-1}$  (45)

$$(Q'_L)_{i,j} = \frac{(Q'_{C22})_{i,j}}{\lambda_i \lambda_j} \left\{ \frac{[\exp([\lambda_i + \lambda_j]\gamma) - 1]}{(\lambda_i + \lambda_j)} - \frac{[\exp(\lambda_i\gamma) - 1]}{\lambda_i} - \frac{[\exp(\lambda_j\gamma) - 1]}{\lambda_j} + \gamma \right\} \quad (46)$$

for  $i = 1$  to  $2n_{ec}$ ,  $j = 1$  to  $2n_{ec}$

$$(Q'_M)_{i,j} = \frac{(Q'_{C22})_{i,j}}{\lambda_j} \left\{ \frac{[\exp([\lambda_i + \lambda_j]\gamma) - 1]}{(\lambda_i + \lambda_j)} - \frac{[\exp(\lambda_i\gamma) - 1]}{\lambda_i} \right\} \quad (47)$$

for  $i = 1$  to  $2n_{ec}$ ,  $j = 1$  to  $2n_{ec}$

$$(Q'_N)_{i,j} = \frac{(Q'_{C22})_{i,j}}{\lambda_i} \left\{ \frac{[\exp([\lambda_i + \lambda_j]\gamma) - 1]}{(\lambda_i + \lambda_j)} - \frac{[\exp(\lambda_j\gamma) - 1]}{\lambda_j} \right\} \quad (48)$$

for  $i = 1$  to  $2n_{ec}$ ,  $j = 1$  to  $2n_{ec}$

(note  $Q_N = Q_M^T$  only if  $Q_{C22} = Q_C^T 22$ )

and  $H_1, H_2$  are given for  $\bar{Q}$  and  $H_3, H_4, H_5$  are given for  $\bar{F}$ .

## 3.3.7

Solution for  $\bar{A}_C$  and  $\bar{B}_C$ 

$$\bar{A}_C = \begin{pmatrix} \bar{G} & 0 \\ 0 & \bar{H} \end{pmatrix} \quad \bar{G}: 2n_{rc} \times 2n_{rc}, \quad \bar{H}: 2n_{ec} \times 2n_{ec} \quad (40)$$

$$\bar{G} = \begin{pmatrix} I & \gamma I \\ 0 & I \end{pmatrix} \quad (50)$$

$$\bar{H} = V \exp(\Delta\gamma) V^{-1} \quad (51)$$

$$\bar{B}_C = \bar{A}_O B_u(\gamma) \quad (52)$$

$$B_u(\gamma) = \begin{pmatrix} B_{u11}(\gamma) & B_{u12} \\ B_{u21}(\gamma) & B_{u22} \end{pmatrix} \quad (53)$$

$$B_{u11}(\gamma) = \begin{pmatrix} -\frac{1}{2}\gamma^2 Br_g \\ \gamma Br_g \end{pmatrix} \quad (54)$$

$$B_{u21}(\gamma) = H_8 \begin{pmatrix} 0 \\ B_{eay} \end{pmatrix} \quad (55)$$

$$H_8 = V \Delta^{-1} [I - \exp(-\Delta \gamma)] V^{-1} \quad (56)$$

and  $B_{u12}$ ,  $B_{u22}$  are as given for  $B_u(t)$ .

### 3.3.8 Comments

The modal weighting matrix  $Q_C$  may be fed in directly (flag NPHYS = 0) or it may be set via subroutine QPHYS to control the q-coordinates directly (NPHYS ≠ 0). In the latter case, a diagonal matrix  $Q_W$  is read in, specifying the individual weightings for each of the q-coordinates (physical coordinates). This is converted to the modal weightings by forming  $E^T Q_W E$ , where  $E$  is the modal transformation matrix.

When  $\tilde{Q}$  has been computed, it is output to a separate file on unit 4, so that it can be used by another program.

## 3.4 The Observer Design Program DECOBR

### 3.4.1 Introduction

The program DECOBR is designed to compute the observer matrices  $A_O$ ,  $K_O$  and  $B_O$  for either a coupled or decoupled observer. The program outputs the matrices to a separate file where they can be picked up by the digital simulation program FDCSIM or stability program STABFO.

3.4.2 The Observer Equation

The observer equation is:

$$\dot{\hat{x}} = A_o \hat{x} + k_o y + b_o u$$

where  $A_o = A_c - K_o C_c$ ,  $\hat{x}$  is the state estimate,  $y$  represents the sensor outputs and  $u$  the control input. The DECOBR program allows the user to specify the required eigenvalues of  $A_o$ . If the decoupled observer is required, the eigenvalues are divided into two groups, the rigid eigenvalues and flexible eigenvalues.

The theory for the observer is given in the report SPAR-R.1134.

3.4.3 Assignment of Decoupled Observer Eigenvalues

Since the order of each decoupled observer is small (two), the eigenvalues can be assigned directly.

$$A_{ri} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \quad C_{rii} = \begin{pmatrix} C_{rii} & 0 \end{pmatrix} \quad K_{ri} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

Let  $(A_{ri} - K_{ri} C_{rii})$  have eigenvalues  $\lambda_1$  and  $\lambda_2$

$$(A_{ri} - K_{ri} C_{rii}) = \begin{pmatrix} -\alpha C_{rii} & 1 \\ -\beta C_{rii} & 0 \end{pmatrix}$$

$$\text{Therefore } \lambda_1(-\alpha C_{rii} + \lambda_1) + \beta C_{rii} = 0$$

$$\text{and } \lambda_2(-\alpha C_{rii} + \lambda_2) + \beta C_{rii} = 0$$

Solving these two equations for  $\alpha$  and  $\beta$  gives:

$$\alpha = -(\lambda_1 + \lambda_2)/C_{rii}$$

$$\beta = \lambda_1 \lambda_2 / C_{rii}$$

$\alpha$  and  $\beta$  will both be real if  $\lambda_1$  and  $\lambda_2$  are both real or are a complex conjugate pair.

3.4.4 The Eigenvalue Assignment Routine

The theory behind the eigenvalue assignment routine ASINEV is given in many standard control textbooks, e.g. Linear Multivariable Control: A Geometric Approach by W. Murray Wonham.

Given that the pair  $(C, A)$  is observable (where  $A$  is an  $n \times n$  matrix, and  $C$  is  $p \times n$ ), the routine will find a matrix  $K$  ( $k$  is  $n \times p$ ) such that  $(A - KC)$  has the desired eigenvalues  $\{\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_n\}$ .

The eigenvalues  $\{\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_n\}$  may be a mixture of real values or adjacent complex conjugate pairs.

A pseudo-random number generator is used in the computations. Since  $K$  is not unique if the dimension  $p$  is greater than one, different forms of  $K$  can be produced by changing the start value or limits on the pseudo-random number generator.

Once  $K$  has been calculated, the routine checks the result by forming  $(A - KC)$  and computing the eigenvalues  $\{\lambda_1, \lambda_2, \dots, \lambda_n\}$ . These eigenvalues cannot be directly compared with the desired eigenvalues  $\{\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_n\}$ , since the order will probably be different. A sorting routine would not really help since the eigenvalues are not expected to match precisely, and values close together will obviously cause problems. However, the polynomial coefficients  $\alpha_i$  can be computed where  $f(s - \lambda_i) = \sum \alpha_i s^i + s^n$  and similarly the coefficients  $\alpha_i$  for  $\{\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_n\}$ . These polynomial coefficients are independent of the order of the eigenvalues and unique for a given set of eigenvalues, so an error value can be obtained by summing the absolute difference between the corresponding polynomial coefficients.

3.4.5 The Polynomial Coefficients Routine

The polynomial coefficients routine COEFF calculates the coefficients  $\{\alpha_i\}$  corresponding to the inputs  $\{\lambda_i\}$ , where:

$$\prod_{i=1}^n (s - \lambda_i) = \alpha_0 + \alpha_1 s + \dots + \alpha_{n-1} s^{n-1} + s^n$$

This routine is used by the eigenvalue assignment routine ASINEV and also by the decoupled observer routine DECUPL.

Define the set of coefficients  $\{\beta_{r,i}\}$  by

$$\prod_{i=1}^r (s + \lambda_i) = \beta_{r,0} + \beta_{r,1}s + \dots + \beta_{r,r-1}s^{r-1} + s^r$$

$$\text{Then } \prod_{i=1}^{r+1} (s + \lambda_i) = (s + \lambda_{r+1}) \prod_{i=1}^r (s + \lambda_i)$$

$$= \beta_{r+1,0} + \beta_{r+1,1}s + \dots + \beta_{r+1,r}s^r + s^{r+1}$$

$$= \lambda_{r+1}\beta_{r,0} + (\beta_{r,0} + \lambda_{r+1}\beta_{r,1})s + (\beta_{r,1} + \lambda_{r+1}\beta_{r,2})s^2 + \dots \\ \dots + (\beta_{r,r-1} + \lambda_{r+1}\beta_{r,r})s^{r-1} + (\beta_{r,r-1} + \lambda_{r+1})s^r + s^{r+1}$$

$$\text{Therefore: } \beta_{r+1,0} = \lambda_{r+1}\beta_{r,0}$$

$$\beta_{r+1,i} = (\beta_{r,i-1} + \lambda_{r+1}\beta_{r,i}) \quad \text{for } i=1, \dots, r-1$$

$$\beta_{r+1,r} = (\beta_{r,r-1} + \lambda_{r+1})$$

$$\text{Therefore, } \prod_{i=1}^r (s - \lambda_i) = \alpha_0 + \alpha_1 s + \dots + \alpha_{n-1} s^{n-1} + s^n$$

$$\text{where } \alpha_i = \alpha_{n-i}, \quad i=0, \dots, n-1$$

$$\text{and } \alpha_{r+1,0} = -\lambda_{r+1}\alpha_{r,0}$$

$$\alpha_{r+1,i} = (\alpha_{r,i-1} - \lambda_{r+1}\alpha_{r,i}) \quad \text{for } i=1, \dots, r-1$$

$$\alpha_{r+1,r} = (\alpha_{r,r-1} - \lambda_{r+1})$$

So the set  $\{\alpha_{r+1,i}\}$  can be calculated from the set  $\{\alpha_{r,i}\}$  until the set  $\{\alpha_{n,i}\}$  is reached. The sequence is easily started from the set  $\{\alpha_{1,i}\}$  since  $\alpha_{1,0} = -\lambda_1$ .

Provided the set  $\{\lambda_i\}$  are all real or complex conjugate pairs, the coefficients  $\{\alpha_i\}$  are all real values. To avoid the use of complex arithmetic (the routine COEFF uses double precision), the

complex conjugate pairs are processed together. The routine assumes that when a complex  $\lambda_i$  is detected, the next value  $\lambda_{i+1}$  is the complex conjugate of  $\lambda_i$ .

Suppose we have the complex conjugate pair

$$\lambda_{r+1} = a + ib, \lambda_{r+2} = a - ib \text{ where } a, b \text{ are real.}$$

$$\begin{aligned} \text{Then: } (s - \lambda_{r+1})(s - \lambda_{r+2}) &= s^2 - 2as + (a^2 + b^2) \\ &\equiv s^2 + \gamma s + \mu \end{aligned}$$

$$\text{where: } \gamma \equiv -2a, \mu \equiv a^2 + b^2$$

$$\begin{aligned} \text{Then: } \prod_{i=1}^{r+2} (s - \lambda_i) &= \alpha_{r+2,0} + \alpha_{r+2,1}s + \dots + \alpha_{r+2,r+1}s^{r+1} + s^{r+2} \\ &= (s^2 + \gamma s + \mu) \prod_{i=1}^r (s - \lambda_i) \end{aligned}$$

$$\text{Therefore: } \alpha_{r+2,0} = \mu \alpha_{r,0}$$

$$\alpha_{r+2,1} = \mu \alpha_{r,1} + \gamma \alpha_{r,0}$$

$$\alpha_{r+2,i} = \alpha_{r,i-2} + \gamma \alpha_{r,i-1} + \mu \alpha_{r,i} \text{ for } i=2, \dots, r-1$$

$$\alpha_{r+2,r} = \mu + \gamma \alpha_{r,r-1} + \alpha_{r,r-2}$$

$$\alpha_{r+2,r+1} = \gamma + \alpha_{r,r-1}$$

Using this procedure, the two complex conjugate pairs can be fed in simultaneously and complex arithmetic is avoided.

The start-up of the routine becomes a little more involved to avoid complex arithmetic. The first three values of  $\lambda_i$  are examined,  $\lambda_1, \lambda_2$  and  $\lambda_3$ . If all three are real, or if one is real and two are complex conjugate, then the initial set of three coefficients can be calculated directly. Otherwise, the first four values of  $\lambda_i$  are used, which may be two complex conjugate pairs or two reals followed by one complex conjugate pair.

Once the coefficients  $\{\alpha_i\}$  have been calculated, the routine COEFF checks the result. If  $s$  is set to any  $\lambda_i$  in the polynomial expression, the result should be zero. The polynomial expression is evaluated in the form:

$$\alpha_0 + s(\alpha_1 + s(\alpha_2 + \dots + s(\alpha_{n-2} + s(\alpha_{n-1} + s))) \dots)$$

for each  $\lambda_i$  in turn, and the magnitude of the result accumulated to form the overall error value, which may be printed out. Complex arithmetic is avoided by simulating it with two separate variables.

## 3.4.6

The Kernel Routine

This subroutine actually calculates the transposed kernel of the transposed matrix, i.e. it calculates matrix B such that:

$$A^T B^T = 0 \text{ or } BA = 0$$

where A is a given matrix and neither A or B are zero.

The assumption is made that A has full rank. If A has dimension  $m \times n$ , then  $m > n$  for B to exist, and B has dimension  $(m-n) \times m$ .

Subroutine KERNEL calls up subroutine REDUCE to perform a series of elementary row operations on  $A^T$ , in order to reduce the left-hand side of the matrix to identity. These elementary row operations can be represented by premultiplying  $A^T$  by E:

$$E A^T = (I : A^T)$$

Subroutine REDUCE performs two sorts of row operations to accomplish this: swapping two rows or multiplying row  $i$  by a constant, adding row  $j$  and storing the result in row  $i$ . Some computation can be saved by taking advantage of the fact that most rows will start with a series of zeros,

especially as the computation progresses and these zeros need not be swapped or multiplied. By this method, subroutine REDUCE initially converts the left-hand side to upper triangular form, then to diagonal form, and finally to identity.

Then the matrix B is set to:

$$B = \left( A_1^T : -I \right)$$

since  $A E^T = \begin{pmatrix} I \\ A_1^T \end{pmatrix}$

and  $B A E^T = \left( A_1^T : -I \right) \begin{pmatrix} I \\ A_1^T \end{pmatrix} = 0$

Since  $E^T \neq 0$ ,  $BA = 0$ .

The subroutine automatically checks the result by multiplying B and A and summing the absolute values of the result. This gives an error value, which is printed out.

### 3.4.7

#### Calculation of the Rank of a Matrix

The RANK subroutine computes the row-rank of a matrix (equals the column rank).

The given array is first copied into a work array, since the calculations will destroy the original data.

A series of elementary row-operations is performed on the work array to set the elements on the diagonal non-zero, and elements below the diagonal to zero. If the matrix does not have full rank, the process will terminate prematurely, since it will not be possible to place a non-zero value on the diagonal by this means. The remaining rows may be set to zero. The rank is then the number of non-zero rows in the work array.

Subroutines ROWOP and SWOP perform the elementary row - operations, these subroutines are also used by the KERNEL subroutine. Subroutine ROWOP multiplies row i by a constant, adds row j and stores the result in row k. Subroutine SWOP interchanges two rows. Again, advantage is taken of the leading zeros in any row.

### 3.5 The Controller Stability Program STABF

#### 3.5.1 Introduction

The program STABF calculates the stability of the spacecraft plus controller system. The spacecraft modes are assumed to be fed directly to the controller, the observer is not in the loop. This program differs slightly from the stability analysis performed by the RICQ program, in that elements of  $\hat{C}$  are not zeroed to decouple the controlled modes and residual modes equations. The controller deadband is assumed to be zero, and no bias is applied.

#### 3.5.2 Equations

The spacecraft dynamics may be written in the form:

$$\ddot{\eta}_r = B_{rg} u_g + B_{rt} u_t$$

$$\ddot{\eta}_{ec} + C_e \dot{\eta}_{ec} + C_{eR} \dot{\eta}_{eR} + \tilde{\Omega}_e \eta_{ec} = B_{ec} u_g + B_{ect} u_t$$

$$\ddot{\eta}_{eR} + C_R \dot{\eta}_{eR} + C_{RC} \dot{\eta}_{ec} + \tilde{\Omega}_R \eta_{eR} = B_{eng} u_g + B_{ert} u_t$$

where the r subscript refers to the rigid modes (all controlled), the ec subscript refers to the elastic controlled modes, the eR subscript refers to the elastic residual modes, the g subscript refers to the gimbals, and the t subscript refers to the thrusters.

(see section 3.2.2)

$$\text{Let: } \boldsymbol{x}_{cr}^T = (\eta_{r1} \ \dot{\eta}_{r1} \ \eta_{r2} \ \dot{\eta}_{r2} \ \dots)$$

$$\boldsymbol{x}_{ce}^T = (\eta_{ec}^T \ \dot{\eta}_{ec}^T)$$

$$\boldsymbol{x}_e^T = (\eta_{er}^T \ \dot{\eta}_{er}^T)$$

$$\boldsymbol{x}_e^T = (\boldsymbol{x}_{ce}^T \ \boldsymbol{x}_a^T)$$

$$\boldsymbol{u}^T = (u_g^T \ u_t^T)$$

Then the dynamic equations may be written:

$$\dot{\boldsymbol{x}}_{cr} = \boldsymbol{A}_{cr} \boldsymbol{x}_{cr} + \boldsymbol{B}_{cr} \boldsymbol{u}$$

$$\dot{\boldsymbol{x}}_e = \boldsymbol{A}_e \boldsymbol{x}_e + \boldsymbol{B}_e \boldsymbol{u}$$

$$\text{where } \boldsymbol{B}_{cr} = (B_{crg} \ B_{crt}), \ \boldsymbol{B}_e = (B_{eg} \ B_{et})$$

$$B_{crg} = \begin{pmatrix} 0 \\ B_{rg1} \\ \vdots \\ 0 \\ B_{rgg} \end{pmatrix}, \ B_{crt} = \begin{pmatrix} 0 \\ B_{rt1} \\ \vdots \\ 0 \\ B_{rtg} \end{pmatrix}, \ B_{eg} = \begin{pmatrix} 0 \\ B_{eg} \\ 0 \\ B_{erg} \end{pmatrix}, \ B_{et} = \begin{pmatrix} 0 \\ B_{et} \\ 0 \\ B_{ert} \end{pmatrix}$$

$$\boldsymbol{A}_{cr} = \begin{pmatrix} 0 & I & 0 & 0 \\ 0 & 0 & -C_{cr} & 0 \\ 0 & 0 & 0 & -C_{er} \\ 0 & 0 & -C_{rc} & -B_{cr} \end{pmatrix}, \quad \boldsymbol{A}_e = \begin{pmatrix} 0 & I & 0 & 0 \\ -B_{eg} & -C_{eR} & 0 & -C_{eR} \\ 0 & 0 & 0 & I \\ 0 & -C_{RC} & -B_{et} & -C_R \end{pmatrix}$$

The dynamic equations can now be discretized:

$$\boldsymbol{x}_{cr,b+1} = \bar{\boldsymbol{A}}_{cr} \boldsymbol{x}_{cr,b} + \bar{\boldsymbol{B}}_{cr} \boldsymbol{u}_b$$

$$\boldsymbol{x}_{e,b+1} = \bar{\boldsymbol{A}}_e \boldsymbol{x}_{e,b} + \bar{\boldsymbol{B}}_e \boldsymbol{u}_b$$

$$\text{where } \bar{\boldsymbol{A}}_{cr} = \exp(\boldsymbol{A}_{cr}\gamma) = \begin{pmatrix} 1 & \gamma & 0 & 0 \\ 0 & 1 & -\frac{C_{cr}}{\Delta} & -\gamma \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{C_{rc}}{\Delta} & 1 \end{pmatrix}$$

$$\bar{\boldsymbol{A}}_e = \exp(\boldsymbol{A}_e\gamma) = V \exp(\Delta\gamma)V^{-1}$$

where  $\Delta$  is the diagonal eigenvalue matrix and  $V$  the eigenvector matrix of  $\boldsymbol{A}_e$ .

$$\bar{\boldsymbol{B}}_{cr} = (\bar{\boldsymbol{B}}_{crg} \ \bar{\boldsymbol{B}}_{crt}), \ \boldsymbol{B}_e = (\bar{\boldsymbol{B}}_{eg} \ \bar{\boldsymbol{B}}_{et})$$

$$\begin{aligned}\bar{B}_{crq} &= \exp(A_{cr}\tau) \int_0^\tau \exp(-A_{cr}s) ds \bar{B}_{crq} \\ &= \begin{pmatrix} \gamma & \frac{1}{2}\gamma^2 & & \\ 0 & -\gamma & -\frac{1}{2}\gamma^2 & 0 \\ & & \ddots & \vdots \\ & & 0 & -\gamma \end{pmatrix} \bar{B}_{crq} \\ \bar{B}_{crt} &= \bar{A}_{cr} \bar{B}_{crt}\end{aligned}$$

$$\begin{aligned}\bar{B}_{eg} &= \exp(A_e\tau) \int_0^\tau \exp(-A_e s) ds \bar{B}_{eg} \\ &= V \Delta^{-1} (\exp(\Delta\tau) - I) V^{-1} \bar{B}_{eg}\end{aligned}$$

$$\bar{B}_{et} = \bar{A}_e \bar{B}_{et}$$

The control  $u_h = F x_{ch}$  may be rewritten:

$$u_h = F_r x_{crh} + (F_e \ 0) x_{eh}$$

If there are  $n_{rc}$  rigid modes and  $n_{ec}$  controlled flexible modes, then  $F_r$  consists of columns 1,  $n_{rc}+1$ , 2,  $n_{rc}+2$ , ---  $n_{rc}$ ,  $2n_{rc}$  of  $F$ , and  $F_e$  consists of columns  $2n_{rc}+1$  to  $2(n_{rc}+n_{ec})$  of  $F$ .

Then the closed loop system becomes:

$$\begin{pmatrix} x_{crh+1} \\ x_{eh+1} \end{pmatrix} = \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix} \begin{pmatrix} x_{crh} \\ x_{eh} \end{pmatrix}$$

$$\text{where: } s_{11} = \bar{A}_{cr} + \bar{B}_{cr} F, \quad s_{12} = \bar{B}_{cr} (F_e \ 0)$$

$$s_{21} = \bar{B}_e F_r, \quad s_{22} = \bar{A}_e + \bar{B}_e (F_e \ 0)$$

The system is stable if and only if all the eigenvalues of:

$$\begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix}$$

have magnitude less than one.

3.5.3 Comments

The control matrix  $F$  is read in from a separate data file, as produced by the control design program RICQ. The controlled modes and control period  $\gamma$  are also picked up from the same file.

The damping matrix  $\hat{C}$  may be changed by the flag IDIAG, for test purposes. If IDIAG=1,  $\hat{C}$  is diagonalized by setting all off-diagonal elements to zero. If IDIAG=2 the (off-diagonal) elements corresponding to  $C_{cr}$  and  $C_{Rc}$  are set to zero, to decouple the controlled modes and residual modes equations. The eigenvalues should then be the same as computed by the RICQ program.

The exact methods used to calculate the matrix exponentials, by eigenvector method, are given in a separate chapter since it is common to a number of programs.

The program uses Double Precision arithmetic. Where complex matrices are required, the program uses separate Double Precision variables to represent the real and imaginary parts.

3.6 The Controller Plus Observer Stability Program  
STABFO3.6.1 Introduction

This program calculates the eigenvalues of the stability matrix for the complete system of spacecraft, controller and observer. The state estimates produced by the observer are used to drive the spacecraft controller, with zero deadband and no bias present.

3.6.2 Equations

The spacecraft dynamic equations may be written in the form:

$$\dot{x}_{cr} = A_{cr}x_{cr} + B_{cr}u$$

$$\dot{x}_e = A_e x_e + B_e u$$

as given in the chapter on the STABF program (*section 3.5.2*).

The sensor output Y is given by:

$$Y = \begin{pmatrix} Y_r \\ Y_e \end{pmatrix}, \quad Y_r = C_r x_{cr} + C_{re} x_{ce} + C_{rR} x_R \\ Y_e = C_e x_{ce} + C_{er} x_R$$

where the matrices are given by equations (4-12a), (3-16a) and (3-16b) in the document SPAR-R.1134.

The observer equation is:

$$\dot{\hat{x}} = A_o \hat{x} + B_o u + k_o Y$$

$$\text{where: } \hat{x} = \begin{pmatrix} \hat{x}_{cr} \\ \hat{x}_{ce} \end{pmatrix}, \quad A_o = \begin{pmatrix} A_1 & A_2 \\ 0 & A_4 \end{pmatrix}, \quad B_o = \begin{pmatrix} B_1 \\ B_2 \end{pmatrix}, \quad k_o = \begin{pmatrix} k_1 & 0 \\ 0 & k_2 \end{pmatrix}$$

The observer error is given by:

$$\epsilon_r = \hat{x}_{cr} - x_{cr}, \quad \epsilon_e = \hat{x}_{ce} - x_{ce}$$

$$\begin{pmatrix} \dot{\epsilon}_r \\ \dot{\epsilon}_e \end{pmatrix} = \begin{pmatrix} (A_{cr} - k_r C_r) & -k_r C_{re} \\ 0 & (A_e - k_e C_e) \end{pmatrix} \begin{pmatrix} \epsilon_r \\ \epsilon_e \end{pmatrix} + \begin{pmatrix} k_r C_{rR} \\ k_e C_{eR} - A_{cr} \end{pmatrix} x_R$$

where the above equation is taken from the document SPAR-R.1134 and modified slightly to include the off-diagonal elements of the damping matrix  $\hat{C}$ .

$$A_{ceR} = \begin{pmatrix} 0 & 0 \\ 0 & -C_{cR} \end{pmatrix}$$

where  $C_{cR}$  is given in the chapter on the STABF program, *section 3.5.2*.

The above equation for the observer error is only valid for the decoupled observer.

$$\text{Then: } \dot{\epsilon} = A_\epsilon \epsilon + A_{\epsilon e} x_e$$

$$\text{where } \epsilon = \begin{pmatrix} \epsilon_e \\ \epsilon_r \end{pmatrix}, \quad A_\epsilon = \begin{pmatrix} A_4 & 0 \\ A_2 & A_1 \end{pmatrix}, \quad A_{\epsilon e} = \begin{pmatrix} 0 & k_e C_{eR} - A_{eR} \\ 0 & k_r C_{rR} \end{pmatrix}$$

Combining the observer error equation with the spacecraft flexible dynamics:

$$\dot{x}_e = A_e x_e + B_e u$$

$$\text{where: } x_e = \begin{pmatrix} x_e \\ \epsilon \end{pmatrix}, \quad A_e = \begin{pmatrix} A_e & 0 \\ A_{ee} & A_e \end{pmatrix}, \quad B_e = \begin{pmatrix} B_e \\ 0 \end{pmatrix}$$

The equations can now be discretized as:

$$x_{cr,b+1} = \bar{A}_{cr} x_{cr,b} + \bar{B}_{cr} u_b$$

$$x_{\epsilon,b+1} = \bar{A}_\epsilon x_{\epsilon,b} + \bar{B}_\epsilon u_b$$

where  $\bar{A}_{cr}$ ,  $\bar{B}_{cr}$  as given in the STABF section,

$$\bar{B}_{\epsilon g} = \exp(A_\epsilon \tau) \int_0^\tau \exp(-A_\epsilon s) ds \begin{pmatrix} B_{eg} \\ 0 \end{pmatrix}, \quad \bar{B}_{\epsilon t} = \bar{A}_\epsilon \begin{pmatrix} B_{et} \\ 0 \end{pmatrix}$$

$$\bar{A}_\epsilon = \exp(A_\epsilon \tau), \quad \bar{B}_\epsilon = \begin{pmatrix} \bar{B}_{\epsilon g} & \bar{B}_{\epsilon t} \end{pmatrix}$$

The feedback control is given by:

$$\begin{aligned} u_b &= F \hat{x}_{ch} \\ &= F_r \hat{x}_{cr,b} + (F_e \ 0) \hat{x}_{\epsilon,b} \\ &= F_r \hat{x}_{cr,b} + F_e x_{\epsilon,b} \end{aligned}$$

$$\text{where } F_\epsilon = (F_e \ 0 \ F_e \ F_r)$$

and  $F_r$ ,  $F_\epsilon$  as specified in the STABF chapter.

$$\text{Then: } \begin{pmatrix} x_{cr,b+1} \\ x_{\epsilon,b+1} \end{pmatrix} = \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix} \begin{pmatrix} x_{cr,b} \\ x_{\epsilon,b} \end{pmatrix}$$

$$\text{where } s_{11} = \bar{A}_{cr} + \bar{B}_{cr} F_r, \quad s_{12} = \bar{B}_{cr} F_\epsilon$$

$$s_{21} = \bar{B}_\epsilon F_r, \quad s_{22} = \bar{A}_\epsilon + \bar{B}_\epsilon F_\epsilon$$

The closed loop system is stable if and only if all the eigenvalues of  $\begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix}$  have magnitude less than one.

## 3.6.3

Comments

The control matrix  $F$ , controlled modes and control period  $T$  are read in from a separate data file, produced by the RICQ program. The residual modes are specified by the main data file, so that the stability can be compared when different residual modes are present.

The observer matrices  $A_o$ ,  $B_o$ ,  $K_o$  are read in from another data file, produced by the DECOBS program. Only the decoupled observer can be used with this program.

If the flag ITEST is set to one, the observer error is artificially set to zero and the eigenvalues of the stability matrix should be the same as the eigenvalues of the STABF program, plus zero eigenvalues.

## 3.7

Common Routines

Some routines are common to a number of programs and are stored as a separate module to be picked up at compile-time. These are described below. Also, the method for calculating general matrix exponentials is given since it is used in a number of different programs, although it is not a separate module.

## 3.7.1

Eigenvalue Module

The EIGENP subroutine finds the eigenvalues and eigenvectors of a given real matrix. The eigenvalues are found by the QR double-step method and the eigenvectors are computed by inverse iteration. In general, the eigenvalues and eigenvectors are complex, and the real and imaginary parts are computed in separate real arrays. Single Precision arithmetic is used, as it appears to be adequate for most cases. The subroutine destroys the original matrix.

The programs which use EIGENP call it via another subroutine, for example EV1. This subroutine copies the original Double Precision array containing the matrix into a Single Precision array required by EIGENP. When EIGENP returns with the eigenvalues and eigenvectors, EV1 may check the results, if required by the option flag IC. If  $\lambda_i$  is an eigenvalue and  $x_i$  an eigenvector of the given matrix A, then the program computes the Euclidian norm of  $(Ax_i - \lambda_i x_i)$ . This value should ideally be zero, and so represents an error value for the ith eigenvalue and eigenvector. The program may optionally print the eigenvalues (real and imaginary parts and magnitude and phase) and the corresponding error value.

## 3.7.2

Matrix Inversion Modules

There are two matrix inversion modules; one for inverting a real matrix (MTINV2) and one for inverting a complex matrix (MINVDC). Both modules use the standard Gauss-Jordan method and produce the determinant as a byproduct.

Both subroutines use Double Precision arithmetic, as Single Precision was found to be inadequate for most cases. The complex variables required by MINVDC are simulated by two Double Precision variables, one each for the real and imaginary parts. Normal computer complex variables consist of Single Precision real and imaginary parts and so are not accurate enough.

While experimenting with the complex inversion module, the identity  $(P + iQ) = (P + QP^{-1}Q)^{-1} - i(Q + PQ^{-1}P)^{-1}$  was considered. If the matrix to be inverted has real and imaginary parts P and Q, then the inverse could be calculated using the real inversion module. However, if it is being used for the inversion of an eigenvector matrix, then the inverses of P and Q will not exist, since the complex eigenvectors occur in complex conjugate pairs for real matrices.

The programs which use the inversion modules call them via another subroutine, for example INVERT for real matrices and COMINV for complex matrices. The INVERT (or COMINV) subroutine first copies the given matrix into a work array, since the inversion module will overwrite it with the inverse. The data may also need to be shuffled around, since the inversion module assumes the array elements are stored continuously in a one-dimensional array, or exactly contained in a two-dimensional array.

When the inverse has been calculated, the program has the option of checking the result (option flag IC). If A is the given matrix and B is the calculated inverse, the program calculates  $(AB-I)$  and forms the sum of the absolute values of the elements. This number gives an indication of the accuracy of the inversion. The subroutine prints the error value and the determinant, and also halts if the determinant is zero.

### 3.7.3

#### Matrix Exponentials

The eigenvalue method is used to calculate the exponential of a general matrix:

$$\exp(At) = V \exp(\Delta t)V^{-1}$$

where V is the (complex) matrix of eigenvectors and  $\Delta$  is the diagonal (complex) matrix of eigenvalues. The eigenvalues are assumed to be distinct.

The same method is used for similar problems, e.g.

$$H_1 = \int_0^r \exp(At) dt = V D^{-1} (\exp(Ar) - I) V^{-1}$$

used by the CON52 program.

In each case, the form is the same:  $VDV^{-1}$  where D is a diagonal matrix which can easily be calculated.

In general, V and D are complex matrices, but the product  $VDV^{-1}$  is expected to be real since A and t are both real. Let  $R + iC = VDV^{-1}$  where R and C are real matrices. Ideally  $C=0$ , but since some accuracy will be lost in the calculations, C will be non-zero.

If r and c are the sums of the absolute values of the elements of R and C respectively, then c should be very small compared to r. These two values therefore give some indication of the accuracy of the calculations.

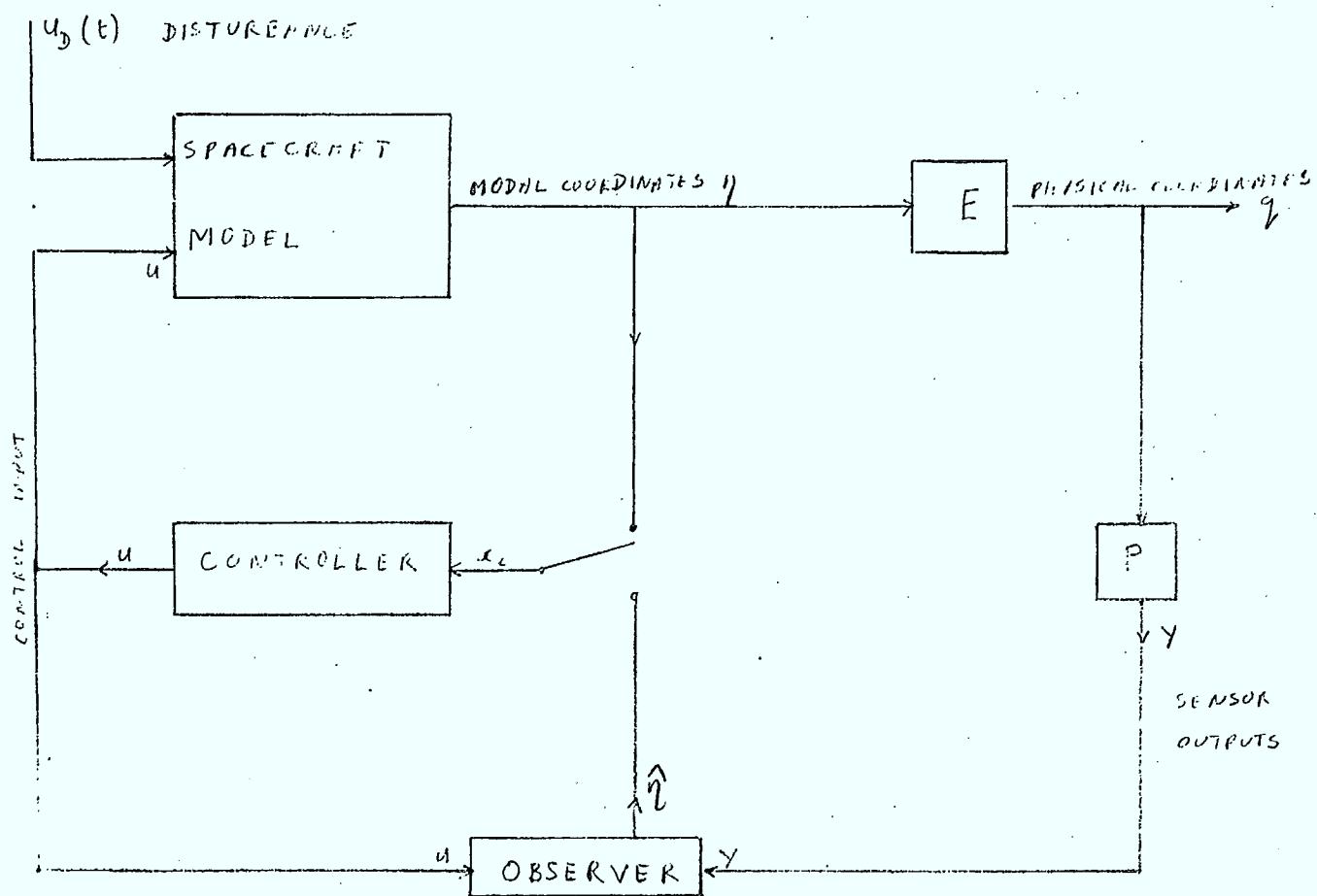


Fig. 3-1: SPACECRAFT - CONTROLLER - OBSERVER SYSTEM

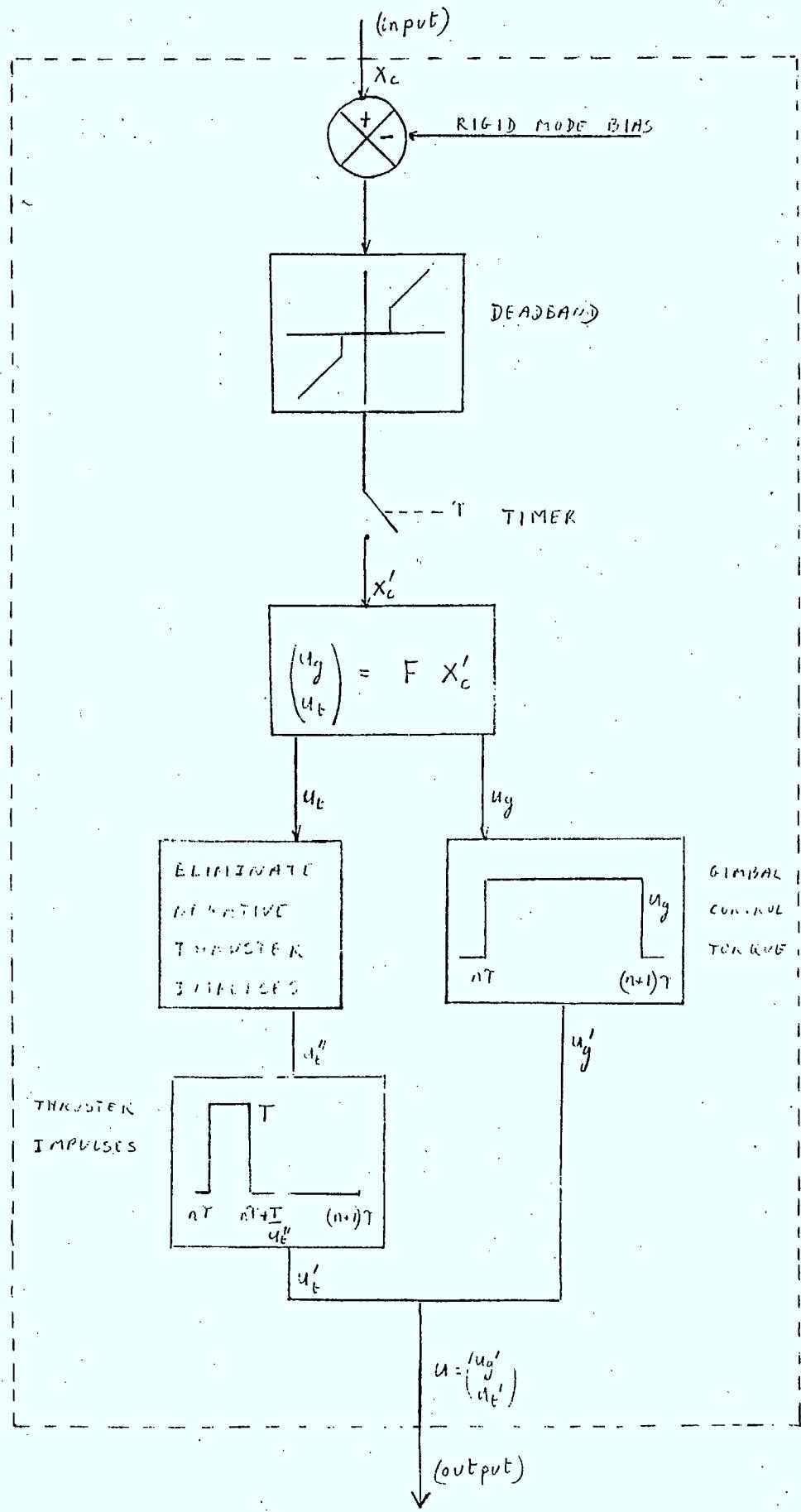


Fig 3-2: THE CONTROLLER  
3-43

TABLE 3-1  
PROGRAM PARAMETERS SET VIA ARRAY VC

VC Index	Parameter	Default Value	Comments
1	TSTART	0.0	Simulation start time, sec.
2	TEND	0.0	Simulation end time, sec.
3	DT	0.01	Simulation step size, sec.
4	IPRT	1	Controls initial output from simulation, 0 = reduced output, non-zero = full output
5	NPSTEP	10	Controls simulation run output. Output occurs once every NPSTEP step, or NPSTEP*DT sec.
6	-	0	Not used
7	NVC	150	Used portion of array VC.
8	INT	0	Determines integration method, 0 = Euler, non-zero = 4th order Runge-Kutta
9	IDBUG	0	Debugging flag. 0 = no debugging output, non-zero values cause debugging output.
10	IVCL	90	Start position in array VC of initial values of state vector V.
11	NDI	12	Size of disturbance vector UD.
12	TP	10.0	Control interval, sec.

8/4/mcla702/3

SPAR-R.1135  
ISSUE A

TABLE 3-1 - (Continued)

## PROGRAM PARAMETERS SET VIA ARRAY VC

VC Index	Parameter	Default Value	Comments
13	QBC	4.51E-6	Solar energy constant Q/C, N/m <sup>2</sup> .
14	GAMMA0	0.0	Initial orbital position $\alpha$ , radians.
15	W0	7.2722E-5	Orbital rate, rad/sec.
16	IDIST	0	Switch on or off disturbance input 0 = on, non-zero = off.
17	UMULT(1)	1.0	Multiplies the i <sup>th</sup> component of the control input vector u by UMULT(i).
26	UMULT(10)	1.0	
27	DEAD(1)	-1.0	Deadband level for each of the 7 controlled rigid modes (a negative level is always exceeded, of course).
33	DEAD (7)	-1.0	
34	REFTIM	0.0	Time at which bias level EREF is switched on, sec.
35	EREF(1)	0.0	Bias values for controlled rigid modes.
41	EREF(7)	0.0	
42	NEG	0	A non-zero value of NEG switches off the conversion of negative thruster pulses, for test purposes.

TABLE 3-1 - (Continued)  
PROGRAM PARAMETERS SET VIA ARRAY VC

VC Index	Parameter	Default Value	Comments
43	IDIAG	0	If IDIAG = 0, the full c matrix is used, as read in. If IDIAG = 1, the off-diagonal elements of c are zeroed. If IDIAG is not 0 or 1, the cross-elements of c, relating controlled modes to residual modes, are zeroed.
44	KILL	0	Number of flexible modes to be 'killed' i.e. artificially held to their initial values.
45	KKILL(1)	0	The first KILL elements of KKILL specify the mode numbers of those flexible elements to be 'killed'.
55	KKILL(11)	0	
56	TSTIME	1.0E6	If simulation times exceeds TSTIME, the debugging flag IDEBUG is set to 1, for test purposes.
57	NCLOSE	0	If NCLOSE = 0, the spacecraft state vector is fed back to the controller. If NCLOSE nonzero, the observer estimates are fed back to the controller.
58	--	-	Not used.
IVC1-1	--	-	
IVC1	V(1)	0.0	Initial values of state vector.
IVC1 +NV-1	V(NV)	0.0	

## 4.0

RESULTS

## 4.1

Comparison of Numerical Simulation and Analytic Stability Programs

A set of computer runs were performed with the numerical simulation program (FDCSIM) and the analytic stability program (STABFO) to check whether the two programs agreed on the stability of the system. Different control matrices ( $F$ ) and different observer matrices were used and the residual modes present in the system were varied.

Since the modal outputs from the numerical simulation are oscillatory, it is not always easy to see whether the outputs are converging or diverging or to compare the relative stability of two runs. To try to overcome this, the Euclidian norm of the state vector has been calculated (square root of the sum of the squares of the modal coordinates and their derivatives). The variable  $SX(1)$  is the norm of just the rigid modes (and their first derivatives),  $SX(2)$  is the norm of just the controlled flexible modes and derivatives,  $SX(3)$  is the norm of just the residual flexible modes and derivatives, and  $SX(4)$  is the norm of all the modes and derivatives.

The numerical simulation was initialized with each rigid mode set to 0.1 to excite the system, and the flexible modes set to zero. All rates were also set to zero initially. The  $SX$  variables were therefore initially at  $SX(1) = 0.26$ ,  $SX(2) = 0.0$ ,  $SX(3) = 0.0$ ,  $SX(4) = 0.26$ . Table 4-1 gives the values of  $SX$  at the end of each run (100 seconds each). If the value of  $SX(4)$  at the end of the run is significantly greater than the initial value (0.26), then the system must be unstable. If it is less than the initial value, the system is stable. Also, by comparing  $SX(1)$ ,  $SX(2)$ ,  $SX(3)$  with their initial values, the stability of just the rigid modes, the controlled flexible modes or the residual flexible modes can be inferred.

The STABFO program computes the eigenvalues of the complete system stability matrix. If the magnitudes of the eigenvalues are all less than unity, then the system is stable, otherwise it is unstable. Table 4-1 lists the magnitude of the maximum eigenvalue (EV1) and the second maximum (EV2). If EV1 = EV2, then the two eigenvalues are complex conjugates.

Two different controllers were used: RICMATE and RICQF. These are described in the Appendix A-2 on controller design. The two observers used were the DECMATS and DECMAT observers, described in the Appendix A-3. Only run 5 used the DECMAT observer, and run 1 used no observer (direct feedback of S/C state vector, which is not possible in real-life). The controller/observer system is designed to control/observe seven rigid modes (1, 2, 4, 5, 6, 7, 8) and four flexible modes (1, 3, 4, 6). The remaining flexible modes (residual modes) may or may not be included in the overall system, as desired.

Table 4-1 shows that the two programs (FDCSIM and STABFO) agree on whether the system is stable or unstable. Also, the table shows that the relative degree of stability of each run is consistent.

The table also shows that the RICMATE controller is stable, with all residual modes present but no observer (direct feedback). The system is also stable with an observer provided no residual modes are present. When even one residual mode is included (mode 11, highest frequency), the system becomes unstable. Run 4 shows the system is also unstable when just the suppressed residual modes are present (these are not suppressed in the rigid mode observers).

## 4.2

Effect of Changing the Control Period

The STABFO program was used to compute the stability of the spacecraft-controller-observer system for various values of the control period.

The full spacecraft model of 7 (controlled) rigid modes and 11 flexible modes was used. The controller/observer system was designed to control/observe all seven rigid modes and four (1, 3, 4, 6) flexible modes.

The control period was set so that the sampling frequency was greater than twice the greatest natural controlled/observed frequency i.e.

$$\omega_s > 2\omega_n$$

The control period was therefore less than  $\pi/\omega_n$ . The frequency of mode 6 is 0.7796 rad/sec, therefore the control period was set less than 4.03 seconds. The values 4.0, 3.0, 2.0, 1.0 seconds were used.

The design procedure for the controller followed the method for the RICQF controller, the only difference being the value of the control period. The DYNACON modal weighting matrix was used by the CON52 program to set up the  $\tilde{Q}$  matrix. The  $\tilde{Q}$  matrix was used as the modal weighting matrix for the RICQ program to generate the control matrix  $F$ . The two other weighting matrices  $R$  and  $R_R$  were each set to diagonal  $\{10^{-6} \dots\}$ . A separate  $F$  matrix was designed for each control period.

Two observers were used; DECMATS and DECMAT. The observers do not depend on the control period, of course. These two observers are described in the Appendix A-3.

Table 4-2 shows that the system is still unstable for these values of control period and that the DECMATS observer gives better results than the DECMAT observer.

## 4.3

Compensator with Fewer Controlled Modes

Due to the large cost of computing the stability of the spacecraft-controller-observer system with the STABFO program (about \$40 per run), a series

of runs were performed with just one controlled rigid mode and one controlled flexible mode. The remaining rigid modes were deleted from the system, but the residual flexible modes present were varied.

The object of these runs was to see if any trends were apparent when certain parameters were varied, for example, whether the stability improved when the observer damping factor ( $\zeta$ ) was changed.

The controller used was based on the design procedure for the RICQF controller. The CON52 program was used to generate the  $\bar{Q}$  matrix, which was used as the modal weighting matrix for the RICQ program. The R weighting matrix was set to diag  $\{10^{-6}, \dots\}$  and  $R_R$  was set to zero.

Table 4-3 shows the results for the system with rigid mode 1 controlled, flexible mode 1 controlled, and one residual mode present (2 or 7). The control period was set to 10 or 20 seconds, and the observer was designed with a frequency  $\omega$  and damping factor  $\zeta$ , as specified. No residual modes were suppressed. The magnitude of the maximum eigenvalue of the stability matrix is given, computed by the STABFO program.

Table 4-4 shows the results for the system with rigid mode 1 controlled, flexible mode 1 controlled and all residual modes present (2 to 11). The control period was set to 10 or 20 seconds, and the observer frequency  $\omega$  damping factor  $\zeta$  were varied. The observer was designed to suppress residual modes 3 and 4, or no modes.

Unfortunately, neither Table 4-3 or Table 4-4 shows a constant trend (in improving stability) when the observer parameters are varied.

8/4/mcl702/54

SPAR-R.1135  
ISSUE A

4.4      Stability Analysis of Controller Plus Observer

Tables 4-5, 4-6, 4-7, 4-8 list the eigenvalues of the system for the DECMATS observer (decoupled observer, suppresses residual modes 2, 5, 7) and the RICMATT or RICQF controller, with either no residual modes or all residual modes present. The magnitude and phase of the eigenvalue is listed and an index specifying the order according to magnitude. The run number corresponds to the run number in Table 4-1.

8/4/mcla702/8

SPAR-R.1135  
ISSUE A

TABLE 4-1  
COMPARISON OF RESULTS OF FDCSIM AND STABFO PROGRAMS

Type of Run			FDCSIM					STABFO		Run Number
Controller	Observer	Residual Modes Present	SX(1)	SX(2)	SX(3)	SX(4)	EV1	EV2	Stability	
RICMATF	None	All	0.019	6.2E-3	0.027	0.034	0.9915	0.9478	Stable	1
RICMATF	DECMATS	All	24.	31.	18.	43.	2.077	1.314	Unstable	2
RICMATF	DECMATS	None	9.0E-3	7.4E-3	0.0	0.012	0.9326	0.9326	Stable	3
RICMATF	DECMATS	2,5,7	0.65	0.39	1.00	1.25	1.359	1.359	Unstable	4
RICMATF	DECMAT	2,5,7	2.4E10	1.9E10	2.1E10	3.8E10	27.11	1.226	Unstable	5
RICMATF	DECMATS	11	1.219	3.512	1.1E-3	3.718	1.508	1.113	Unstable	6
RICMATF	DECMATS	10,11	—	—	—	—	1.907	1.253	Unstable	7
RICQF	DECMATS	None	1.1E-2	4.7E-3	0.0	0.012	0.8972	0.7481	Stable	8
RICQF	DECMATS	All	—	—	—	—	1.789	1.485	Unstable	9

8/4/mcla702/5.

SPAR-R.1135  
ISSUE A

TABLE 4-2  
EFFECT OF CHANGING THE CONTROL PERIOD

Observer	Control Period (sec.)	Maximum Eigenvalue (magnitude)
DECMAT no suppression	4.0	11.75
	3.0	6.182
	2.0	3.547
	1.0	2.465
DECMATS (suppression modes 2,5,7)	4.0	1.517
	3.0	1.387
	2.0	1.999
	1.0	2.082
No observer (direct feedback)	4.0	0.9967
	3.0	0.9976
	2.0	0.9984
	1.0	0.9992

8/4/mcla702/6

SPAR-R.1135  
ISSUE A

TABLE 4-3

RESULTS WITH ONE RIGID MODE, ONE CONTROLLED FLEXIBLE MODE  
AND ONE RESIDUAL MODE

Residual Mode Present	Control Period (sec.)	$\omega$ rad/sec.	$\zeta$	Maximum Eigenvalue (magnitude)
2	10	0.1	0.1	1.123
		0.3	0.1	0.9470
		0.5	0.1	0.9801
		0.1	0.9	1.020
		0.3	0.9	0.9537
		0.5	0.9	0.9615
2	20	0.1	0.1	1.263
		0.3	0.1	0.9371
		0.5	0.1	0.9490
		0.1	0.9	1.030
		0.3	0.9	0.9830
		0.5	0.9	0.9658
7	10	0.1	0.1	1.003
		0.3	0.1	1.126
		0.5	0.1	1.110
		0.1	0.9	0.9286
		0.3	0.9	0.8604
		0.5	0.9	0.8973
7	20	0.1	0.1	1.112
		0.3	0.1	0.8260
		0.5	0.1	0.8829
		0.1	0.9	0.5853
		0.3	0.9	0.4660
		0.5	0.9	0.4797

8/4/mcla702/7

SPAR-R.1135  
ISSUE A

TABLE 4-4

RESULTS WITH ONE RIGID MODE, ONE CONTROLLED FLEXIBLE MODE  
AND ALL RESIDUAL MODES

Control Period (sec.)	Suppressed Modes	$\omega$ rad/sec.	$\zeta$	Maximum Eigenvalue (magnitude)
10	3, 4	0.1	0.1	1.659
	3, 4	0.3	0.1	65.17
	3, 4	0.5	0.1	35.92
	3, 4	0.1	0.9	4.659
	3, 4	0.3	0.9	15.08
	3, 4	0.5	0.9	13.61
	(no observer)			0.9915
20	3, 4	0.5	0.1	132.7
	3, 4	0.1	0.9	5.982
	None	0.5	0.1	4.522
	None	0.1	0.9	6.004
	(no observer)			0.9831

STABILITY ANALYSIS

PERFORMED BY STABFD  
PROGRAM

RICMATE CONTROLLER

DECMATS OBSERVER

ALL RESIDUAL MODES

PRESENT

(RUN 2)

TABLE 4-5

STABILITY ANALYSIS

OF RUN 2

ORDER	MAGNITUDE	PHASE (D)
1	.2077E+01	.1800E+03
2	.1314E+01	.6711E+02
4	.1314E+01	.6711E+02
6	.1000E+01	.1024E+03
7	.1000E+01	.1024E+03
8	.9032E+00	.1800E+03
9	.8955E+00	.1483E+03
10	.8955E+00	.1483E+03
12	.1243E+01	.2416E+02
13	.1243E+01	.2416E+02
15	.9367E+00	.7212E+02
16	.9367E+00	.7212E+02
17	.8159E+00	.8254E+02
18	.8159E+00	.8254E+02
20	.6016E+00	.1169E+03
21	.6016E+00	.1169E+03
22	.6951E+00	.8102E+02
23	.6951E+00	.8102E+02
24	.9200E+00	.3818E+02
25	.9200E+00	.3818E+02
26	.7507E+00	.5430E+02
27	.7507E+00	.5430E+02
28	.4651E+00	.1800E+03
29	.4572E+00	.1190E+03
30	.4572E+00	.1190E+03
31	.8372E+00	.3650E+01
32	.8372E+00	.3650E+01
33	.6372E+00	.3650E+01
34	.7910E+00	.1495E+02
35	.7910E+00	.1495E+02
36	.7755E+00	0.
37	.7098E+00	.1994E+02
38	.7098E+00	.1994E+02
39	.6676E+00	.2352E+02
40	.6676E+00	.2352E+02
41	.6626E+00	0.
42	.6470E+00	0.
43	.6000E+00	.4259E+02
44	.4000E+00	.4259E+02
45	.3612E+00	0.
46	.2301E+00	.9337E+02
47	.2301E+00	.9337E+02
48	.1322E+00	0.
49	.1535E+00	0.
50	.7511E+01	.3354E+02
51	.7511E+01	.3354E+02
52	.6347E+01	0.
53	.6401E+01	0.
54	.2529E+01	.1800E+03
55	.1765E+01	.5705E+02
56	.1765E+01	.5705E+02
57	.5406E+02	0.
58	.1001E+01	0.
59	.4036E+02	.1892E+02
60	.4036E+02	.1892E+02
61	.4067E+04	0.
62	.3319E+04	.1800E+03
63	.1064E+05	.1800E+03

## STABILITY ANALYSIS

PERFORMED BY STABFO

PROGRAM

RICMATE CONTROLLER

DECIMATS OBSERVER

NO RESIDUAL MODES

PRESENT

(RUN 3)

TABLE 4-6

## STABILITY ANALYSIS

OF RUN 3

ORDER	MAGNITUDE	PHASE (D)
4	.707AE+00	.1353E+03
	.7078E+00	-.1353E+03
1	.9326E+00	.7393E+02
	.9326E+00	-.7393E+02
9	.3782E+00	.1800E+03
10	.3367E+00	.1800E+03
	.2662E+00	.8873E+02
11	.2662E+00	-.8873E+02
	.8003E+00	.4369E+02
2	.8003E+00	-.4369E+02
	.7350E+00	.1668E+02
3	.7350E+00	-.1668E+02
	.6896E+00	.8343E+01
5	.6896E+00	-.8343E+01
	.5957E+00	.2565E+02
8	.5957E+00	-.2565E+02
	.6430E+00	.2305E+02
7	.6430E+00	-.2305E+02
	.6730E+00	.2189E+02
6	.6730E+00	-.2189E+02
	.7016E-01	.1800E+03
13	.5826E-01	.1800E+03
	.7334E-01	.7302E+02
24	.7334E-01	-.7302E+02
	.6975E-01	.7974E+02
12	.6975E-01	-.7974E+02
	.6473E-01	.6970E+02
22	.6473E-01	-.6970E+02
	.6161E-01	.7730E+02
23	.6161E-01	-.7730E+02
	.6796E-01	.7565E+02
15	.6796E-01	-.7565E+02
	.6763E-01	.7411E+02
16	.6763E-01	-.7411E+02
	.6625E-01	.7512E+02
21	.6625E-01	-.7512E+02
	.6705E-01	.7484E+02
20	.6705E-01	-.7484E+02
	.6726E-01	.7499E+02
17	.6726E-01	-.7499E+02
	.6710E-01	.7494E+02
19	.6710E-01	-.7494E+02
	.6721E-01	.7494E+02
19	.6721E-01	-.7494E+02

	ORDER	MAGNITUDE	PHASE (D)
STABILITY ANALYSIS	1	.8972E+00	.7120E+02
PERFORMED BY STABFO		.8972E+00	.7120E+02
PROGRAM	2	.7481E+00	.2067E+02
		.7481E+00	.2067E+02
RICQE CONTROLLER	7	.3363E+00	.1607E+03
DEC MATS. OBSERVER		.3363E+00	.1607E+03
N.O. RESIDUAL MODES	3	.6260E+00	.3150E+02
PRES ENT		.6260E+00	.3150E+02
(RUN. 8)	4	.5568E+00	.2792E+02
		.5568E+00	.2792E+02
	5	.4578E+00	.4495E+02
		.4578E+00	.4495E+02
	8	.2805E+00	.2047E+02
		.2805E+00	.2047E+02
	6	.4348E+00	.2889E+02
		.4348E+00	.2889E+02
	9	.1220E+00	.2657E+02
		.1220E+00	.2657E+02
	23	.1604E+01	0.
	22	.9262E+01	0.
	24	.1518E+01	0.
	21	.3776E+01	0.
	10	.7333E+01	.7303E+02
		.7333E+01	.7303E+02
	11	.6975E+01	.7974E+02
		.6975E+01	.7974E+02
	19	.6474E+01	.6970E+02
		.6474E+01	.6970E+02
	20	.6160E+01	.7731E+02
		.6160E+01	.7731E+02
	13	.6758E+01	.7500E+02
		.6758E+01	.7500E+02
	12	.6796E+01	.7429E+02
		.6796E+01	.7429E+02
	18	.6622E+01	.7473E+02
		.6622E+01	.7473E+02
	17	.6705E+01	.7404E+02
		.6705E+01	.7404E+02
	16	.6718E+01	.7502E+02
		.6718E+01	.7502E+02
	14	.6726E+01	.7500E+02
		.6726E+01	.7500E+02
	15	.6721E+01	.7494E+02
		.6721E+01	.7494E+02

STABILITY ANALYSIS

PERFORMED BY STABFO

PROGRAM.

RICQF CONTROLLER,

DECMATS OBSERVGR

ALL RESIDUAL MODES

PRESENT

(RUN 9)

TABLE 4-8

STABILITY ANALYSIS

OF RUN 9

ORDER	MAGNITUDE	PHASE (D)
1	.1769E+01	.1800E+03
2	.1485E+01	.6519E+02
3	.1485E+01	.6519E+02
4	.1253E+01	.2661E+02
7	.1253E+01	.2661E+02
4	.9709E+00	.1037E+03
7	.9709E+00	.1037E+03
5	.9045E+00	.1431E+03
6	.9045E+00	.1431E+03
5	.9465E+00	.6840E+02
6	.9465E+00	.6840E+02
6	.9395E+00	.3843E+02
7	.9395E+00	.3843E+02
9	.8364E+00	.8235E+02
10	.8364E+00	.8235E+02
11	.7632E+00	.8832E+02
12	.7632E+00	.8832E+02
14	.6140E+00	.1696E+03
15	.6140E+00	.1696E+03
10	.8330E+00	.2094E+02
11	.8330E+00	.2094E+02
8	.8528E+00	.5067E+01
16	.5505E+00	.1227E+03
17	.5505E+00	.1227E+03
20	.4644E+00	.1800E+03
12	.7463E+00	0.
17	.5443E+00	.6361E+02
13	.5443E+00	.6361E+02
22	.3856E+00	.1242E+03
15	.3856E+00	.1242E+03
18	.5984E+00	0.
18	.5115E+00	.2761E+02
19	.5115E+00	.2761E+02
19	.4704E+00	.3141E+02
20	.4704E+00	.3141E+02
21	.3967E+00	.3218E+02
22	.3967E+00	.3218E+02
24	.2688E+00	.6699E+02
23	.2688E+00	.6699E+02
25	.3255E+00	0.
26	.1575E+00	.3549E+01
27	.1575E+00	.3549E+01
28	.5806E-01	0.
27	.8708E-01	0.
31	.2318E-01	.1800E+03
29	.3815E-01	0.
30	.2458E-01	0.
34	.4940E-02	.7245E+02
32	.4940E-02	.7245E+02
32	.8299E-02	0.
35	.2954E-02	0.
33	.5404E-02	0.
36	.2060E-04	0.
37	.2035E-04	.1800E+03
38	.1002E-06	.1800E+03

## 5.0

CONCLUSION

The stability analysis program and the numerical simulation program both agree on whether the spacecraft-controller-observer system is stable or not, which should give some confidence in the results of these programs.

The controller was designed to be driven by the true spacecraft modes directly and this system is stable even when residual modes are present. Of course, the spacecraft modes are not directly available in practice.

The full spacecraft-controller-observer system is stable when the spacecraft residual modes are artificially eliminated i.e. when the 'design model' is used. In practice, residual modes will be present, and these cause the system to become unstable i.e. when the 'evaluation model' is used.

The controller used was a previous version to the one given in the report SPAR-R.1134. Implementing this controller might help to overcome the instability.

The selection of the controlled/observed modes and the modal weighting matrix  $\tilde{Q}$  were based on the results of the DYNACON study, which was intended for attitude control only. The control scheme presented here is used for stationkeeping, and attempts to control both the attitude and the N-S and E-W position errors. Therefore the stability of the system should be improved if the controlled/observed modes and the  $\tilde{Q}$  matrix are recomputed to account for stationkeeping control.

The control period  $\tau$  was only varied over a small range, from 1 sec to 10 secs. This parameter could obviously be varied over a much larger range (perhaps going as low as 1 millisecond) to see how the stability changes.

8/4/mcl702/56

SPAR-R.1135  
ISSUE A

The observer eigenvalues could also be changed (perhaps they should not all have been set equal to the same value). Alternatively, an optimal observer could be used.

Further simulations would obviously help to eliminate some of these areas of concern and additional work is therefore warranted. However, financial limitations preclude any further activity at this time.

8/4/mcl702/57

SPAR-R.1135  
ISSUE A  
APPENDIX A

APPENDIX A

DESIGN DATA AND RESULTS

8/4/mcl702/58.

SPAR-R.1135  
ISSUE A  
APPENDIX A

## APPENDIX A

### DESIGN DATA AND RESULTS

#### 1.0 THE MODAL WEIGHTING MATRIX

The modal weighting matrix  $Q_C$  is based on the modal output weighting matrix  $\hat{Q}$  given in the DYNACON memorandum of August 25, 1982. Two values have been added on the diagonal in columns/rows 1 and 2, to give some weighting to the translational motion in the X and Y directions. A value of  $10^{-6}$  was chosen, since many other elements in the matrix are of this magnitude. The third diagonal element was left at zero, as this corresponds to the Z (radial) motion, which is uncontrollable.

The matrix has also been partitioned and expanded with zero blocks, since the required  $Q_C$  matrix also weights the state vector rates. These rate weightings are set to zero. See Table A-1.

#### 2.0 THE CONTROLLERS RICMATE AND RICQF

RICMATE and RICQF are actually the names of the data files containing different versions of the feedback control matrix F. These data files were created by the RICQ program and may be used directly by other programs to pick up the F matrix.

In both cases, seven rigid modes were to be controlled (1, 2, 4, 5, 6, 7, 8) and four flexible modes (1, 3, 4, 6). The control period was 10 seconds.

##### (a) RICMATE

The RICMATE controller was generated directly by the RICQ program. The modal weighting matrices ( $Q_C$  and  $Q_{CK}$ ) were set to the values given by DYNACON, modified to include

8/4/mcl702/59

SPAR-R.1135  
ISSUE A  
APPENDIX A

the 3 translational modes - see Appendix A section 1.0. The weighting matrices R and  $R_R$  were set to  $10^{-6}I$ , where I is the identity matrix.

The sequence of iterations was found to converge adequately with the index K set to 25. The initial difference in the F matrix (sum of the absolute difference in elements of F<sub>25</sub> and F<sub>24</sub>) was 143.12, and the P matrix difference was 7.83E-3. The final F difference was 0.1005 and the final P difference was 8.82E-6.

The F matrix and corresponding stability analysis is given in Table A-2 and Table A-3 respectively.

(b) RICOF

The DYNACON modal weighting matrix (as used for the RICMATF controller) was used as the Q<sub>c</sub> matrix for the CON52 program, which computed the corresponding  $\bar{Q}$  matrix. This  $\bar{Q}$  was then used as the Q<sub>c</sub> and Q<sub>cK</sub> matrices by the RICO program to produce the F matrix. The weighting matrix R was set to  $10^{-6}I$ , and the control spillover matrix  $R_R$  was set to zero.

The iteration control index was again set to 25. The initial F matrix difference was 376.83, and the P difference was 0.207. The final F difference was 0.0994, and final P difference was 2.91E-5.

The  $\bar{Q}$  matrix, F matrix and stability analysis is given in Table A-4, Table A-5 and Table A-6 respectively.

## 3.0

THE DECOUPLED OBSERVERS DECMATS AND DECMAT

DECMATS and DECMAT are actually the names of the data files containing different versions of the observer. These data files are created by the DECOBR program, and may be used directly by other programs to pick up the observer matrices  $A_o$ ,  $K_o$ ,  $B_o$ .

In both cases, the decoupled observer was designed with seven rigid modes to be observed (1, 2, 4, 5, 6, 7, 8) and four flexible modes (1, 3, 4, 6). The control period does not affect the observer design, of course.

The decoupled rigid modes form a simple second order system. If the damping factor is specified as  $\zeta$ , and frequency is  $\omega$ , then the corresponding eigenvalues are:

$$\lambda = -\zeta\omega \pm i\omega\sqrt{1-\zeta^2}$$

The frequency  $\omega$  was selected to be a decade lower than the lowest frequency flexible mode not being observed or suppressed. Modes 2, 5 and 7 were intended to be suppressed, so mode 8 was the critical mode. Therefore  $\omega$  was set to  $3.0/10 = 0.3$  rad/sec. The damping factor  $\zeta$  was chosen to be 0.9. Therefore the rigid mode eigenvalues were set at  $-0.2700 \pm i0.1308$ . The flexible mode eigenvalues were also set to this value.

The DECMATS observer was designed to suppress flexible modes 2, 5 and 7. The DECMAT observer was designed without any suppression of residual modes.

The DECMATS observer is given in Table A-7.

## 4.0

THE COUPLED OBSERVER

The program DECOBR has the option of producing a coupled or a decoupled observer. The eigenvalue assignment subroutine ASINEV appears to work well in the decoupled case, where it assigns the eigenvalues to an 8x8 matrix for the flexible modes. Unfortunately, it breaks down in the coupled case, where it has to assign the eigenvalues to a 22x22 matrix. Part of the problem may be the inversion of the T matrix, which may have a very small determinant (e.g.  $-7 \times 10^{-78}$ ). This suggests the problem is not well defined for this case.

Some possible improvements to the ASINEV subroutine are:

- (a) Compute eigenvalues in Double Precision (EIGENP subroutine uses single precision, the rest of the program is in Double Precision).
- (b) The computation of P requires powers of  $A_{OT}$  to be computed. Instead of repeatedly multiplying  $A_{OT}$  by itself, the eigenvalue method could be used:

$$(A_{OT})^k = V \Delta^k V^{-1}$$

where  $\Delta$  is the diagonal eigenvalue matrix of  $A_{OT}$ , and V is the eigenvector matrix.

- (c)  $k_T * T^{-1}$  could be computed directly instead of first computing the inverse of T and then multiplying by  $k_T$ .

TABLE A-1

(96)

THE MODAL WEIGHTING MATRIX  $Q_c$ 

SPAR-R.1135

ISSUE A

APPENDIX A

MATRIX QC	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1.0000E+06	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	1.0000E+06	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
4	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
5	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
11	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
12	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
13	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
14	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
15	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
16	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

MODAL WEIGHTING MATRIX Q<sub>c</sub> (CONTINUED)

SPAR-R.1135

ISSUE A

APPENDIX A

17	0.	0.	0.	0.9230E=10	-3.4520E=07	2.5390E=06	-3.2950E=06	4.3310E=09
	0.	0.	0.	0.	0.	0.	0.	0.
	2.0180E=06	-2.3560E=08	8.7460E=07	-8.0750E=06	4.5690E=06	3.8910E=06	6.7720E=10	8.0330E=07
	4.0600E=08	1.7230E=06	2.3250E=06	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
18	0.	0.	0.	-3.8200E=07	2.1070E=09	-3.4670E=08	2.5290E=08	-7.9730E=07
	0.	0.	0.	0.	0.	0.	0.	0.
	-2.3560E=08	1.3500E=07	3.7970E=07	1.0660E=07	-5.7790E=08	-2.6850E=08	-3.4980E=06	-3.7460E=09
	2.2120E=06	-1.3080E=08	-1.8880E=03	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
19	0.	0.	0.	-1.2370E=06	1.1220E=09	1.5650E=08	-3.6280E=09	-3.1980E=06
	0.	0.	0.	0.	0.	0.	0.	0.
	8.7460E=09	3.7970E=07	1.1150E=06	-6.1950E=08	2.2830E=08	5.2370E=09	-9.5480E=06	-6.3530E=09
	6.0610E=06	-6.1870E=10	5.4200E=09	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
20	0.	0.	0.	-3.7370E=10	-7.1230E=07	-1.4760E=05	2.7240E=06	-1.9990E=08
	0.	0.	0.	0.	0.	0.	0.	0.
	-8.0750E=06	1.0660E=07	-4.1950E=08	4.2470E=05	-2.2030E=05	3.4170E=07	3.9660E=07	3.2990E=06
	-1.9510E=07	6.0670E=07	3.4120E=06	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
21	0.	0.	0.	-7.3630E=10	-2.5980E=07	7.2950E=06	-1.1970E=06	9.2900E=09
	0.	0.	0.	0.	0.	0.	0.	0.
	4.5690E=06	-5.7790E=08	2.2830E=08	-2.2030E=05	1.1760E=05	3.2620E=06	-1.4310E=07	-2.7740E=07
	1.0640E=07	4.8600E=07	1.3930E=06	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
22	0.	0.	0.	-4.5310E=09	-6.0140E=06	-3.5500E=06	-3.0010E=06	-9.6300E=09
	0.	0.	0.	0.	0.	0.	0.	0.
	3.8910E=06	-2.6850E=08	5.2370E=09	3.4170E=07	3.2620E=06	3.5140E=05	6.9430E=07	1.4290E=05
	1.9410E=08	6.1860E=06	-2.2560E=07	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
23	0.	0.	0.	6.7910E=06	-1.3600E=07	-2.1570E=07	-3.3350E=08	-2.1160E=07
	0.	0.	0.	0.	0.	0.	0.	0.
	6.7720E=10	-3.4980E=06	-9.5480E=06	3.9660E=07	-1.4310E=07	6.9430E=07	9.3140E=05	3.6280E=07
	-5.8720E=05	9.6750E=08	-5.7150E=08	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
24	0.	0.	0.	3.1030E=09	-3.5320E=06	-3.1700E=06	1.1880E=05	-2.9090E=09
	0.	0.	0.	0.	0.	0.	0.	0.
	8.0330E=07	-3.7460E=09	-6.3530E=09	3.2990E=06	-2.7740E=07	1.4290E=05	3.6280E=07	6.9820E=06
	-3.9360E=08	-4.3560E=06	-9.3370E=06	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
25	0.	0.	0.	-5.6420E=06	7.1960E=09	7.2670E=08	-1.9610E=08	-1.4020E=06
	0.	0.	0.	0.	0.	0.	0.	0.
	4.0600E=08	2.2120E=06	6.0610E=06	-1.9510E=07	1.0640E=07	1.9410E=08	-5.8720E=05	-3.9360E=08
	3.7040E=05	-8.0490E=09	3.2330E=08	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
26	0.	0.	0.	1.6660E=09	5.9830E=06	3.8660E=06	-8.3640E=05	-9.2290E=11
	0.	0.	0.	0.	0.	0.	0.	0.
	1.7230E=06	-1.3080E=08	-6.1870E=10	-6.0670E=07	4.8600E=07	4.1860E=06	9.6750E=08	-4.3560E=06
	-5.0490E=09	3.0300E=05	3.8060E=05	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
27	0.	0.	0.	-6.0460E=10	1.0100E=05	7.2860E=06	-1.2470E=04	1.6770E=08
	0.	0.	0.	0.	0.	0.	0.	0.
	2.3250E=06	-1.8880E=08	5.4200E=09	-3.4120E=06	1.3930E=06	-2.2560E=07	-5.7150E=08	-9.3370E=06
	3.2330E=08	5.6060E=05	8.7090E=05	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
28	0.	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
29	0.	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.	0.

(CONTINUED NEXT PAGE)

SPAR-R-1135  
ISSUE A  
APPENDIX A

MODAL WEIGHTING MATRIX  $Q_a$  (CONTINUED)

TABLE A-2 THE RICMATE CONTROL MATRIX

CONTROL MATRIX F PRODUCED BY RICQ PROGRAM ( $u = -F x_c$ )

ROW 1	-1.0411E-02	1.0142E+06	1.4355E-05	-1.5894E-01	-1.3373E-01	2.0568E+00	2.4684E-06	-3.7665E-01
	2.0814E-04	7.6859E-04	-1.7109E+00	-1.9022E+00	2.4327E+01	-5.1360E-05	-1.5629E-03	9.6771E-06
	-2.0846E-03	2.4955E-02	-1.9847E-01	1.5869E-05	-1.1739E-01	3.2390E-02		
ROW 2	-6.1942E-06	-1.7173E+02	1.3253E-01	-5.2547E-05	-1.7687E-04	-1.1805E-05	2.1006E+00	-1.2244E-04
	-4.5295E-01	1.5596E+00	-5.8406E-04	-2.2198E-03	-1.8541E-04	2.4578E+01	-8.6736E-05	3.3499E-02
	-1.2131E-04	4.0130E-05	1.5793E-05	-1.0559E-01	2.8195E-04	4.6870E-05		
	1.7700E-01	-3.0192E-01	1.5302E-01	3.6544E-01	7.6142E-01	5.8154E-02	-5.1429E-02	2.9450E+00
ROW 3	-7.8703E+00	3.9120E+00	1.6605E+01	2.2577E+01	4.1136E-01	-3.2961E-01	4.9373E-02	8.9634E-03
	6.1529E-01	-9.2141E-05	5.6634E-01	-2.9508E-02	1.1735E+00	-2.0915E-01		
	4.4736E-01	3.0181E-01	-1.3302E-01	2.0528E-02	-4.0467E-01	1.7695E-02	5.1481E-02	1.7163E+01
ROW 4	7.8684E+00	-3.9068E+00	-1.9290E+00	-1.7688E+01	1.6221E-01	3.3029E-01	-3.9505E-02	-9.1066E-03
	-9.3235E-01	1.3402E-01	5.5849E-01	2.9721E-02	3.1235E-02	7.9669E-03		
	-4.4789E-01	3.0074E-01	-1.3202E-01	-1.9566E-02	4.0707E-01	-1.7631E-02	5.1504E-02	-1.7184E+01
ROWS	7.8524E+00	-3.8698E+00	1.9645E+00	1.7744E+01	-1.6167E-01	3.2481E-01	5.9500E-02	-9.0191E-03
	9.5594E-01	-1.3452E-01	-3.4188E-01	2.9845E-02	-2.3815E-02	-7.4715E-03		
ROW 6	-1.7661E-01	-3.0063E-01	1.3202E-01	-3.6621E-01	-7.6292E-01	-5.8254E-02	-5.1356E-02	-2.9301E+00
	-7.8310E+00	3.8646E+00	-1.6632E+01	-2.2612E+01	-4.1182E-01	-3.2949E-01	-4.9301E-02	9.1621E-03
	-6.1868E-01	9.6595E-03	-5.6227E-01	-3.0057E-02	-1.1777E+00	2.0902E-01		
	2.5644E-02	-6.7645E-02	-1.4204E-01	-6.1458E-02	9.4464E-02	2.7733E-02	3.0656E-02	1.3034E+00
ROW 7	-1.5253E+00	-3.7036E+00	-3.5976E+00	1.8601E+00	1.6551E-01	3.2059E-02	-8.4465E-02	3.6415E-03
	6.8960E-02	-2.1219E-01	-6.4232E-01	-6.9259E-01	4.8961E-01	-3.6154E-01		
	2.8987E-02	6.7886E-02	1.4239E-01	-6.2193E-02	9.4042E-02	2.7843E-02	-3.0779E-02	1.3145E+00
ROW 8	1.3271E+00	3.7051E+00	-3.6186E+00	1.8367E+00	1.6662E-01	-3.2233E-02	-8.2559E-02	-3.7151E-03
	6.5534E-02	-2.1233E-01	-6.2218E-01	6.9403E-01	4.8518E-01	-3.6423E-01		
ROW 9	-2.8649E-02	6.7695E-02	1.4204E-01	6.1458E-02	-9.4464E-02	-2.7733E-02	-3.0656E-02	-1.3034E+00
	1.5253E+00	3.7036E+00	-3.5976E+00	-1.8601E+00	-1.6551E-01	-3.2059E-02	-8.4465E-02	-3.6415E-03
	6.8960E-02	-2.1219E-01	-6.4232E-01	-6.9259E-01	-4.8961E-01	3.6154E-01		
	-2.8987E-02	6.7886E-02	-1.4239E-01	-6.2193E-02	-9.4042E-02	-2.7843E-02	3.0779E-02	-1.3145E+00
ROW 10	-1.3271E+00	-3.7051E+00	3.6186E+00	-1.8367E+00	-1.6662E-01	3.2233E-02	-8.2559E-02	-3.7151E-03
	6.5534E-02	2.1233E-01	6.2218E-01	6.9403E-01	4.8518E-01	3.6423E-01		

CONTROL PERIOD = 10 SECS.

DYNACON MODAL WEIGHTING MATRIX.

 $R = \text{DIAG}(10^{-6})$ ,  $R_w = \text{DIAG}(10^{-6})$ 

(-F) STORED IN FILE RICMATF.

## STABILITY ANALYSIS OF CONTROL MATRIX (RICMATF)

## EIGENVALUES OF CONTROLLED MODES

ORDER	MAGNITUDE	PHASE (DEGREES)
4	.7074E+00	.1353E+03
	.7074E+00	-.1353E+03
1	.9326E+00	.7393E+02
	.9326E+00	-.7393E+02
9	.5782E+00	.1800E+03
	.5567E+00	.1800E+03
2	.8003E+01	.4964E+02
	.8003E+01	-.4964E+02
10	.2662E+00	.8475E+02
	.2662E+00	-.8475E+02
3	.7350E+00	.1668E+02
	.7350E+00	-.1668E+02
5	.6846E+00	.8343E+01
	.6896E+00	-.8343E+01
7	.6450E+00	.2305E+02
	.6430E+00	-.2305E+02
6	.6730E+00	.2189E+02
	.6730E+00	-.2189E+02
8	.5957E+00	.2565E+02
	.5957E+00	-.2565E+02
11	.7016E-01	.1800E+03
12	.5826E-01	.1800E+03

## EIGENVALUES OF RESIDUAL MODES

ORDER	MAGNITUDE	PHASE (DEGREES)
1	.9915E+00	.8662E+02
	.9915E+00	-.8662E+02
5	.4719E+00	.1692E+03
	.4719E+00	-.1692E+03
4	.5901E+00	.1665E+03
	.5901E+00	-.1665E+03
2	.9461E+00	.3544E+02
	.9461E+00	-.3544E+02
3	.7556E+00	.2446E+01
	.7556E+00	-.2446E+01
6	.5265E-02	.4465E+01
	.5265E-02	-.4465E+01
7	.7950E-05	.1161E+02
	.7950E-05	-.1161E+02

TABLE A-4

SPAR-R.1135

ISSUE A

APPENDIX A

(103)

## Q MATRIX PRODUCED BY CONS2 PROGRAM.

## MATRIX QBAR

8 22 BY 22

ROW 1	1.0000E-05,	0.	0.	0.	0.	0.	0.
	0.	0.	0.	5.0000E-05,	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	1.0000E-05,	0.	0.	0.	0.	0.
Row 2	0.	0.	0.	0.	5.0000E-05,	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	1.6540E-05,	7.0190E-09,	-1.6840E-08,	
ROW 3	-1.6690E-09,	1.2810E-04,	0.	0.	0.	8.2700E-05,	
	-3.5095E-08,	-8.4200E-08,	-8.3450E-09,	6.4050E-04,	-3.7840E-09,		
	-3.5737E-06,	7.2936E-10,	-3.5032E-09,	-2.0984E-08,	-3.7020E-05,		
	-3.9324E-10,	-6.8484E-09,					
	0.	0.	0.	-7.0190E-09,	2.1930E-05,	1.5350E-05,	
ROW 4	-1.3880E-04,	-5.3420E-08,	0.	0.	0.	-3.5095E-08,	
	1.0965E-04,	7.6750E-05,	-6.9400E-04,	-2.6710E-07,	-2.6509E-06,		
	3.0554E-09,	7.7970E-07,	-6.6778E-06,	-1.4679E-05,	3.3824E-08,		
	-5.5102E-07,	-9.1555E-06,					
	0.	0.	0.	-1.6840E-08,	1.5350E-05,	5.8910E-05,	
ROWS	-9.3740E-05,	-1.2440E-07,	0.	0.	0.	-8.4200E-08,	
	7.6750E-05,	2.9455E-04,	-4.6870E-04,	-6.2200E-07,	1.9404E-05,		
	4.5872E-08,	1.5622E-05,	-3.6060E-06,	1.1014E-04,	4.6618E-07,		
	-1.4995E-05,	-5.0231E-06,					
	0.	0.	0.	-1.6690E-09,	-1.3880E-04,	-9.3740E-05,	
ROW 6	1.7910E-03,	-9.7170E-09,	0.	0.	0.	-8.3450E-09,	
	-6.9400E-04,	-4.6870E-04,	8.9550E-03,	-4.8585E-08,	-2.5147E-05,		
	-1.0856E-08,	-2.6766E-06,	-3.7561E-06,	-1.4348E-04,	-1.0724E-07,		
	2.7479E-06,	-4.5600E-06,					
	0.	0.	0.	1.2810E-04,	-5.3420E-08,	-1.2440E-07,	
ROW 7	-9.7170E-09,	1.9990E-03,	0.	0.	0.	6.4050E-04,	
	-2.6710E-07,	-6.2200E-07,	-4.8585E-08,	9.9950E-03,	3.3003E-08,		
	-1.5017E-05,	2.2339E-08,	-1.1925E-08,	1.8972E-07,	-1.5556E-04,		
	-2.0666E-08,	-1.4009E-08,					
	5.0000E-05,	0.	0.	0.	0.	0.	0.
ROW 8	0.	0.	0.	3.3333E-04,	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	5.0000E-05,	0.	0.	0.	0.	0.
ROW 9	0.	0.	0.	0.	3.3333E-04,	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.	0.
ROW 10	0.	0.	0.	0.	0.	0.	0.
	-8.3450E-09,	6.4050E-04,	0.	0.	0.	5.5133E-04,	
	-2.3397E-07,	-5.6133E-07,	-5.5633E-08,	4.2700E-03,	-1.6146E-08,		
	1.8058E-06,	6.4504E-09,	-4.3493E-08,	-1.3618E-07,	-2.1688E-04,		
	1.0500E-08,	-2.9948E-09,					
	0.	0.	0.	-3.5095E-08,	1.0965E-04,	7.6750E-05,	
ROW 11	-6.9400E-04,	-2.6710E-07,	0.	0.	0.	-2.3397E-07,	
	7.3100E-04,	5.1167E-04,	-4.6267E-03,	-1.7807E-06,	-1.1292E-05,		
	-2.7932E-09,	8.2944E-06,	-5.3863E-05,	-9.6978E-05,	1.9593E-07,		
	2.0019E-05,	-3.5094E-06,					

(CONTINUED ON NEXT PAGE)

## Q MATRIX CONTINUED

	0.	0.	-8.4200E=08,	7.6750E=05,	2.9455E=04,
ROW 12	-4.6670E=04,	-6.2200E=07,	0.	0.	-5.6133E=07,
	5.1167E=04,	1.9637E=03,	-3.1247E=03,	-4.1467E=06,	8.2463E=05,
	-1.8048E=08,	1.8066E=04,	-3.2504E=05,	7.1427E=04,	2.7335E=06,
	3.7748E=04,	2.2409E=06,			
	0.	0.	-8.3450E=09,	-6.9400E=04,	-4.6870E=04,
ROW 13	8.9550E=03,	-4.8585E=08,	0.	0.	-5.5633E=08,
	-4.6267E=03,	-3.1247E=03,	5.9700E=02,	-3.2390E=07,	-1.0675E=04,
	2.6279E=09,	-3.2224E=05,	-2.9094E=05,	-9.3021E=04,	-6.2927E=07,
	-6.9193E=05,	-2.4076E=06,			
	0.	0.	6.4050E=04,	-2.6710E=07,	-6.2200E=07,
ROW 14	-4.8585E=08,	9.9950E=03,	0.	0.	4.2700E=03,
	-1.7807E=06,	-4.1467E=06,	-3.2390E=07,	6.6633E=02,	1.3993E=07,
	7.5880E=06,	2.5240E=07,	-9.7466E=08,	1.2306E=06,	-9.1134E=04,
	5.1148E=07,	-1.2787E=09,			
	0.	0.	-3.7840E=09,	-2.6509E=06,	1.9404E=05,
ROW 15	-2.5147E=05,	3.3003E=08,	0.	0.	-1.6146E=08,
	-1.1292E=05,	8.2463E=05,	-1.0675E=04,	1.3993E=07,	1.2613E=05,
	2.9533E=08,	4.6573E=06,	1.8640E=06,	5.8045E=05,	1.8620E=07,
	-2.4870E=05,	5.6591E=06,			
	0.	0.	-3.5737E=06,	3.0554E=09,	4.5872E=08,
ROW 16	-1.0856E=08,	-1.5017E=05,	0.	0.	1.8058E=06,
	-2.7932E=09,	-1.8048E=08,	2.6279E=09,	7.5880E=06,	2.9533E=08,
	4.3832E=06,	-2.7524E=08,	-2.0281E=09,	6.6849E=09,	4.7220E=06,
	-2.7525E=07,	6.8956E=09,			
	0.	0.	7.2936E=10,	7.7970E=07,	1.5622E=05,
ROW 17	-2.6766E=06,	2.2339E=08,	0.	0.	6.4504E=09,
	8.2944E=06,	1.8066E=04,	-3.2224E=05,	2.5240E=07,	4.6573E=06,
	-2.7524E=08,	1.7976E=04,	3.3367E=06,	9.2437E=05,	3.9994E=07,
	3.6559E=05,	3.3644E=06,			
	0.	0.	-5.5032E=09,	-6.6778E=06,	-3.6060E=06,
ROW 18	-3.7561E=06,	-1.1925E=08,	0.	0.	-4.3493E=08,
	-5.3863E=05,	-3.2504E=05,	-2.9094E=05,	-9.7466E=08,	1.8640E=06,
	-2.0281E=09,	3.3367E=06,	1.4615E=04,	2.5843E=05,	1.4850E=08,
	-7.1476E=06,	2.8740E=05,			
	0.	0.	-2.0984E=08,	-1.4879E=05,	1.1014E=04,
ROW 19	-1.4348E=04,	1.8972E=07,	0.	0.	-1.3618E=07,
	-9.6978E=05,	7.1427E=04,	-9.3021E=04,	1.2306E=06,	5.8045E=05,
	6.6849E=09,	9.2437E=05,	2.5843E=05,	4.8062E=04,	1.3186E=06,
	1.4860E=04,	8.8677E=07,			
	0.	0.	-3.7020E=05,	3.3824E=08,	4.6618E=07,
ROW 20	-1.0724E=07,	-1.5556E=04,	0.	0.	-2.1688E=04,
	1.9593E=07,	2.7335E=06,	-6.2927E=07,	-9.1134E=04,	1.8620E=07,
	4.7220E=06,	3.9994E=07,	1.4850E=08,	1.3186E=06,	1.1495E=04,
	1.6715E=07,	-2.9431E=09,			
	0.	0.	-3.9324E=10,	-5.5182E=07,	-1.4995E=05,
ROW 21	2.7479E=06,	-2.0666E=08,	0.	0.	1.0500E=08,
	2.0019E=05,	3.7748E=04,	-6.9193E=05,	5.1148E=07,	-2.4870E=05,
	-2.7525E=07,	3.6559E=05,	-7.1476E=06,	1.4860E=04,	1.6715E=07,
	6.8155E=04,	7.2173E=06,			
	0.	0.	-6.8484E=09,	-9.1555E=06,	-5.0231E=06,
ROW 22	-4.5600E=06,	-1.4009E=08,	0.	0.	-2.9948E=09,
	-3.5994E=06,	2.2409E=06,	-2.4076E=06,	-1.2787E=09,	5.6591E=06,
	6.8956E=09,	3.3644E=06,	2.8740E=05,	8.8677E=07,	-2.9431E=09,
	7.2173E=06,	2.3383E=04,			

TABLE A-5 THE RICQF CONTROL MATRIX

CONTROL MATRIX F PRODUCED BY RICQ PROGRAM ( $u = -Fx_c$ )

ROW 1	-1.8257E-02	-8.2767E-06	-5.3552E-06	-1.0584E-01	-9.4367E-02	1.3295E+00	-4.2735E-06	-4.9849E-01
	-8.0934E-05	1.0876E-04	-1.4935E+00	-1.5233E+00	2.1019E+01	-7.9715E-05	5.2649E-03	2.7710E-05
	2.4920E-03	2.2058E-02	-1.6015E-01	1.9224E-04	-3.7568E-02	4.2078E-02		
ROW 2	-7.9777E-06	-3.2186E-02	7.2925E-02	4.0977E-06	-6.3499E-05	-3.0714E-05	1.3369E+00	-1.8918E-04
	-6.1559E-01	1.1202E+00	-8.2575E-05	-1.0080E-03	-4.0226E-04	2.1094E+01	-1.5097E-04	2.0095E-02
	6.5827E-05	8.5447E-05	-1.8482E-03	-1.8973E-02	2.1728E-04	4.2078E-04		
ROW 3	5.6699E-01	-6.3369E-01	2.4926E-01	5.4786E-01	1.2266E+00	7.5108E-02	-6.0445E-02	1.0971E+01
	-1.2100E+01	4.8097E+00	1.4108E+01	2.3893E+01	5.3308E-01	-3.5669E-01	2.6644E-01	5.9180E-02
	8.1631E-01	-2.3340E-01	3.1728E+00	3.3282E-01	-5.3864E-01	-5.5688E-01		
ROW 4	9.2942E-01	6.3354E-01	-2.4937E-01	1.9479E-01	-7.3463E-01	5.8604E-02	6.0586E-02	2.4170E+01
	1.2100E+01	-4.8048E+00	5.4810E-01	-2.1391E+01	3.0356E-01	3.5725E-01	1.4025E-01	-5.9316E-02
	-2.1667E+00	3.4797E-01	3.7054E-01	-3.3797E-01	1.4521E+00	2.0078E-01		
ROW 5	-9.3029E-01	6.3069E-01	-2.4738E-01	-1.9329E-01	7.3876E-01	-5.8747E-02	6.0623E-02	-2.4196E+01
	1.2042E+01	-4.7562E+00	-5.1512E-01	2.1468E+01	-3.0390E-01	3.6019E-01	-1.4344E-01	-6.1802E-02
	2.1722E+00	-3.4760E-01	-3.8053E-01	-3.3982E-01	-1.4385E+00	-1.9625E-01		
ROW 6	-5.6712E-01	-6.3055E-01	2.4750E-01	-5.4795E-01	-1.2283E+00	-7.5049E-02	-6.0766E-02	-1.0981E+01
	-1.2042E+01	4.7514E+00	-1.4092E+01	-2.3898E+01	-5.3310E-01	-3.6076E-01	-2.6393E-01	6.1938E-02
	-8.1867E-01	2.3247E-01	-3.1659E+00	3.4497E-01	5.4407E-01	5.5456E-01		
ROW 7	2.9668E-02	-7.6931E-02	-2.1110E-01	-8.3119E-02	9.7909E-02	5.8793E-03	1.7501E-02	1.6304E+00
	-1.4564E+00	-4.0169E+00	-4.0240E+00	1.6932E+00	1.5750E-02	-5.4439E-03	-5.5552E-02	-1.9052E-02
	-3.2756E-02	-2.1422E-01	-8.7595E-01	-9.8764E-01	4.2162E-01	-3.2698E-01		
ROW 8	2.9983E-02	7.7177E-02	2.1186E-01	-8.4397E-02	9.5731E-02	6.2421E-03	-1.7703E-02	1.6400E+00
	1.4587E+00	4.0227E+00	-4.0455E+00	1.6548E+00	1.9593E-02	4.3680E-03	-5.2249E-02	1.9195E+02
	-3.6356E-02	-2.1567E-01	-8.3697E-01	9.9057E-01	4.1732E-01	-3.3579E-01		
ROW 9	2.9668E-02	7.6931E-02	2.1110E-01	8.3119E-02	-9.7909E-02	-5.8793E-03	-1.7501E-02	-1.6304E+00
	1.4564E+00	4.0169E+00	-4.0240E+00	1.6932E+00	-1.5750E-02	5.4439E-03	5.5552E-02	1.9052E-02
	3.2756E-02	2.1422E-01	-8.7595E-01	-9.8764E-01	-4.2162E-01	3.2698E-01		
ROW 10	2.9983E-02	-7.7177E-02	-2.1186E-01	8.4397E-02	-9.5731E-02	-6.2421E-03	1.7703E-02	-1.6400E+00
	1.4587E+00	-4.0227E+00	4.0455E+00	-1.6548E+00	-1.9593E-02	-4.3680E-03	5.2249E-02	-1.9195E+02
	3.6356E-02	2.1567E-01	8.3697E-01	-9.9057E-01	-4.1732E-01	3.3579E-01		

CONTROL PERIOD = 10 SECs.  
(-F) STORED IN FILE RICQ.F.Q MODAL WEIGHTING MATRIX.  $R = \text{DIAG}(10^{-6})$ ,  $R_R = \text{DIAG}(10^{-6})$

TABLE A-6  
STABILITY ANALYSIS OF CONTROL MATRIX (RICQF)

## EIGENVALUES OF CONTROLLED MODES

ORDER	MAGNITUDE	PHASE (DEGREES)
1	.8972E+00	.7120E+02
	.8972E+00	=.7120E+02
2	.7481E+00	.2067E+02
	.7481E+00	=.2067E+02
7	.3363E+00	.1607E+03
	.3363E+00	=.1607E+03
3	.6260E+00	.3158E+02
	.6260E+00	=.3158E+02
4	.5568E+00	.2792E+02
	.5568E+00	=.2792E+02
5	.4578E+00	.4495E+02
	.4578E+00	=.4495E+02
8	.2805E+00	.2847E+02
	.2805E+00	=.2847E+02
6	.4348E+00	.2889E+02
	.4348E+00	=.2889E+02
9	.1220E+00	.2657E+02
	.1220E+00	=.2657E+02
12	.1684E-01	0.
11	.5262E-01	0.
13	.1518E-01	0.
10	.5776E-01	0.

## EIGENVALUES OF RESIDUAL MODES

ORDER	MAGNITUDE	PHASE (DEGREES)
1	.9915E+00	.8662E+02
	.9915E+00	=.8662E+02
7	.4719E+00	.1692E+03
	.4719E+00	=.1692E+03
5	.5901E+00	.1065E+03
	.5901E+00	=.1065E+03
2	.9461E+00	.3544E+02
	.9461E+00	=.3544E+02
4	.7556E+00	.2446E+01
	.7556E+00	=.2446E+01
6	.5285E-02	.4485E+01
	.5285E-02	=.4485E+01
3	.7950E-05	.1161E+02
	.7950E-05	=.1161E+02

TABLE A-7

THE DECMATS OBSERVER ( $\omega = 0.3 \text{ r/s}$ ,  $\zeta = 0.9$ , suppress modes 2, 5, 7)

		OBSERVER A, ARRAY	$(A_0 = A_c - k_c C_c)$
Row 1	$-5.40000000E+01$ ,	$1.00000000E+00$ ,	0.
	0,	0,	0.
	$-2.44541023E+01$ ,	0,	$-1.00776457E+01$ ,
	0,	$-4.28275862E+02$ ,	0.
	$-1.85468490E+05$ ,	0,	$-9.79512485E+02$ ,
	$-6.37598098E+05$ ,	$-4.94732461E+02$ ,	$1.17535077E+01$ ,
	0,	0,	0.
Row 2	$-9.00086400E-02$ ,	0,	0.
	0,	0,	0.
	$-4.07607498E+02$ ,	0,	$-1.67976885E+02$ ,
	0,	$-7.13861628E+03$ ,	0.
	$-3.09143825E+06$ ,	0,	$-1.63267753E+02$ ,
	$-1.06276551E+05$ ,	$-8.24633259E+03$ ,	$1.95910601E+02$ ,
	0,	0,	0.
Row 3	0,	0,	$-5.40000000E+01$ ,
	1.00000000E+00,	$2.12372176E+01$ ,	0.
	$-9.01177170E+05$ ,	0,	$-1.01835910E+03$ ,
	0,	$-7.91058264E+05$ ,	0.
	$-2.99793103E+02$ ,	0,	$-1.73140309E+04$ ,
	$-1.25850178E+01$ ,	$2.66115339E+03$ ,	$-2.17027348E+04$ ,
	0,	0,	0.
Row 4	0,	0,	$-9.00086400E-02$ ,
	0,	$3.53987606E+02$ ,	0.
	$-1.50210614E+05$ ,	0,	$-1.69742810E+04$ ,
	0,	$-1.31855701E+05$ ,	0.
	$-4.99703139E+03$ ,	0,	$-2.88594884E+05$ ,
	$-2.09770433E+02$ ,	$4.43568143E+04$ ,	$-3.61746972E+05$ ,
	0,	0,	0.
Row 5	0,	0,	0.
	0,	$-5.40000000E+01$ ,	$1.00000000E+00$ ,
	$2.29146690E+04$ ,	0,	$5.49678909E+04$ ,
	0,	$5.44750385E+05$ ,	0.
	$2.79560150E+02$ ,	0,	$4.91783594E+04$ ,
	$5.87147570E+01$ ,	$-5.26045744E+04$ ,	$2.57055201E+03$ ,
	0,	0,	0.
Row 6	0,	0,	0.
	0,	$-9.00086400E+02$ ,	0.
	$3.81947814E+05$ ,	0,	$9.16219464E+05$ ,
	0,	$9.08004468E+06$ ,	0.
	$4.65978312E+03$ ,	0,	$8.19718008E+05$ ,
	$9.78673227E+02$ ,	$-8.76827075E+05$ ,	$4.26466464E+04$ ,
	0,	0,	0.

## DECIMALS. OBSERVER A. ARRAY. CONTINUED (1)

	0.	0.	0.
	0.	0.	0.
Row 7	-5.40000000E+01,	1.00000000E+00,	-3.78103152E+01,
	0.	-6.24068768E+02,	0.
	-5.77134670E+05,	0.	3.02028653E+01,
	-1.10011461E+04,	2.04446991E+01,	3.05226361E+00,
	0.	0.	0.
	0.	0.	0.
	0.	0.	0.
	-9.00086400E+02,	0.	-6.30232416E+02,
	0.	-1.04021446E+02,	0.
Row 8	-9.61983459E+06,	0.	5.03429413E+02,
	-1.83370037E+05,	3.40777697E+02,	5.08759438E+01,
	0.	0.	0.
	0.	0.	0.
	0.	0.	0.
	0.	0.	0.
Row 9	1.00000000E+00,	-3.88487269E+02,	-5.40000000E+01,
	-5.96615077E+05,	0.	-3.11922117E+01,
	-1.56688967E+03,	1.59924114E+00,	-7.40309536E+02,
	0.	0.	0.
	0.	0.	0.
	0.	0.	0.
Row 10	0.	-6.47540940E+03,	-9.00086400E+02,
	-9.94453921E+06,	0.	-5.19920102E+02,
	-2.61173348E+04,	2.66565778E+01,	-1.23396768E+02,
	0.	0.	0.
	0.	0.	0.
	0.	0.	0.
	0.	0.	0.
Row 11	-5.28101959E+06,	-5.40000000E+01,	1.00000000E+00,
	1.16647465E+04,	0.	4.31828917E+02,
	0.	-6.62402074E+02,	-3.76226959E+01,
	0.	0.	0.
	0.	0.	0.
	0.	0.	0.
Row 12	-8.80254427E+07,	-9.00086400E+02,	7.19783954E+03,
	1.94431106E+05,	0.	-6.27105127E+02,
	0.	-1.16410944E+02,	0.
	0.	0.	0.

DEC.MATS OBSERVER A<sub>0</sub> ARRAY CONTINUED (ii)

	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
Row 13	-5.40000000E+01,	1.00000000E+00,	1.71573472E+04,		
	2.01066320E+01,	*1.35058518E+04,	1.88972692E+04,		
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
Row 14	-9.00086400E-02,	0.	2.85983238E+05,		
	3.35142704E-02,	*2.25119139E-05,	3.14984722E+05,		
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
Row 15	-7.94320120E+03,	1.19853590E+01,	-3.73369859E+02,		
	1.00000000E+00,	0.	6.02716559E+01,		
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
Row 16	-1.87685017E+01,	2.74196213E+00,	-8.50925255E+01,		
	0.	1.00000000E+00,	1.38380563E+01,		
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
Row 17	5.58185218E+02,	*8.33032098E+01,	2.58861272E+01,		
	0.	0.	-4.19974493E+00,		
	0.	0.	8.00000000E+00,		
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
Row 18	0.	0.	0.	0.	0.
	1.36485363E+02,	*2.04562967E+01,	6.31947491E+02,		
	0.	0.	-1.03743209E+00,		
	1.00000000E+00,	0.	0.	0.	0.

DECMATS OBSERVER A<sub>0</sub> ARRAY CONTINUED (iii)

	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
Row 19	0.	0.	0.	0.	0.
	2.82673703E-03,	-5.03816074E-02,			7.23523774E-05,
	-1.84609310E-03,	0.			-2.55996263E-01,
	0.				0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
Row 20	0.	0.	0.	0.	0.
	-3.24758256E-02,	-3.40080923E-01,			1.05930888E-01,
	0.	-3.40891823E-03,			-1.71033338E+00,
	0.				0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
Row 21	0.	0.	0.	0.	0.
	-4.40594041E-02,	3.56359985E-01,			2.06454564E-01,
	0.	0.			3.36414019E+00,
	0.				-1.71116506E-02,
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.
Row 22	0.	0.	0.	0.	0.
	2.22539784E-02,	-3.31305530E-01,			1.02707532E-01,
	0.	0.			-2.28213144E+00,
	0.				0.
	0.	0.	0.	0.	0.
	-4.21471433E-02,				

SPAR-R.1135  
ISSUE A  
APPENDIX A

THE DECMATS OBSERVER

OBSERVER K<sub>o</sub> MATRIX

3.21046373E+01,

Row 1

0.  
0.  
0.  
0.  
0.

5.35128656E+00,

Row 2

0.  
0.  
0.  
0.

3.21046373E+01,

Row 3

0.  
0.  
0.  
0.

5.35128656E+00,

Row 4

0.  
0.  
0.  
0.

5.93798109E+02,

Row 5

0.  
0.  
0.  
0.

9.89758522E+01,

Row 6

0.  
0.  
0.  
0.

0.  
0.  
0.  
0.

5.15759312E+02,

Row 7

0.  
0.  
0.  
0.

0.  
0.  
0.  
0.

8.59681375E+01,

Row 8

0.  
0.  
0.  
0.

0.  
0.  
0.  
0.

2.69595607E+02,

Row 9

0.  
0.  
0.  
0.

0.  
0.  
0.  
0.

DECMATS OBSERVER K<sub>6</sub> MATRIX CONTINUED. (i)

	0.	0.	0.
	0.	0.	0.
Row 10	0.	0.	0.
	0.	0.	0.
	0.	0.	0.
	0.	0.	0.
Row 11	0.	0.	0.
	0.	0.	0.
	0.	0.	0.
	0.	0.	0.
Row 12	0.	0.	0.
	0.	0.	0.
	0.	0.	0.
	0.	0.	0.
Row 13	7.80234070E+01,	0.	0.
	0.	0.	0.
	0.	0.	0.
Row 14	1.30051495E+01,	0.	0.
	0.	0.	0.
	0.	0.	0.
Row 15	0.	0.	0.
	0.	0.	0.
	2.19766290E+00,	2.42155529E+00,	3.98899648E+01,
	6.70527764E+01,	-4.05200516E+01,	-8.75108186E+00,
Row 16	0.	0.	0.
	0.	0.	0.
	5.34397860E+01,	5.25533102E+01,	9.63696077E+00,
	1.46622325E+03,	-9.87230039E+02,	-1.22553495E+02,
Row 17	0.	0.	0.
	0.	0.	0.
	-1.55851786E+01,	-1.62444492E+01,	-2.82336488E+00,
	-4.52086026E+02,	2.87532958E+02,	4.47857762E+01,
Row 18	0.	0.	0.
	0.	0.	0.
	-3.78109676E+00,	-3.65086596E+00,	-6.86672176E+01,
	-1.02965534E+02,	6.97167465E+01,	1.74755630E+00,
Row 19	0.	0.	0.
	0.	0.	0.
	-6.36697115E+01,	-8.72364842E+01,	-1.22118636E+01,
	-2.47213216E+01,	1.15508519E+01,	-3.03002611E+01,
Row 20	0.	0.	0.
	0.	0.	0.
	-7.52924752E+00,	-6.86249180E+00,	-1.33852999E+00,
	-1.90055112E+02,	1.39647391E+02,	2.45970149E+01,

## DECMATS OBSERVER KO MATRIX CONTINUED (ii)

	0.	0.	0.
	0.	0.	0.
Row 21	0.	1.25767909E+01,	2.20616035E+00,
	1.21354598E+01,	-2.23672429E+02,	-2.46624121E+01,
	<u>3.51652924E+02,</u>		
	0.	0.	0.
	0.	0.	0.
Row 22	0.	-6.23878447E+00,	-1.12665356E+00,
	-6.21406487E+00,	1.14741192E+02,	1.17450751E+01,
	<u>-1.74519287E+02,</u>		

8/4/mcl702/63

SPAR-R.1135  
ISSUE A  
APPENDIX B

APPENDIX B

GRAPHS OF MODAL COORDINATES VERSUS TIME

The following set of computer graphs were obtained from the FDCCSIM numerical simulation program. The 'run number' corresponds to the run number used in Table 4-1. The rigid modes were initialized at 0.1, flexible modes at zero. Each run was 100 seconds long, and the disturbance torque was turned off.

Graphs 1, 2, 3, and 4 correspond to run number 1: the RICMATE controller is used with feedback taken directly from the spacecraft model, the observer is not in the loop. The DECMATS observer is present, however, and is just tracking the spacecraft model outputs. All residual modes are present. Graph 1 shows the spacecraft rigid mode 1 and the observer estimate of rigid mode 1. Similarly, Graph 2 shows the spacecraft and observer estimate of rigid mode 8. As can be seen, the controller brings the rigid modes back to zero in about 30 or 40 seconds. Graphs 3 and 4 show the spacecraft flexible modes 1 and 6, and corresponding observer output. The flexible modes are initially excited by the control action on the rigid modes, but the controller also damps out the flexible mode oscillations. The observer seems to track the spacecraft modes reasonably well, except possibly for rigid mode 8.

Graphs 5, 6, 7 and 8 correspond to run number 3: the DECMATS observer is used to drive the RICMATE controller and no residual modes are present. Graphs 5 and 6 show the spacecraft rigid modes 1 and 8 and corresponding observer output and Graphs 7 and 8 show flexible modes 1 and 6 and observer output. The first opportunity for a control input occurs at time  $t = 0$  seconds, but at this instant the observer is at its initial value, zero. Since the controller takes its input from the observer, there is therefore no control action at time  $t = 0$ . The rigid modes remain at 0.1 until the next opportunity for a control action, at  $t = 10$  seconds. By this time, the observer has had a chance to respond and in fact, the rigid mode estimates are very close to the correct values. The controller can now bring

8/4/mcl702/64

SPAR-R.1135  
ISSUE A  
APPENDIX B

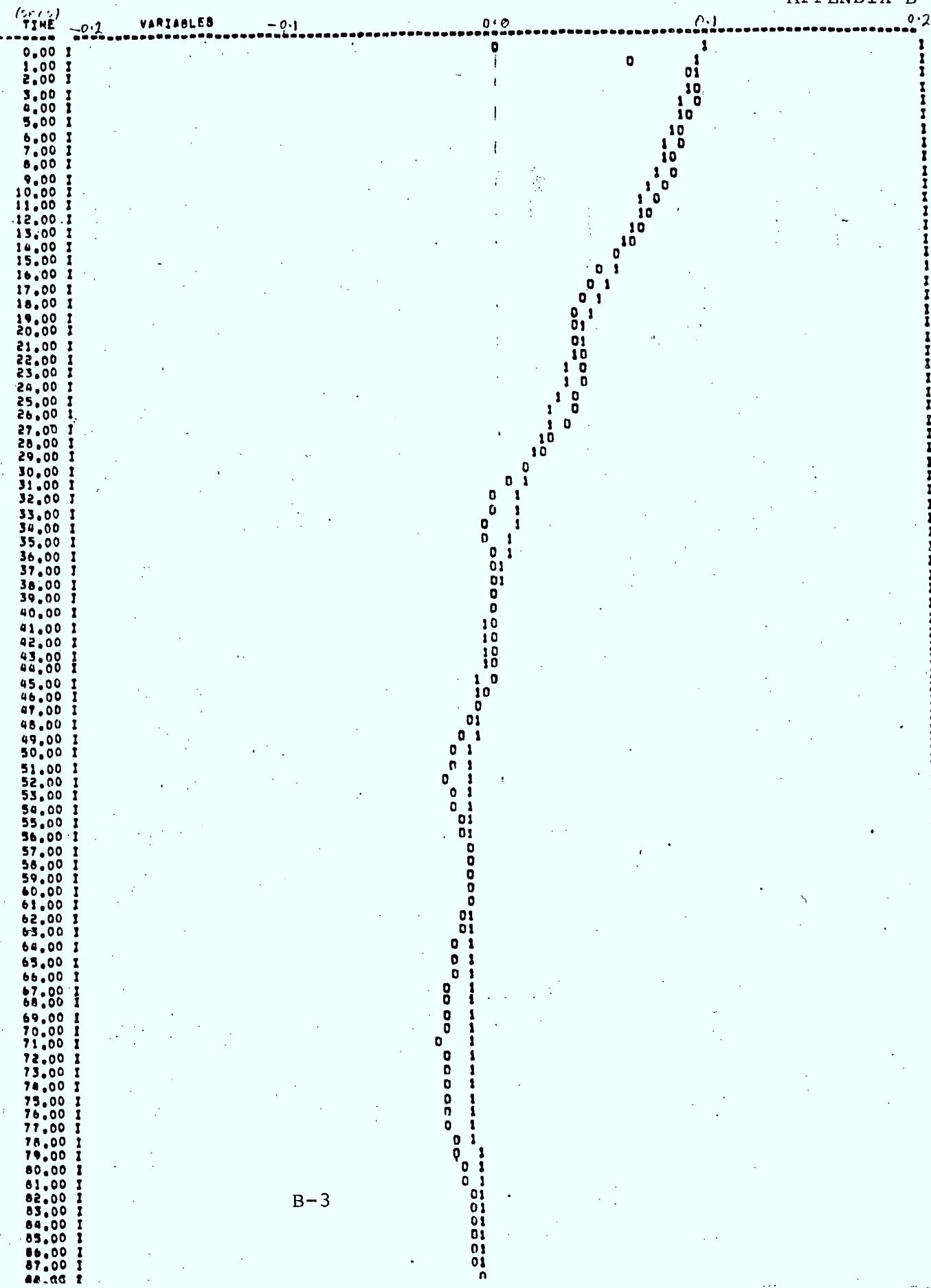
the rigid modes back to zero, at the same time exciting the flexible modes, which eventually die down. As shown by the graph, the observer estimates remain very close to the correct values.

Graphs 9, 10, 11 and 12 correspond to run number 6: this is the same as run number 3 except residual mode 11 is present. Graphs 9 and 10 show rigid modes 1 and 8, and Graphs 11 and 12 show flexible modes 1 and 6. As can be seen, the modes are now unstable and the observer is no longer able to track the spacecraft modes.

Graphs 13, 14, 15 and 16 correspond to run number 8: this is the same as run number 3 except the RICQ controller is used instead of RICMATF. Graphs 13 and 14 show rigid modes 1 and 8, and Graphs 15 and 16 show flexible modes 1 and 6. As can be seen, there is some difference between runs 3 and 8, although it is not clear which run is 'best'.

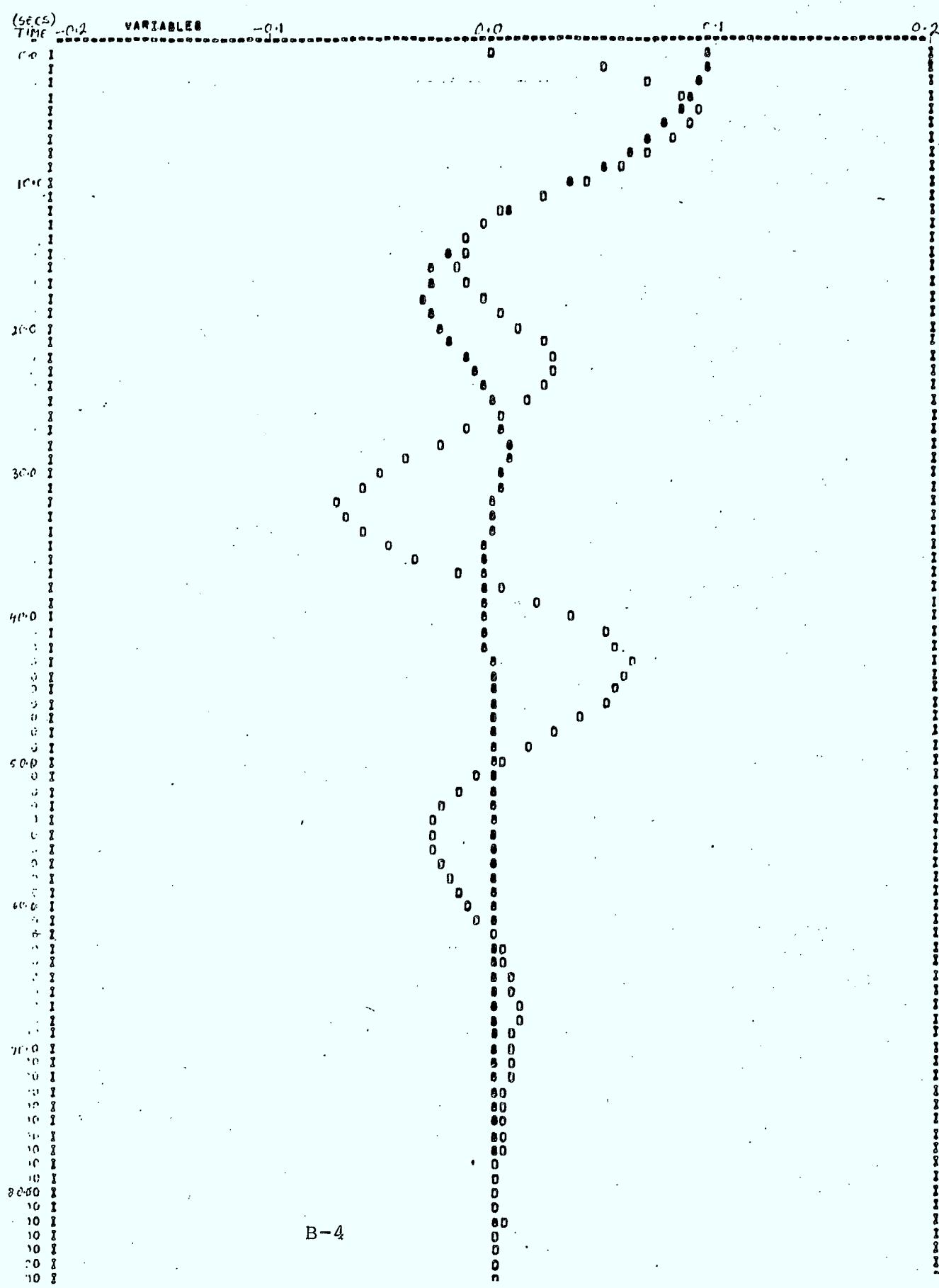
## Graph 1: PLOT OF RIGID MODE + (SYMBOL 4) AND CORRESPONDING

observer output (symbol 0). Run number 1.

SPAR-R.1135  
ISSUE A  
APPENDIX B

Graph 2: Plot of rigid mode 8 (symbol 8) and corresponding  
observer output (symbol 0). Run number 1.

SPAR-R.1135  
ISSUE A  
APPENDIX B

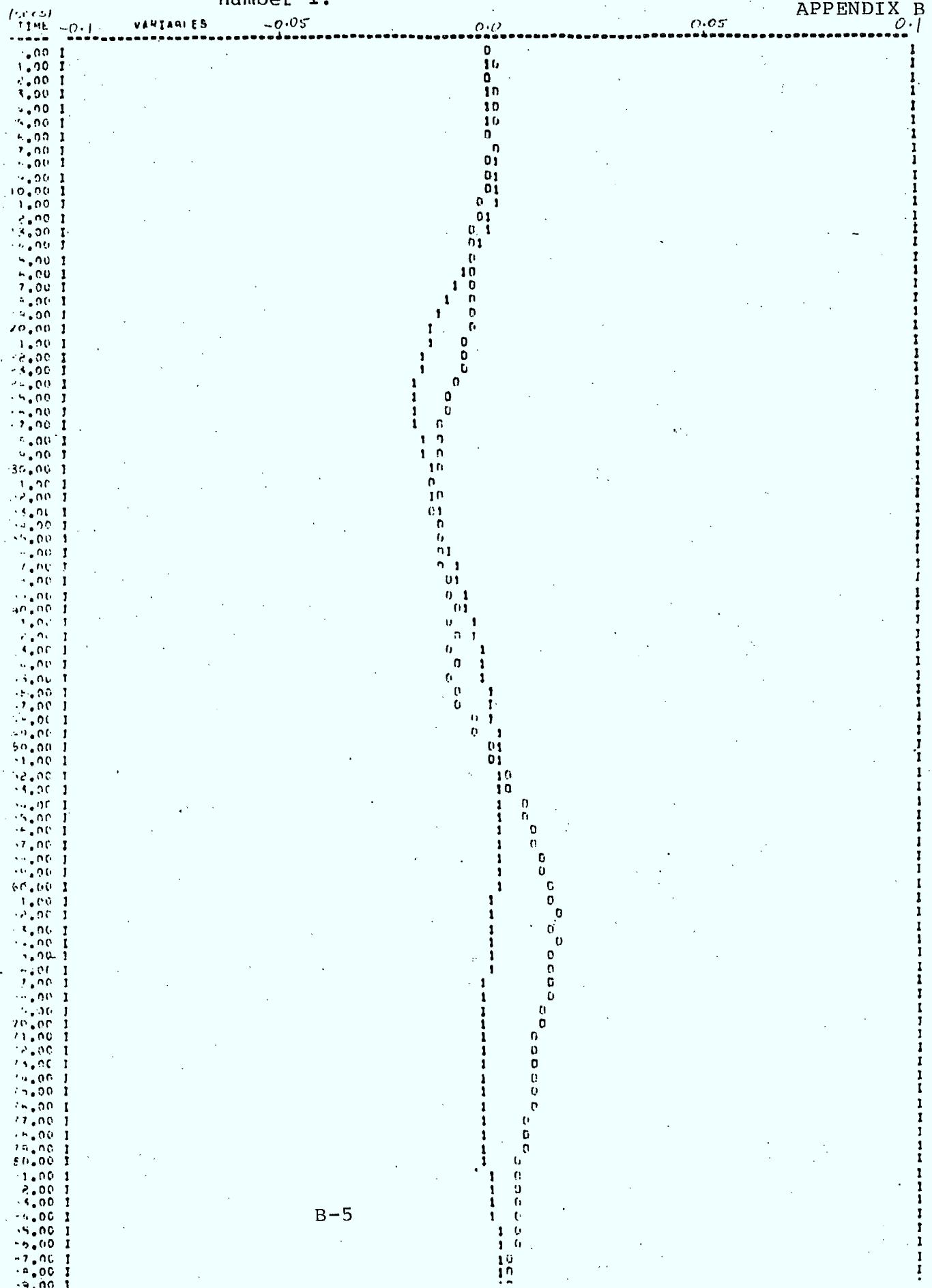


corresponding observer output (symbol 0). Run SPAR-R.1135

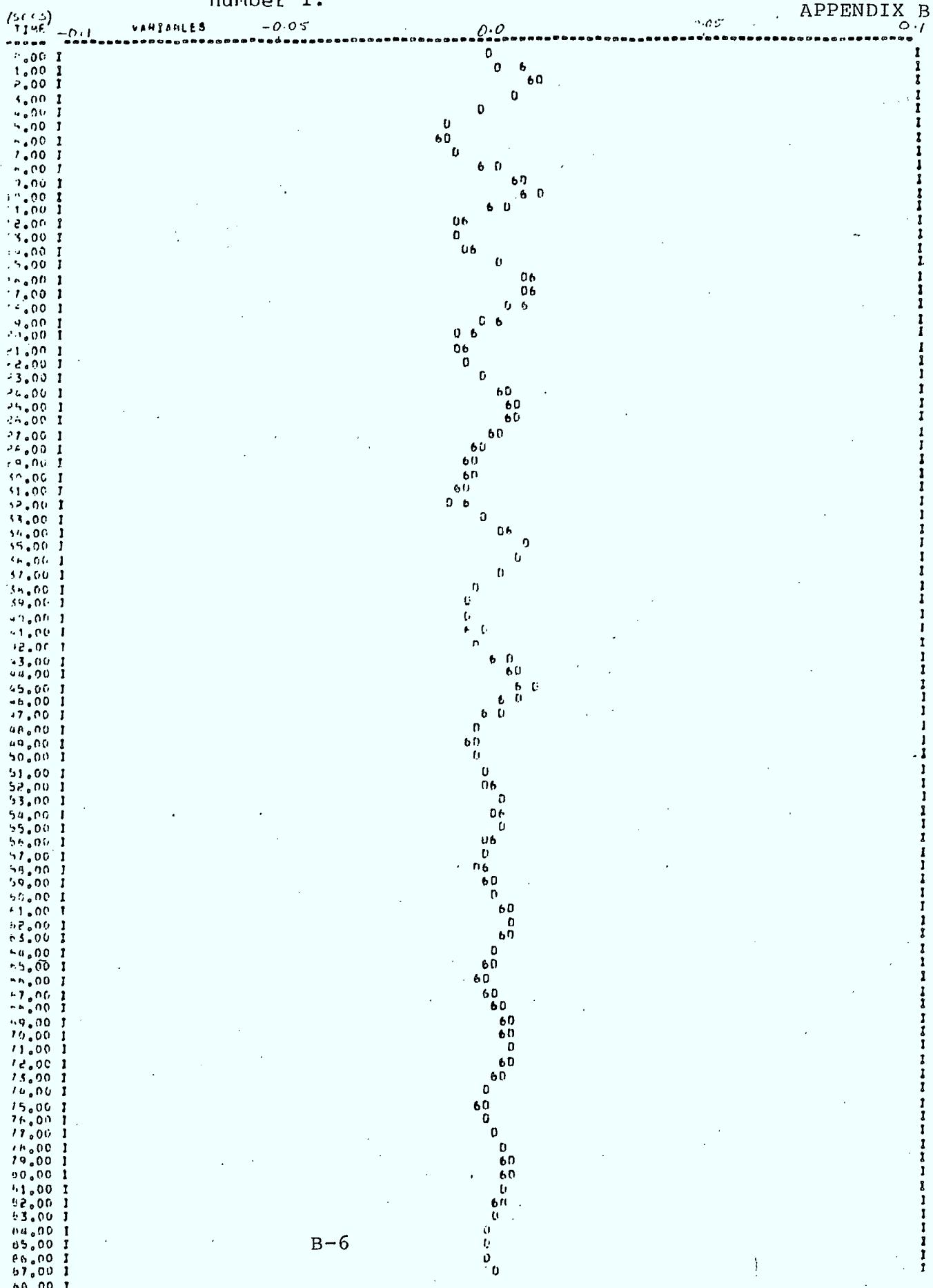
number 1.

ISSUE A  
APPENDIX B

0.1



Graph 4: Plot of flexible mode 6 (symbol 6) and corresponding observer output (symbol 0). Run SPAR-R.1135  
ISSUE A  
APPENDIX B

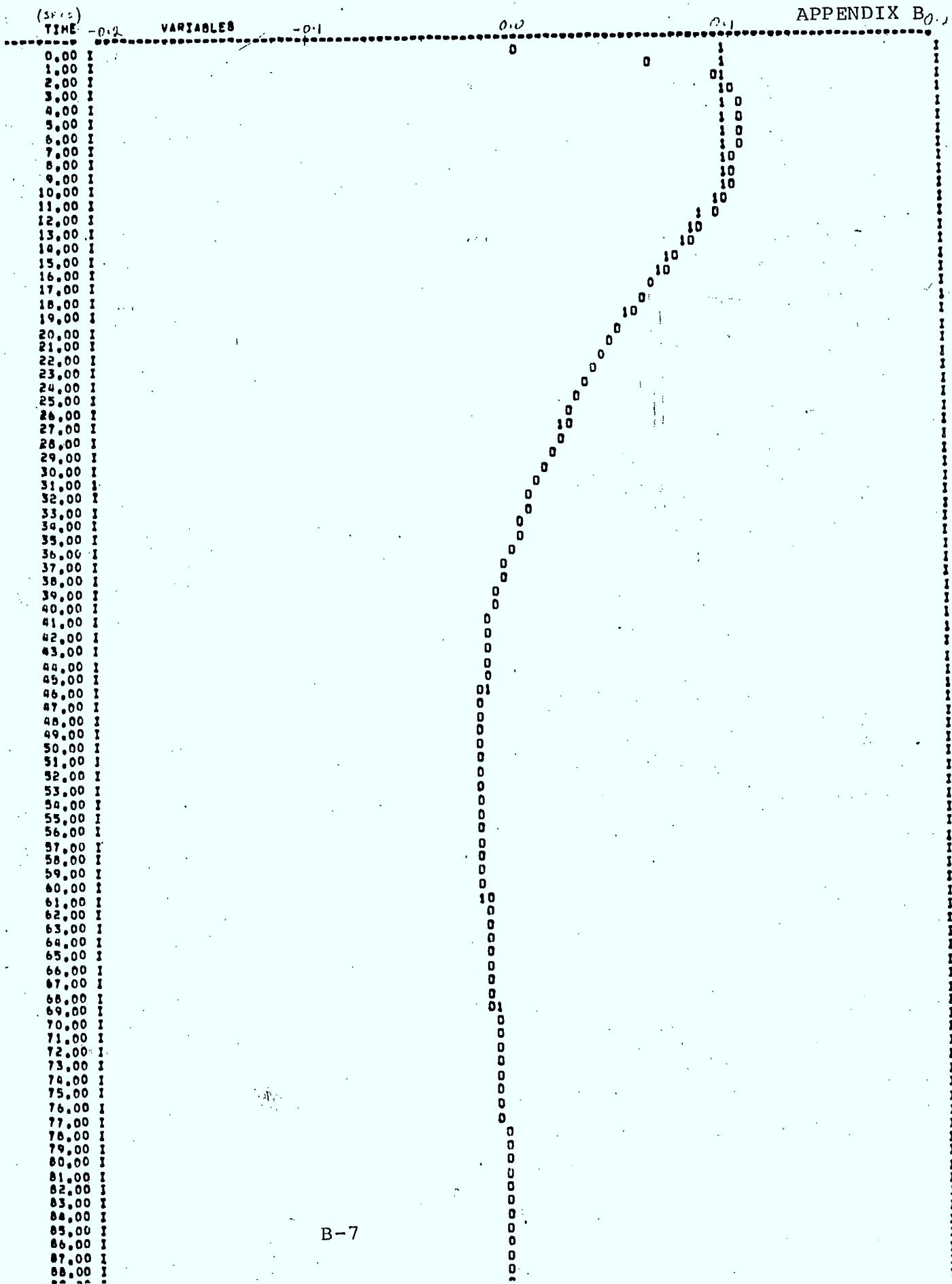


Graph 5: Plot of rigid mode 1 (symbol 1) and corresponding  
observer output (symbol 0). Run number 3.

SPAR-R.1135

ISSUE A

APPENDIX B



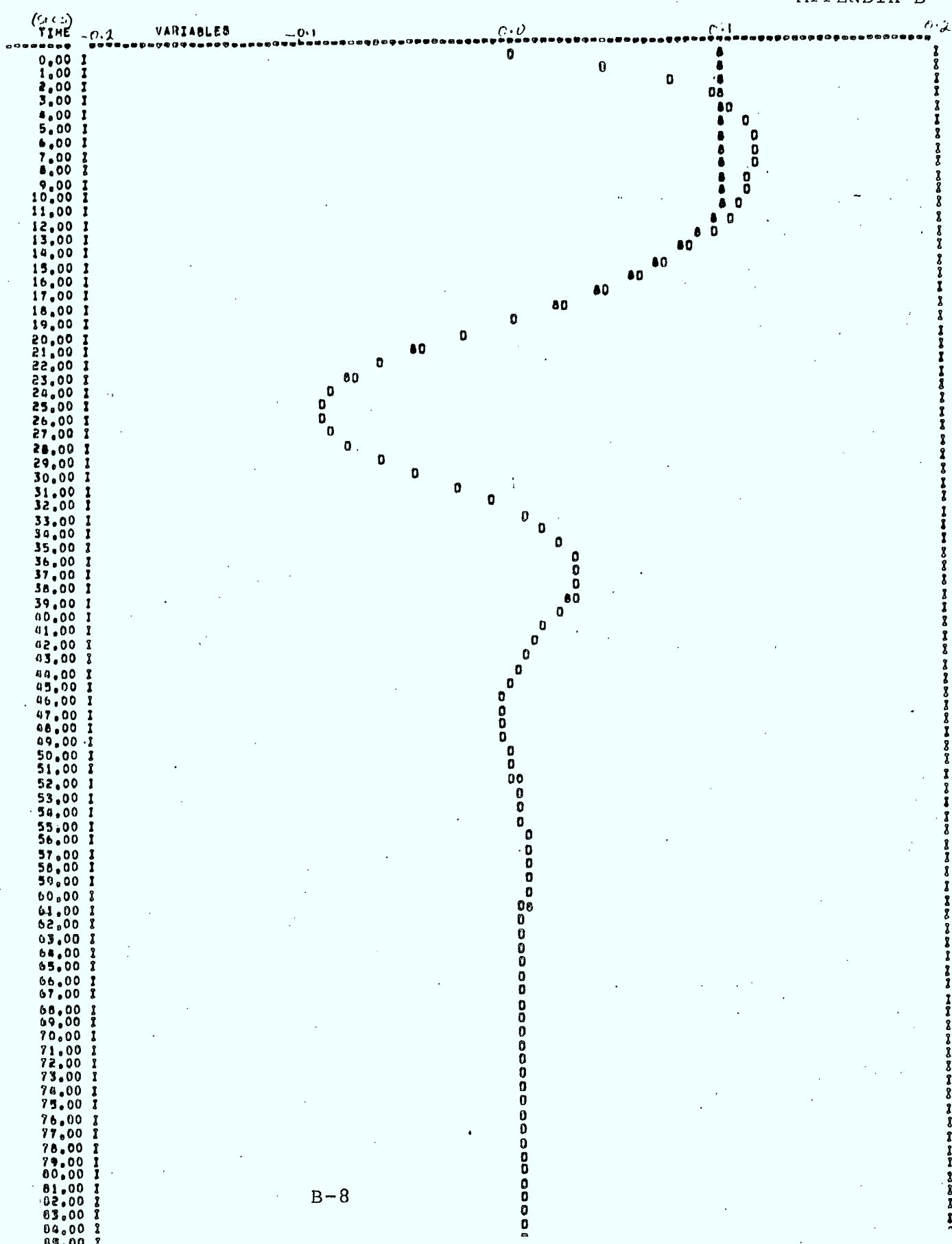
Graph 6: Plot of rigid mode 8 (symbol 8) and corresponding

observer output (symbol 0). Run number 3.

SPAR-R.1135

ISSUE A

APPENDIX B



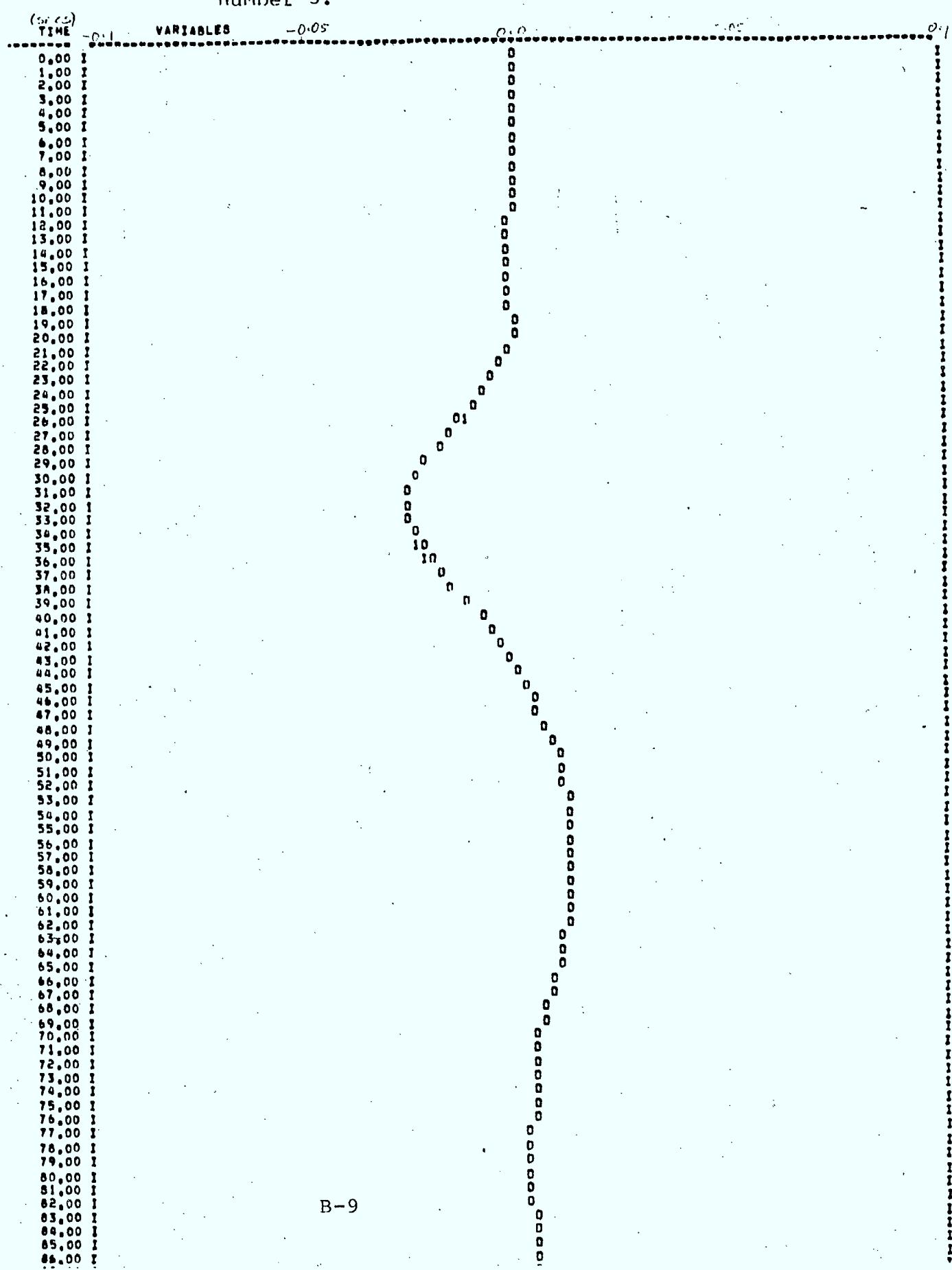
Graph 1: Plot of flexible mode 1 (symbol 1) and

corresponding observer output (symbol 0).

RSPAR-R.1135

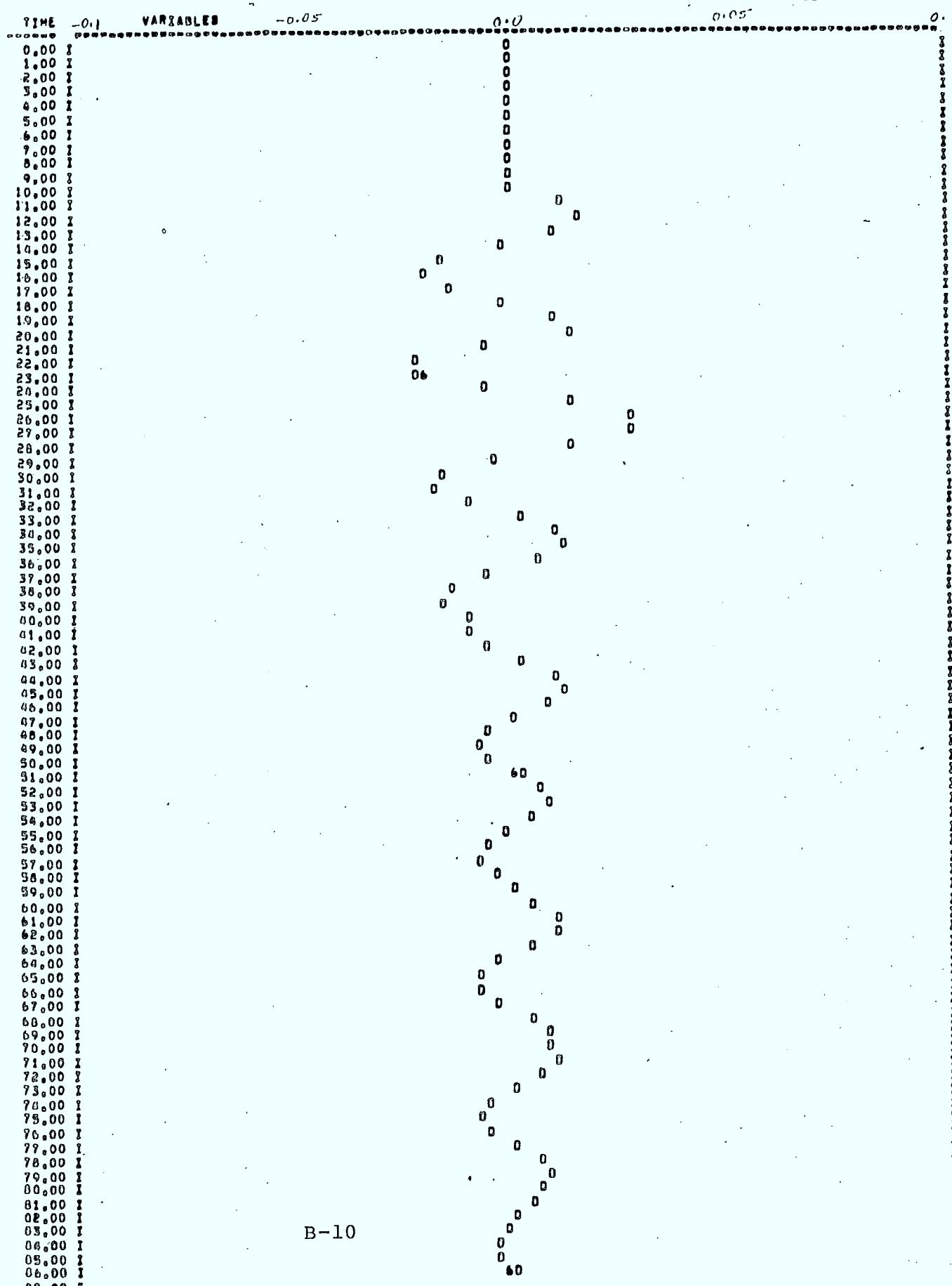
ISSUE A

APPENDIX B



Graph 8: Plot of flexible mode 6 (symbol 6) and  
corresponding observer output (symbol 0). RSPAR-R.1135  
number 3.

ISSUE A  
APPENDIX B

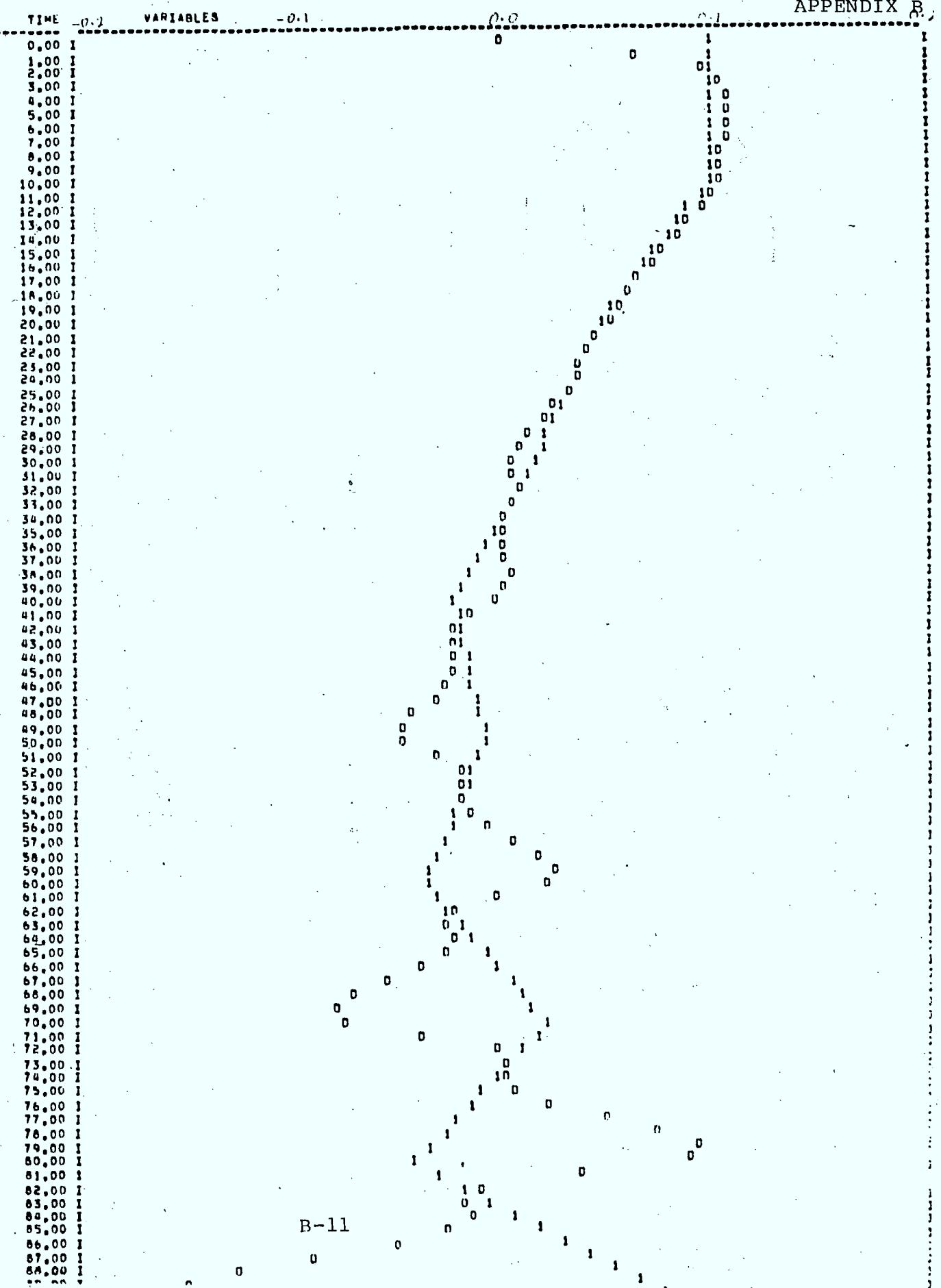


Graph 9: Plot of rigid mode 1 (symbol 1) and corresponding  
observer output (symbol 0). Run number 6.

SPAR-R.1135

ISSUE A

APPENDIX B



## THE DECMATS OBSERVER

OBSERVER B<sub>0</sub> MATRIX

	0.	,	0.	,	0.	,	0.	,
Row 1	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
Row 2	0.	,	0.	,	0.	,	8.41000000E-03,	,
	8.41000000E-03,	,	0.	,	0.	,	-8.41000000E-03,	,
	8.41000000E-03,	,	0.	,	0.	,	-8.41000000E-03,	,
	-8.41000000E-03,	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
Row 3	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
Row 4	0.	,	0.	,	0.	,	-1.45700000E-02,	,
	1.45700000E-02,	,	0.	,	0.	,	-1.45700000E-02,	,
	-1.45700000E-02,	,	0.	,	0.	,	1.45700000E-02,	,
	-1.45700000E-02,	,	0.	,	0.	,	0.	,
Row 5	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
Row 6	0.	,	0.	,	0.	,	5.72900000E-03,	,
	-5.72900000E-03,	,	0.	,	0.	,	5.72900000E-03,	,
	-2.86900000E-02,	,	0.	,	0.	,	2.86900000E-02,	,
	-2.86900000E-02,	,	0.	,	0.	,	0.	,
Row 7	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
Row 8	0.	,	0.	,	0.	,	3.80600000E-03,	,
	3.81100000E-03,	,	0.	,	0.	,	-3.80600000E-03,	,
	-1.90600000E-02,	,	0.	,	0.	,	1.90600000E-02,	,
	1.90800000E-02,	,	0.	,	0.	,	0.	,
Row 9	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
Row 10	0.	,	0.	,	0.	,	1.02200000E-02,	,
	-7.07800000E-03,	,	0.	,	0.	,	-1.02700000E-02,	,
	9.29900000E-03,	,	0.	,	0.	,	-9.29900000E-03,	,
	-9.28400000E-03,	,	0.	,	0.	,	0.	,
Row 11	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
	0.	,	0.	,	0.	,	0.	,
Row 12	6.94400000E-03,	,	0.	,	0.	,	1.28900000E-03,	,
	4.53900000E-05,	,	0.	,	0.	,	-1.29300000E-03,	,
	-2.67500000E-04,	,	0.	,	0.	,	2.67500000E-04,	,
	2.70200000E-04,	,	0.	,	0.	,	0.	,

out of D value

SPAR-R.1135  
ISSUE A  
APPENDIX A

(117)

DECIMATS OBSERVER B<sub>0</sub> MATRIX CONTINUED.

	0.	0.	0.	0.	0.	0.
Row 13	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	6.79100000E-08,	6.92100000E-03,	8.07500000E-04,			
	8.08100000E-04,	8.09400000E-04,	8.10000000E-04,			
Row 14	9.73500000E-04,	9.72500000E-04,	9.73500000E-04,			
	9.72500000E-04,					
	0.	0.	0.	0.	0.	0.
Row 15	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
Row 16	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
Row 17	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
Row 18	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	0.	0.	0.	0.	0.	0.
	-5.55300000E-04,	-2.19900000E-06,	6.53000000E-03,			
Row 19	-3.47800000E-03,	3.48800000E-03,	-6.53900000E-03,			
	-4.82600000E-02,	-4.78300000E-02,	4.82600000E-02,			
	4.78300000E-02,					
	-1.50000000E-06,	-2.57700000E-03,	-3.36800000E-03,			
Row 20	3.37000000E-03,	3.41900000E-03,	-3.42100000E-03,			
	-1.08700000E-01,	1.08500000E-01,	1.08700000E-01,			
	-1.08500000E-01,					
	8.51800000E-04,	1.73100000E-06,	-2.48500000E-02,			
Row 21	2.63900000E-02,	-2.65300000E-02,	2.49900000E-02,			
	5.09500000E-02,	5.07900000E-02,	-5.09500000E-02,			
	-5.07900000E-02,					
	4.83800000E-03,	-2.42200000E-06,	6.47500000E-04,			
Row 22	-3.01400000E-03,	3.02500000E-03,	6.35800000E-04,			
	-2.41800000E-01,	-2.41300000E-01,	2.41800000E-01,			
	2.41300000E-01,					

8/4/mc1702/62

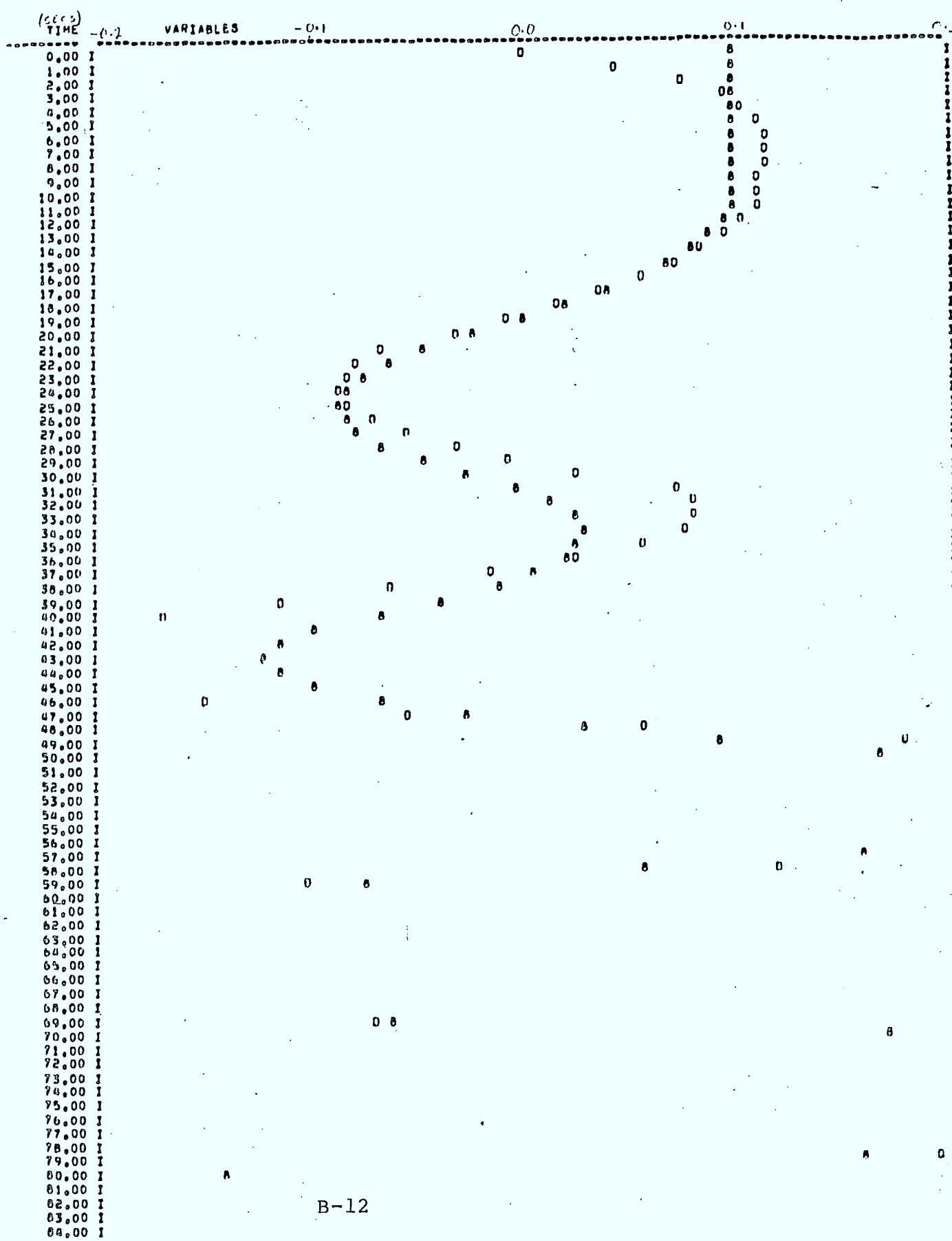
SPAR-R.1135  
ISSUE A  
APPENDIX B

APPENDIX B

GRAPHS OF MODAL COORDINATES VERSUS TIME

Graph 10: Plot of rigid mode 8 (symbol 8) and corresponding  
observer output (symbol 0). Run number 6.

SPAR-R.1135  
ISSUE A  
APPENDIX B



corresponding observer output (symbol 0).  
number 6.

Run  
SPAR-R.1135  
ISSUE A  
APPENDIX B

TIME	-0.1	VARIABLES	0.0	0.1
0.00	I			
1.00	I			
2.00	I			
3.00	I			
4.00	I			
5.00	I			
6.00	I			
7.00	I			
8.00	I			
9.00	I			
10.00	I			
11.00	I			
12.00	I			
13.00	I			
14.00	I			
15.00	I			
16.00	I			
17.00	I			
18.00	I			
19.00	I			
20.00	I			
21.00	I			
22.00	I			
23.00	I			
24.00	I			
25.00	I			
26.00	I			
27.00	I			
28.00	I			
29.00	I			
30.00	I			
31.00	I			
32.00	I			
33.00	I			
34.00	I			
35.00	I			
36.00	I			
37.00	I			
38.00	I			
39.00	I			
40.00	I			
41.00	I			
42.00	I			
43.00	I			
44.00	I			
45.00	I			
46.00	I			
47.00	I			
48.00	I			
49.00	I			
50.00	I			
51.00	I			
52.00	I			
53.00	I			
54.00	I			
55.00	I			
56.00	I			
57.00	I			
58.00	I			
59.00	I			
60.00	I			
61.00	I			
62.00	I			
63.00	I			
64.00	I			
65.00	I			
66.00	I	0	0	
67.00	I		1	
68.00	I	1		
69.00	I			
70.00	I			
71.00	I			
72.00	I	1		
73.00	I		0	
74.00	I			
75.00	I			
76.00	I			
77.00	I			
78.00	I			
79.00	I			
80.00	I			
81.00	I			
82.00	I			
83.00	I			
84.00	I			
85.00	I			
86.00	I	1		
87.00	I	1		

Graph 12: Plot of flexible mode 6 (symbol 6) and

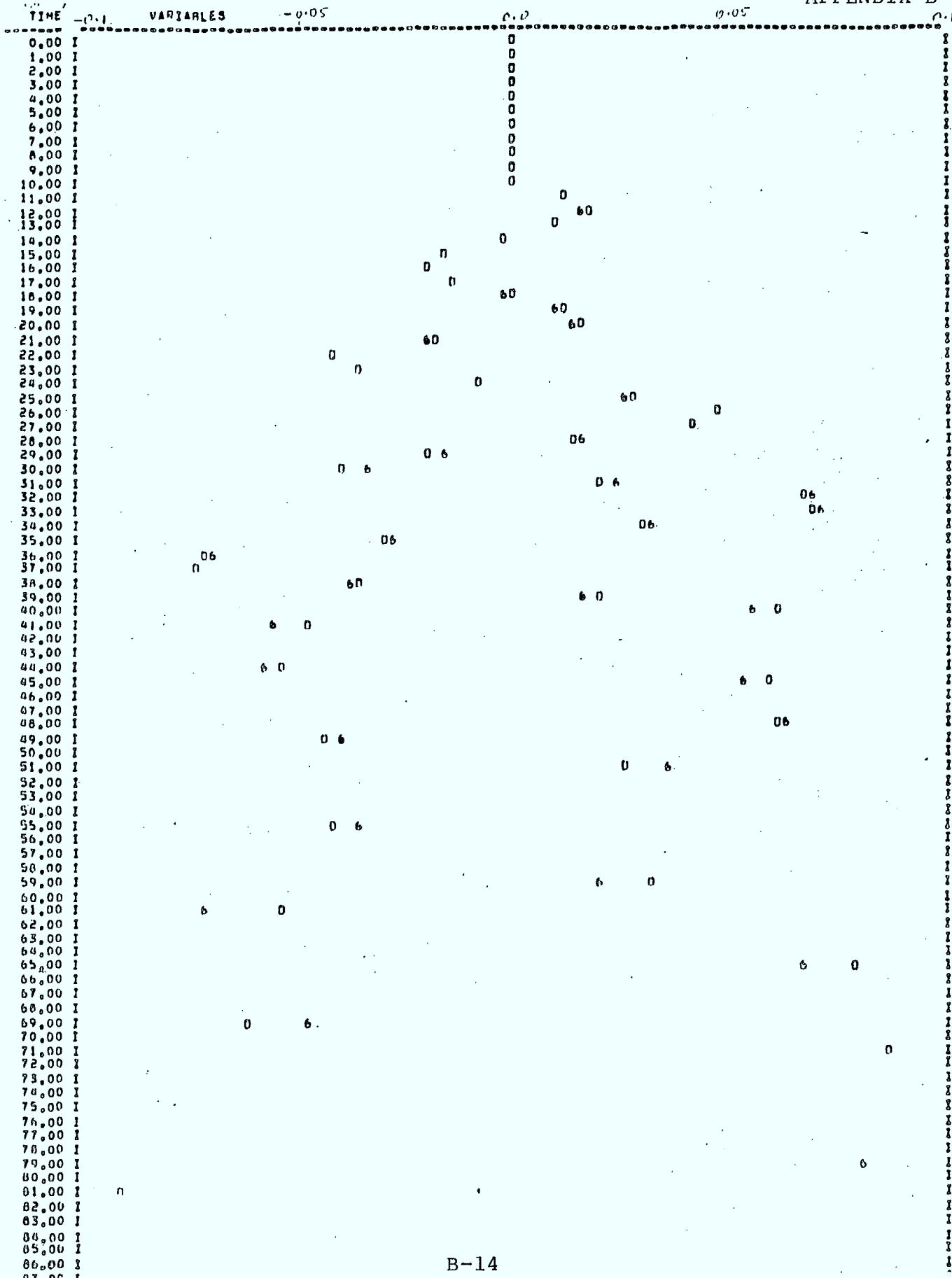
corresponding observer output (symbol 0). Run

SPAR-R.1135

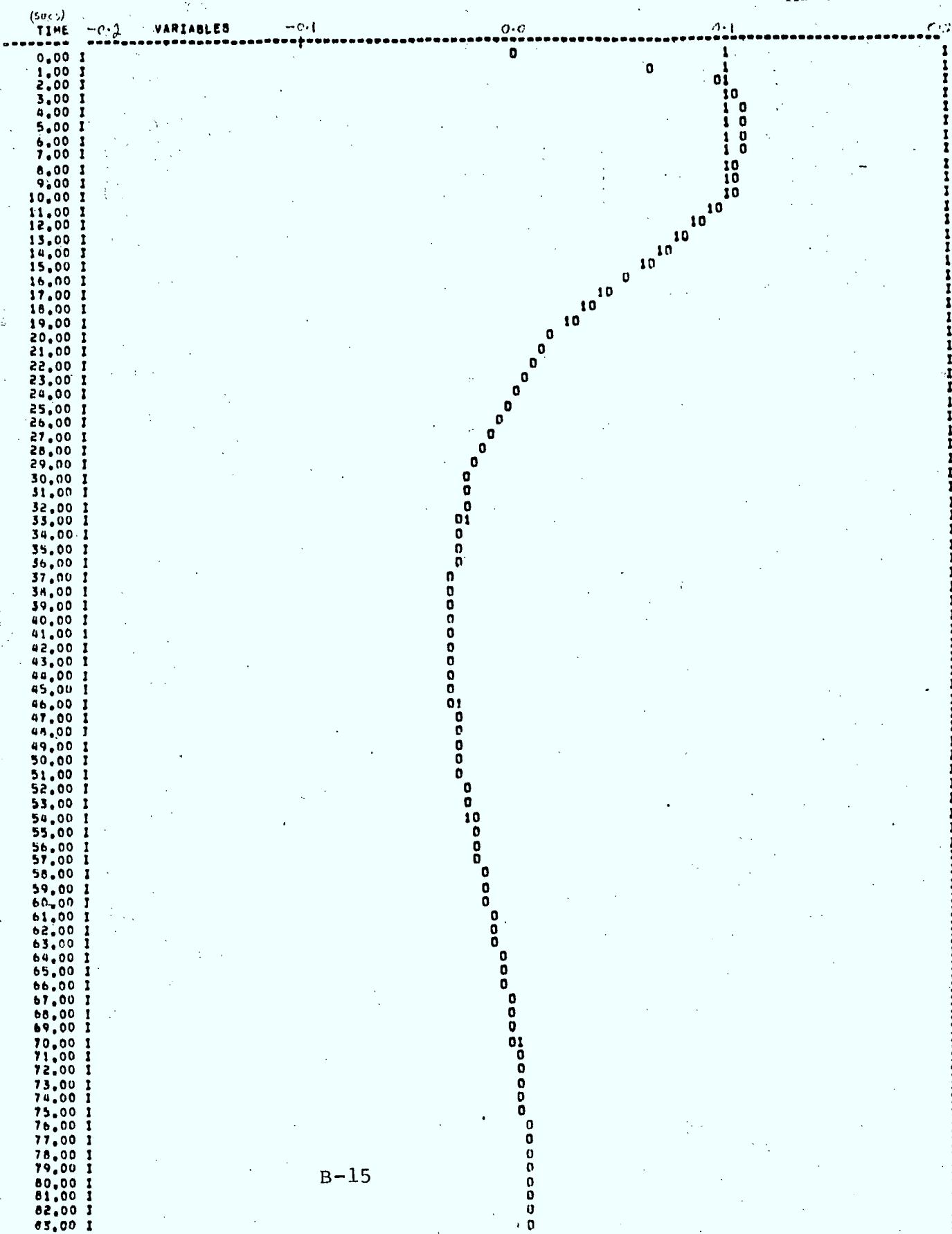
ISSUE A

APPENDIX B

number 6.

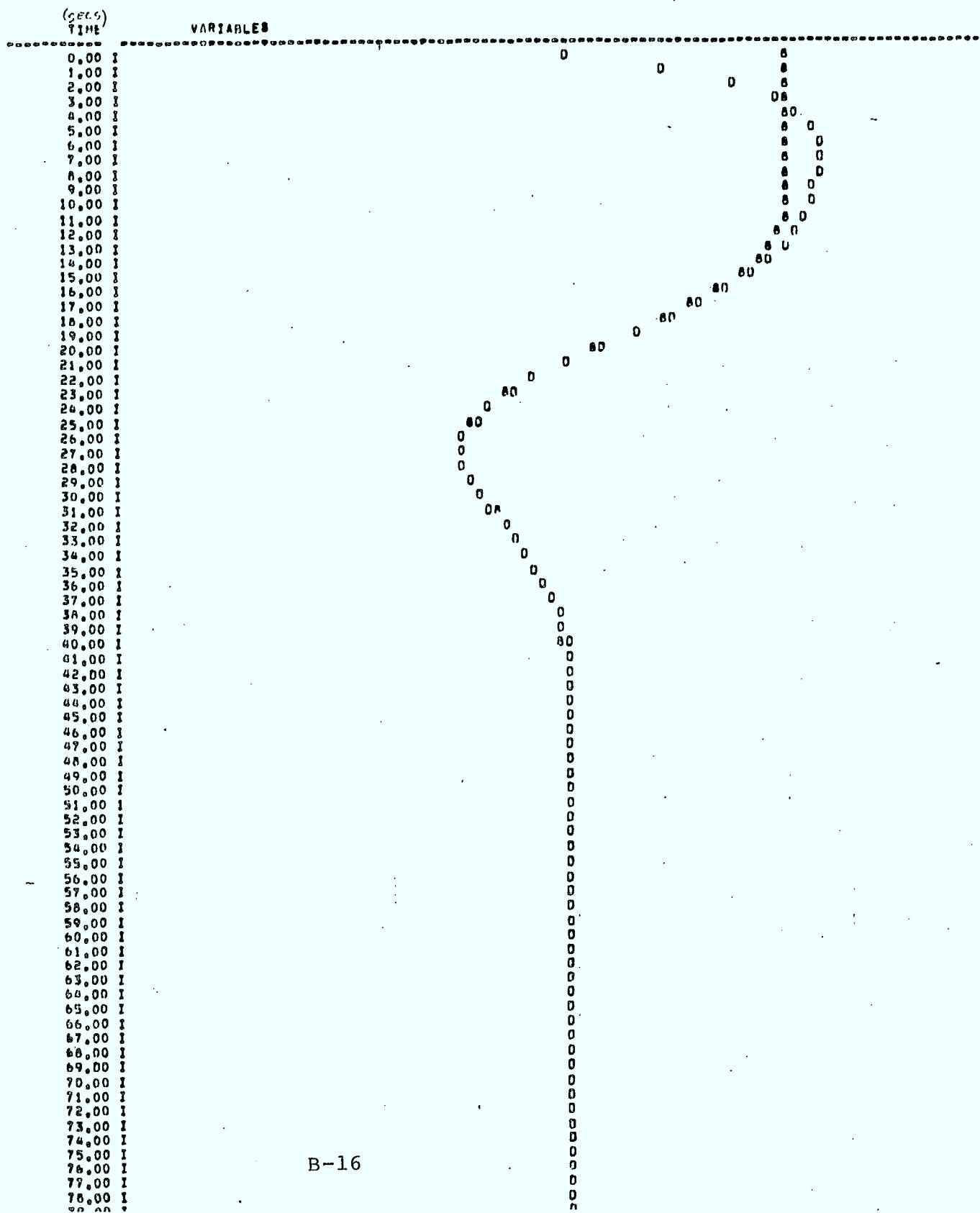


Graph 13: Plot of rigid mode 1 (symbol 1) and corresponding  
 observer output (symbol 0). Run number 8. SPAR-R.1135  
 ISSUE A  
 APPENDIX B



Graph 14: Plot of rigid mode 8 (symbol 8) and corresponding  
observer output (symbol 0). Run number 8.

SPAR-R.1135  
ISSUE A  
APPENDIX B

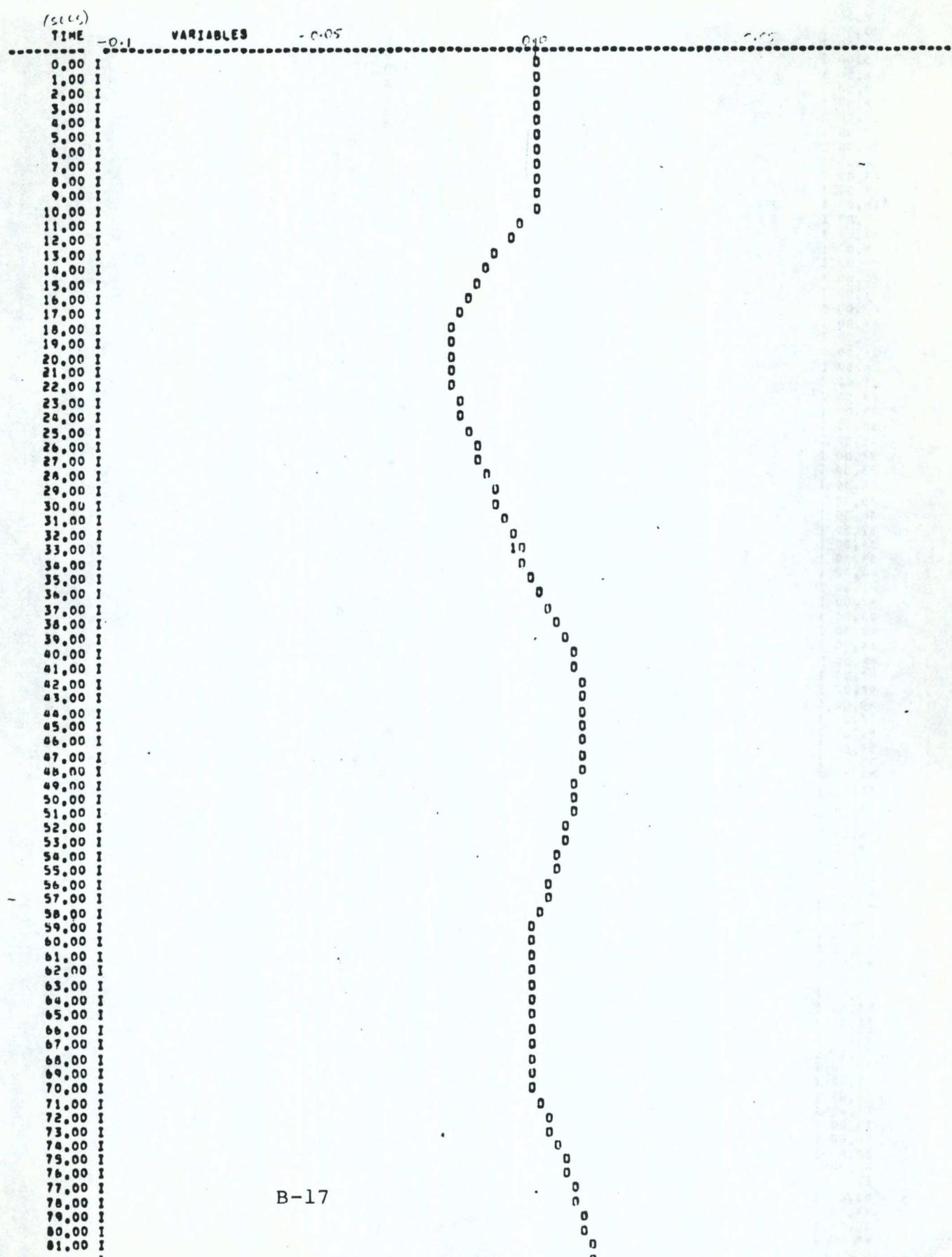


Graph 15: Plot of flexible mode 1 (symbol 1) and

corresponding observer output (symbol 0). Run

number 8.

SPAR-R.1135  
ISSUE A  
APPENDIX B

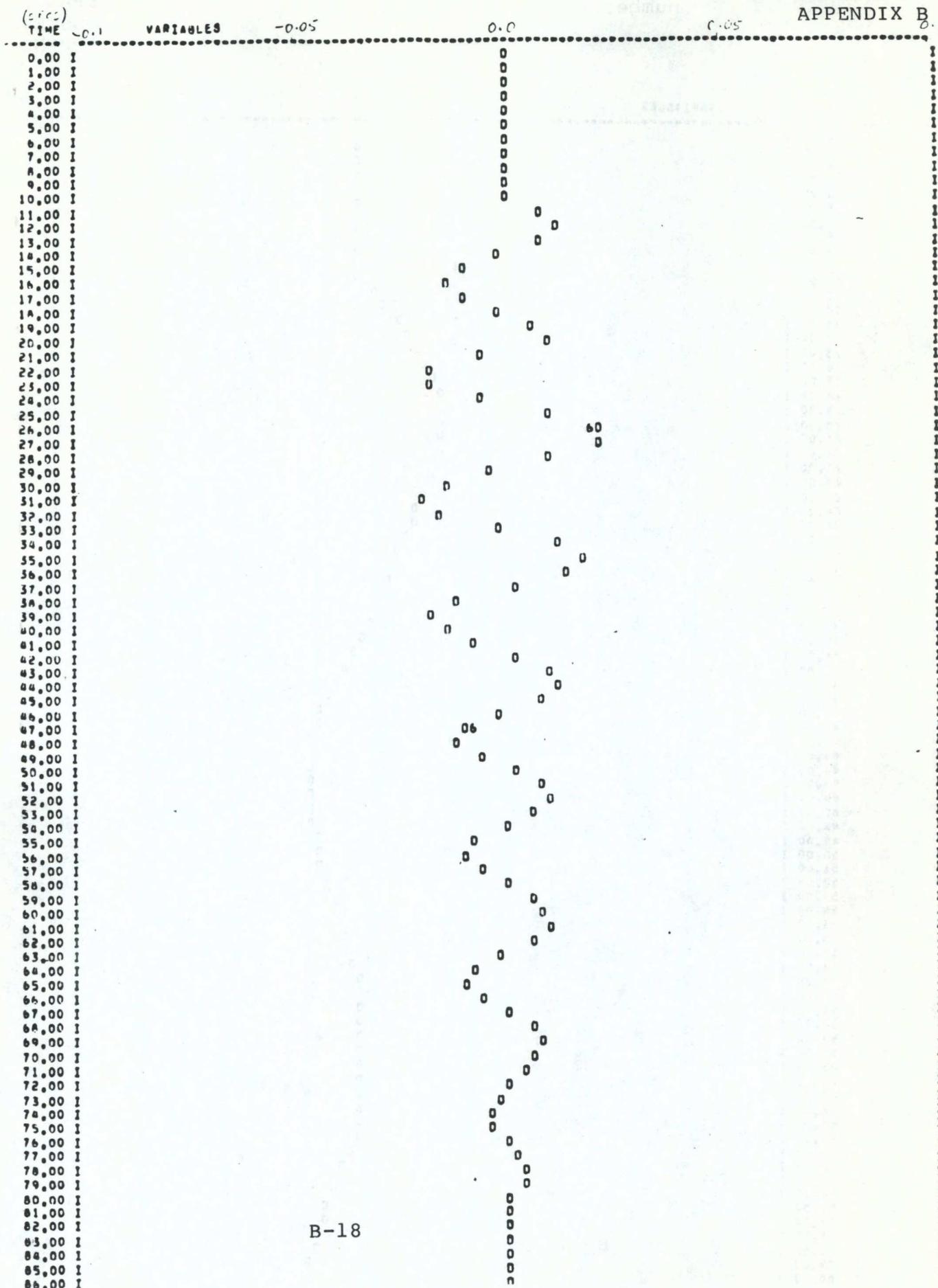


Graph 16: Plot of flexible mode 6 (symbol 6) and

corresponding observer output (symbol 0). Run

number 8.

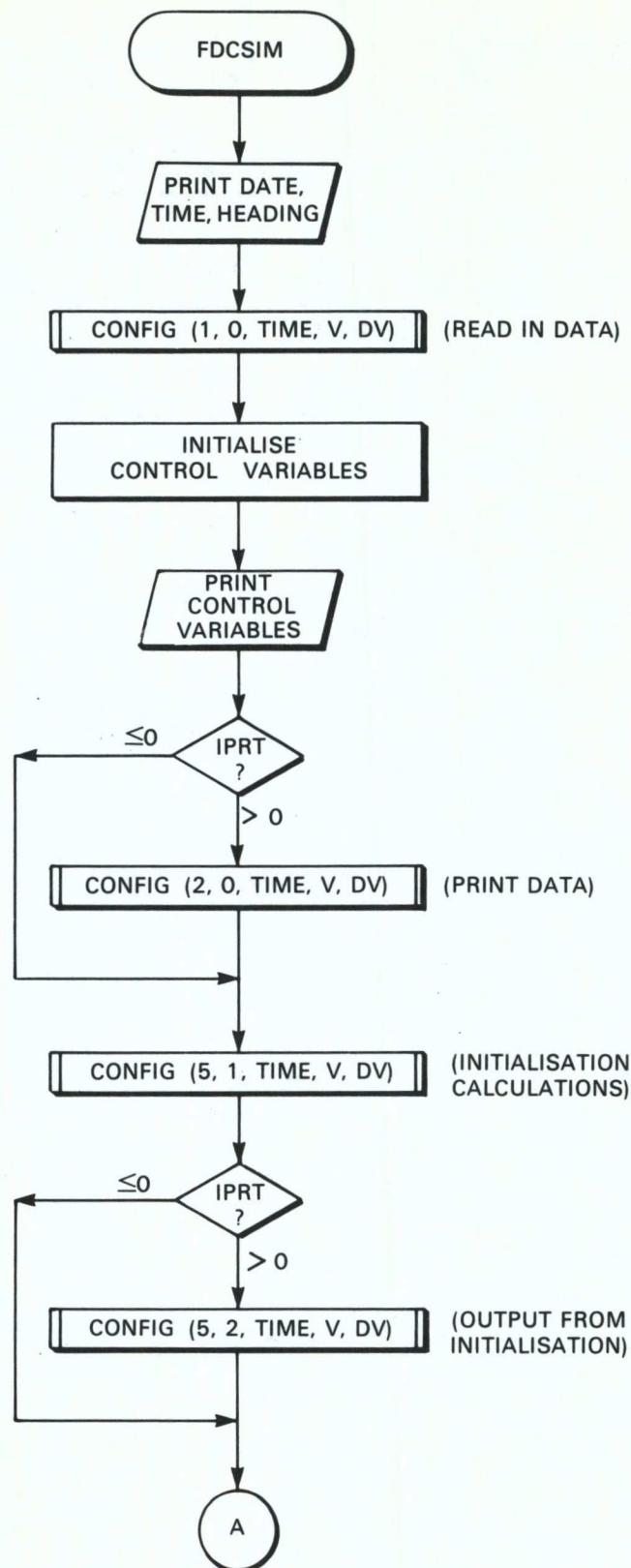
SPAR-R.1135  
ISSUE A  
APPENDIX B-1



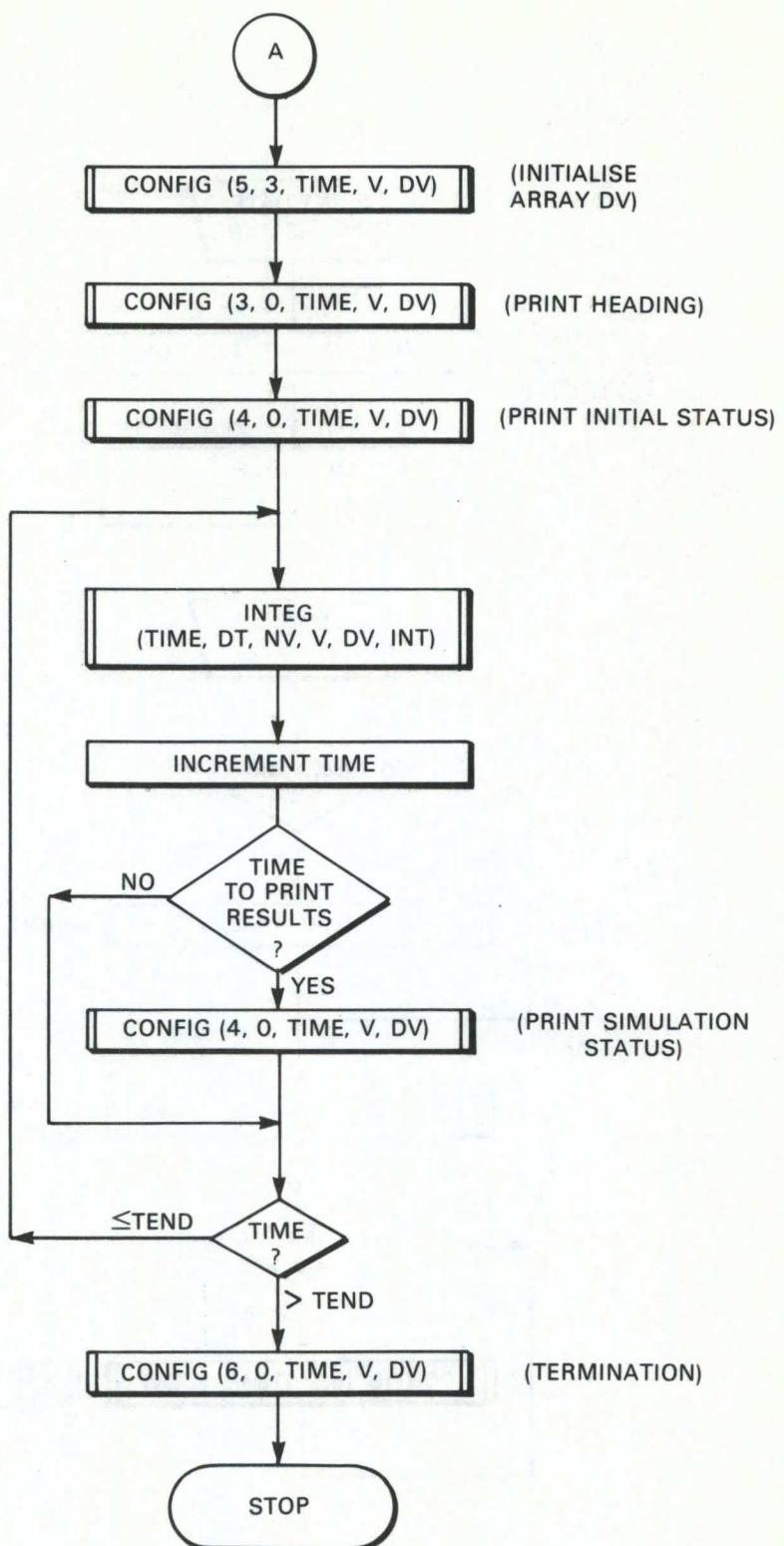
8/4/mcl702/65

SPAR-R.1135  
ISSUE B

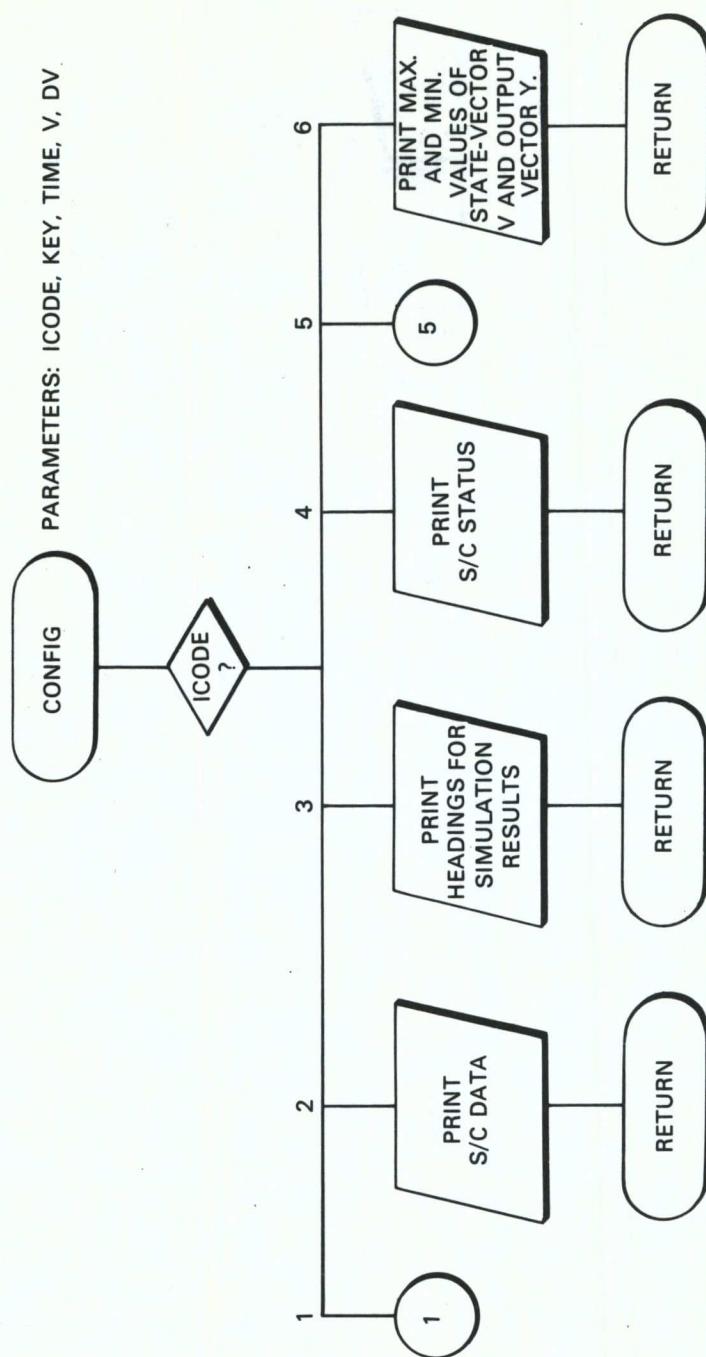
APPENDIX C  
PROGRAM FLOWCHARTS



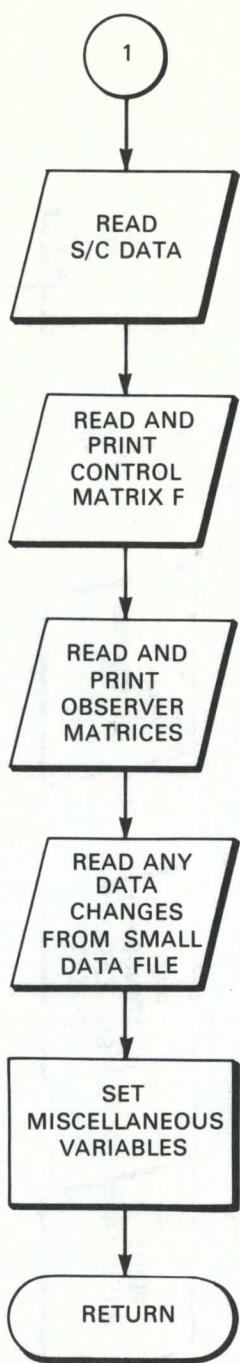
PROGRAM FDCSIM  
MAIN PROGRAM



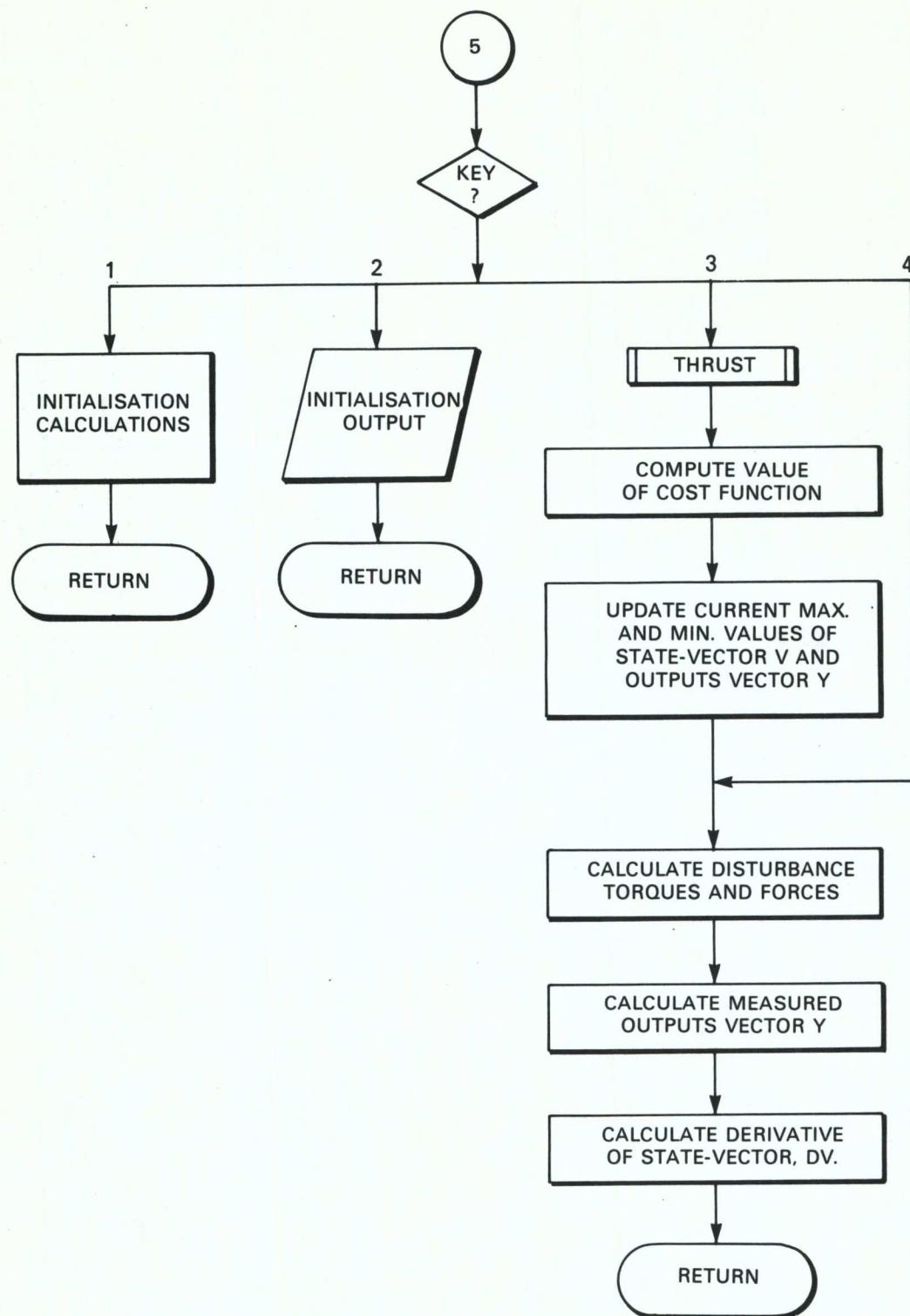
**PROGRAM FDCCSIM CONTINUED  
MAIN PROGRAM CONTINUED**



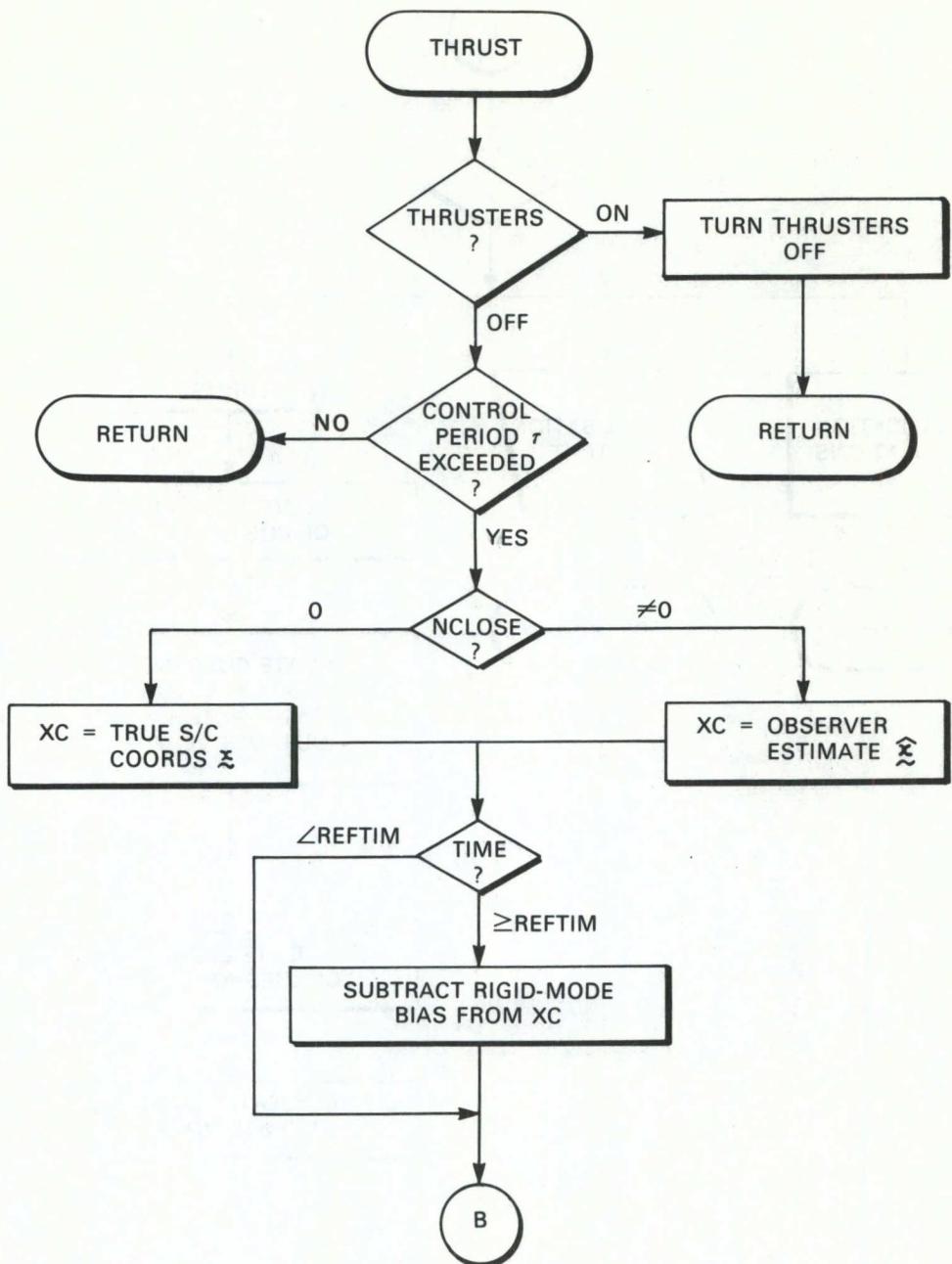
**PROGRAM FDCSIM CONTINUED  
SUBROUTINE CONFIG**



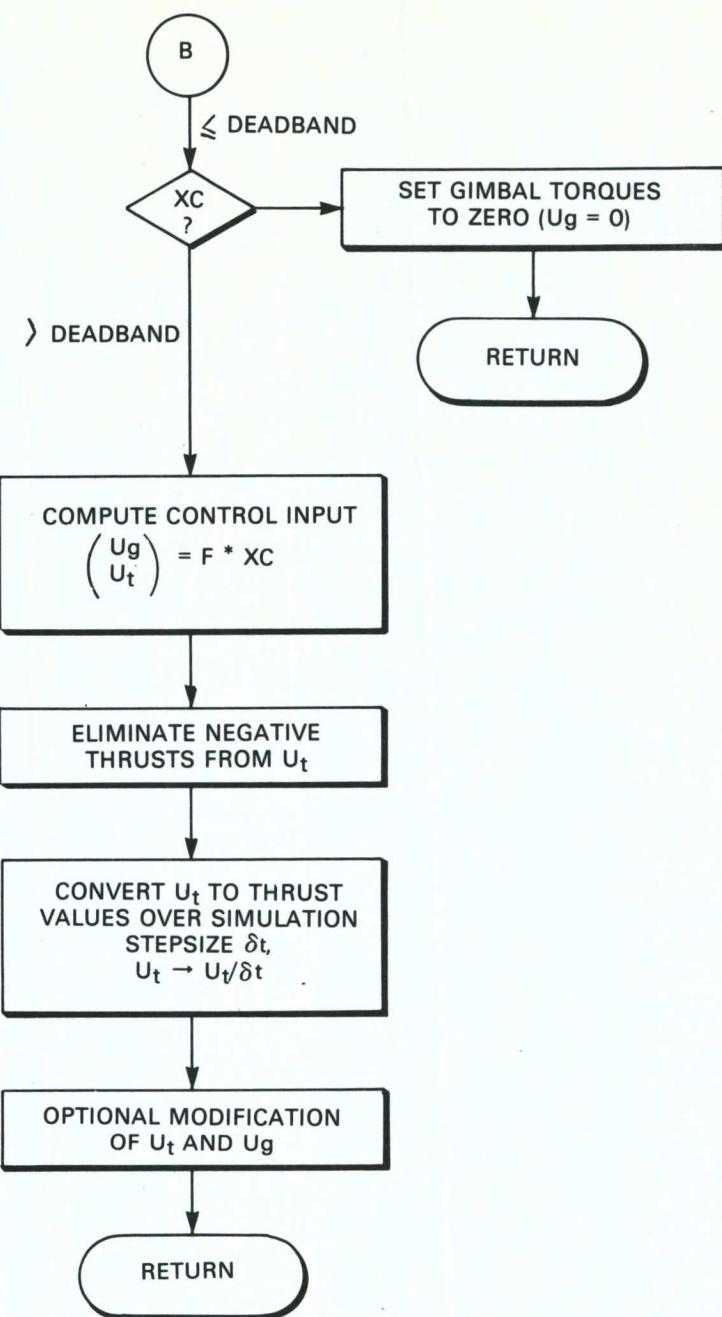
**PROGRAM FDCSIM  
SUBROUTINE CONFIG CONTINUED**



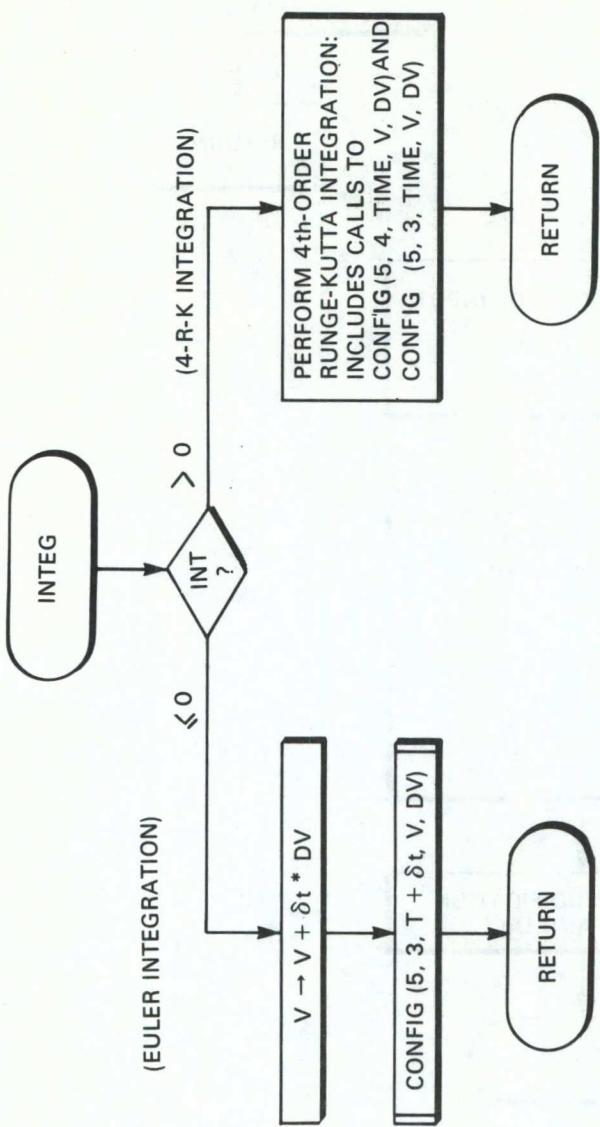
PROGRAM FDGSIM  
SUBROUTINE CONFIG CONTINUED



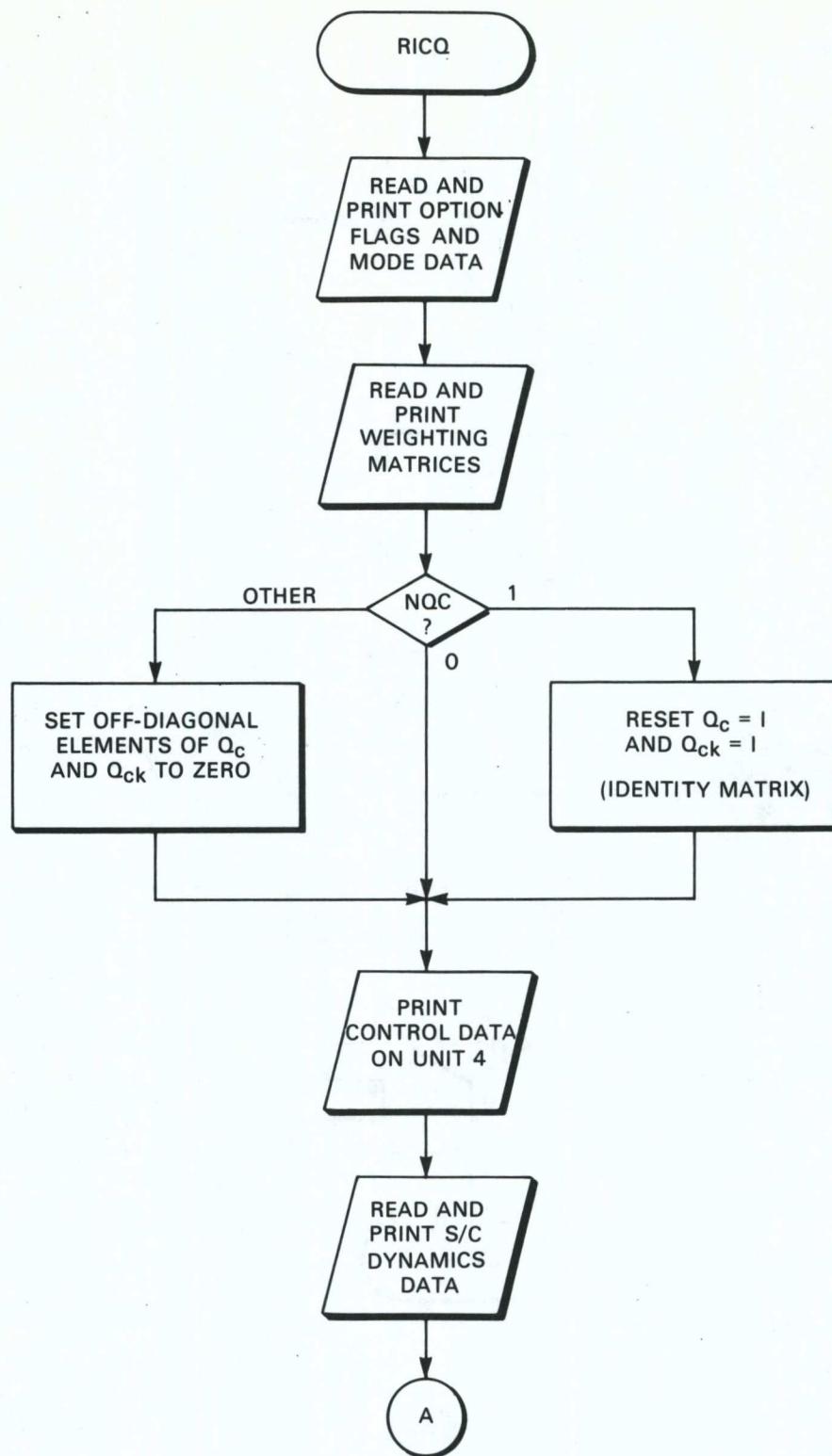
PROGRAM FDCSIM  
SUBROUTINE THRUST



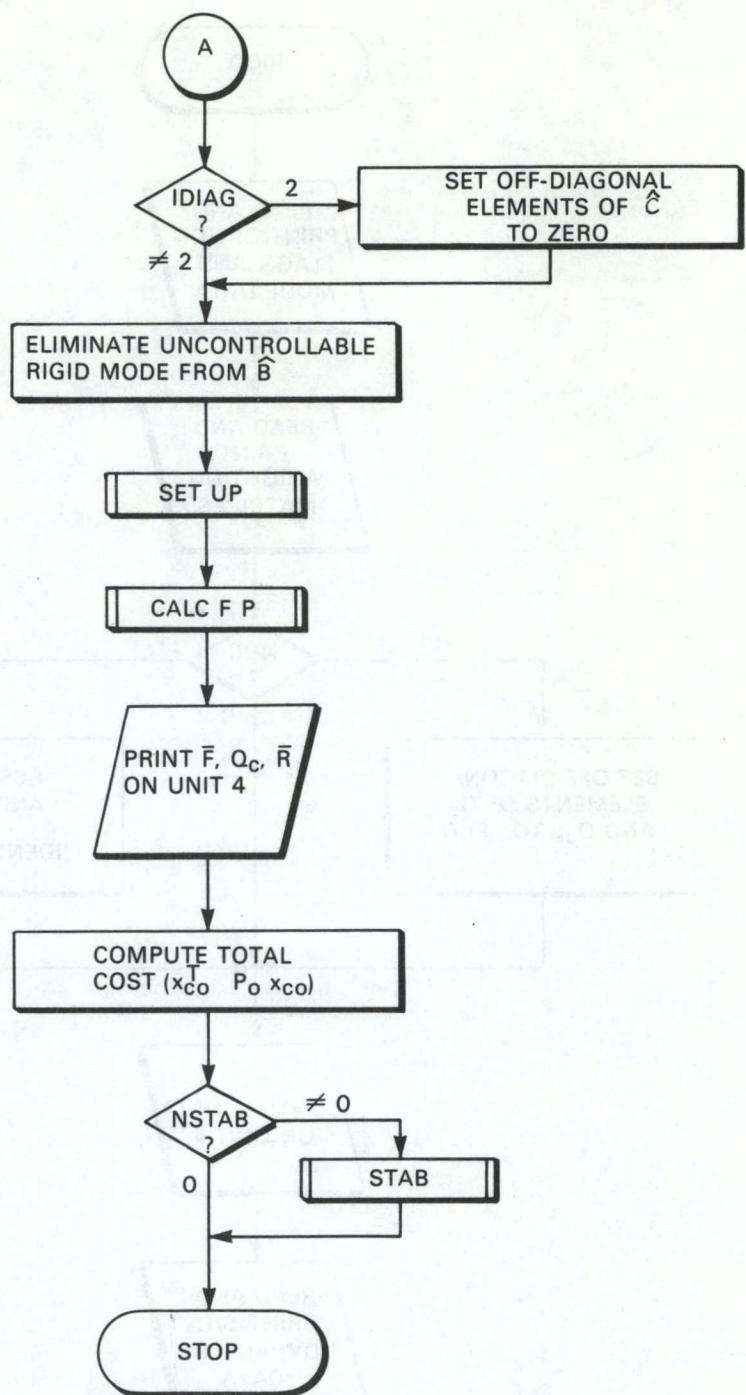
**PROGRAM FDCCSIM  
SUBROUTINE THRUST CONTINUED**



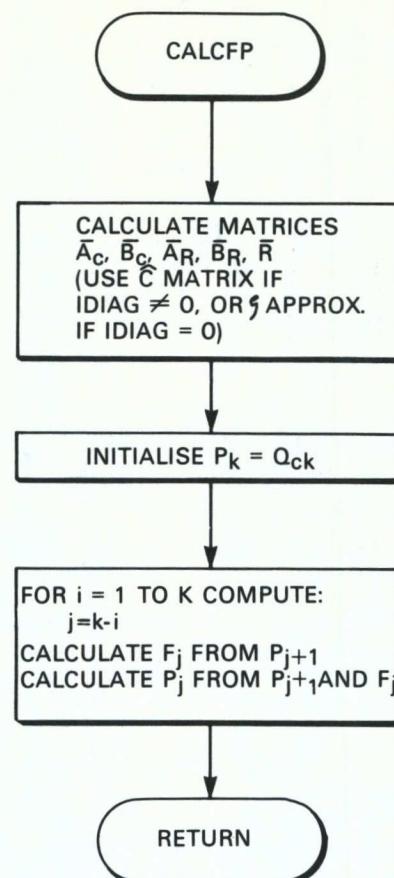
**PROGRAM FDCSIM  
SUBROUTINE INTEG**



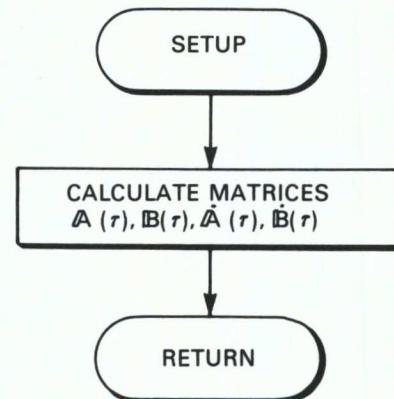
PROGRAM RICQ  
MAIN PROGRAM



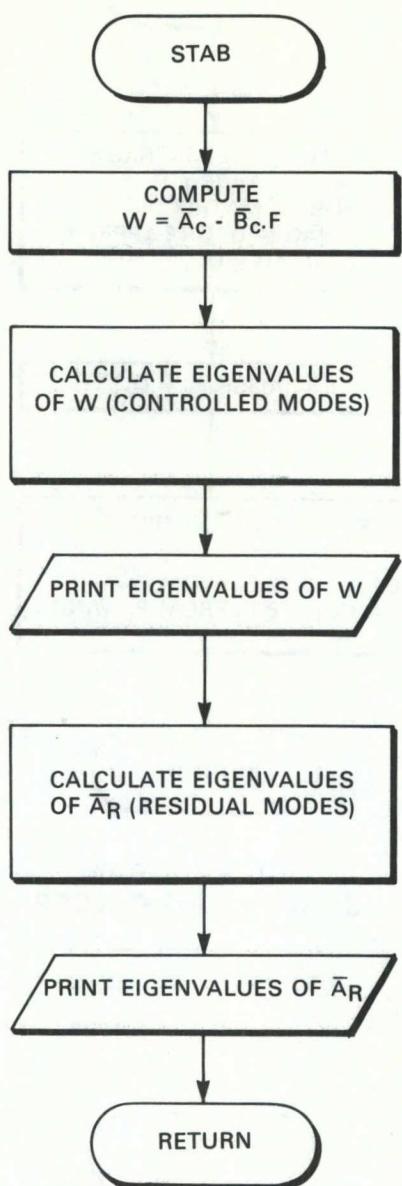
PROGRAM RICQ  
MAIN PROGRAM CONTINUED



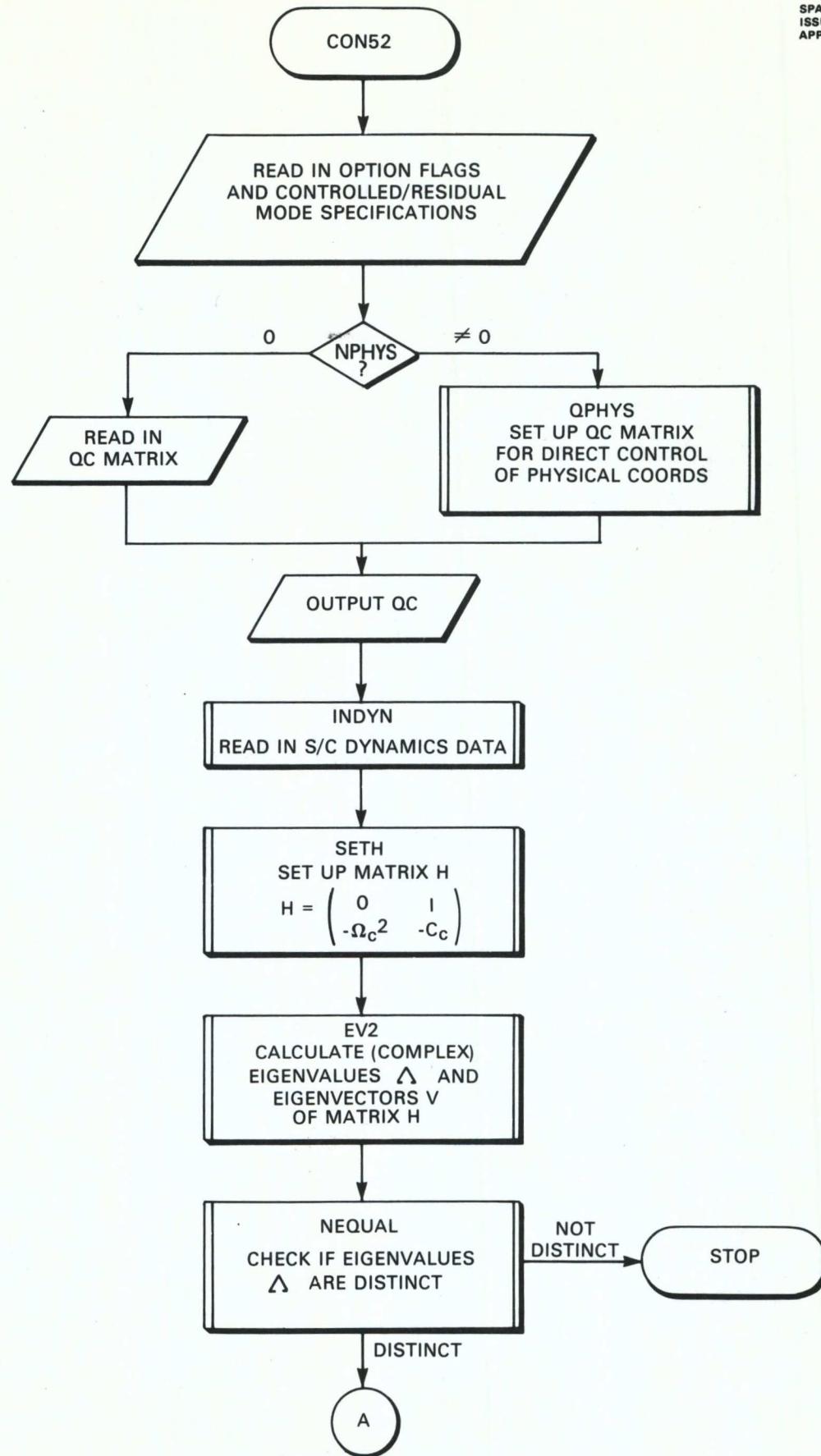
**RICQ PROGRAM  
SUBROUTING CALCFF**



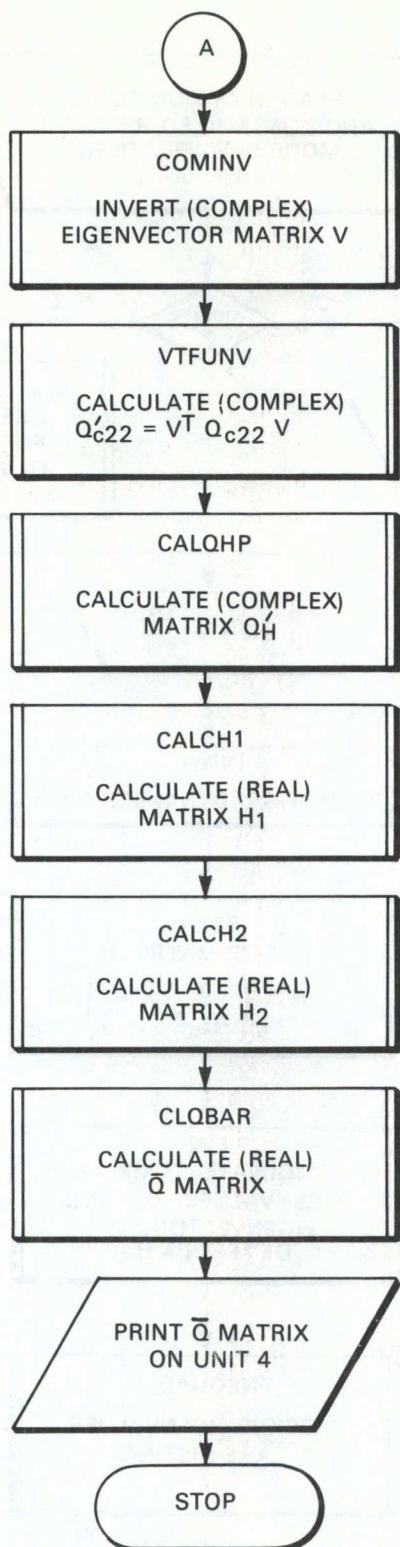
**SUBROUTINE SETUP**



PROGRAM RICQ  
SUBROUTINE STAB

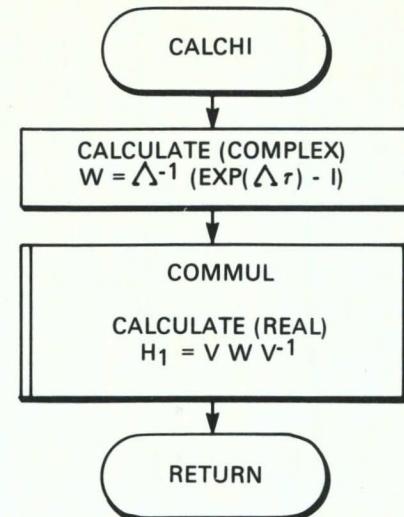


PROGRAM CON52  
MAIN PROGRAM

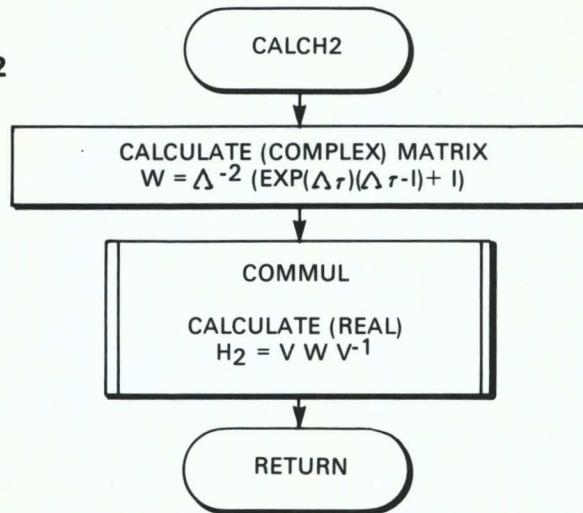


**PROGRAM CON52  
MAIN PROGRAM CONTINUED**

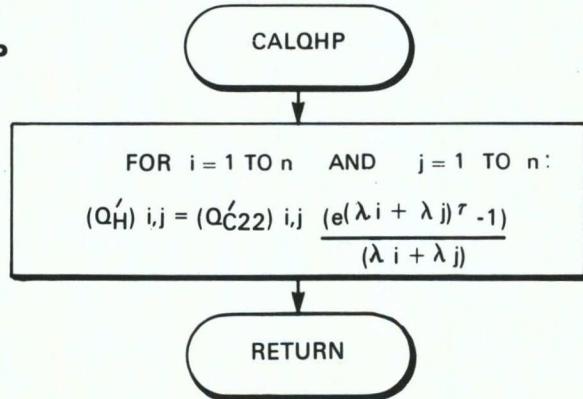
SUBROUTINE CALCH1



SUBROUTINE CALCH2



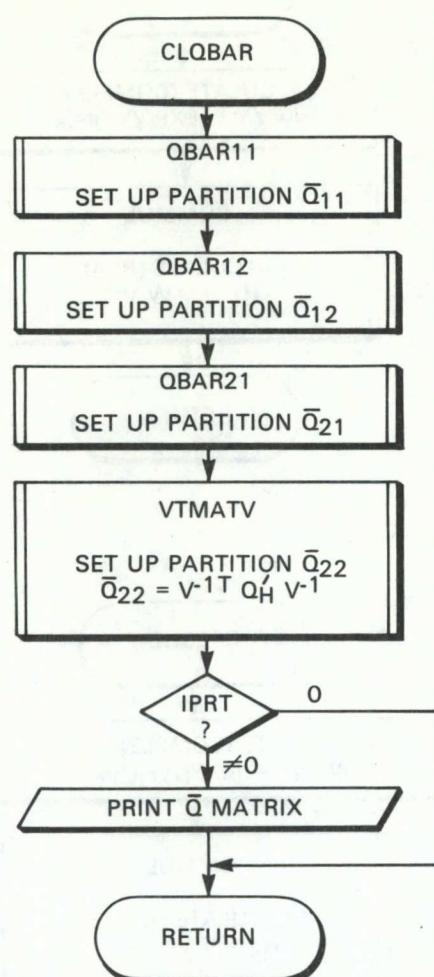
SUBROUTINE CALQHP



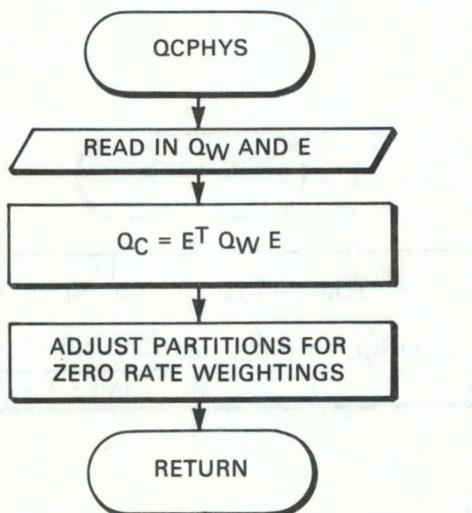
PROGRAM CON52

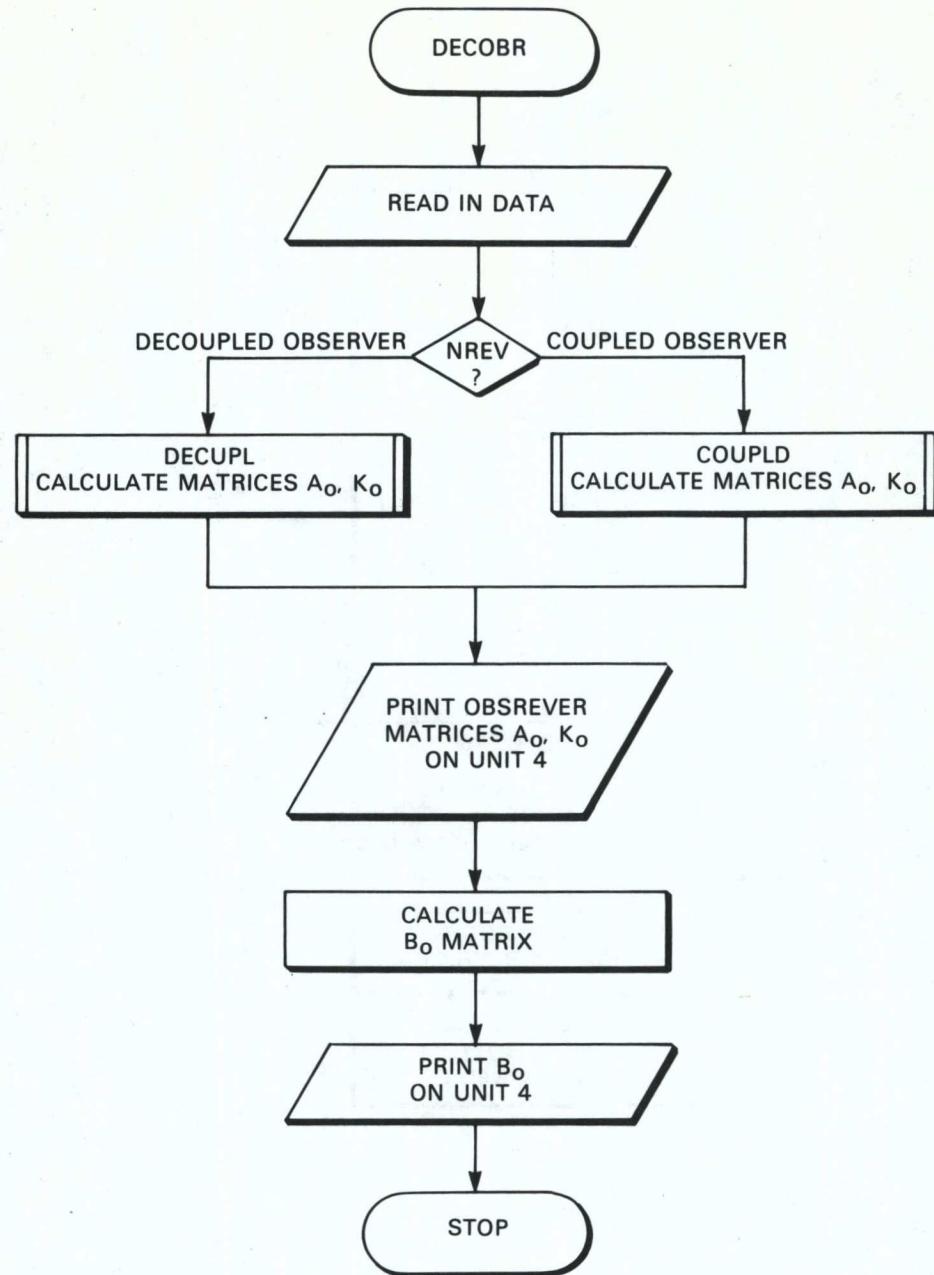
**PROGRAM CON52**

**SUBROUTINE CLQBAR**

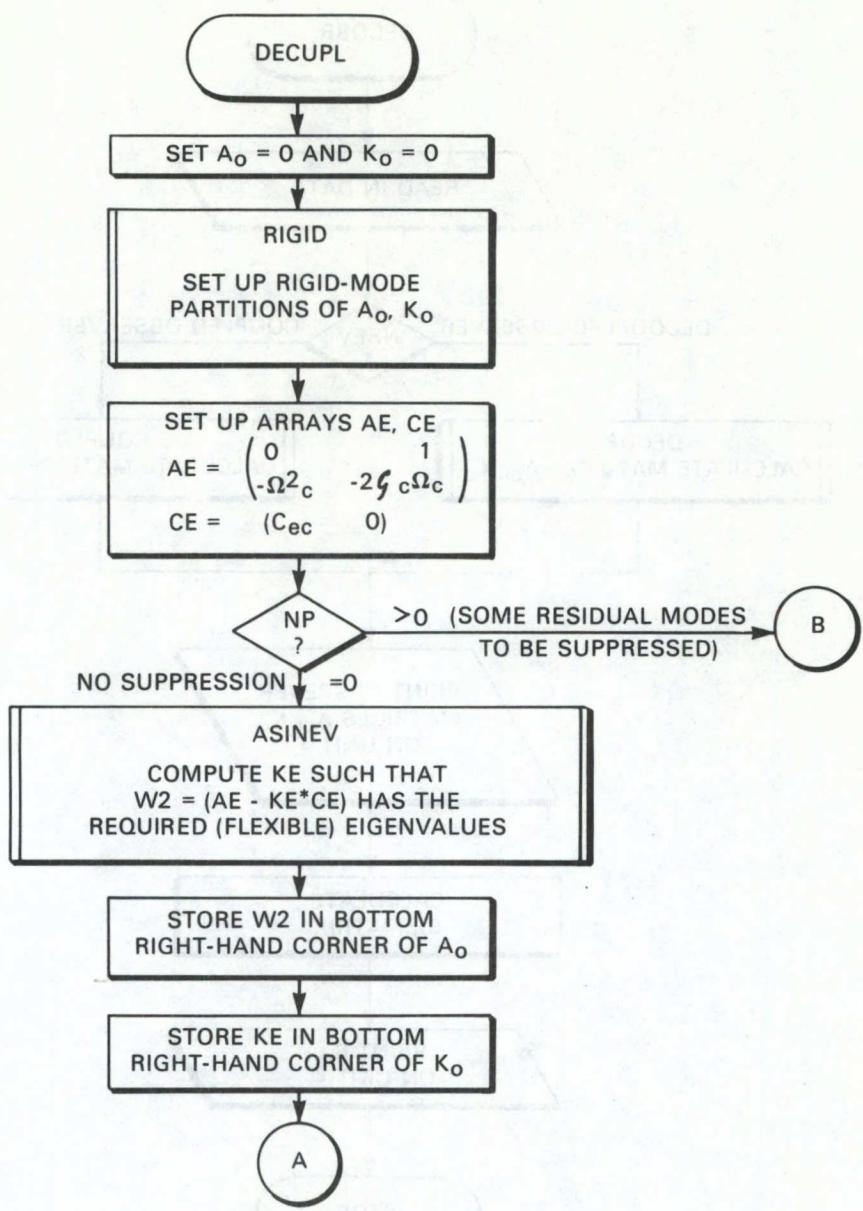


**SUBROUTINE QCOPHYS**

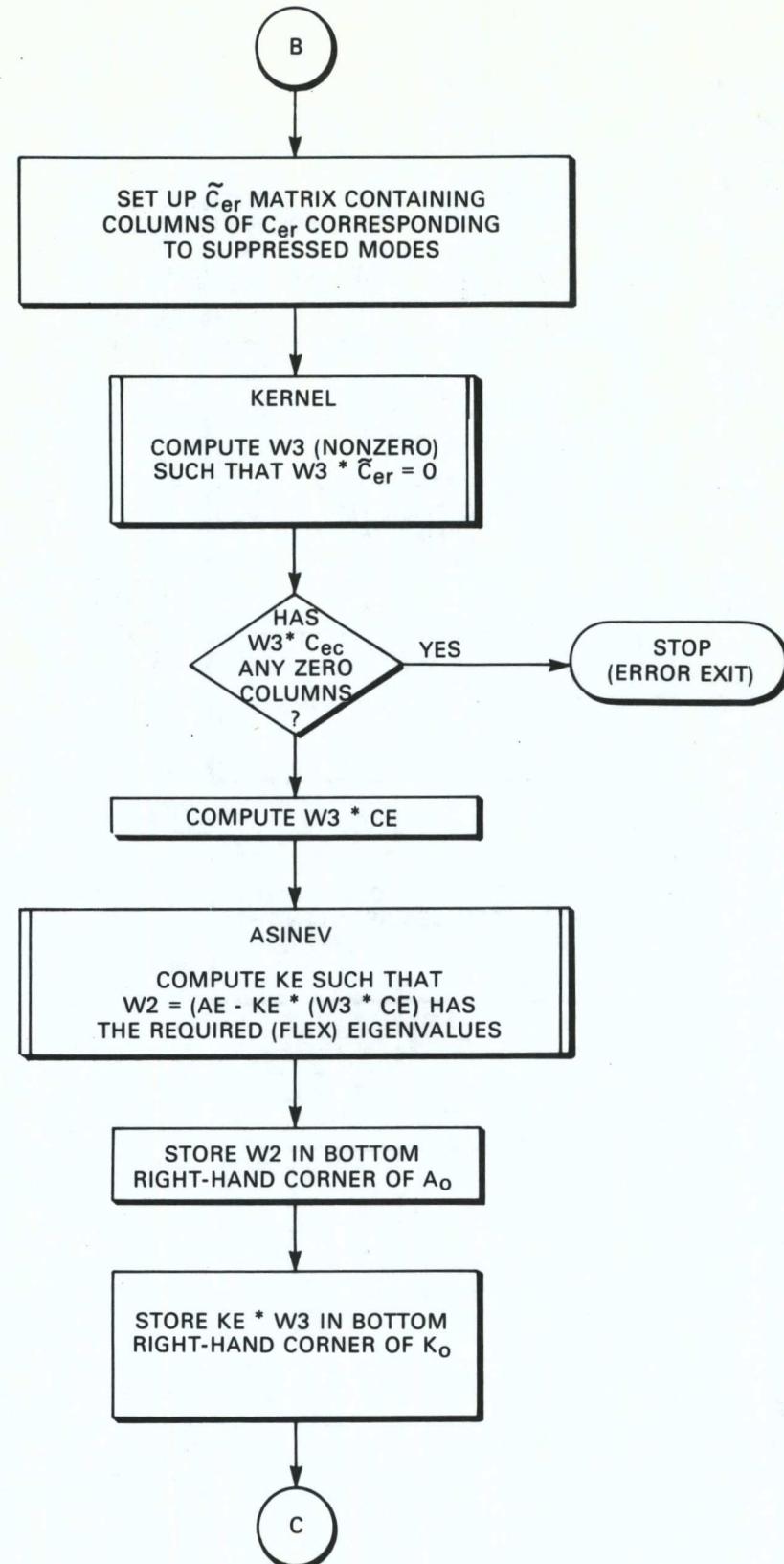




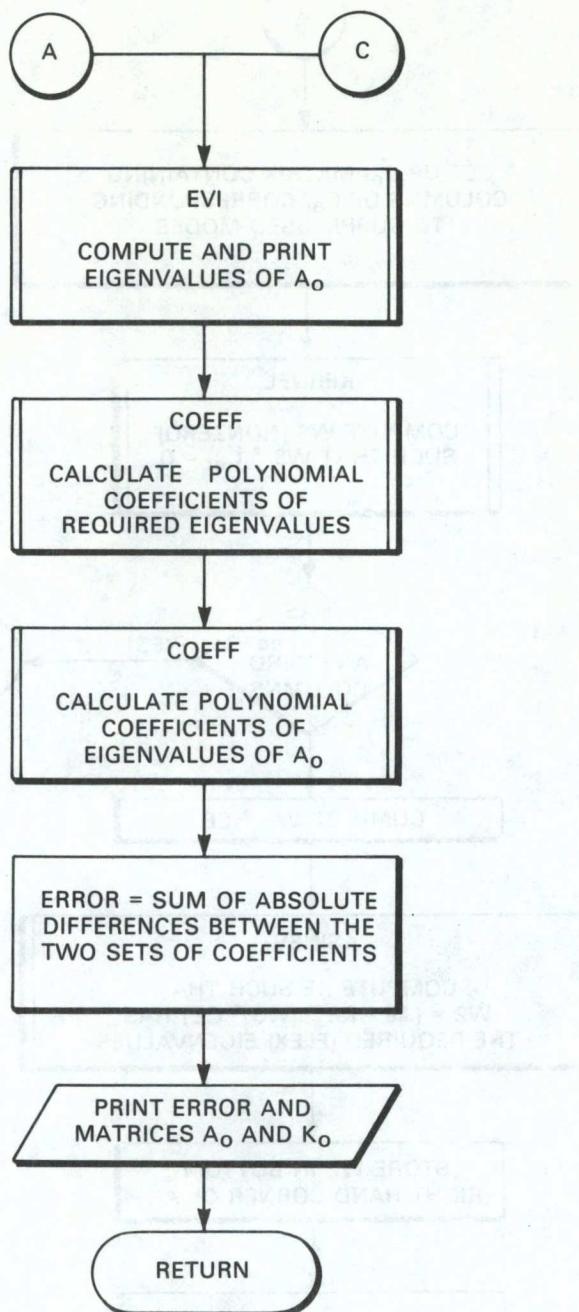
**PROGRAM DECOBR  
MAIN PROGRAM**



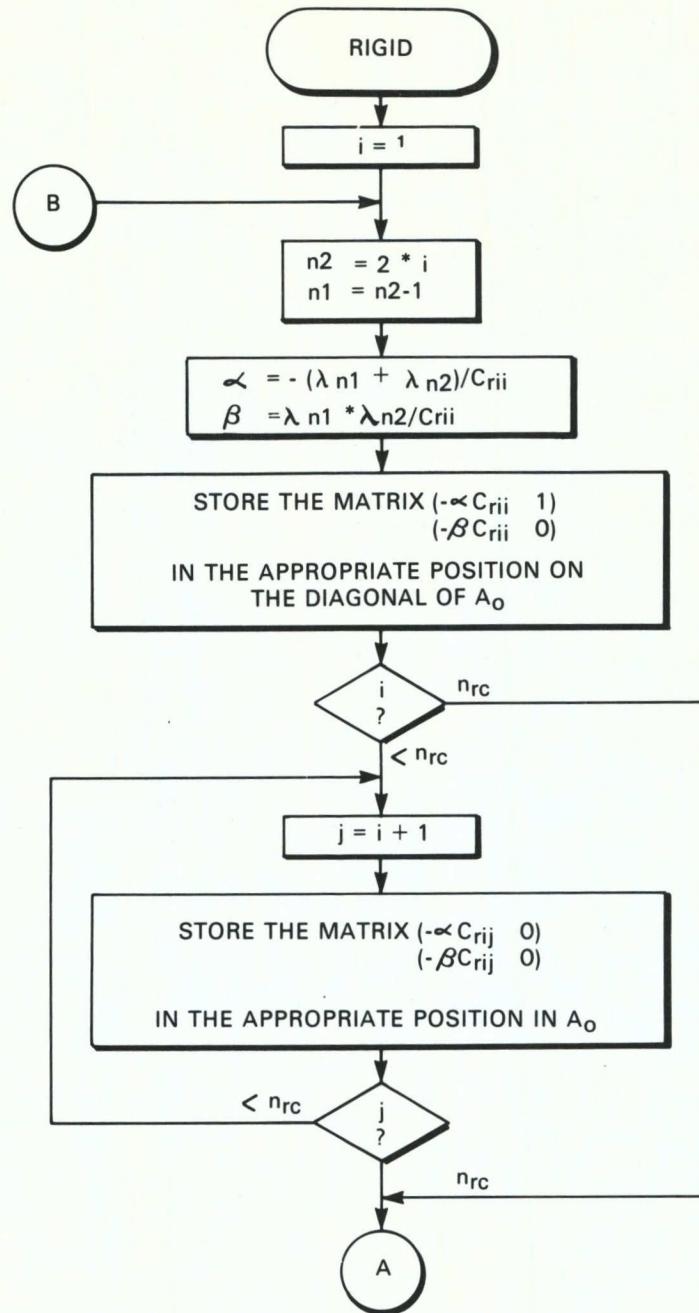
**PROGRAM DECOBR  
SUBROUTINE DECUPL**



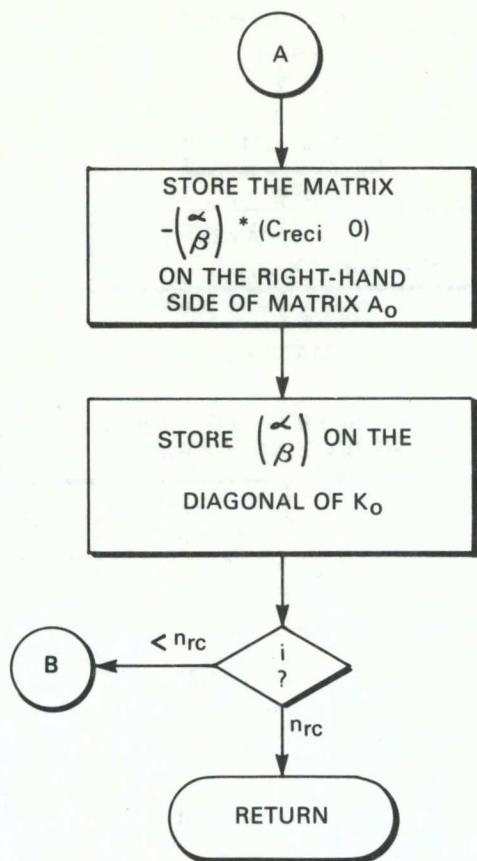
PROGRAM DECOBR  
SUBROUTINE DECUPL CONTINUED



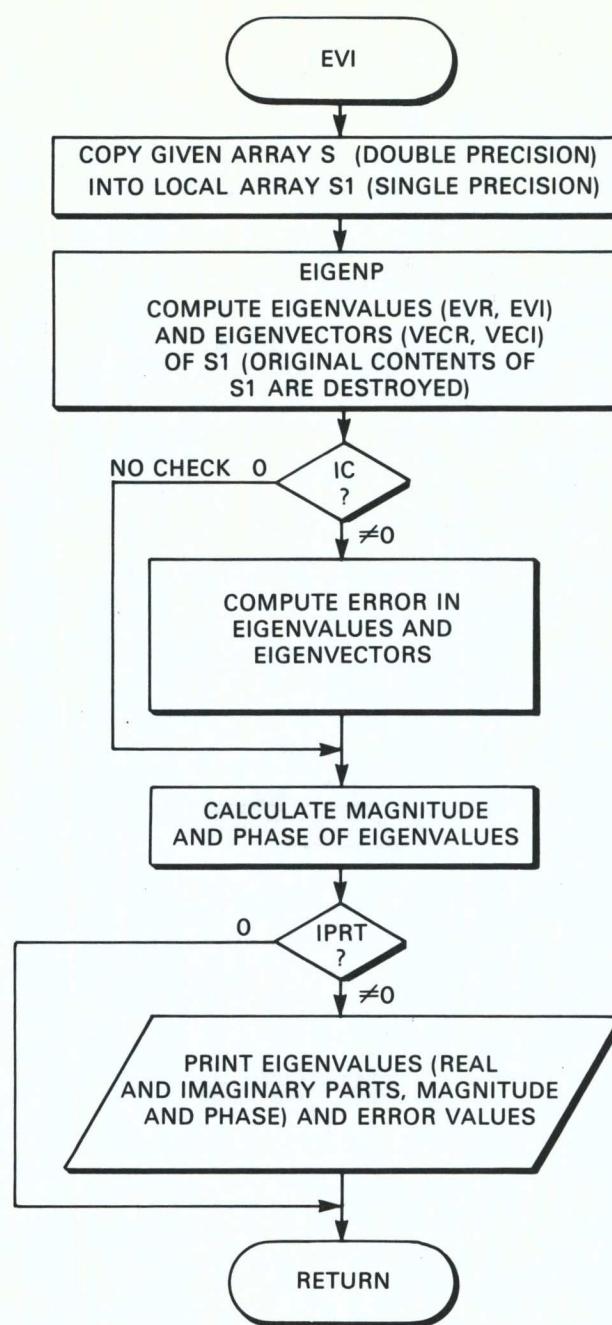
**PROGRAM DECOBR  
SUBROUTINE DECUPL CONTINUED**



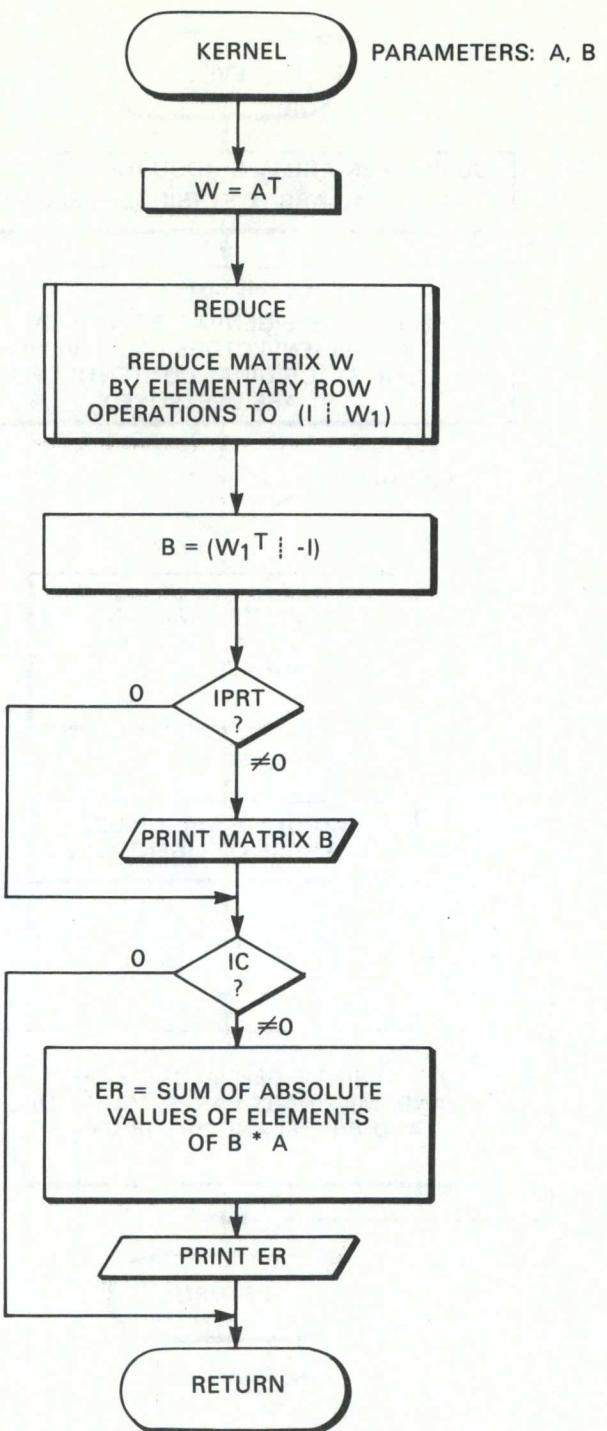
### PROGRAM DECOBR SUBROUTINE RIGID



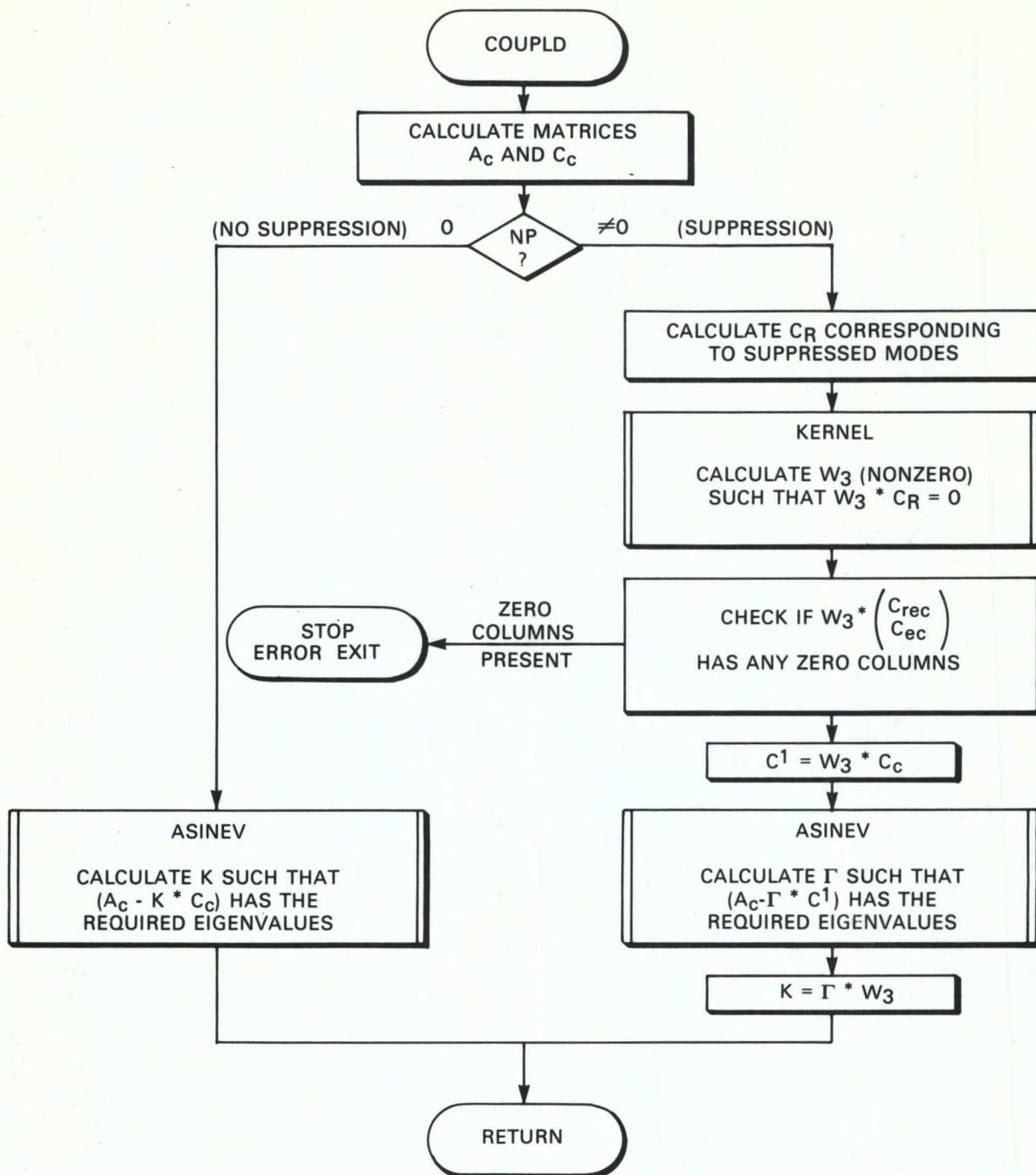
PROGRAM DECOBR  
SUBROUTINE RIGID CONTINUED



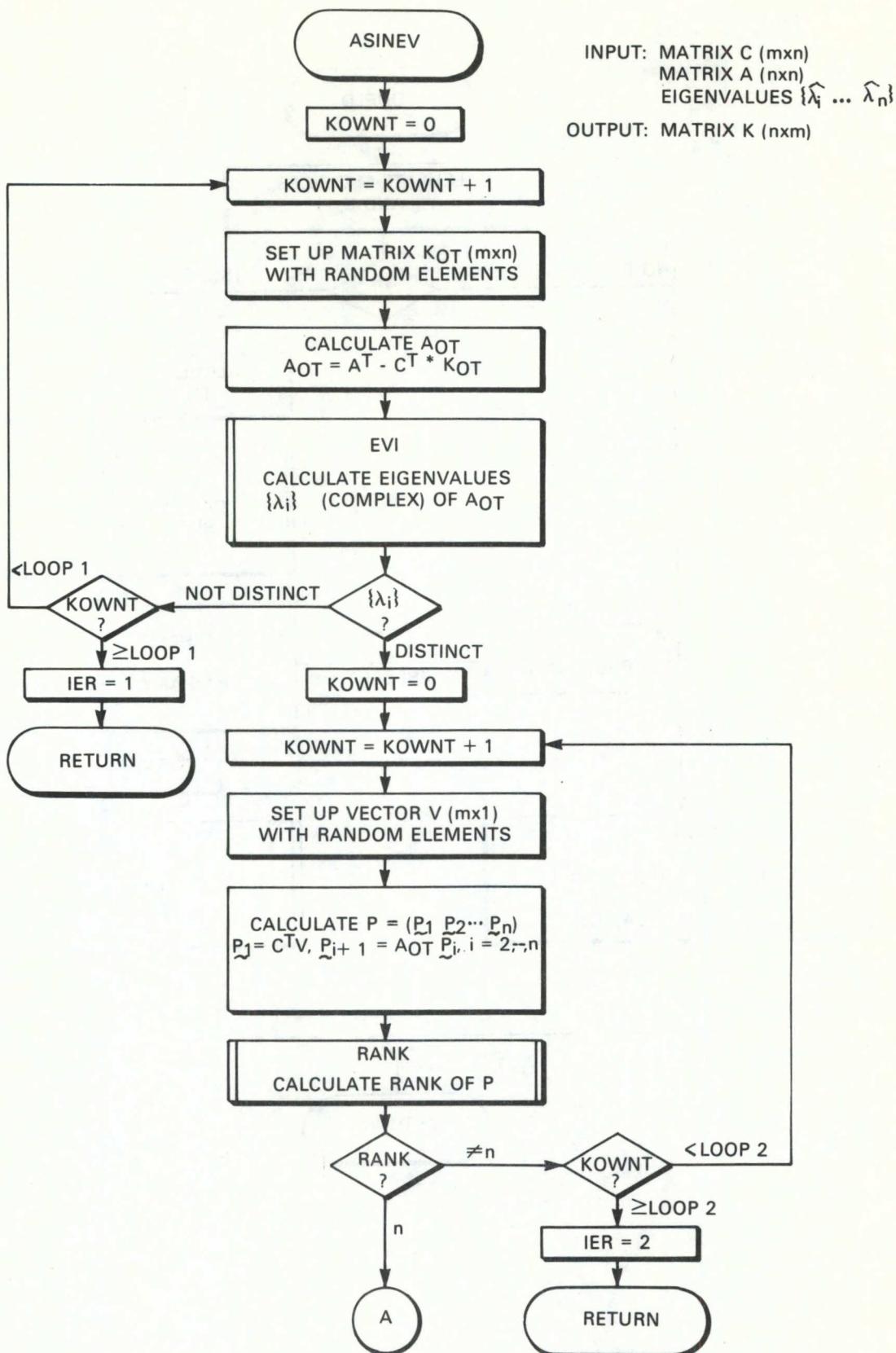
**PROGRAM DECOBR  
SUBROUTINE EVI**



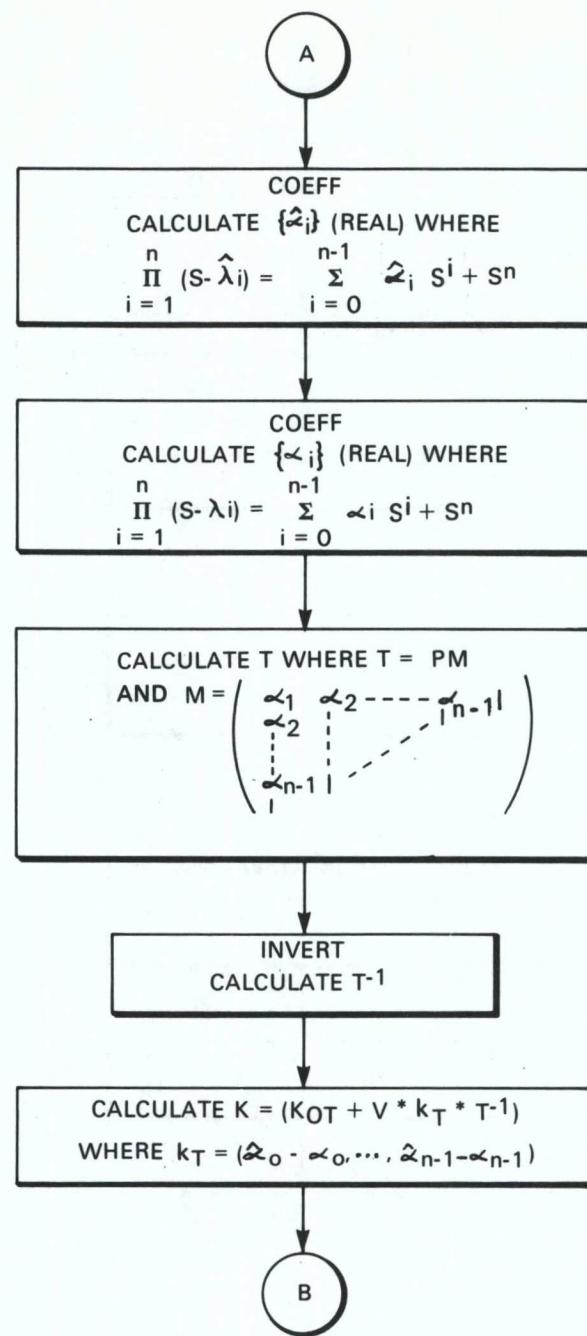
**PROGRAM DECOBR  
SUBROUTINE KERNEL**



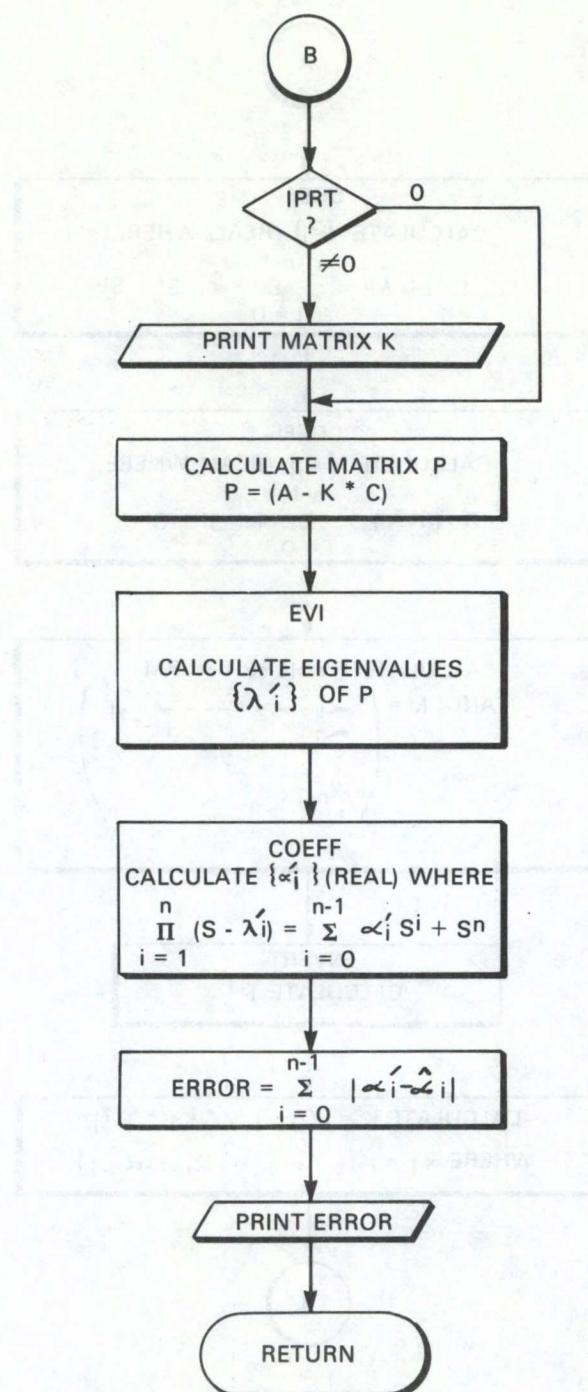
**PROGRAM DECOBR  
SUBROUTINE COUPLD**



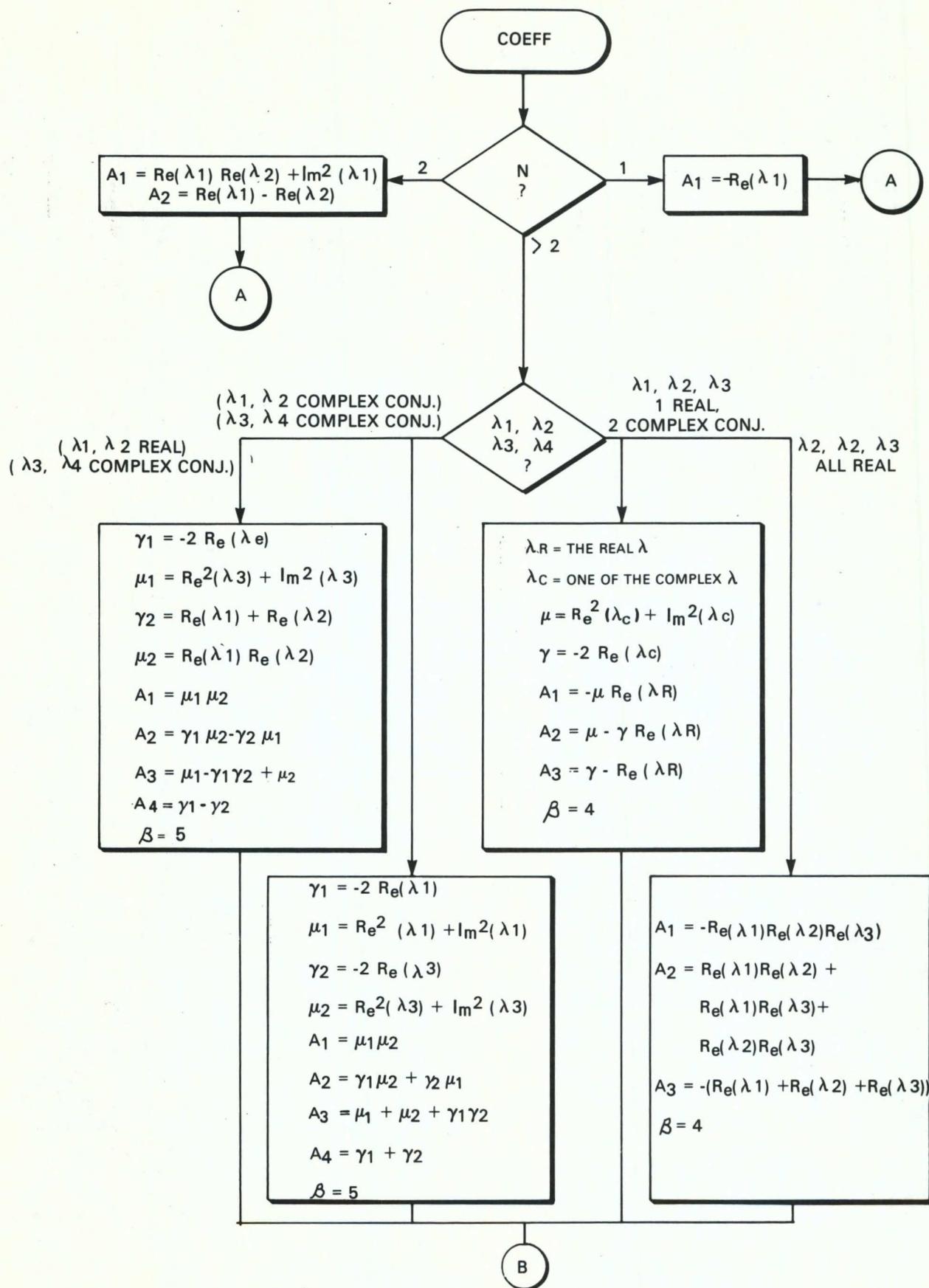
**PROGRAM DECOBR**  
**SUBROUTINE ASINEV**

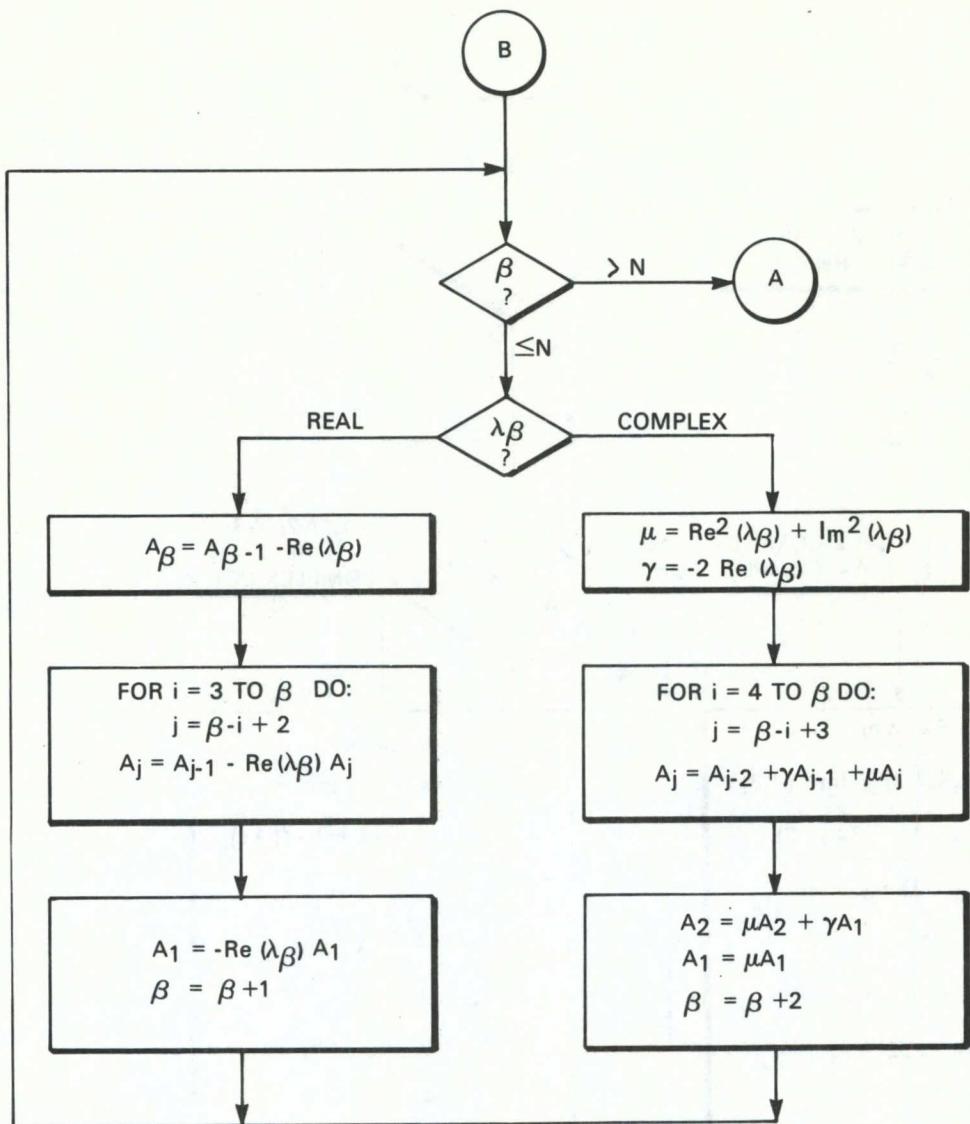


**PROGRAM DECOBR  
SUBROUTINE ASINEV CONTINUED**

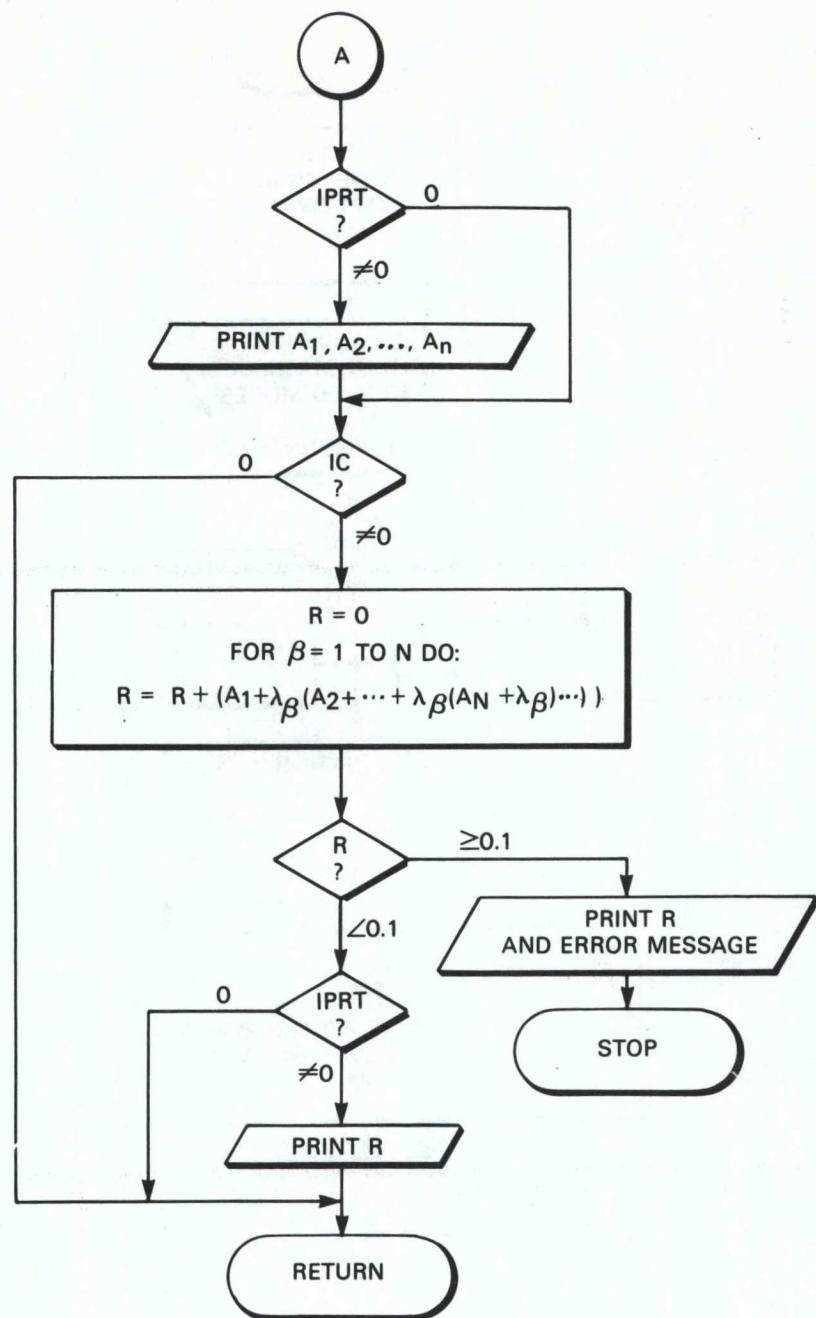


PROGRAM DECOBR  
SUBROUTINE ASINEV CONTINUED

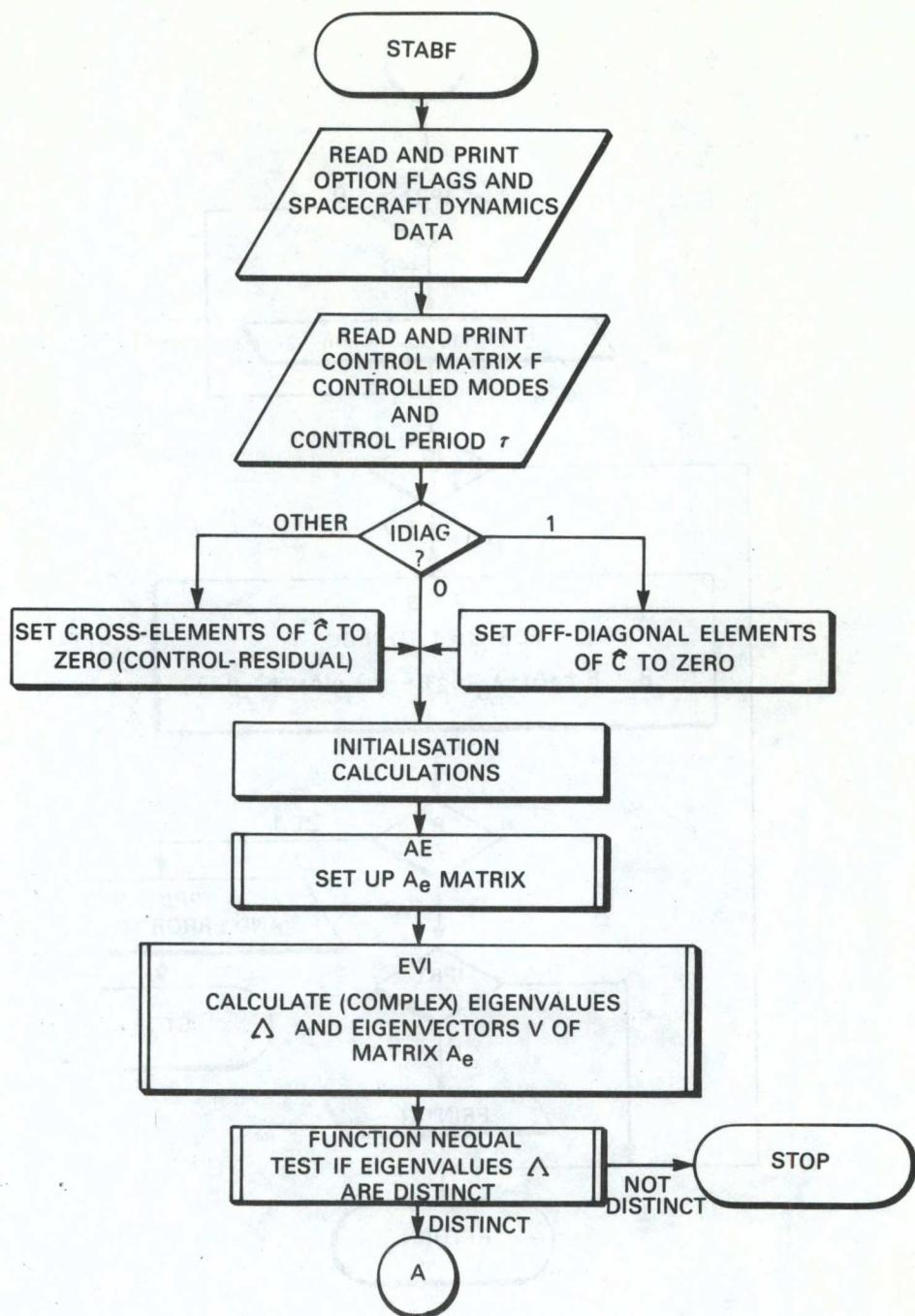




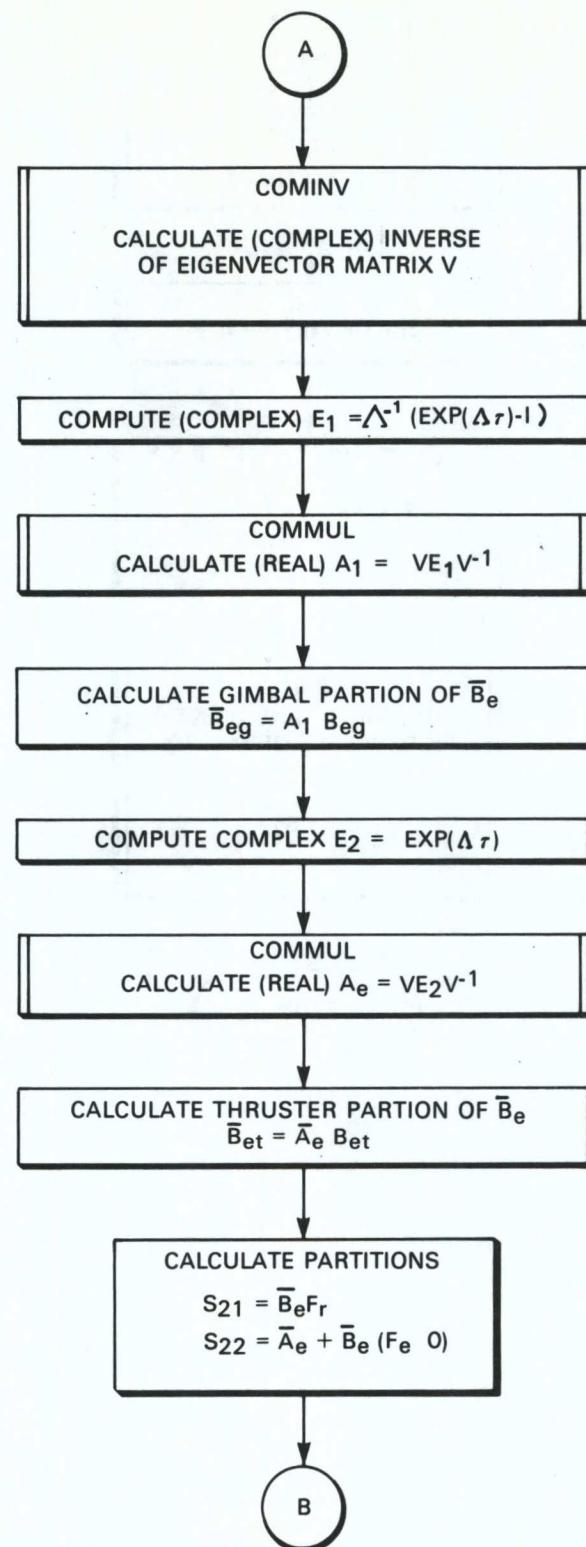
PROGRAM DECOBR  
SUBROUTINE COEFF CONTINUED



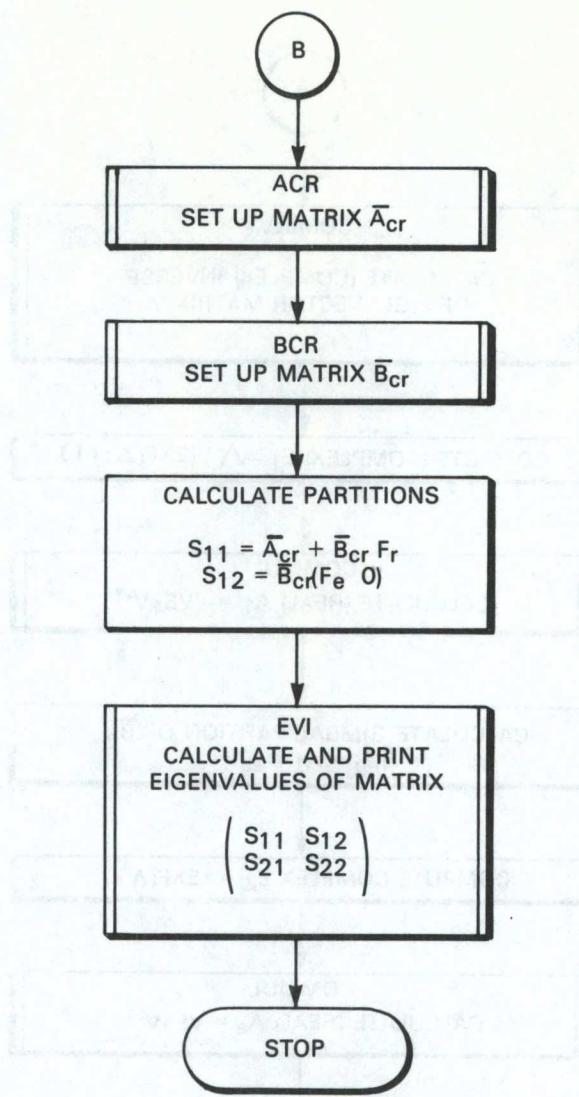
**PROGRAM DECOBR  
SUBROUTINE COEFF CONTINUED**



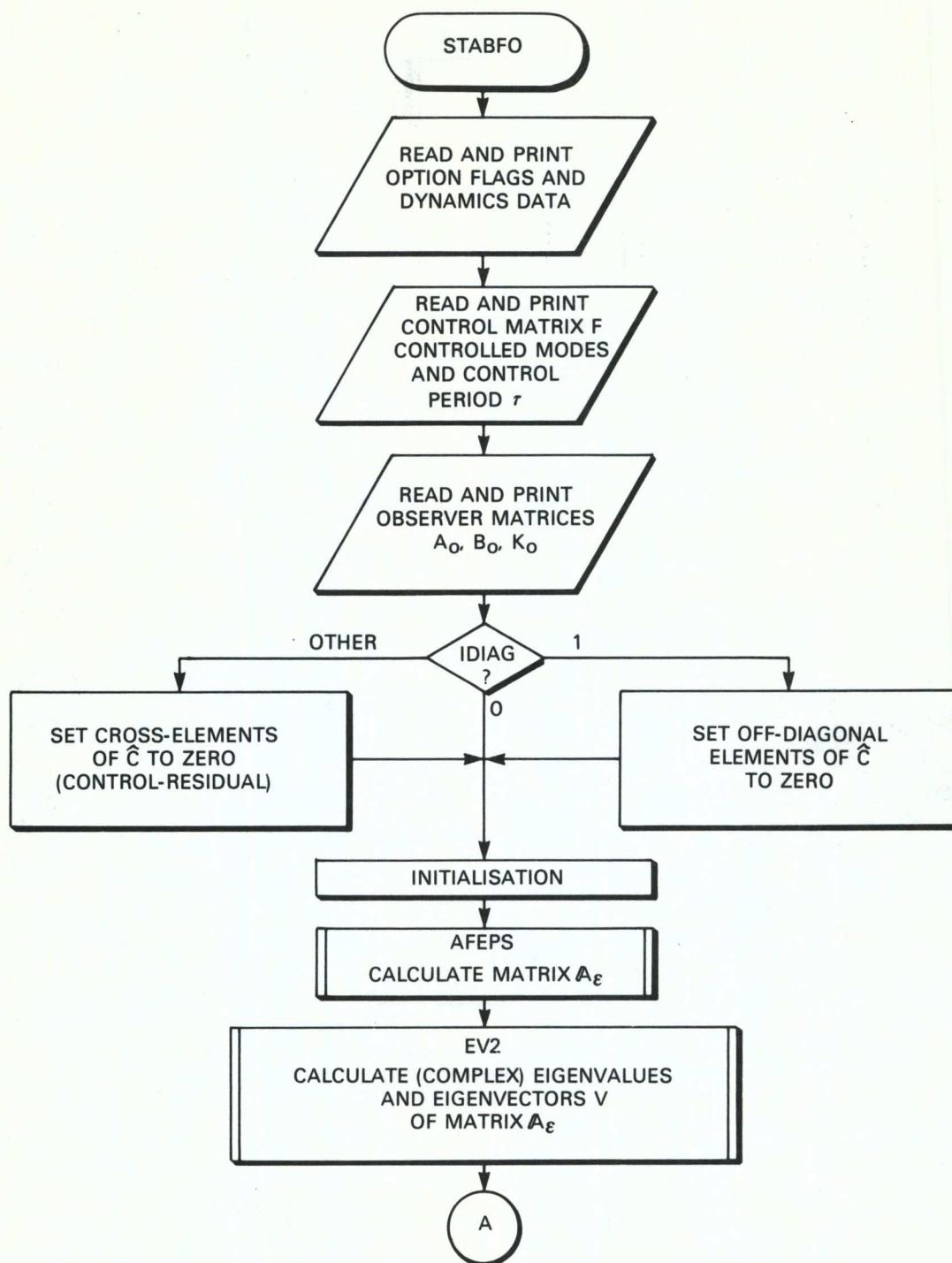
PROGRAM STABF  
MAIN PROGRAM



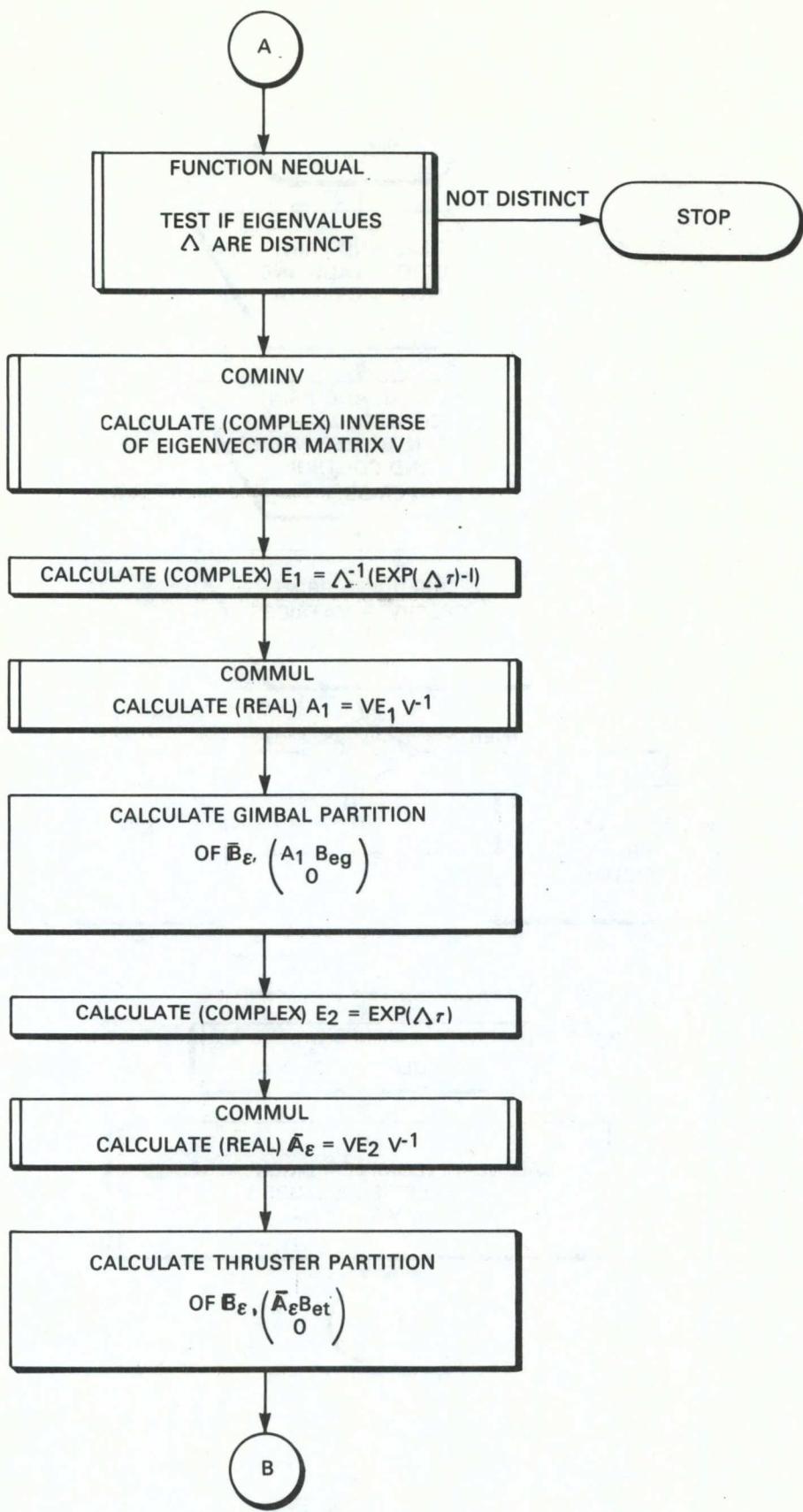
STABF PROGRAM  
MAIN PROGRAM CONTINUED



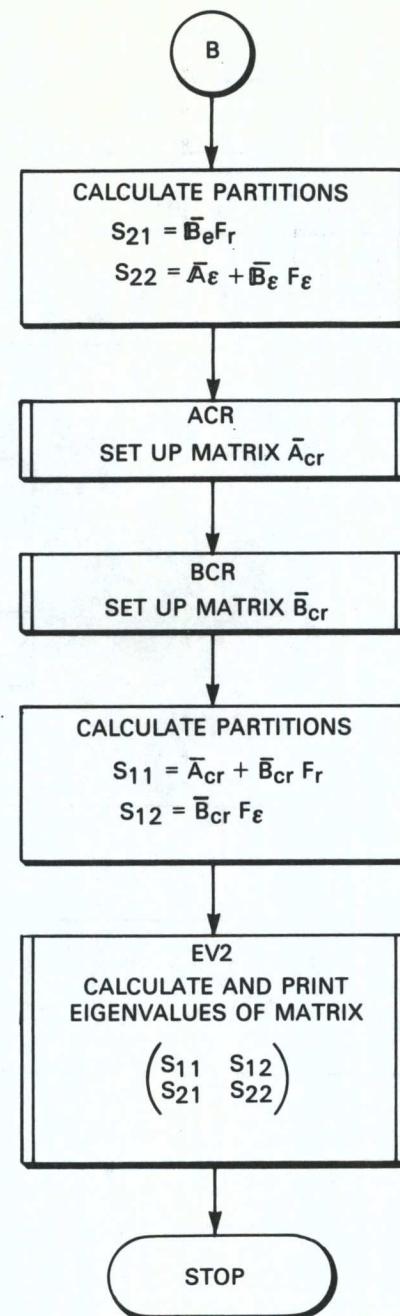
**STABF PROGRAM  
MAIN PROGRAM CONTINUED**



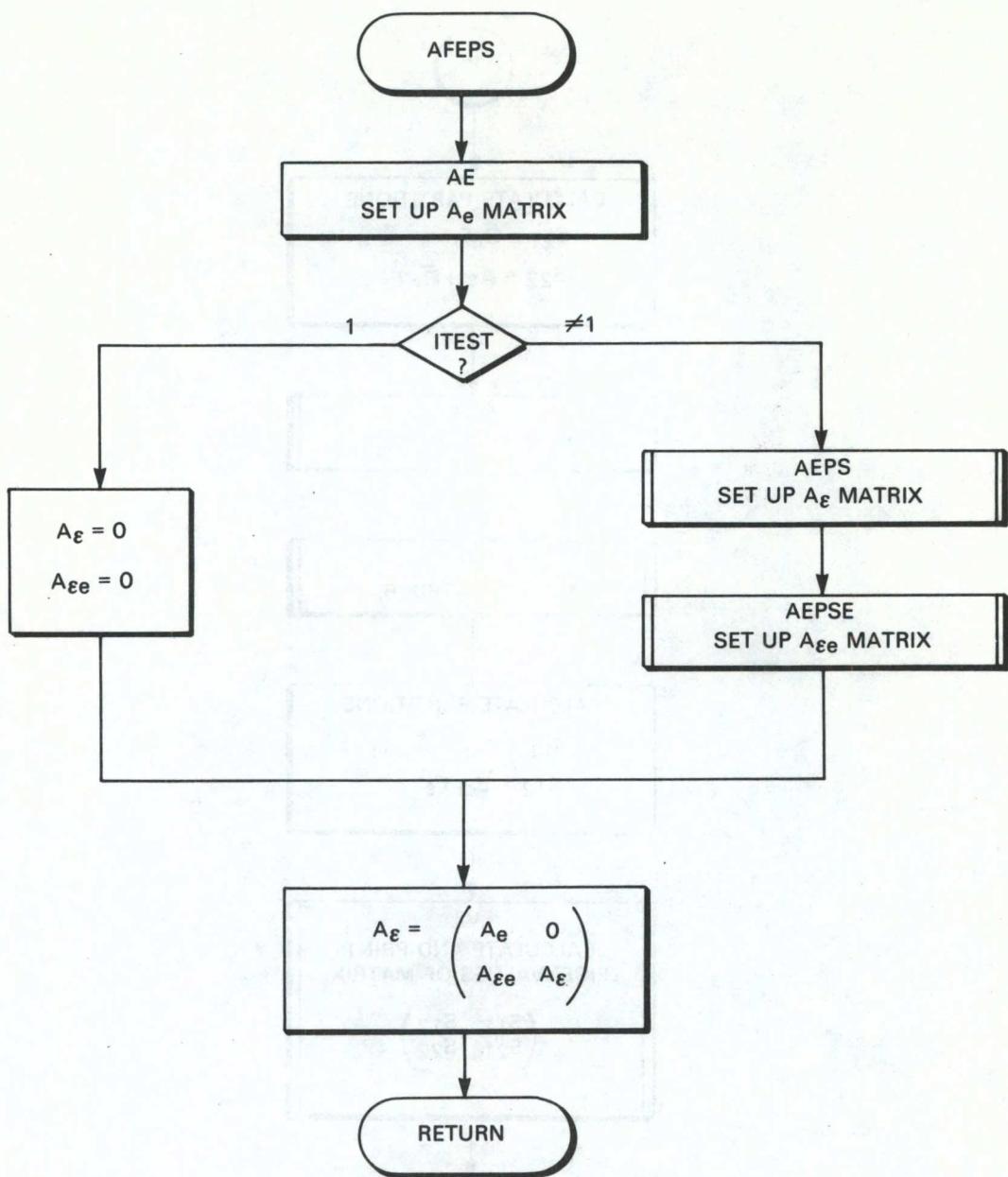
**STABFO PROGRAM  
MAIN PROGRAM**



STABFO PROGRAM  
MAIN PROGRAM CONTINUED



**STABFO PROGRAM  
MAIN PROGRAM CONTINUED**



STABFO PROGRAM  
SUBROUTINE AFEPS

CACC / CCAC



38704

## OPTIMAL STATIONKEEP AND ATTITUDE CONTROL OF FLEXIBLE SPACECRAFT

P  
91  
C655  
058  
1983  
pt.2

DATE DUE  
DATE DE RETOUR

LOWE-MARTIN No. 1137

