

AD-A041 436

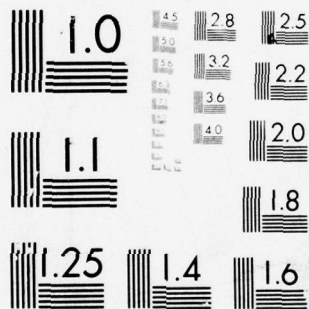
FRANK J SEILER RESEARCH LAB UNITED STATES AIR FORCE --ETC F/G 19/5
HIGH ANGLE OF ATTACK FLIGHT CONTROL USING STOCHASTIC MODEL REFE--ETC(U)
MAY 77 R B ASHER, D GOEBEL
FJSRL-TR-77-0010

UNCLASSIFIED

NL

1 OF 2
AD
A041436





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A



AD A041436



AD NO.
DDC FILE COPY

2

12

FRANK J. SEILER RESEARCH LABORATORY

SRL-TR-77-0010

MAY 1977

HIGH ANGLE OF ATTACK FLIGHT CONTROL USING
STOCHASTIC MODEL REFERENCE ADAPTIVE CONTROL

COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION



SCIENTIFIC

APPROVED FOR PUBLIC RELEASE;
DISTRIBUTION UNLIMITED.

PROJECT 2304

AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

DDC
JUL 11 1977

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

319 920

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

Abstract (cont)

→ of uncertain aerodynamic coefficients. In particular, the partitioning approach of adaptive control is extended to the implicit model following problem. This is then used to solve the problem of high angle of attack flight control.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

HIGH ANGLE OF ATTACK FLIGHT CONTROL
USING STOCHASTIC MODEL REFERENCE ADAPTIVE CONTROL

by

⁺Robert B. Asher
Frank J. Seiler Research Laboratory
U.S. Air Force Academy, CO 80840

and

⁺⁺Dieter Goebel
Wissenschaftlicher Direktor
BAKWT, Mannheim, FRG

⁺Research Associate, Applied Mathematics Division

⁺⁺Visiting Engineer, Frank J. Seiler Research Laboratory, when work was completed.

Table of Contents

	<u>Page</u>
List of Figures -----	ii
Abstract -----	1
I. Introduction -----	1
II. Problem Statement -----	2
III. Control Law Development -----	5
IV. Ideal Aircraft Model -----	9
V. Simulation Results -----	12
References -----	14
Appendix A. Equations of Motion -----	43
Appendix B. Model Following Program, Part 1 -----	46
Appendix C. Model Following Program, Part 2 -----	89

Approved For		Phase Section	☒
Date		Staff Section	☐
UNCLASSIFIED			
AUTHORIZATION			
BY			
INSTRUCTION/AVAILABILITY CODES			
ONE	AVAIL. 200 OF SPECIAL		
A	23		

H2

List of Figures

	<u>Page</u>
1. Case 1 Aircraft Responses in Velocity -----	15
2. Case 1 Aircraft Responses in Angle of Attack -----	16
3. Case 1 Aircraft Responses in Pitch Rate -----	17
4. Case 1 Aircraft Responses in Sideslip Angle -----	18
5. Case 1 Aircraft Responses in Roll Rate -----	19
6. Case 1 Aircraft Responses in Yaw Rate -----	20
7. Case 1 Aircraft Responses in Bank Angle -----	21
8. Case 1 Aircraft Responses in Pitch Angle -----	22
9. Case 1 Closed Loop Elevator Deflection -----	23
10. Case 1 Closed Loop Aileron Deflection -----	24
11. Case 1 Closed Loop Rudder Deflection -----	25
12. Probabilities vs Time -----	26
13. Case 2 Aircraft Responses in Pitch Rate -----	27
14. Case 2 Aircraft Responses in Sideslip -----	28
15. Case 2 Aircraft Responses in Roll Rate -----	29
16. Case 2 Aircraft Responses in Yaw Rate -----	30
17. Case 2 Aircraft Responses in Bank Angle -----	31
18. Case 2 Closed Loop Elevator Deflection -----	32
19. Case 2 Closed Loop Aileron Deflection -----	33
20. Case 2 Closed Loop Rudder Deflection -----	34
21. Standard Deviation of Velocity -----	35
22. Standard Deviation of Angle of Attack -----	36
23. Standard Deviation of Pitch Rate -----	37
24. Standard Deviation of Sideslip -----	38
25. Standard Deviation of Roll Rate -----	39
26. Standard Deviation of Yaw Rate -----	40
27. Standard Deviation of Bank Angle -----	41
28. Standard Deviation of Pitch Angle -----	42

Abstract

High angle of attack flight control is of utmost importance to military aircraft in air combat maneuvering. Flight in this regime has in recent years caused many high performance aircraft to be lost due to departure of the aircraft. The aerodynamics in this regime are highly nonlinear. The problem is compounded by the fact that the aerodynamics are not well known. This paper considers the use of adaptive control in order to perform model following of an "ideal" aircraft in the presence of uncertain aerodynamic coefficients. In particular, the partitioning approach of adaptive control is extended to the implicit model following problem. This is then used to solve the problem of high angle of attack flight control.

I. Introduction

High angle of attack flight may cause undesirable aircraft dynamic response. This dynamic response has been the cause for the loss of many high performance swept-wing aircraft due to stall-departure problems. Aircraft departure can be defined [1] as an uncommanded and/or uncontrollable dynamic response of the aircraft. It is manifested as either a divergent rolling-side slipping oscillation of large amplitude, i.e., wing rock, or a large rapid yaw generally followed by a rapid roll, i.e., nose slice. For example, reference [1] documents a simulation in which an A-7 airplane in a turn at high angle of attack flight exhibited nose slice with a yaw rate buildup of approximately 65 deg/sec. In an actual flight this would have caused a spin from which the probability of recovery would have been low. When this condition is encountered, it is unanticipated and the pilot is under physical and mental stress; the aircraft and pilot will most likely be lost if departure occurs during combat and certainly will be lost under 15,000 ft altitudes regardless of spin recovery characteristics [2].

The importance of departure led to a symposium at the Air Force Flight Dynamics Laboratory in 1971 on AFFDL Stall/Post Stall/Spin Symposium. Since then many articles have appeared in the study of departure [1,2,3,4,5]. Numerous studies have been made to identify the aerodynamic coefficients in high angle of attack regime. Reference [6] is a complete study of this.

However, in order to collect real time data for proper identification, it is necessary to subject the aircraft to divergence. Wind tunnel data, although valuable, still have residual errors due to differences in unsteady flow between the wind tunnel and the actual flight condition. It is desirable to accurately have knowledge as to the coefficients for control purposes; in fact, it is necessary.

This paper considers the control of aircraft at high angles of attack in the presence of uncertainty in the aerodynamic coefficients. In particular, the nonlinear equations of motion are linearized about a given flight condition. The nonlinear aerodynamics are included in the model. An ideal aircraft model for the given flight condition is developed. This ideal model varies with changes in flight condition. An adaptive implicit model following control law is developed in order to keep the actual aircraft close to the ideal response. Uncertainties in the aerodynamic coefficients are eliminated by the adaptive estimator.

This paper is divided into five sections. The next section contains the problem statement. Section III gives the development of the control law. Section IV contains a discussion as to how the ideal model for the A-7 was chosen. A different aircraft may have a different "ideal" model. Section V contains simulation results for the A-7 aircraft using the control law in the paper. Section VI yields the conclusions.

II. Problem Statement

The equations of motion of an aircraft linearized about a given angle of attack, α_0 ; Euler angles between gravity oriented inertial axis and aircraft body axis, $\theta_{p0}, \phi_0, \psi_0$; the flight path angle, γ_0 ; angular velocities, p_0, q_0 , and r_0 ; nominal forward velocity, U_0 ; sideslip angle, β_0 , and given as

$$\dot{x} = A(\mu_1, t)x + B(\mu_2, t)u$$

where

$$x^T = \{u_V, \alpha - \alpha_0, q, \beta - \beta_0, p, r, \phi, \theta_p - \theta_{p0}\}^T,$$

$$u^T = \{\delta_e, \delta_a, \delta_R\}^T$$

where u_v is the perturbed total linear velocity; $\alpha - \alpha_0$ is the perturbed angle of attack; p , q , and r are the perturbed angular velocities; $\beta - \beta_0$ is the perturbed sideslip angle; $\theta_p - \theta_{p0}$ is the perturbed pitch angle and ϕ is the roll angle. The controls are assumed to be deflections in elevator, δ_e , in aileron, δ_a , and in rudder, δ_R . The matrices A and B are given in Appendix A as well as the definitions of μ_1 and μ_2 . The parameter vector μ_1 and μ_2 contain the aerodynamic coefficients which are assumed uncertain except for an a priori probability density function. The coefficients are assumed constant over the time interval that the linearization is assumed valid. The time dependency of the A and B matrices depict the temporally changing linearization.

It is assumed that a noisy measurement of the states is available, i.e.,

$$y_m = Cx + v \quad (2)$$

where v is zero mean white noise with covariance $E\{v(t)v(\tau)^T\} = V_R\delta(t-\tau)$ and C is defined in Appendix A.

From Equation (A.2), it may be noted that the longitudinal and lateral modes of the aircraft are highly coupled. Furthermore, as may be noted by expansion of the equations of motion, there are many destabilizing terms in the equations. Consequently, it is an extremely difficult multivariable task for the pilot to prevent departure at high angles of attack. This is especially true in air combat maneuvering as the physical and mental stress occupies the pilot's attention. Thus, it is desirable to obtain a closed loop control law which will prevent departure. One method is reported in [5] where the author develops a feedback control to eliminate perturbations about a nominal trajectory for a deterministic system. The approach in this paper is to design a feedback control law based on model following of an ideally responding aircraft. The advantage is that the model following control more closely gives a control law that will yield a desirable response. Thus, for example, decoupling of the yaw-roll problem in nose slice may be approximately obtained by placing this feature into the ideal model. The destabilizing terms may be compensated by use of the model. The adaptive control law will compensate for uncertainties in the aerodynamic coefficients by real time learning. The

development of the ideal model will be explained in Section IV with a specific example given for the A-7 in a particular flight condition. The form of the model is

$$\dot{z} = A_m(t)z \quad (3)$$

where the time varying A_m matrix corresponds to the ideal model changing due to the different flight conditions. Since implicit model following [10] is to be accomplished, the performance index is taken to be

$$J = E \left\{ \int_{t_0}^{t_f} [(\dot{y}_0 - A_m y_0)^T Q_p (\dot{y}_0 - A_m y_0) + u^T R_p u] dt \right\} \quad (4)$$

where y_0 is the output vector (not to be confused with the measurement vector)

$$y_0 = C_0 x \quad (5)$$

where C_0 is a time invariant distribution matrix and y_0 is a general term since x and z may not be of the same dimensions and where R_p weights the control surface deflections and is a positive definite matrix, Q_p weights excursions from the model response, and t_f is chosen as the interval over which the linearization and constancy of the aerodynamic coefficients are assumed valid. The optimal control would fall into the class of dual control problems. However, additional uncertainties in the model other than those accounted for, unsteady flow problems, as well as the survivability dictates that the dual control may not be used. This is because additional control responses due to the identification aspect of dual control may, because of additional uncertainties, cause an extremely undesirable response leading, perhaps, to an aggravation of the divergence problem. This may in an extreme case lead to the loss of the aircraft. Thus, an adaptive open loop feedback controller will be chosen. There are two major techniques that may be used. The first in reference [7] yields the optimal open loop feedback controller for the problem with uncertain parameters. The computational burden of the technique is larger than the second technique of [8] even though it will lead to better

performance as it is an optimal technique. Since optimality certainly must, in this application, include computational burden, the technique as given in [8] will be extended to the model following problem.

The partitioning technique as in [8] consists of using a control law found by solving the μ -conditional control problem and then weighting the μ -conditional control with the probability density of μ conditioned on the measurements. The technique includes the measurement conditional probability density function for μ in the solution for the control gains. This yields the optimal open-loop feedback control solution, but it has the disadvantage that the equations for the control gain differ at each measurement. In this paper a control law with as little computational burden as possible consistent with the uncertainty problem and good performance is desired. Consequently, the partitioning algorithm will be chosen as the adaptive control method in this paper.

III. Control Law Development

The solution to the partitioned adaptive control model reference problem may be found by using state augmentation techniques. The new state ζ is defined as

$$\zeta^T = [x^T, z^T].$$

This yields a state equation as

$$\dot{\zeta} = \bar{A}(\mu_1)\zeta + \bar{B}(\mu_2)u \quad (6)$$

where

$$\bar{A}(\mu_1) = \begin{bmatrix} A(\mu_1) & 0 \\ 0 & A_m \end{bmatrix}$$

and

$$\bar{B}(\mu_2) = \begin{bmatrix} B(\mu_2) \\ 0 \end{bmatrix}.$$

The performance index may be easily rewritten as

$$J = E \left\{ \int_{t_0}^t \left[x^T \bar{Q}(\mu_1) x + 2u^T s(\mu_1, \mu_2) x + u^T \bar{R}(\mu_2) u \right] dt \right\} \quad (7)$$

where

$$\begin{aligned} \bar{Q}(\mu_1) &= [C_0 A(\mu_1) - A_m C_0]^T Q_p [C_0 A(\mu_1) - A_m C_0], \\ s(\mu_1, \mu_2) &= B(\mu_2)^T C_0^T Q_p [C_0 A(\mu_1) - A_m C_0], \end{aligned} \quad (8)$$

and

$$\bar{R}(\mu_2) = B(\mu_2)^T C_0^T Q_p C_0 B(\mu_2) + R_p.$$

Thus, it may be noted that the integral under the performance index as well as the system dynamics are functions of μ_1 and μ_2 .

The partitioned adaptive control law may be found by solving for the deterministic control gain conditioned on μ_1 and μ_2 , for the μ_1, μ_2 conditional Kalman filter estimate, $\hat{x}(t|\mu_1, \mu_2, \psi_t)$, and for the conditional density, $p(\mu_1, \mu_2|Y_t)$ where $Y_t = \{y(\tau), t_0 \leq \tau \leq t\}$, and using as a control

$$u(t) = \int_{R_{\theta_1}} \int_{R_{\theta_2}} \bar{K}(t|\mu_1, \mu_2) \hat{x}(t|\mu_1, \mu_2, \psi_t) p(\mu_1, \mu_2|Y_t) d\mu_1 d\mu_2. \quad (9)$$

If μ_1 and μ_2 are defined over discrete ranges, then equation (9) may be rewritten as

$$\begin{aligned} u(t) &= \sum_{i=1}^{\ell_1} \sum_{j=1}^{\ell_2} \bar{K}(t|\mu_{1i}, \mu_{2j}) \hat{x}(t|\mu_{1i}, \mu_{2j}, Y_t) \\ &\quad \cdot P_r(\mu_{1i}, \mu_{2j}|Y_t) \end{aligned} \quad (10)$$

where $P_r((\cdot))$ denotes the probability of the event $(\cdot)$. The control gain may be determined by the solution of the μ_{1i} and μ_{2j} conditional deterministic problem, i.e.,

$$\bar{K}(t|\mu_{1i}, \mu_{2j}) = -\bar{R}(\mu_{2j})^{-1}[S(\mu_{1i}, \mu_{2j}) + B(\mu_{2j})^T P(\mu_{1i}, \mu_{2j}, t)] \quad (11)$$

where

$$\begin{aligned} \dot{P}(\mu_{1i}, \mu_{2j}, t) = & P(\mu_{1i}, \mu_{2j}, t)[A(\mu_{1i}) - B(\mu_{2j})\bar{R}(\mu_{2j})^{-1} \\ & \cdot S(\mu_{1i}, \mu_{2j})] + [A(\mu_{1i}) \\ & - B(\mu_{2j})\bar{R}(\mu_{2j})^{-1}S(\mu_{1i}, \mu_{2j})]^T P(\mu_{1i}, \mu_{2j}, t) \\ & - P(\mu_{1i}, \mu_{2j}, t)B(\mu_{2j})\bar{R}(\mu_{2j})^{-1}B(\mu_{2j})^T P(\mu_{1i}, \mu_{2j}, t) \\ & + \bar{Q}(\mu_{1i}) - S(\mu_{1i}, \mu_{2j})^T \bar{R}(\mu_{2j})^{-1}S(\mu_{1i}, \mu_{2j}) \end{aligned} \quad (12)$$

$$V_i = 1, 2, \dots, \ell_1$$

$$V_j = 1, 2, \dots, \ell_2$$

with final condition

$$P(\mu_{1i}, \mu_{2j}, t_f) = 0, \quad V_i, j.$$

The filter equations are the standard Kalman equations

$$\begin{aligned} \dot{\hat{x}}(t|\mu_{1i}, \mu_{2j}, Y_t) = & A(\mu_{1i})\hat{x}(t|\mu_{1i}, \mu_{2j}, Y_t) \\ & + B(\mu_{2j})\bar{K}(t|\mu_{1i}, \mu_{2j})\hat{x}(t|\mu_{1i}, \mu_{2j}, Y_t) \\ & + K_G(t|\mu_{1i}, \mu_{2j})[y_m - C\hat{x}(t|\mu_{1i}, \mu_{2j}, Y_t)] \end{aligned} \quad (13)$$

with

$$\hat{x}(t_0|\mu_{1i}, \mu_{2j}) = \hat{x}(t_0)$$

where

$$K_G(t|\mu_{1i}) = V(t|\mu_{1i})C^T V_R^{-1} \quad (14)$$

with

$$\begin{aligned} \dot{V}(t|\mu_{1i}) &= A(\mu_{1i})V(t|\mu_{1i}) + V(t|\mu_{1i})A(\mu_{1i})^T \\ &\quad - V(t|\mu_{1i})C^T V_R^{-1} C V(t|\mu_{1i}) \\ i &= 1, 2, \dots, \ell_1 \\ j &= 1, 2, \dots, \ell_2 \end{aligned} \quad (15)$$

and

$$V(t_0|\mu_{1i}) = V(t_0).$$

The conditional probability density function for μ_{1i} and μ_{2j} may be computed via

$$\begin{aligned} \Pr(\mu_{1i}, \mu_{2j}, Y_t) &= \\ &= \frac{P(\mu_{1i})P(\mu_{2j}) \exp \left\{ \int_{t_0}^t x^T(t|\mu_{1i}, \mu_{2j}, Y_t) C^T V_R^{-1} y_m(t) dt - \int_{t_0}^t \|C\hat{x}(t|\mu_{1i}, \mu_{2j}, Y_t)\|^2_{V_R^{-1}} dt \right\}}{\sum_{i=1}^{\ell_1} \sum_{j=1}^{\ell_2} P(\mu_{1i})P(\mu_{2j}) \exp \left\{ \int_{t_0}^t \hat{x}^T(t|\mu_{1i}, \mu_{2j}, Y_t) C^T V_R^{-1} y(t_0) dt - \int_{t_0}^t \|C\hat{x}(t|\mu_{1i}, \mu_{2j}, Y_t)\|^2_{V_R^{-1}} dt \right\}} \\ i &= 1, 2, \dots, \ell_1 \\ j &= 1, 2, \dots, \ell_2 \end{aligned} \quad (16)$$

where $P(\mu_{1i})$ and $P(\mu_{2j})$ are the a priori probabilities for μ_{1i} and μ_{2j} , respectively. The simulations use the proper discrete form of the equation as in reference [9].

The procedure is, thus, to first obtain a set of linearized equations where the linearization is taken over the current flight condition. This linearization must be updated frequently as the changing flight regime may drastically change the dynamic response. It is assumed that over a given time period the aerodynamic coefficients are constant. This assumption is valid

over the same region that the linearization assumption is valid. However, the aerodynamic coefficients are not exactly known. It is assumed that the uncertain coefficients are defined over a discrete range. This procedure defines the linear equations as in equation (1). The ideal model which is dependent on the angle of attack and sideslip angle is utilized along with the measurement equation as in (2) along with the determination of the output equation (5) in order to define the remaining equations for dynamic response. The control weighting matrix R_p and model matching weighting matrix Q_p must be determined. The final time, t_f , must be determined. This is the maximum time that the linearization will be assumed valid.

Thus, all the equations necessary for control law design are now assumed to be available. Equations (13-15) are used to obtain the μ_1 and μ_2 conditional state estimates for the aircraft state. This is used in equation (16) to find the probability of each μ_1 and μ_2 . The control gain as in (11) is calculated, and the control is determined from equation (10).

An approximate law may be obtained by finding the steady state gains $\bar{K}$ and the steady state Kalman filter and using these in the control computations.

IV. Ideal Aircraft Model

This section uses the equations for the A-7 as given in Appendix A in order to discuss the reasons for departure of the A-7 and in order to yield an ideal model with better response in the high angle of attack regime.

The equations of motion for the actual A-7 are given in Appendix A. They are formulated using wind axes and the aerodynamic derivatives are evaluated at a prestall flight condition of $\alpha_0 = 19$ deg and $\beta_0 = 6$ deg. The incremental change in rolling moment L'_i and in yaw moment N'_i are calculated by

$$L'_i = \frac{L_i + (I_{xz}/I_x)N_i}{1 - (I_{xz}^2/I_x I_y)}$$

and

$$N'_i = \frac{N_i + (I_{xz}/I_x)L_i}{1 - (I_{xz}^2/I_x I_y)}$$

(17)

where i denotes the particular state variable and where I_x and I_y are body axes moments of inertia, I_{xz} is the respective cross product of inertia, and L_i and N_i are the aerodynamic moments about the conventional aircraft body axes. As mentioned in reference [1], the major coupling which affects departure for the A-7 is provided by the kinematic terms

$$Z_p = \beta_0 \cos \alpha_0$$

and

(18)

$$Z_r = \beta_0 \sin \alpha_0$$

(see equation A.2), and the aerodynamic terms L'_α and N'_α . The kinematic terms arise due to the rotating coordinate system. The aerodynamic terms L'_α and N'_α as well as L'_β and N'_β are given as functions of α and β in Figure 9 of reference [1]. In the regime of α_{stall} (the stall angle of attack) these aerodynamic terms change sign and, therefore, an aerodynamic term stabilizing at small α can contribute to the tendency for departure at high α .

The A-7 has a typical nose slice departure. The influence of the main aerodynamic derivatives on this departure is discussed at length in reference [1]. Therefore, in the following, only points pertinent to finding the ideal model are discussed. This discussion is based upon many simulation runs as well as physical insight.

The influence of the "effective derivatives"

$$Z_p = \beta_0 \cos \alpha_0$$

and

(19)

$$Z_r = \beta_0 \sin \alpha_0$$

is not a major influence on the ideal model. Consequently, the ideal model contains these terms in their original form. In some of the simulations they were zeroed and the results were not changed significantly.

For static lateral stability in yaw, N'_β should be positive. Figure 9 in reference [1] shows that a negative N'_β can be expected for an angle of attack greater than 17 deg. For a flight with high angle of attack and small side-

slip, $N'_\beta < 0$ will increase the sideslip angle and will, therefore, increase N'_α which is, in the high angle of attack, always destabilizing. The result is the high yaw rate of the aircraft.

For static lateral stability in roll, L'_β should be negative. This means that a positive unwanted bank angle (right wing down) will induce a positive sideslip which causes a negative rolling moment with a decrease in bank angle as a result. A negative L'_β and a negative L'_α are the primary reasons for departure of the A-7 after a rapid yaw. The L'_α is negative if $\alpha > 23$ deg and the magnitude increases with sideslip.

The desired model for implicit model following was obtained with several goals in mind. The terms in the state model were chosen with the following results. The yaw moment due to sideslip should be positive, $N'_\beta > 0$. This will decrease the sideslip and therefore the destabilizing influence of L'_α and N'_α . The roll moment due to sideslip should be negative, $L'_\beta < 0$. An artificial static roll stability ($L'_\theta < 0$) was introduced instead of the "natural" static roll stability ($L'_\beta < 0$) which can contribute to departure.

The model was chosen with the above criteria satisfied. Simulations were conducted to help choose and to verify the model picked. These were compared to the actual A-7 in the same flight regime with significant improvement. The model matrix chosen for the flight condition about $\alpha_0 = 19^\circ$ and $\beta_0 = 6^\circ$ is

$$A_m = \begin{bmatrix} -0.0634 & -22.68 & 0 & -5.766 & 0 & 0 & 3.187 & -32.024 \\ -0.0009 & -0.323 & 1.0 & 0 & -0.0995 & -0.0338 & 0 & 0 \\ 0 & -3.577 & -0.386 & 0 & -0.0032 & 0.0025 & 0 & 0 \\ 0 & 0.0122 & 0 & -0.1062 & 0.3216 & -0.9469 & 0.1166 & 0.0129 \\ 0 & 0 & 0 & -1.0 & -0.849 & 0.3323 & -0.5 & 0 \\ 0 & 0 & 0 & 1.5 & 0.0193 & -0.1276 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0 & 0.3397 & 0 & -0.0116 \\ 0 & 0 & 1.0 & 0 & 0 & 0 & 0.0104 & 0 \end{bmatrix} \quad (20)$$

The available controls are aileron deflection, elevator deflection, and rudder deflection. The thrust is constant over the flight.

The next section yields the results.

V. Simulation Results

The control law was used to find the adaptive implicit model following law for the A-7 about the $\alpha_0 = 19^\circ$ and $\beta_0 = 6^\circ$ case. The model used is defined in equation (20). Measurements of all the states corrupted by white noise were assumed available. The standard deviations for the measurement noise are as follows: velocity perturbation, 1.25 ft/sec; angle of attack perturbation, 0.005 rads; pitch rate, 0.01 rad/sec; sideslip perturbation, 0.005 rad; roll rate, 0.01 rad/sec; yaw rate, 0.01 rad/sec; roll angle, 0.005 rad; and pitch angle perturbation, 0.005 rad. In each figure for this case, the nomenclature A-7 corresponds to an open loop ($\delta_e = \delta_R = \delta_a = 0$) simulation of the A-7 with an initial yaw rate of -10 deg/sec. As is shown in reference [1] a control input typical of an actual pilot response does not control the departure. The nomenclature M0 corresponds to an ideal model which the control is calculated to follow, and the nomenclature MF corresponds to the actual A-7 response with the closed loop control calculated in this paper. Equal control weighting was used.

It is assumed that the coefficients L'_α , N'_α and L'_p are uncertain. These coefficients have a major impact on lateral directional stability. The true value of the parameters were 3.09, -1.486, and -0.849, respectively. It was assumed that the possible parameter vectors, $\{L'_\alpha, N'_\alpha, L'_p\}^T$, were 1.9, 1.45, 1.0, 0.55 and 0.1 times the true values. That is, the possible parameter set was contained within a set of five possible values. The adaptation took place on these three aerodynamic coefficients.

Figures 1-8 show the radical difference in response using the control law derived in this paper. The actual A-7 shows a buildup of roll rate (Figure 5) followed by a rapid increase in bank angle (Figure 7). This type of behavior can indeed cause the loss of the aircraft. The responses using the closed loop control show that divergence is prevented. The response is very adequate using this control. Figures 9-11 show the control deflections required for divergence prevention for this lateral directional case.

Figure 12 shows the probabilities of each parameter being the true parameter. It takes less than 1.75 seconds to adapt upon the correct parameter with probability 0.8.

Case two was chosen to show the coupling between longitudinal and lateral dynamics. This case starts with a 5 deg/sec pitch rate initial condition (Figure 13). Figures 14-16 show a buildup in the A-7 response in the lateral modes due to the initial longitudinal pitch rate initial condition at the high angle of attack regime. Figure 17 vividly depicts the buildup of bank angle as the aircraft goes into departure without the control law used. These figures also show that with the control law applied with deflections in Figures 18-20 that the aircraft is prevented from departure. The control laws in essence yield a soft decoupling of modes while controlling the aircraft.

Figures 21-28 show the standard deviations of the estimation error for each of the conditional Kalman filters. The true parameter is number 3 in these figures.

Several additional simulations were conducted with different flight conditions as well as noise sequences. Each result is very similar to these typical results.

VI. Conclusions

The control law and philosophy of flight control developed within is shown to be an excellent method of divergence prevention in the high angle of attack regime. The control laws found by finding steady state gains for the filters as well as the control gains may be readily implemented along with the probability estimator for uncertain coefficients. The control law was simulated in detail and shows excellent promise for control in a dangerous flight regime.

The model development philosophy points out many key problems in the high angle of attack regime. This information of itself is valuable.

References

1. D.E. Johnston and J.R. Hogge, "Nonsymmetric Flight Influence on High Angle of Attack Handling and Departure," AIAA Journal of Aircraft, Vol. 13, No. 2, pp. 112-118, February 1976.
2. W. Bihrl, Jr. and R.C. Meyer, "F-14A High Angle of Attack Characteristics," AIAA Journal of Aircraft, Vol. 13, No. 8, pp. 576-583, August 1976.
3. R. Weissman, "Preliminary Criteria for Predicting Departure Characteristics/Spin Susceptibility of Fighter-Type Aircraft," AIAA Journal of Aircraft, Vol. 10, No. 4, pp. 214-219, April 1973.
4. R.F. Stengel and P.W. Berry, "Stability and Control of Maneuvering High Performance Aircraft," Proceedings of the 1976 AIAA Guidance and Control Conference, AIAA Paper 76-1973, pp. 485-492.
5. K.L. Johnson and T.B. Willen, "Anticipated Spin Susceptibility Characteristics of the A-10 Aircraft," AIAA Journal of Aircraft, Vol. 13, No. 6, pp. 406-412, June 1976.
6. W.E. Hall, Jr., N.K. Gupta, and R.G. Smith, "Identification of Aircraft Stability and Control Coefficients for the High Angle of Attack Regime," Systems Control, Inc., Technical Report #2, March 1974.
7. R.B. Asher, C.S. Sims, and H.R. Sebesta, "Optimal Open-Loop Feedback Control for Linear Systems with Unknown Parameters," Information Sciences, Vol. 11, No. 3, pp. 265-277, 1976.
8. D.G. Lainiotis, "Partitioning: A Unifying Framework for Adaptive Systems, II: Control," Proceedings of the IEEE, Vol. 64, No. 8, pp. 1182-1198, August 1976.
9. D.G. Lainiotis, "Optimal Adaptive Estimation: Structure and Parameter Adaptation," IEEE Transactions on Automatic Control, Vol. AC-16, No. 2, pp. 160-170, April 1971.
10. E. Kreindler and D. Rothschild, "Model Following in Linear Quadratic Optimization," AIAA Journal, Vol. 14, No. 7, pp. 835-842, July 1976.
11. D.E. Johnston, I.L. Ashkenas, and J.R. Hogge, "Investigation of Flying Qualities of Military Aircraft at High Angles of Attack, Vol. I: Technical Results," AFFDL-TR-74-61, June 1974.
12. D.E. Johnston, J.R. Hogge, and G.L. Teper, "Investigation of Flying Qualities of Military Aircraft at High Angles of Attack, Vol. II: Appendices," AFFDL-TR-74-61, June 1974.

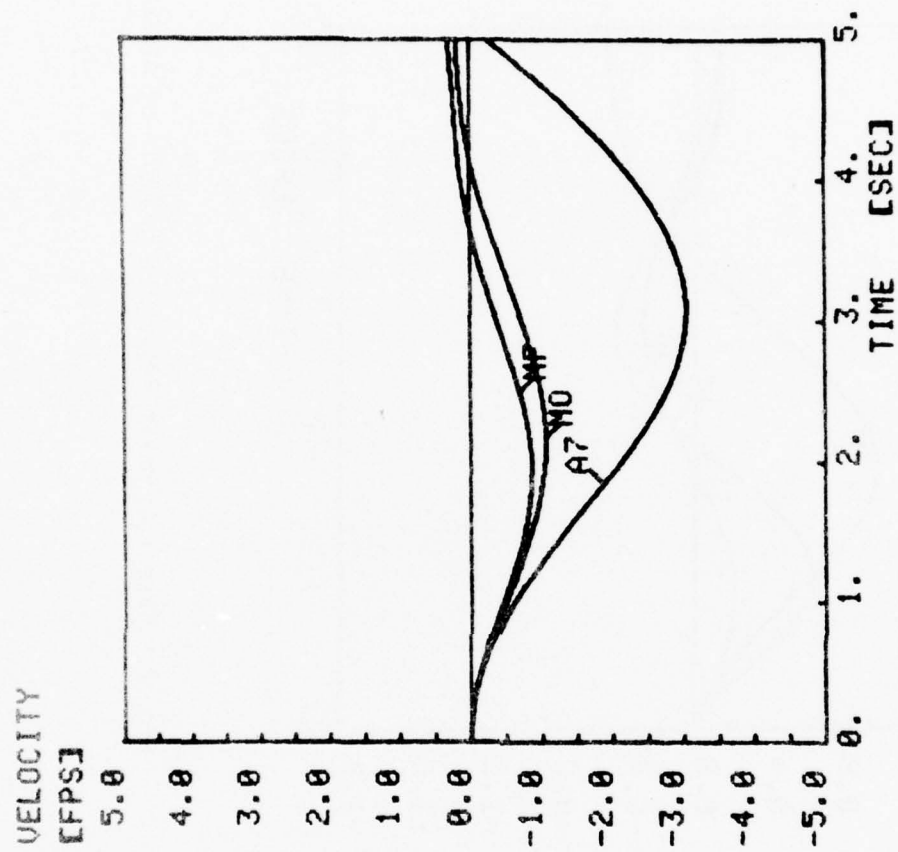


Fig. 1. Case 1 Aircraft Responses in Velocity

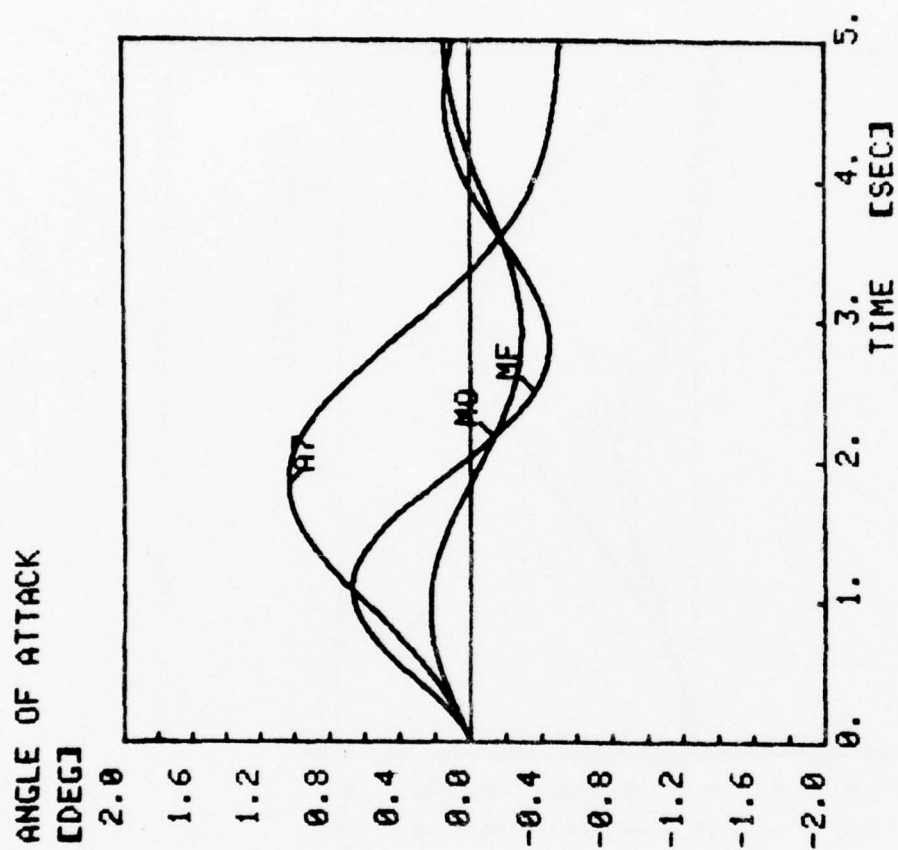


Fig. 2. Case 1 Aircraft Responses in Angle of Attack

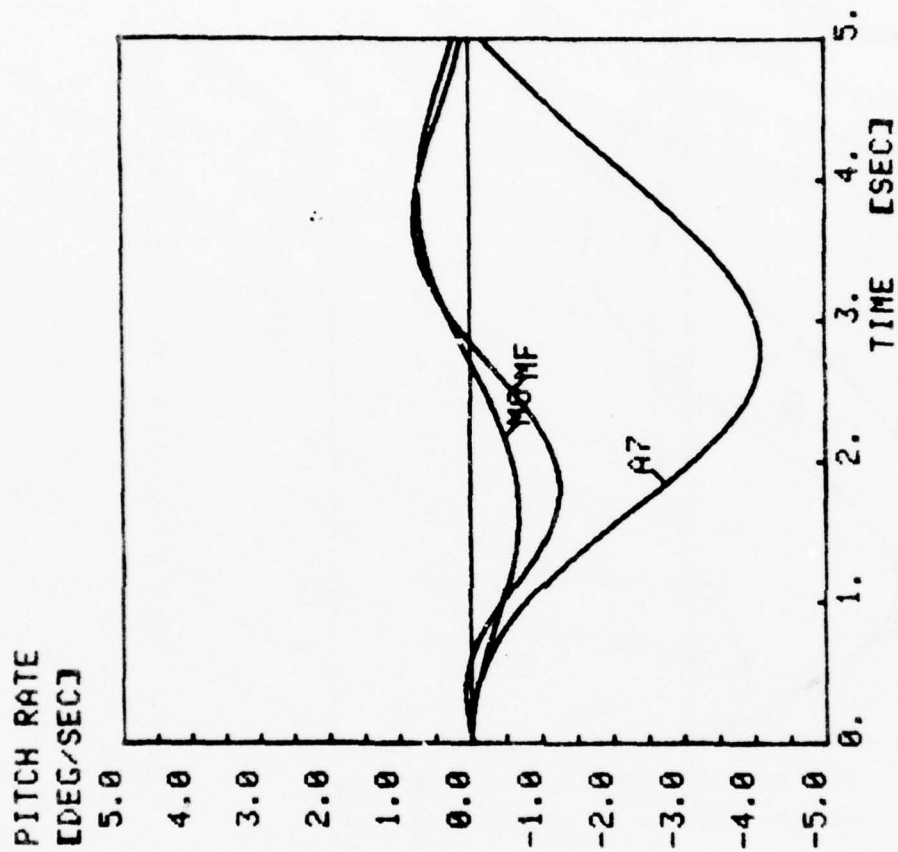


Fig. 3. Case 1 Aircraft Responses in Pitch Rate

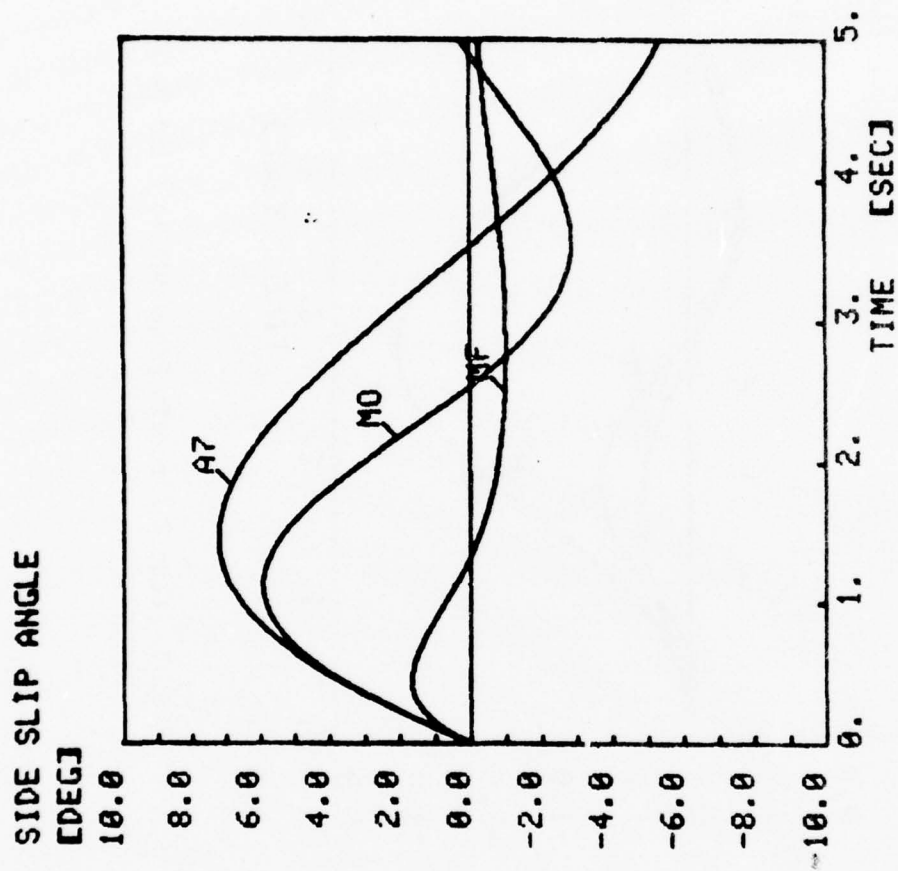


Fig. 4. Case 1 Aircraft Responses in Sideslip Angle

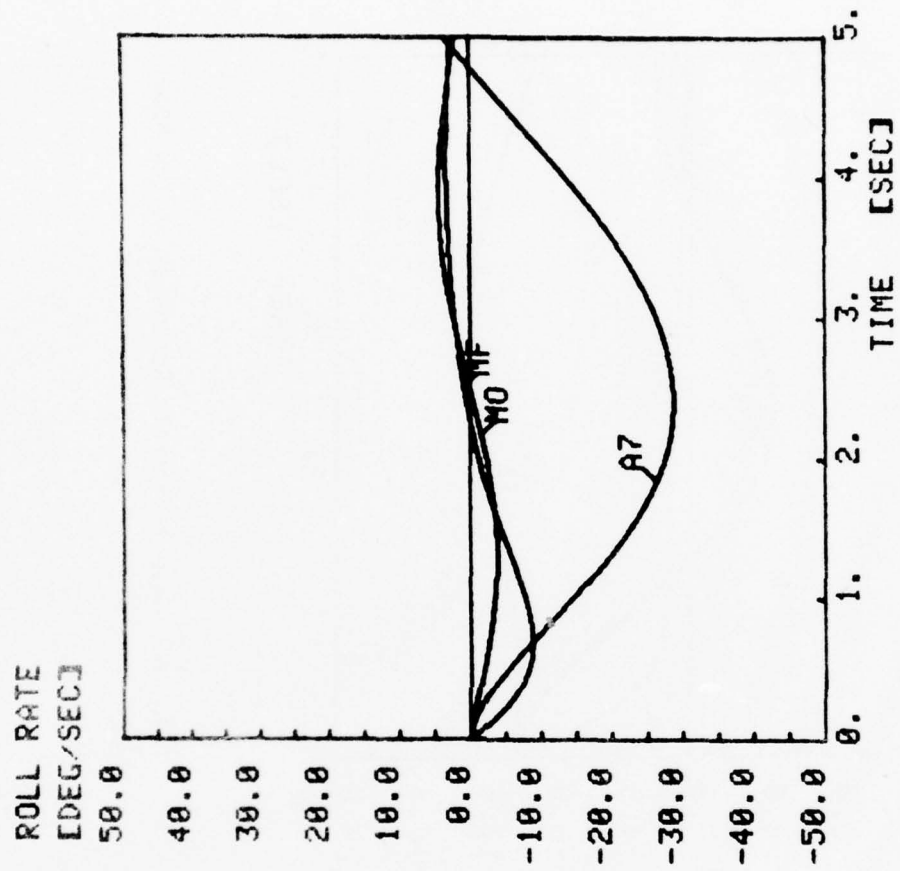


Fig. 5. Case 1 Aircraft Responses in Roll Rate

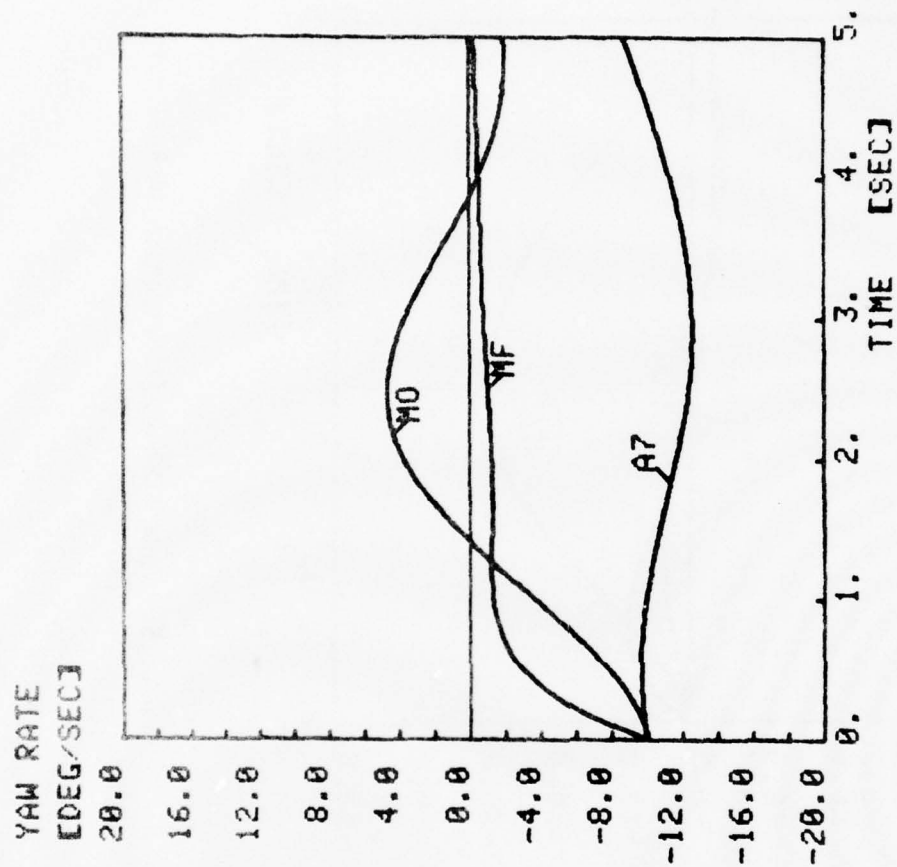


Fig. 6. Case 1 Aircraft Responses in Yaw Rate

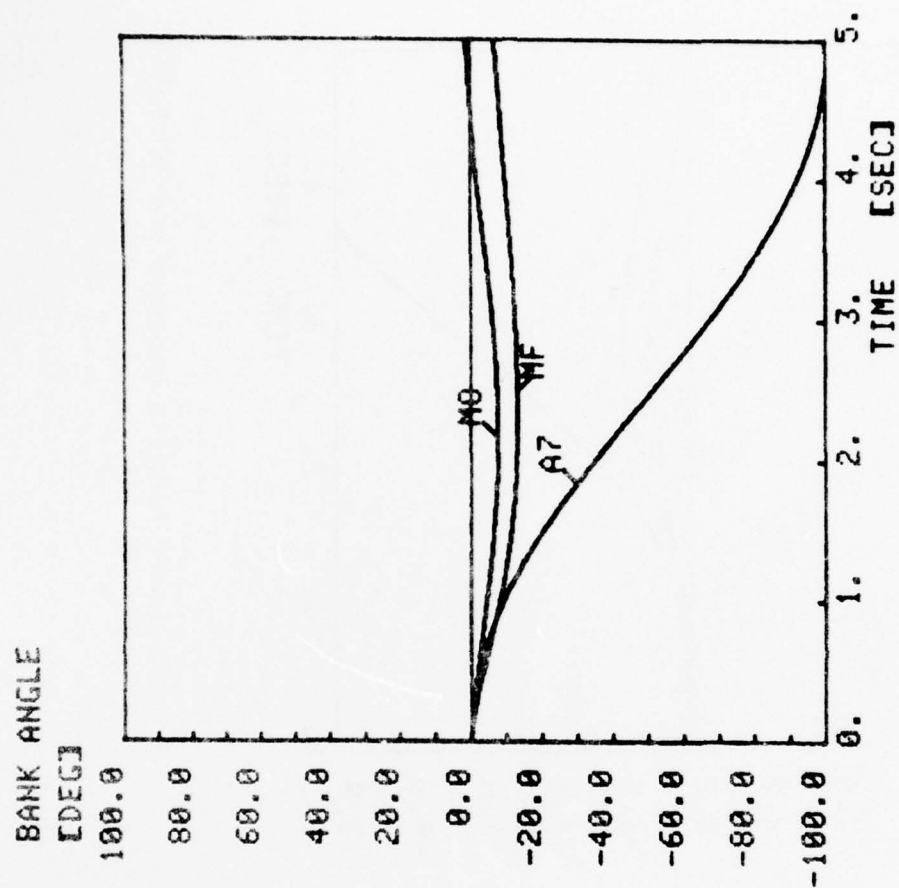


Fig. 7. Case 1 Aircraft Responses in Bank Angle

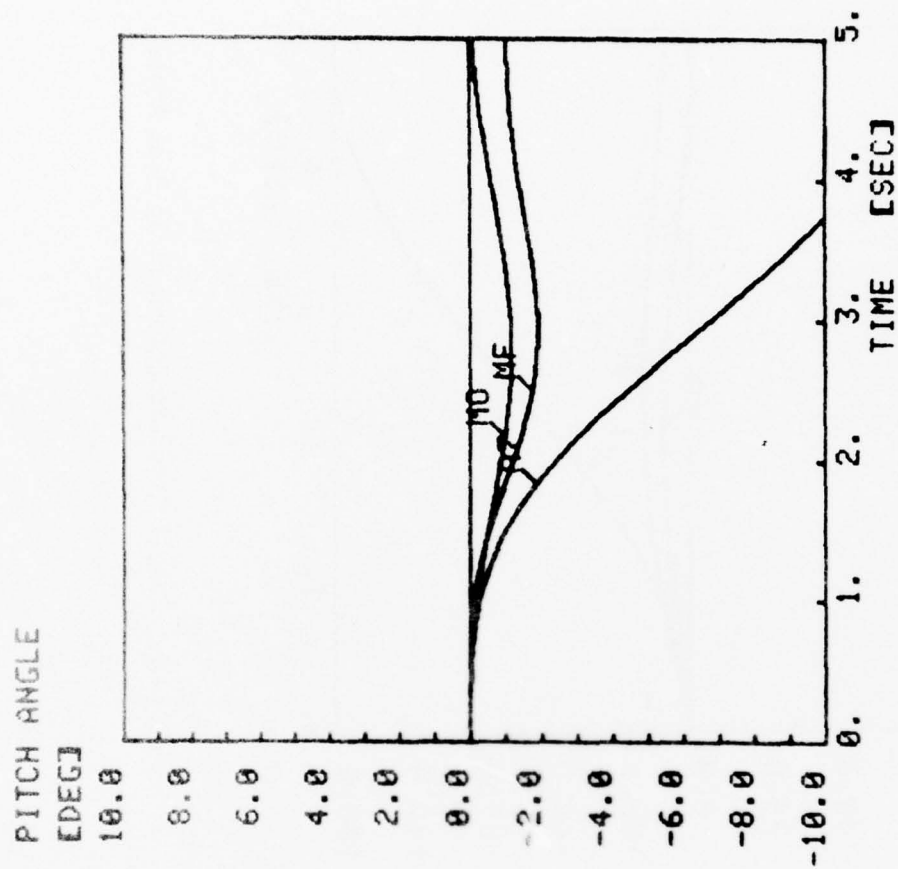


Fig. 8. Case 1 Aircraft Responses in Pitch Angle

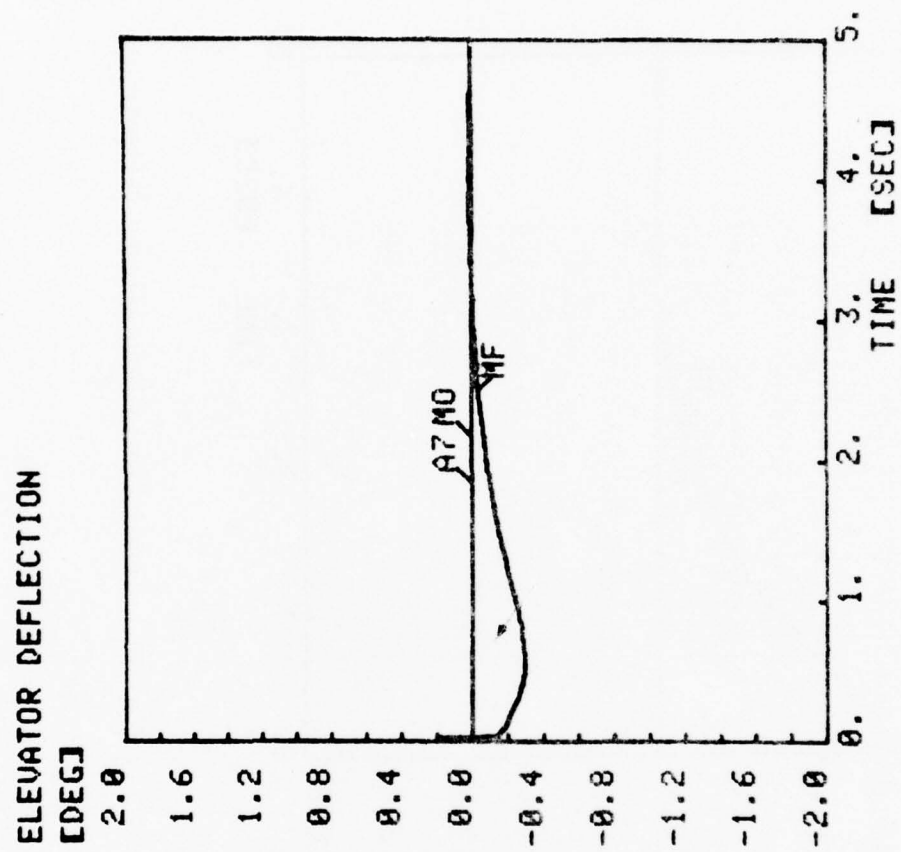


Fig. 9. Case 1 Closed Loop Elevator Deflection

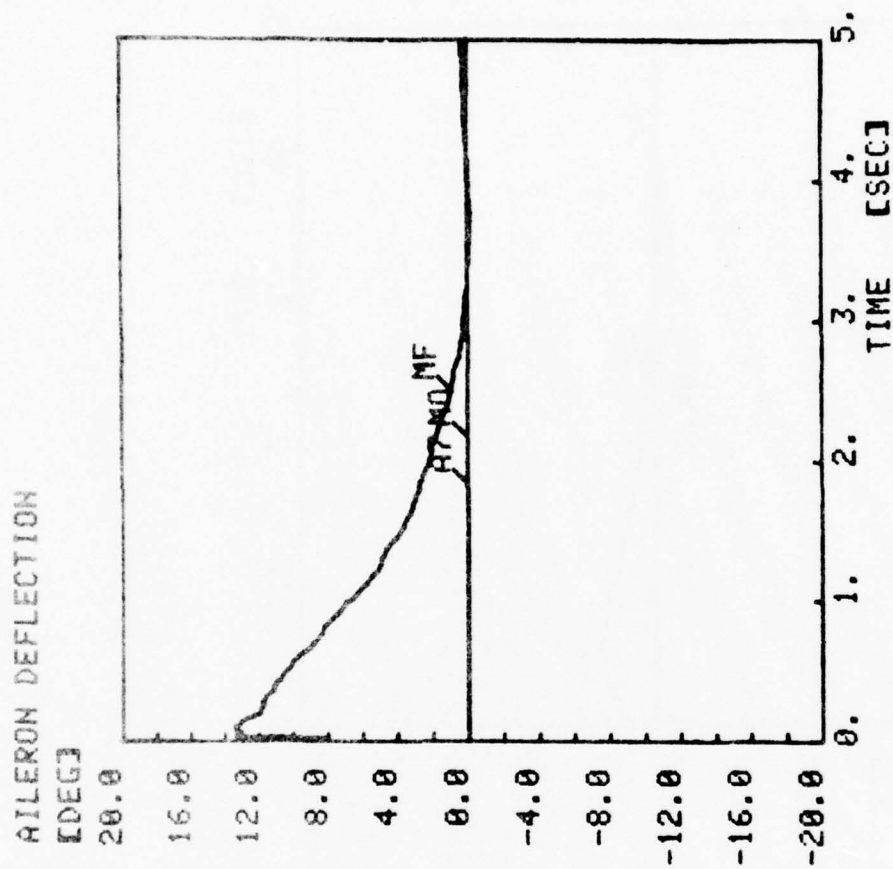


Fig. 10. Case 1 Closed Loop Aileron Deflection

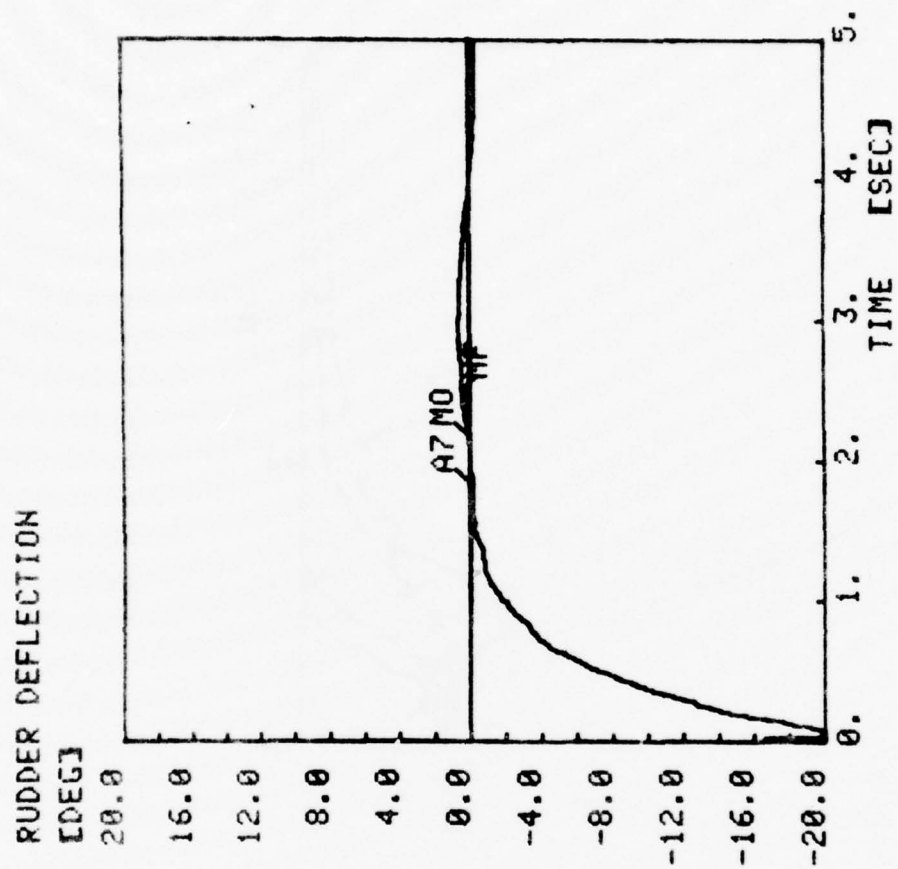


Fig. 11. Case 1 Closed Loop Rudder Deflection

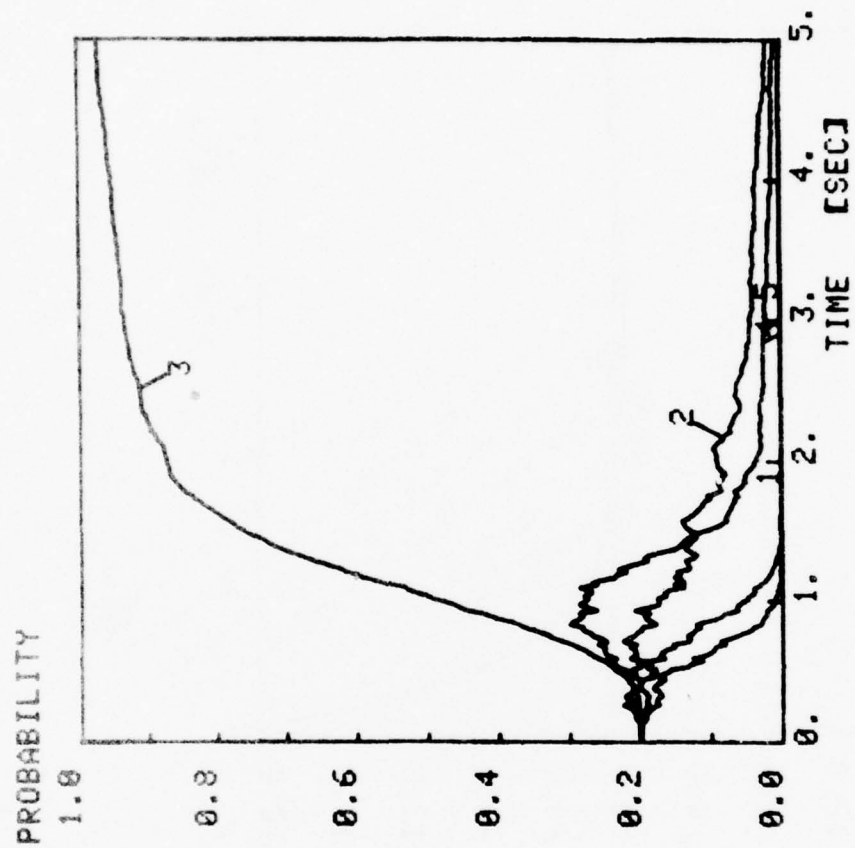


Fig. 12. Probabilities vs Time

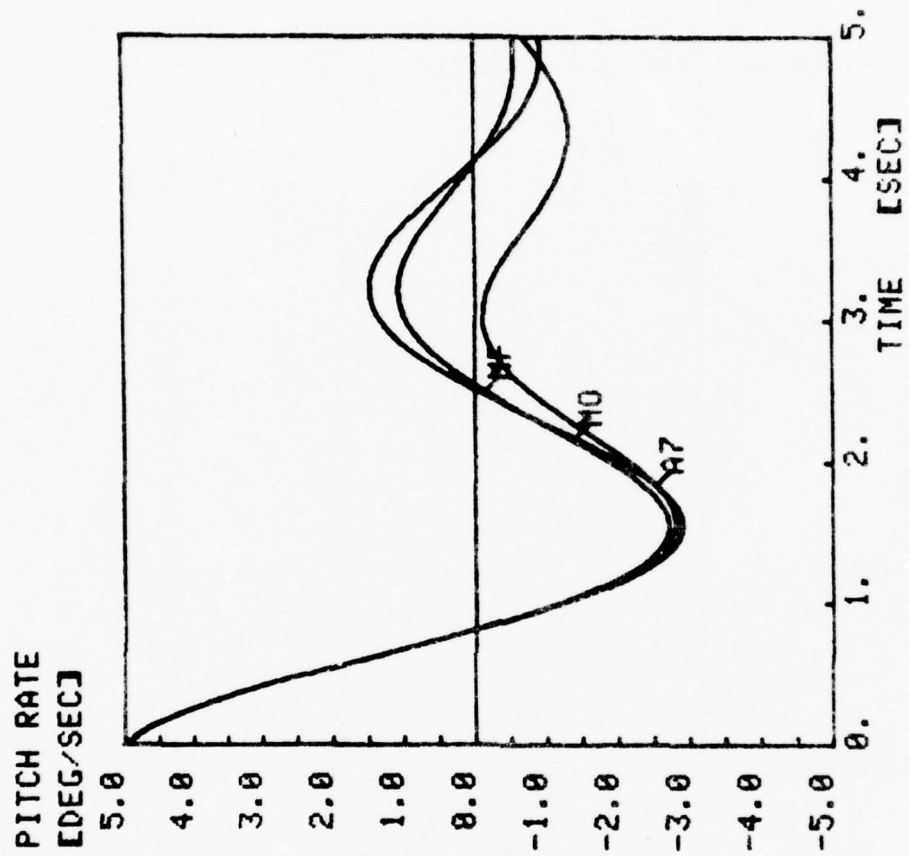


Fig. 13. Case 2 Aircraft Responses in Pitch Rate

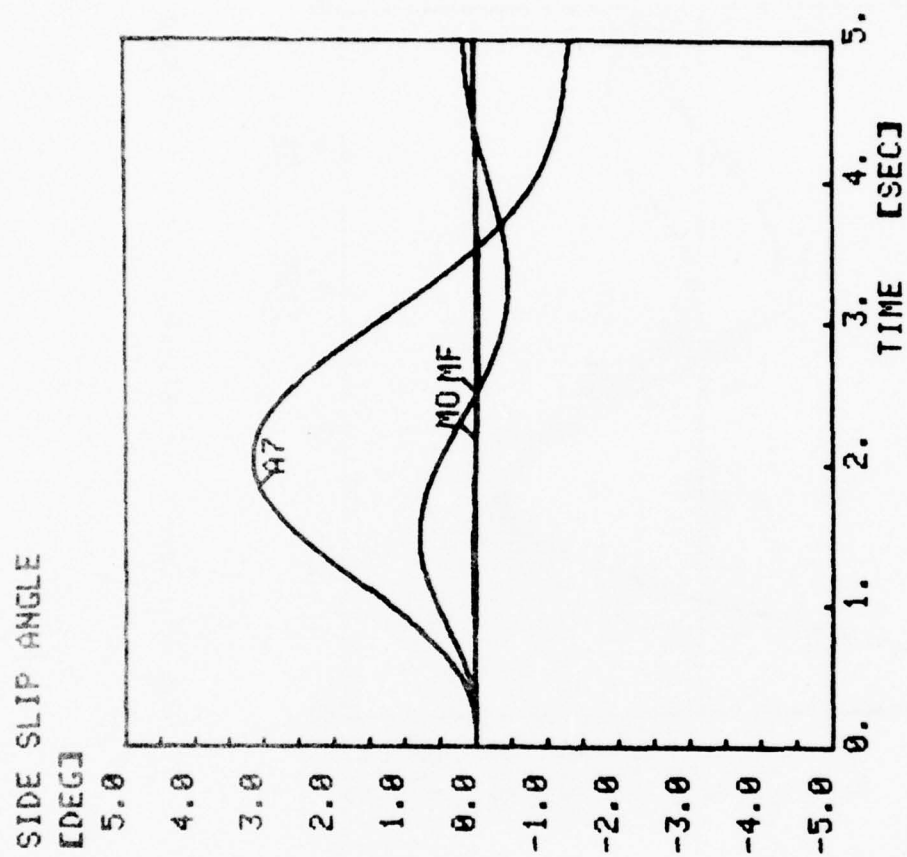


Fig. 14. Case 2 Aircraft Responses in Sideslip

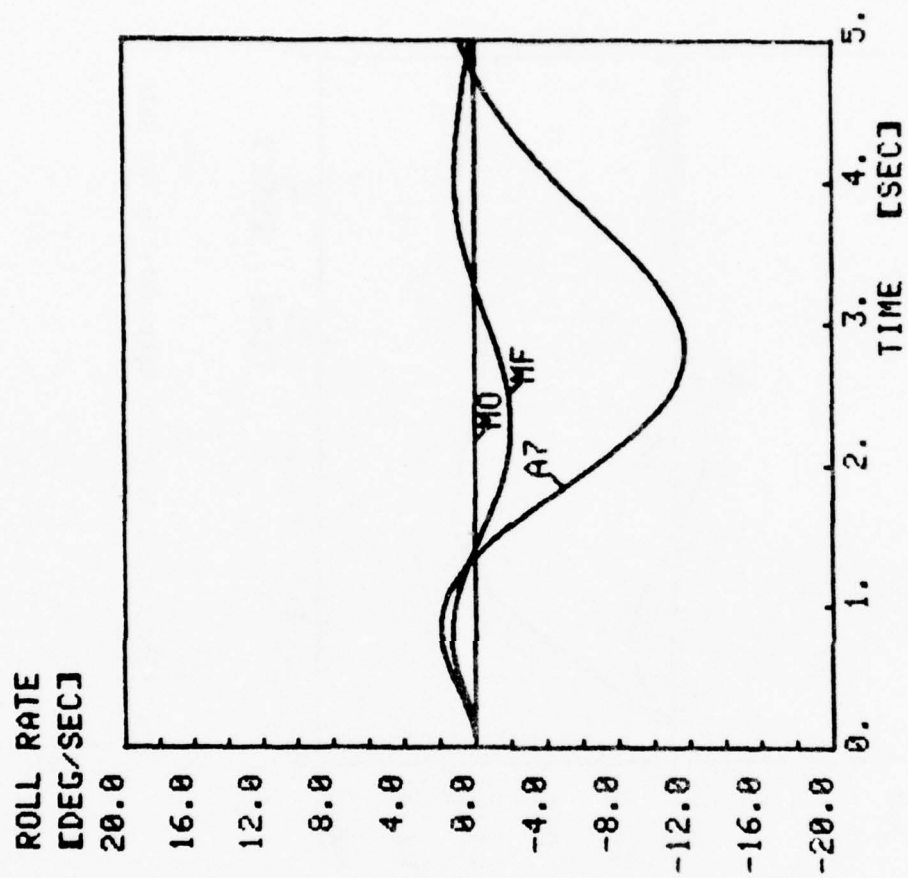


Fig. 15. Case 2 Aircraft Responses in Roll Rate

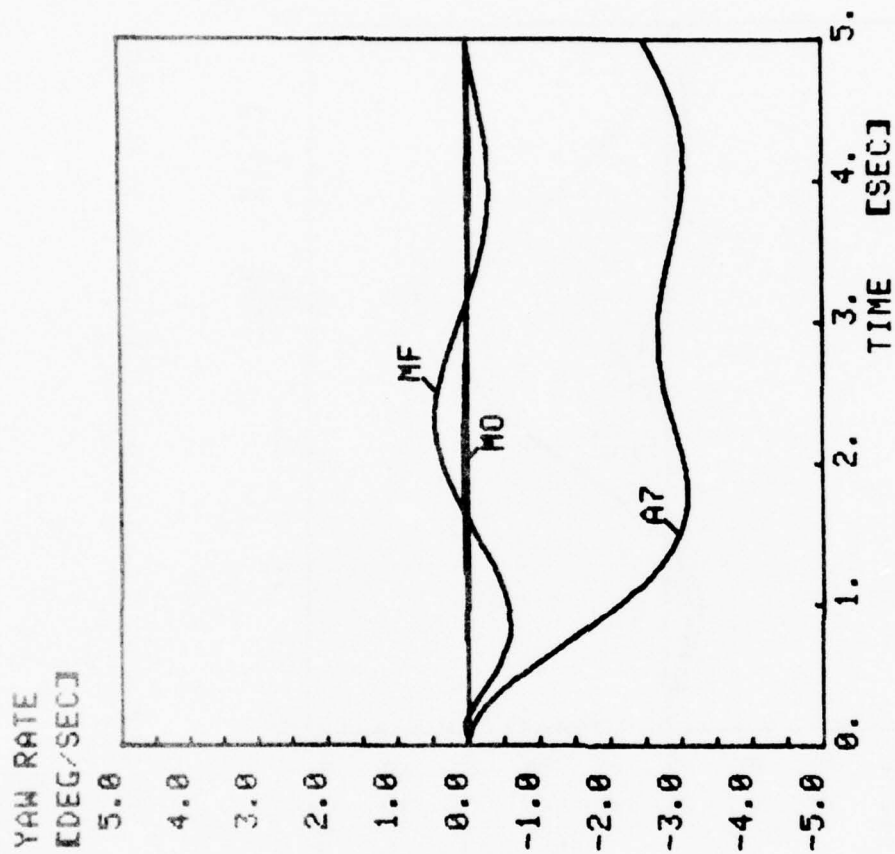


Fig. 16. Case 2 Aircraft Responses in Yaw Rate

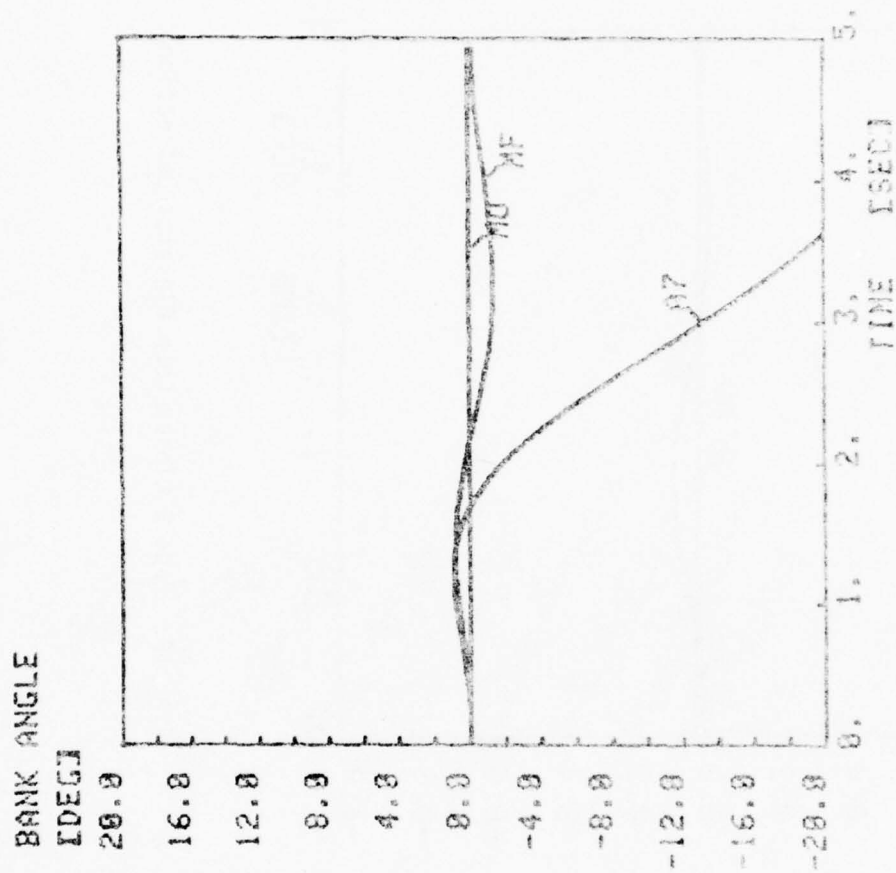


Fig. 17. Case 2 Aircraft Responses in Bank Angle

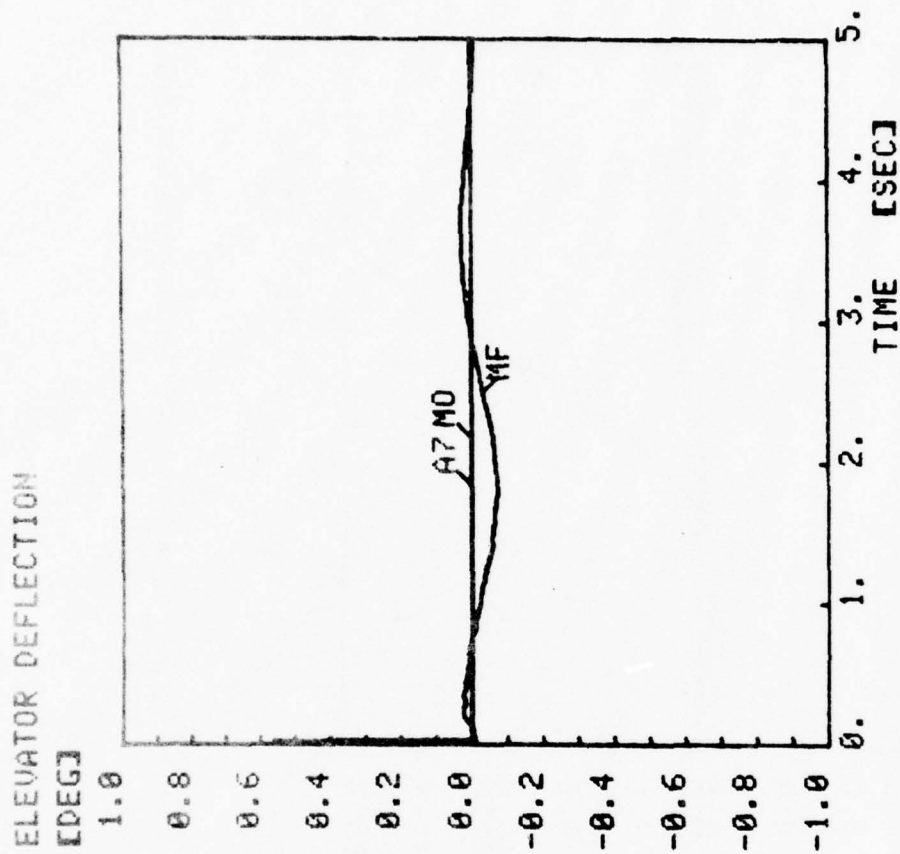


Fig. 18. Case 2 Closed Loop Elevator Deflection

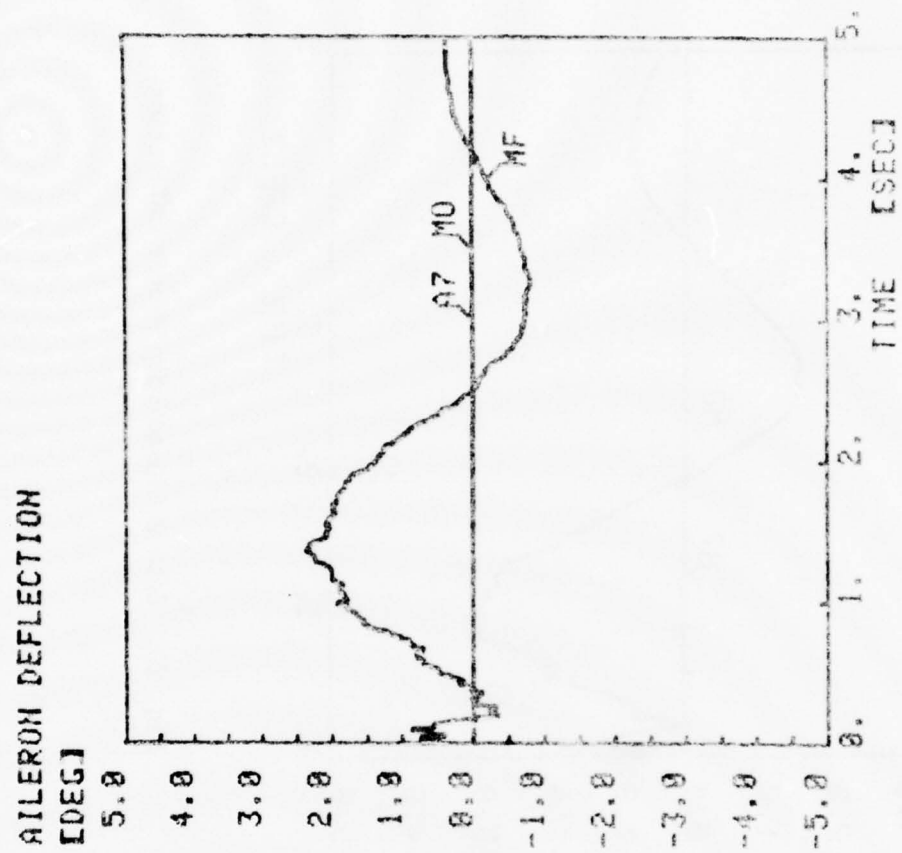


Fig. 19. Case 2 Closed Loop Aileron Deflection

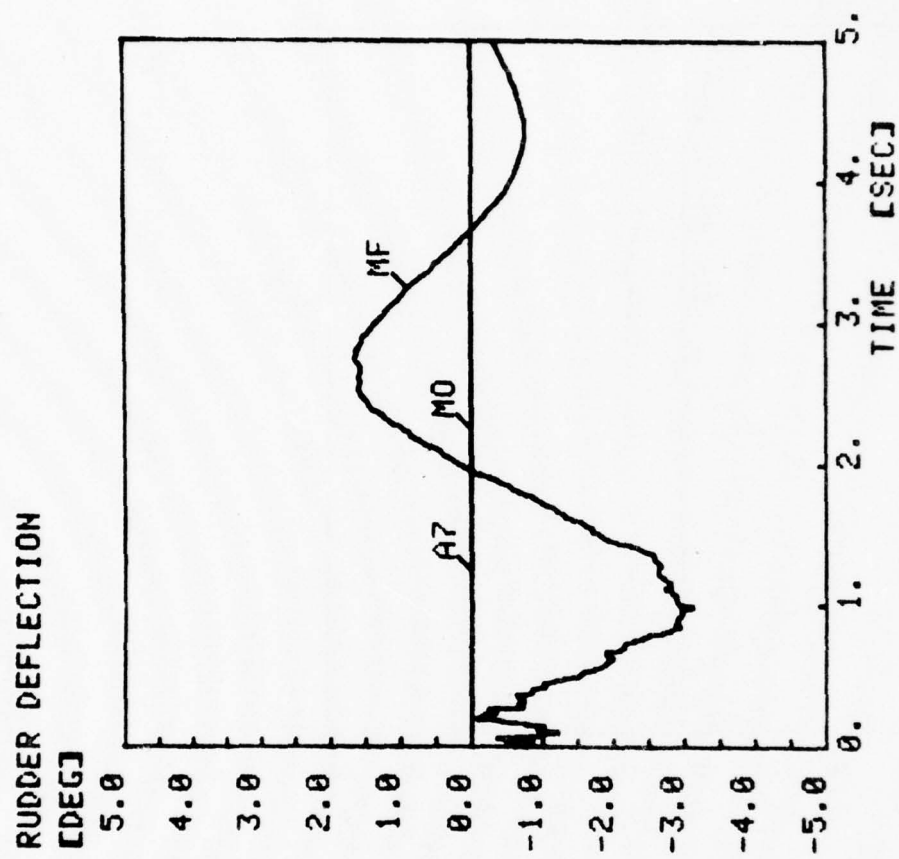


Fig. 20. Case 2 Closed Loop Rudder Deflection

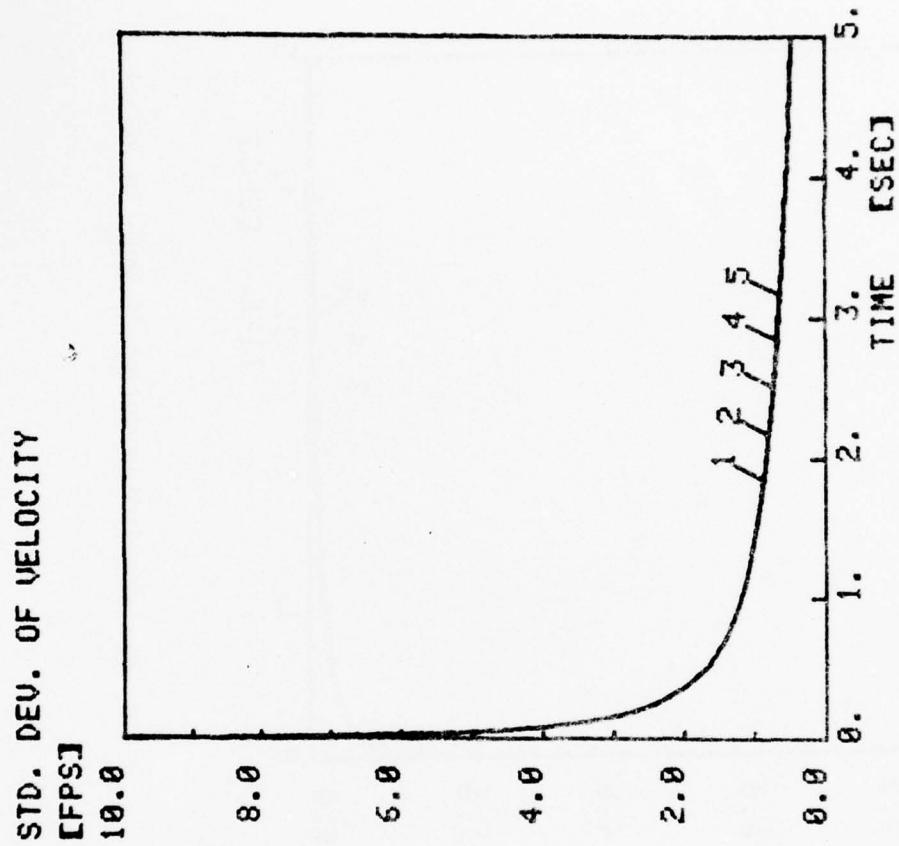


Fig. 21. Standard Deviation of Velocity

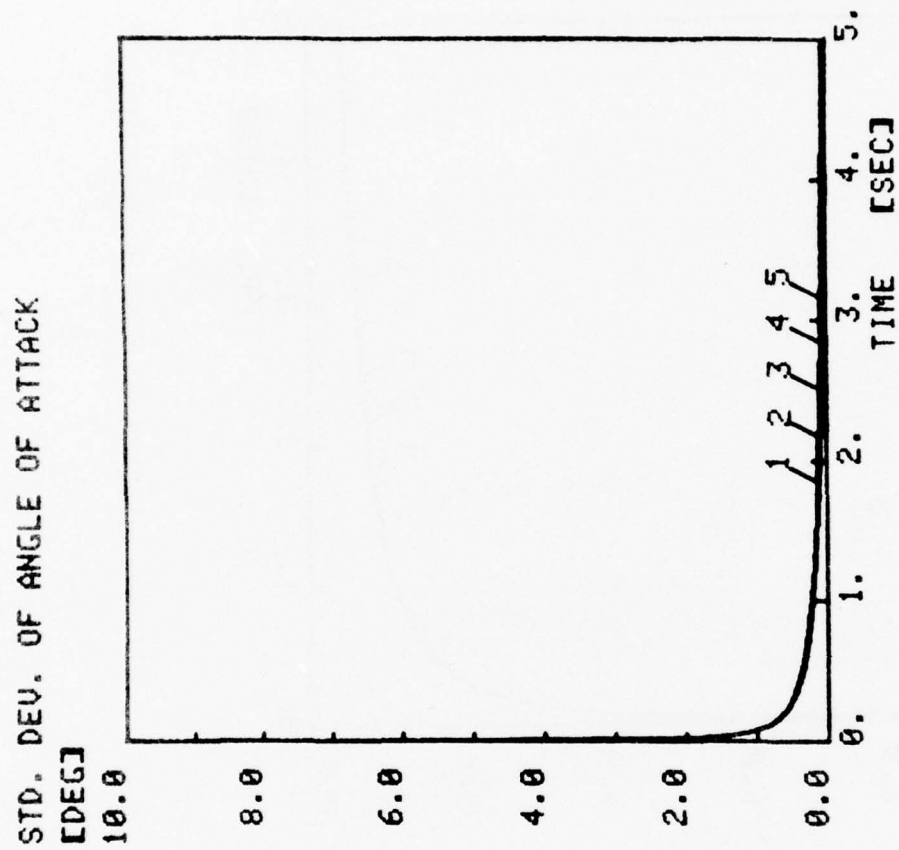


Fig. 22. Standard Deviation of Angle of Attack

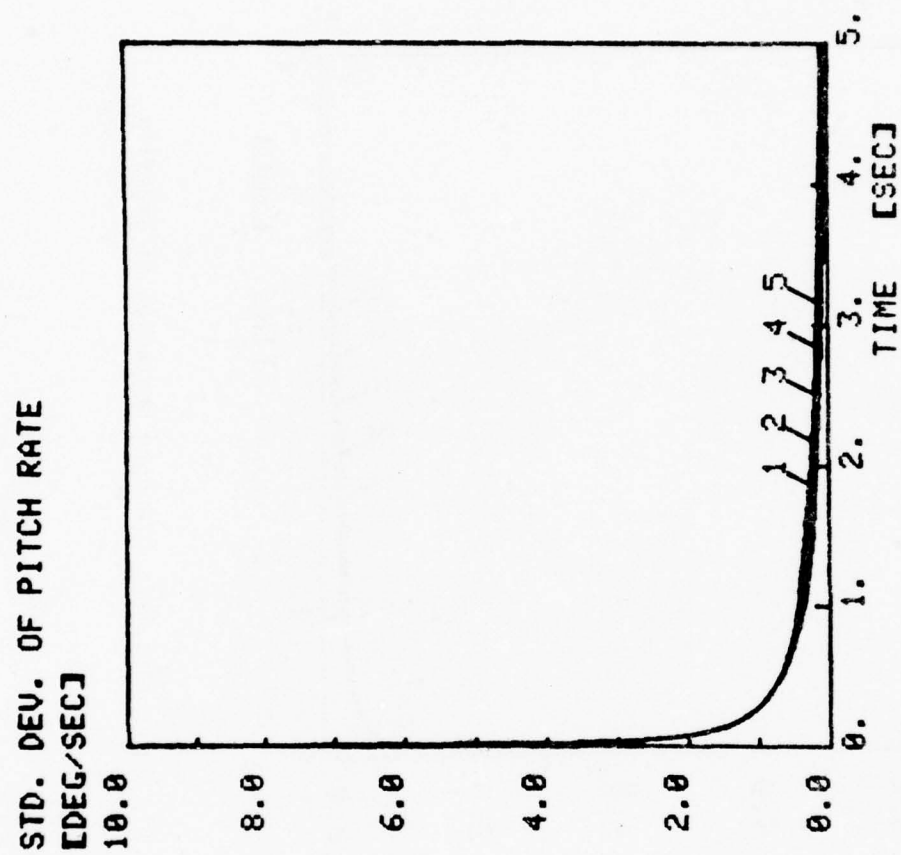


Fig. 23. Standard Deviation of Pitch Rate

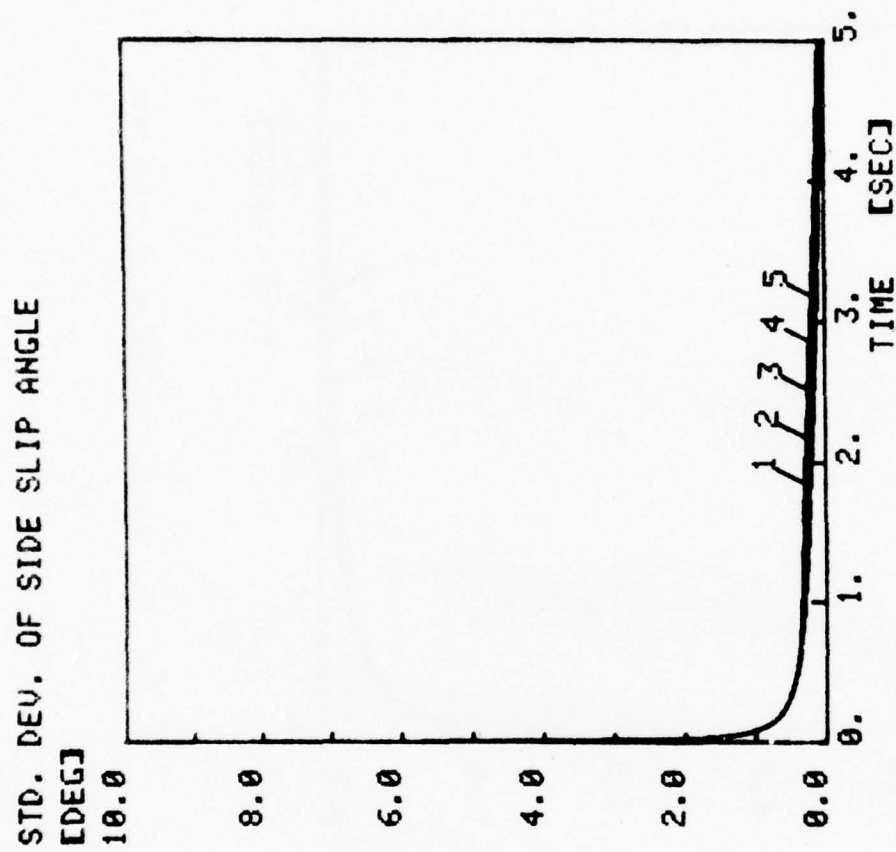


Fig. 24. Standard Deviation of Sideslip

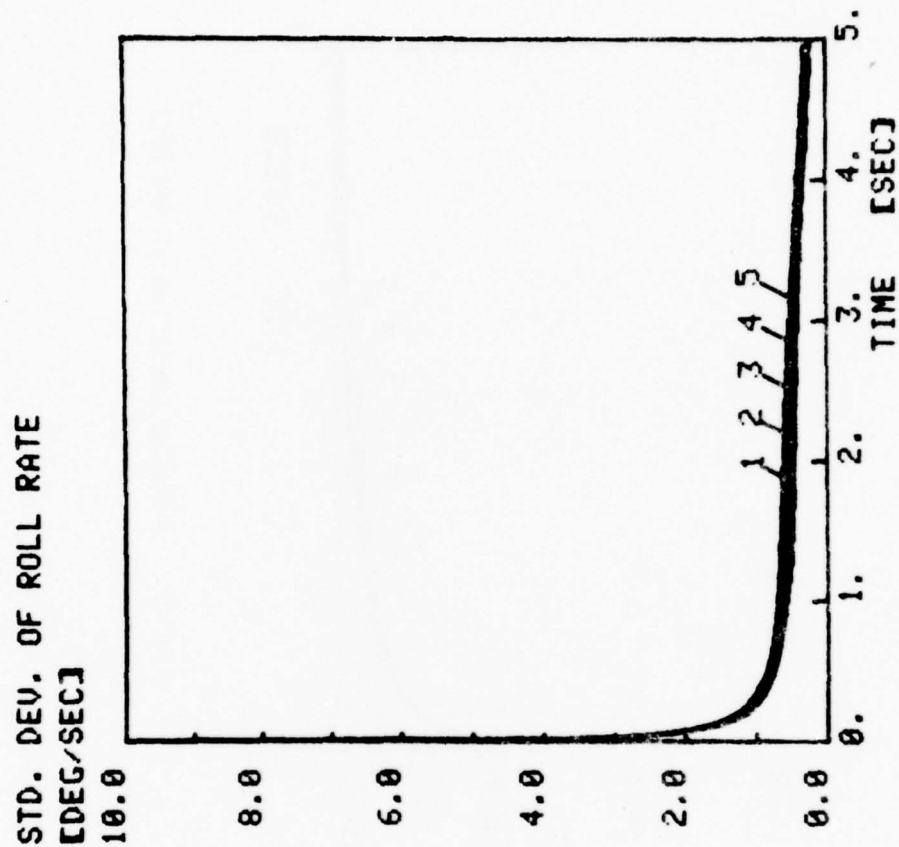


Fig. 25. Standard Deviation of Roll Rate

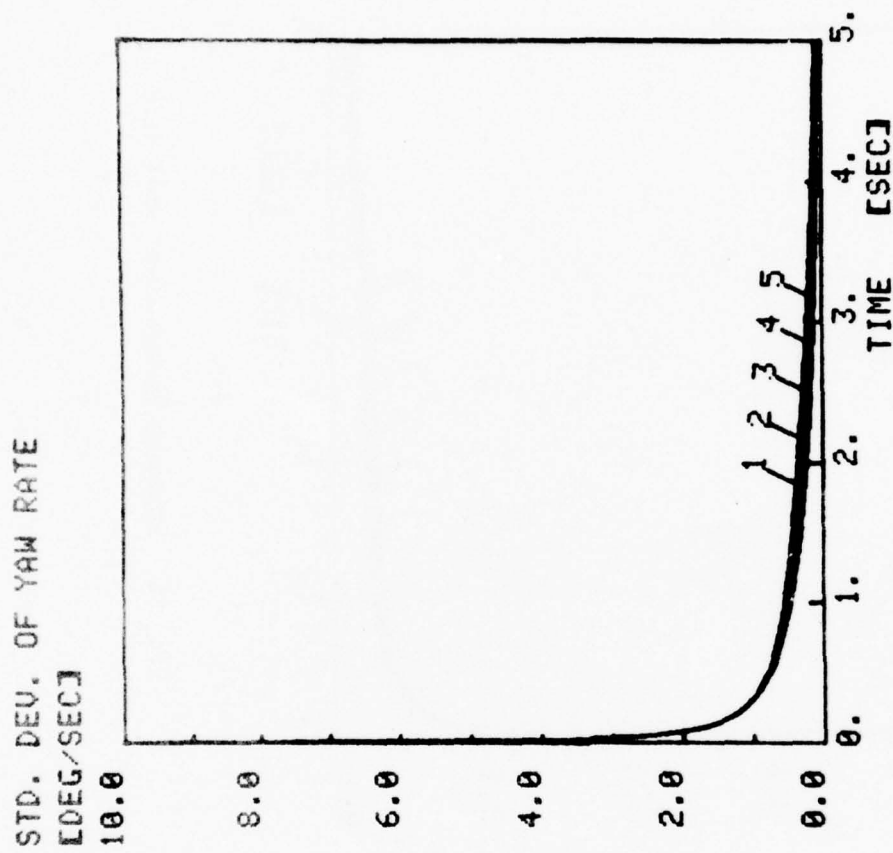


Fig. 26. Standard Deviation of Yaw Rate

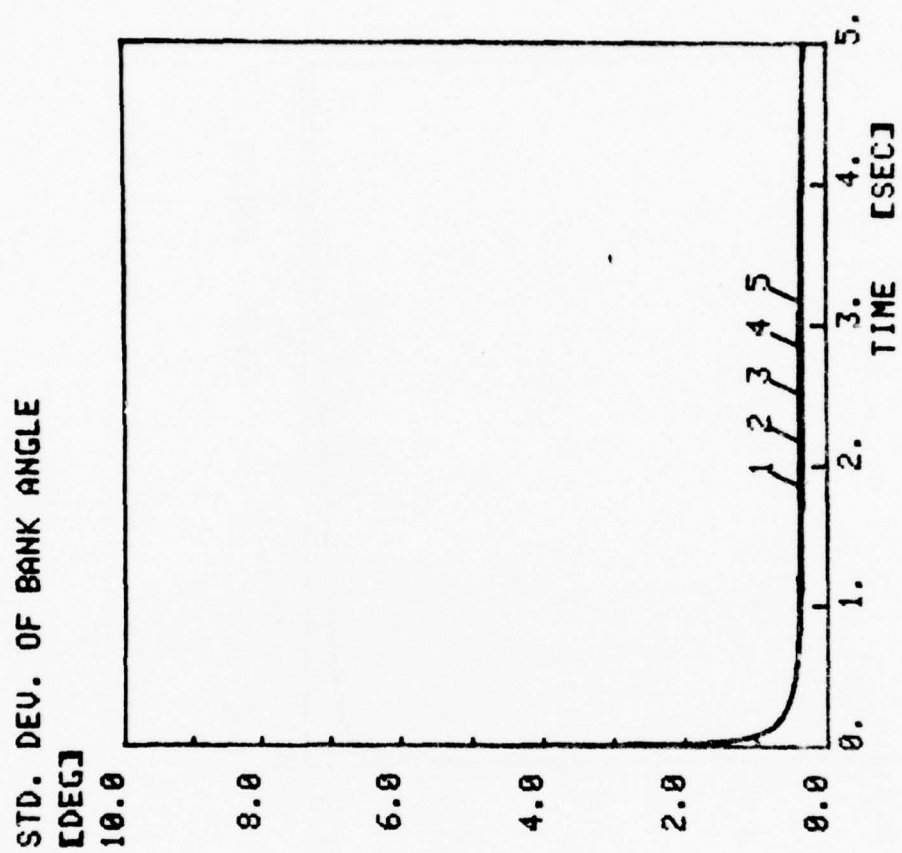


Fig. 27. Standard Deviation of Bank Angle

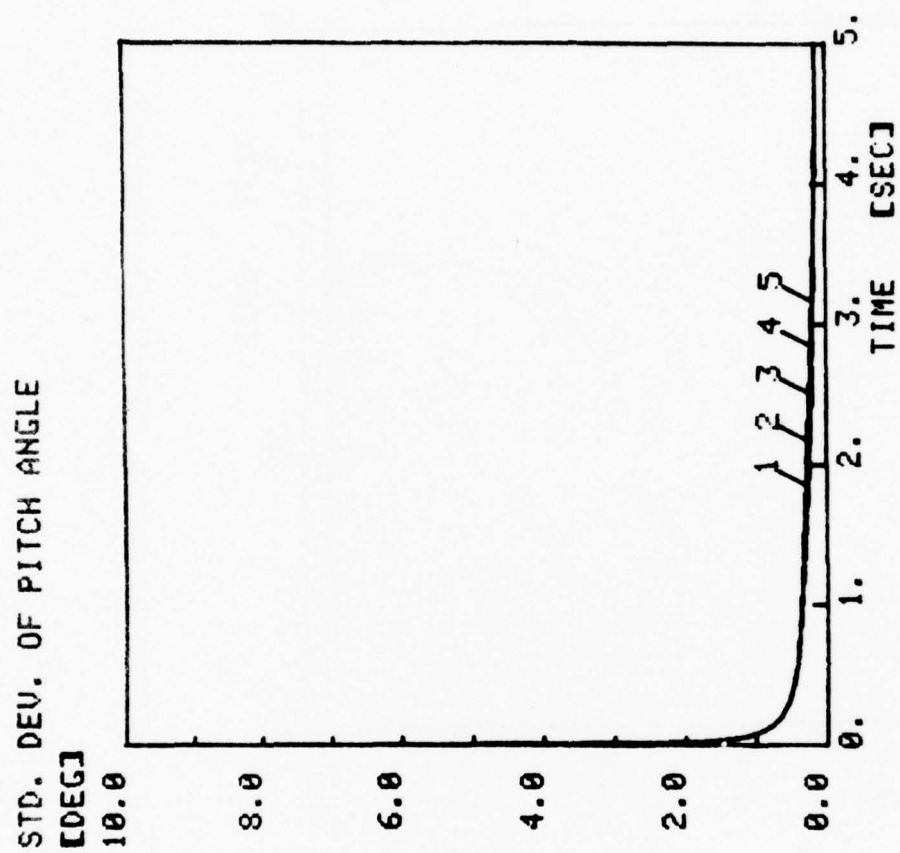


Fig. 28. Standard Deviation of Pitch Angle

APPENDIX A Equations of Motion

The equations of motion for the aircraft are

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}u \quad (\text{A.1})$$

where

$$\mathbf{A} = \begin{bmatrix} X_u & X_w U_0 & 0 & x_\beta & 0 & 0 & g\beta_0 \cos \theta_0 & -g \cos \gamma_0 \\ Z_u/U_0 & Z_w & +1 & 0 & -\beta_0 \cos \alpha_0 & -\beta_0 \sin \alpha_0 & 0 & 0 \\ 0 & M_\alpha + M_w Z_\alpha & M_a + M_\alpha & M_\beta & -r_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & +r_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & 0 & 0 \\ 0 & Y_\alpha/U_0 & 0 & Y_v & \sin \alpha_0 & -\cos \alpha_0 & \frac{g'}{U_0} \cos \theta_0 & \frac{g}{U_0} \beta_0 \sin \gamma \\ 0 & L'_\alpha & 0 & L'_\beta & L'_p & L'_r & 0 & 0 \\ 0 & N'_\alpha & 0 & N'_\beta & N'_p & N'_r & 0 & 0 \\ 0 & 0 & 0 & 0 & +1 & \tan \theta_0 & 0 & \frac{r_0}{\cos^2 \theta_0} \\ 0 & 0 & +1 & 0 & 0 & 0 & r_0 & 0 \end{bmatrix} \quad (\text{A.2})$$

and

$$B = \begin{bmatrix} X_{\delta e} & 0 & X_{\delta r} \\ Z_{\delta e}/U_0 & 0 & 0 \\ M_{\delta e} & 0 & 0 \\ Y_{\delta e}/U_0 & 0 & Y_{\delta r}/U_0 \\ L'_{\delta e} & L'_{\delta a} & L'_{\delta r} \\ N'_{\delta e} & N'_{\delta a} & N'_{\delta r} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (A.3)$$

with the state vector as given in equation (1) and where X_i, Y_i, Z_i is the expression for $a(\)/a_i$, u_v is the perturbed total linear velocity, U_0 is the free stream velocity, r_0 is nominal yaw rate, p_0 is the nominal roll rate, g is gravity, M_i is $\partial M/\partial i$, γ_0 is the flight path angle, and δe , δa , and δr are the elevator, aileron and rudder deflections, respectively. The aerodynamic coefficients may be found in references [11, 12].

The values for the matrices about the α, β point chosen for the simulations are

$$A = \begin{bmatrix} -.0634 & -22.68 & 0 & -5.766 & 0 & 0 & +3.187 & -32.024 \\ -.00087 & -.323 & +1. & 0 & -.0995 & -.0338 & 0 & 0 \\ 0 & -3.577 & -.386 & -.9 \times 10^{-7} & -.00818 & +.0025 & 0 & 0 \\ 0 & +.0122 & 0 & -.1062 & +.3216 & -.9469 & +.1166 & +.0129 \\ 0 & +3.09 & 0 & -4.45 & -.849 & +.3323 & 0 & 0 \\ 0 & -1.486 & 0 & -.1885 & +.0193 & -.1276 & 0 & 0 \\ 0 & 0 & 0 & 0 & +1. & -.3396 & 0 & -.0116 \\ 0 & 0 & +1. & 0 & 0 & 0 & +.0104 & 0 \end{bmatrix} \quad (A.4)$$

$$B = \begin{bmatrix} -0.1025 & 0 & 0.698 \\ -0.057 & 0 & 0 \\ -2.92 & 0 & 0 \\ -0.0037 & 0 & 0.0255 \\ -0.292 & 0.431 & 1.4 \\ 0.1095 & 0.031 & -0.998 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (A.5)$$

For the C matrix in the measurement equation (compare Eq. 2, p. 3), a unit matrix (8x8) was used in the simulation.

APPENDIX B
Model Following Program

-Part 1-
Control Calculations and Flight Simulations

1. Important Parameters

INT(1): - not used

INT(2): (Case) 1: open loop, ramp δ_e
2: close loop $\phi \rightarrow \delta_e$, step ϕ_c
3: as case 2 & $u \rightarrow \delta_T$
4: open loop $r(t=0) = -10$ deg/sec
5: open loop $q(t=0) = 5$ deg/sec

INT(3): MODE -2:
-1: run preparation
0: simulation run
+1: end of simulation

INT(4): NO 1: A7, MO & MF simulation
2: MO & MF simulation
3: MF simulation
NF 1: A7 output file
2: MO output file
3: MF output file

INT(5): 0: A7 derivatives const
1: nonlinear simulation (in older program version only)

INT(6): KEN 0: first call of several subroutines
1: not first call of several subroutines

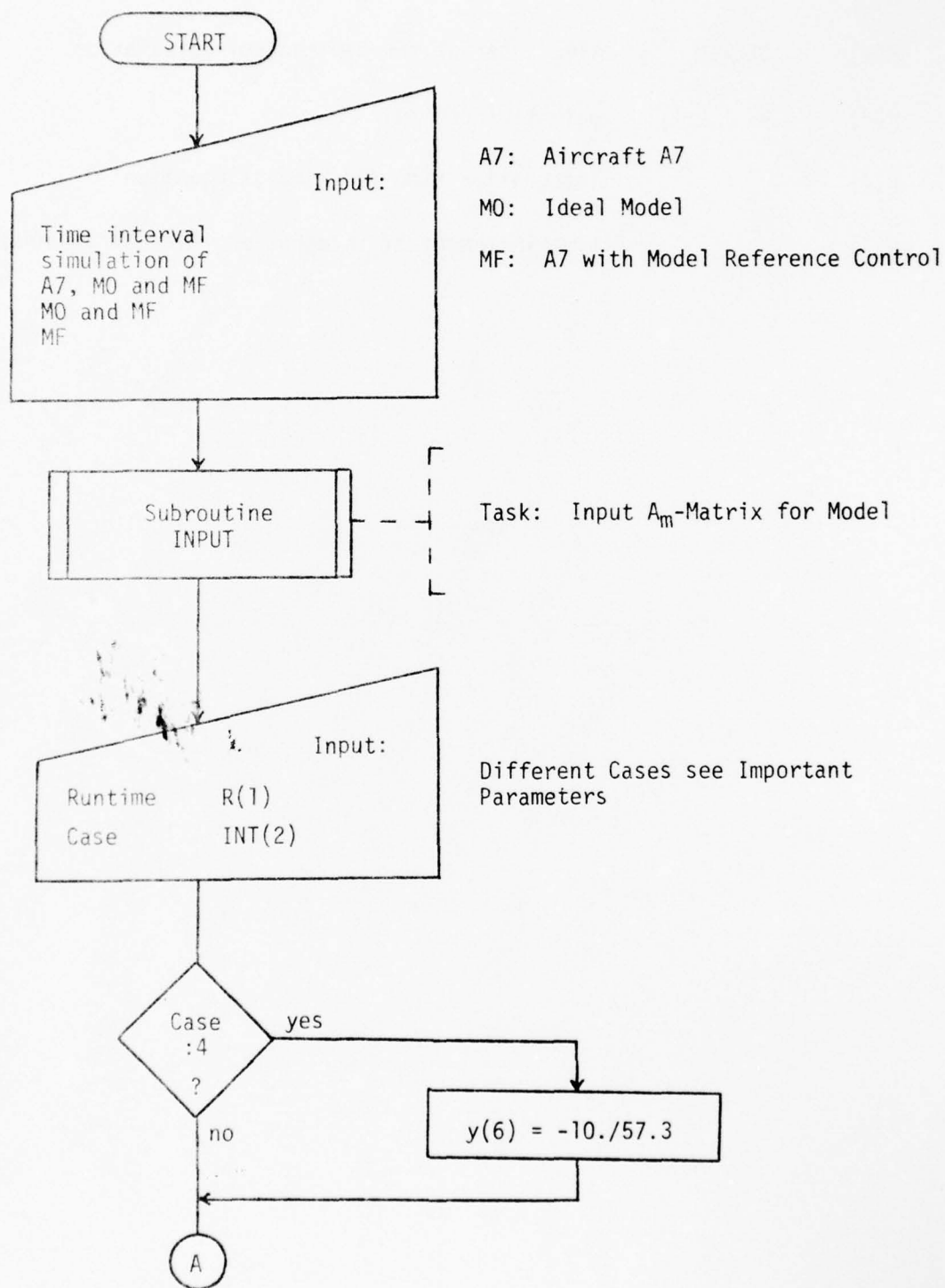
INT(7): IHALF max number of bisections of the initial time increment in $\dot{x}$ - calculation

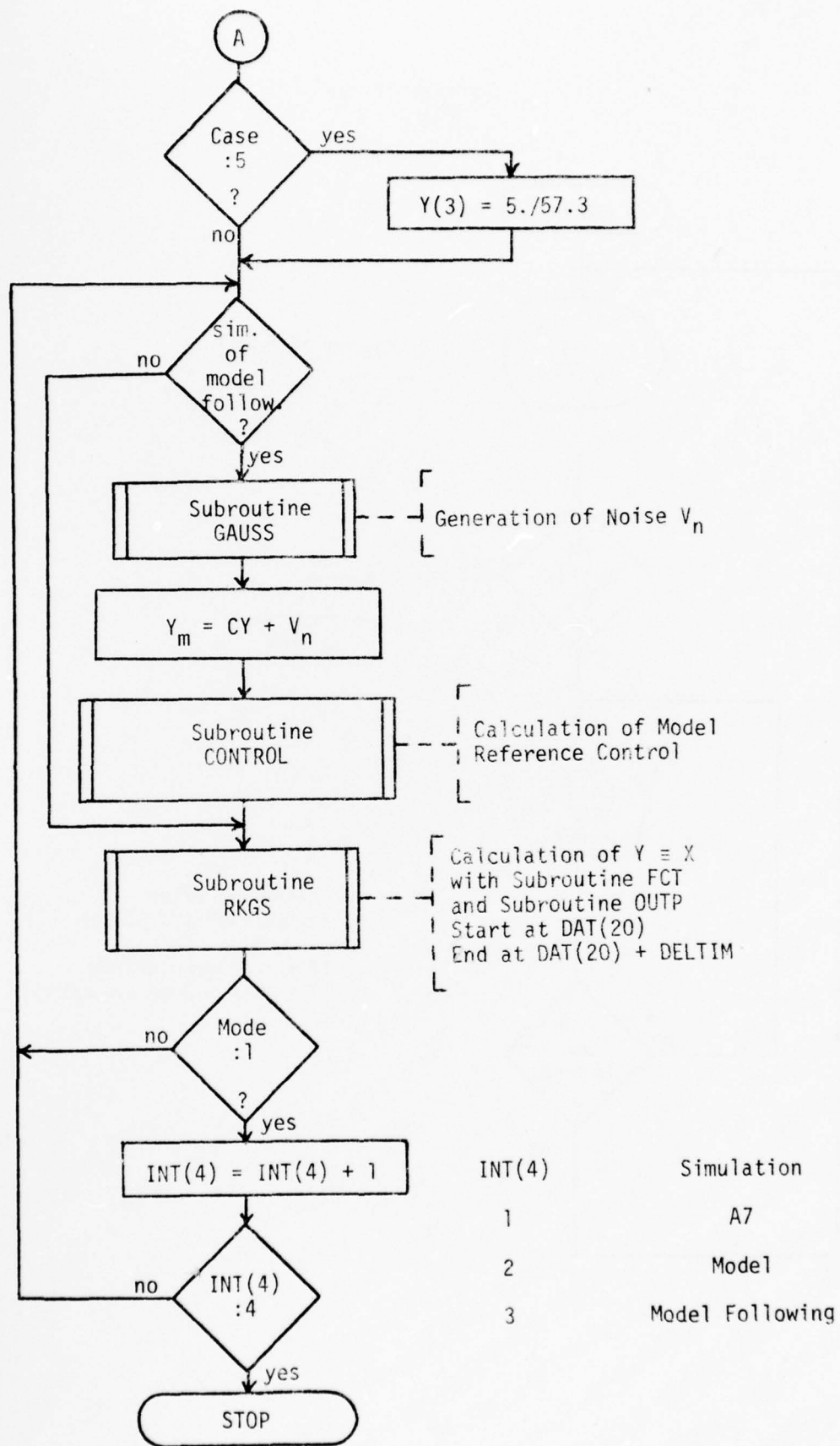
INT(8): - not used

INT(9): - not used

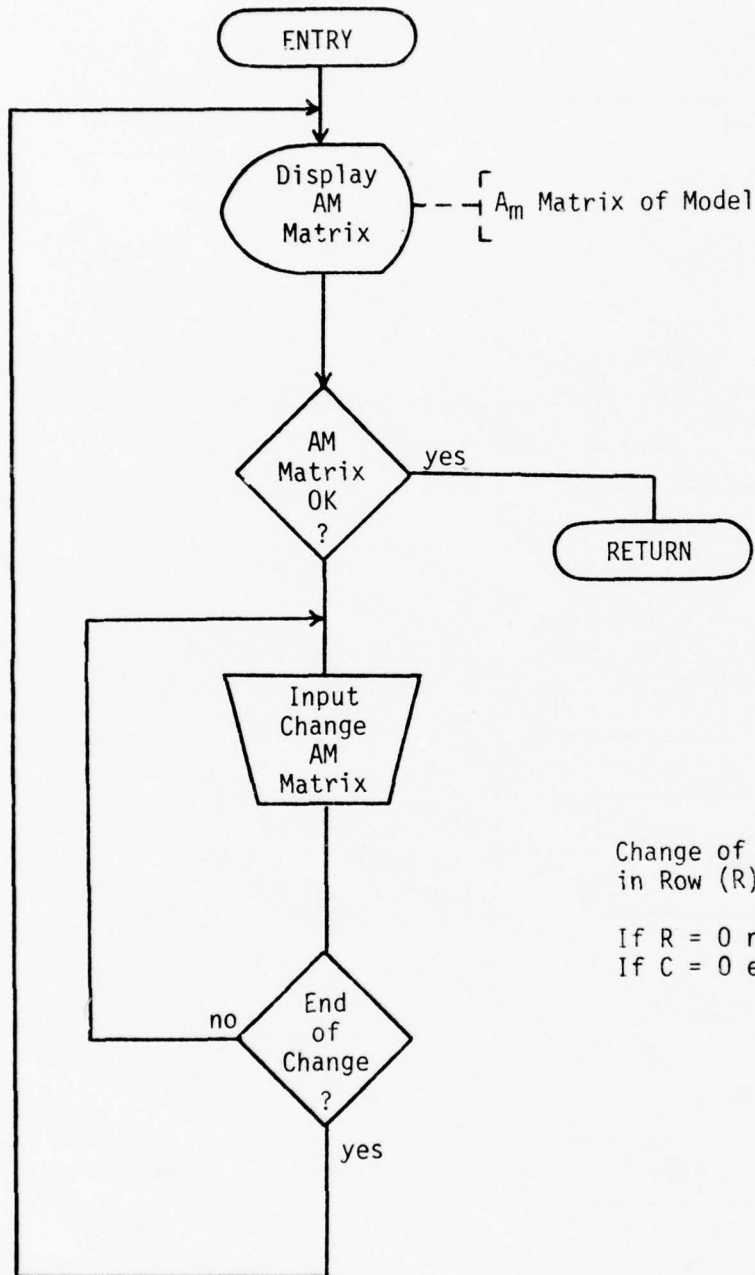
INT(10):	-	not used
R(1)	XM	runtime
R(2)	DELTIM	time interval for new control calculation
R(3)	W	weight of control
R(4)		integration time for Riccati equation
R(5)		time increment of integration (Riccati equation)

2. Flow Chart Program, Part 1





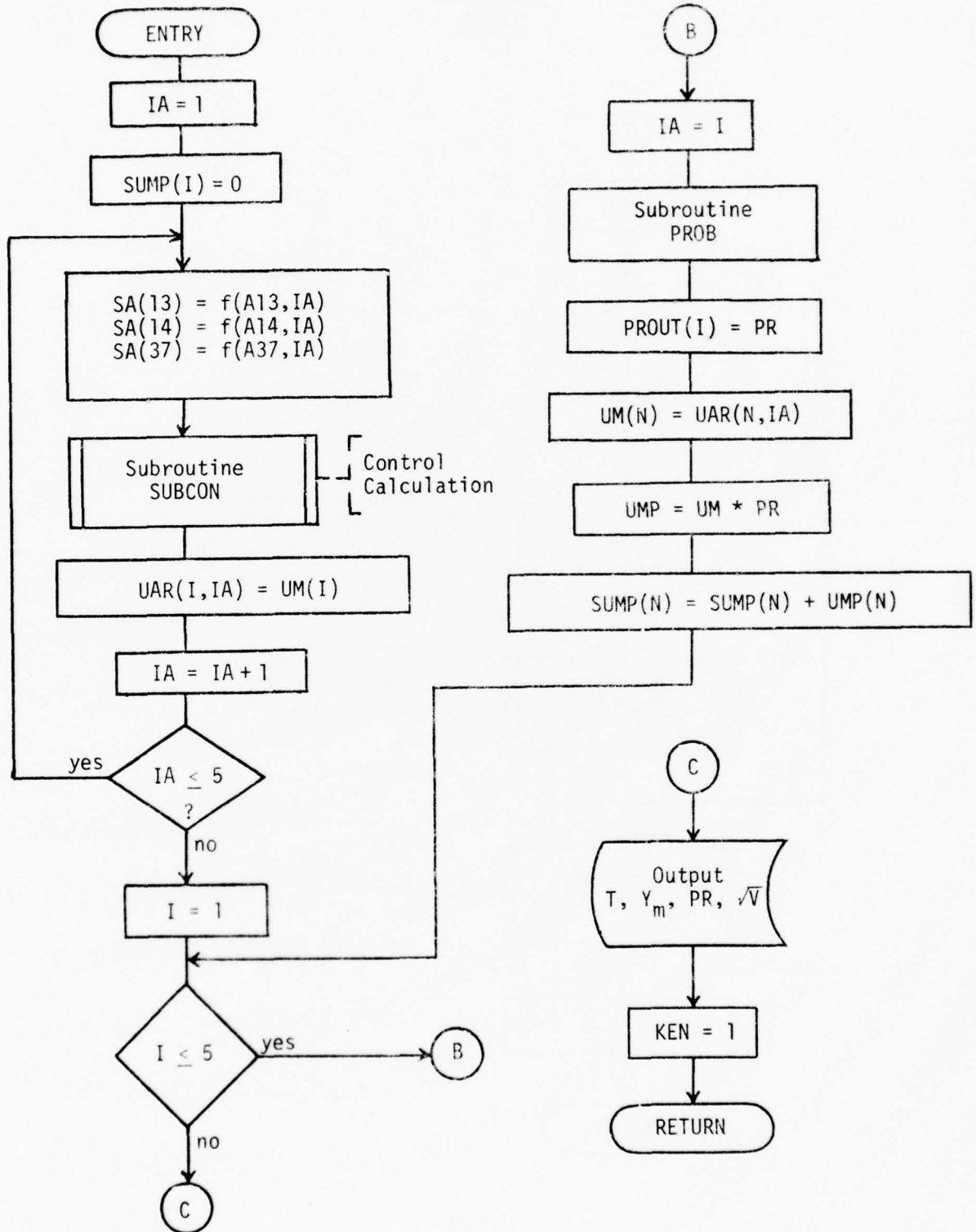
Subroutine
INPUT



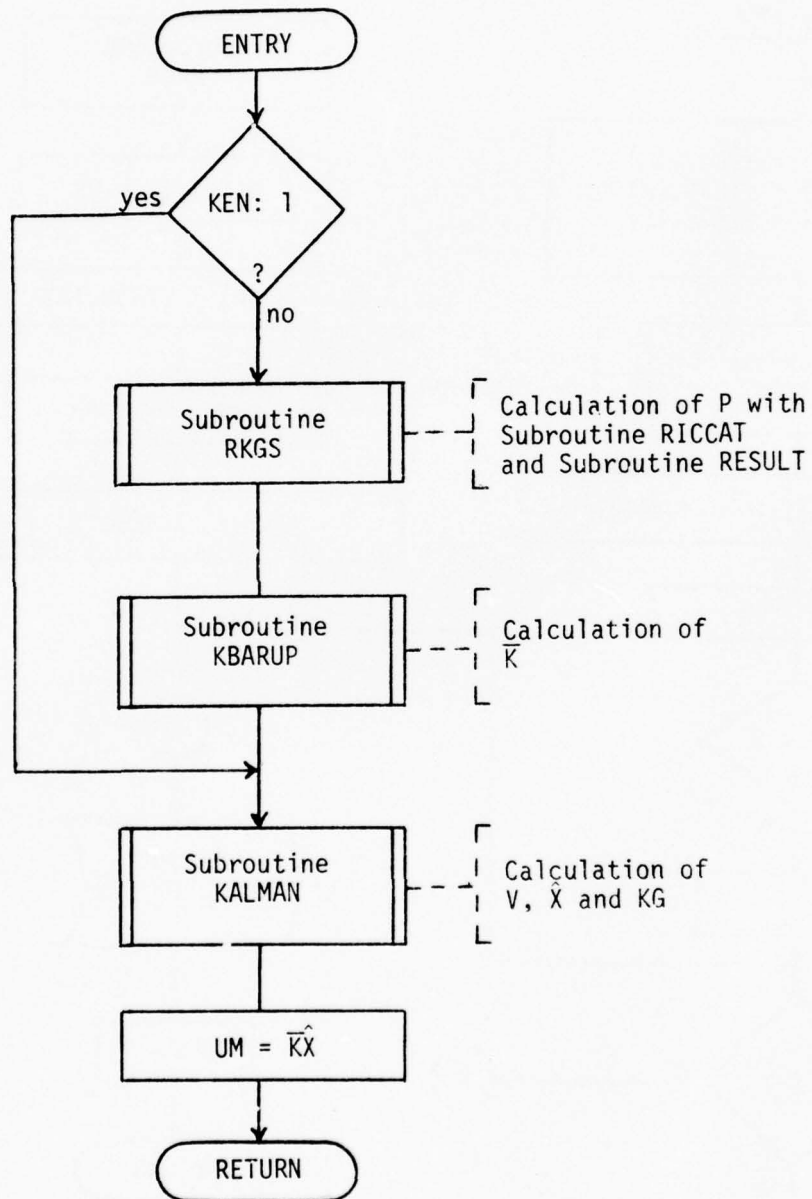
Change of value
in Row (R) and Column (C)

If R = 0 next column
If C = 0 end of changing

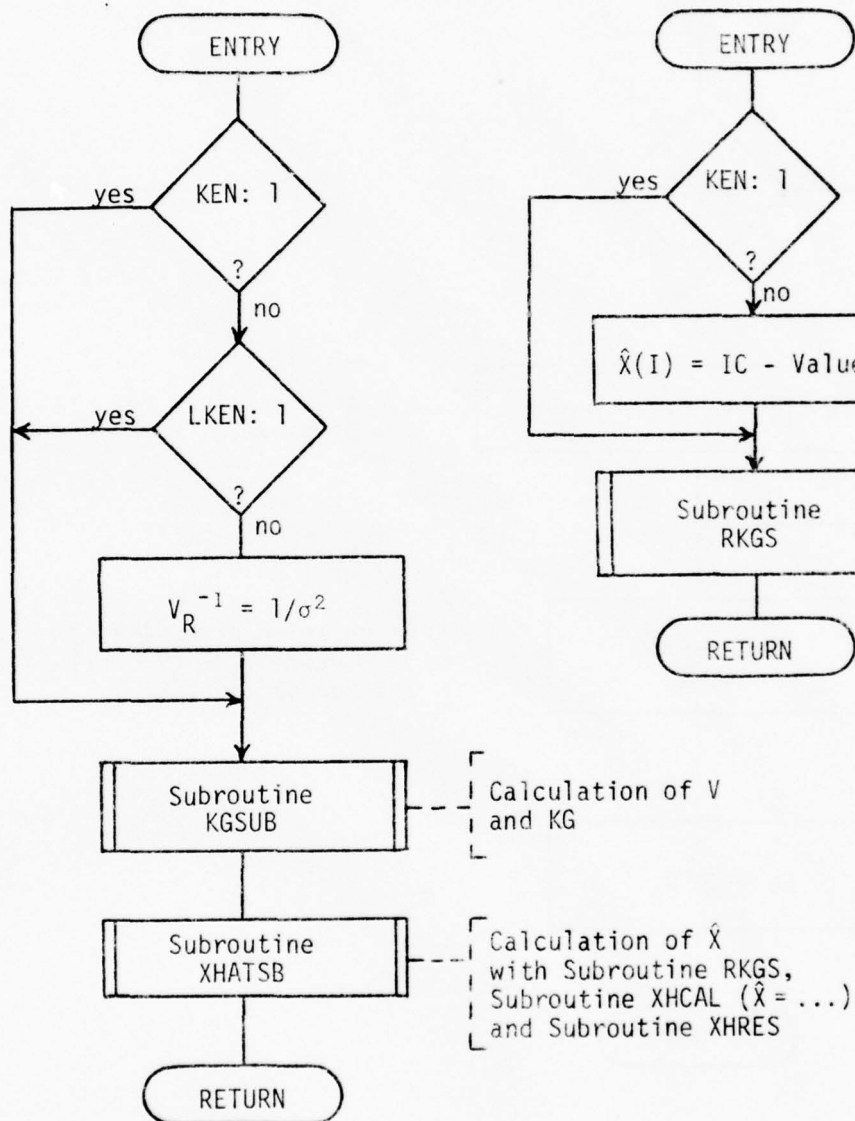
Subroutine
CONTRL



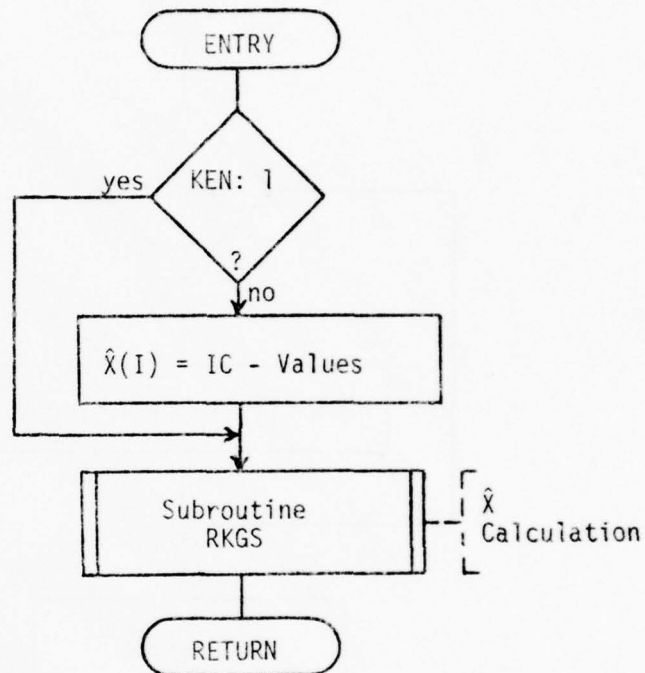
Subroutine
SUBCON



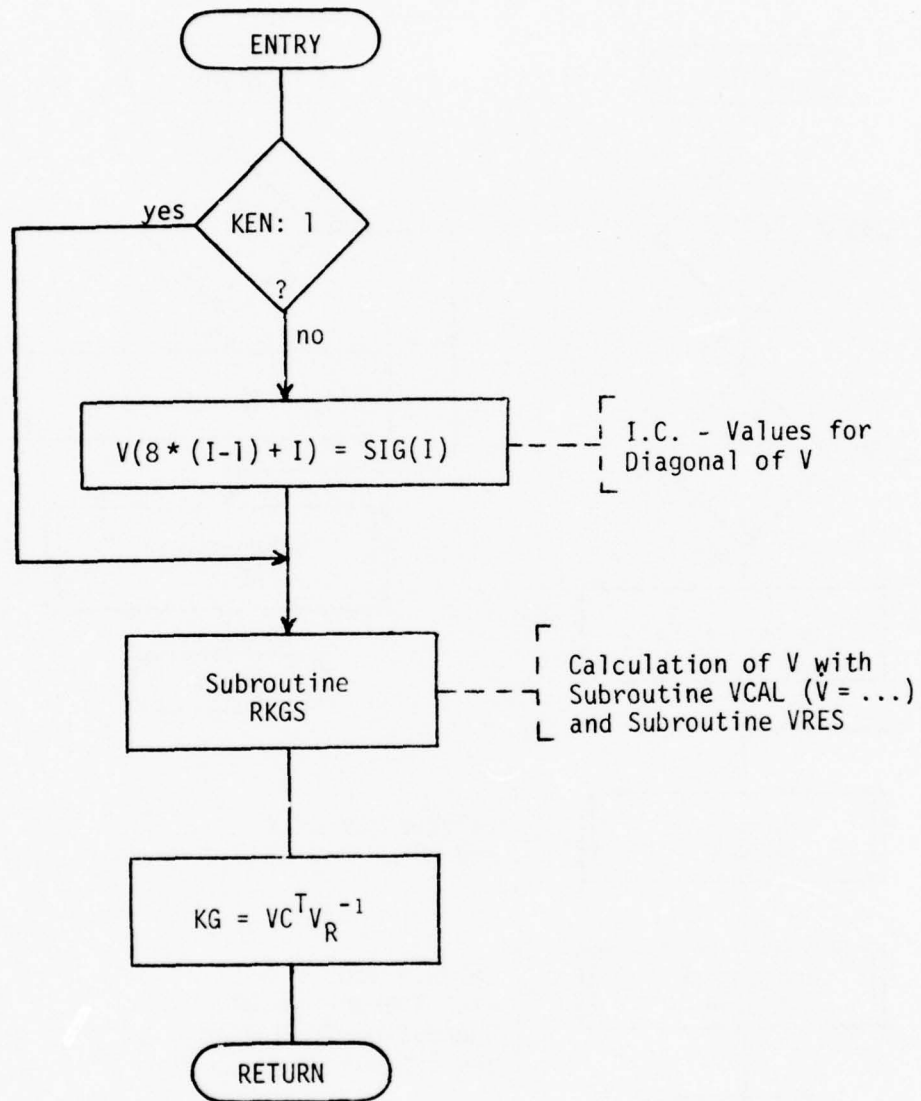
Subroutine
KALMAN



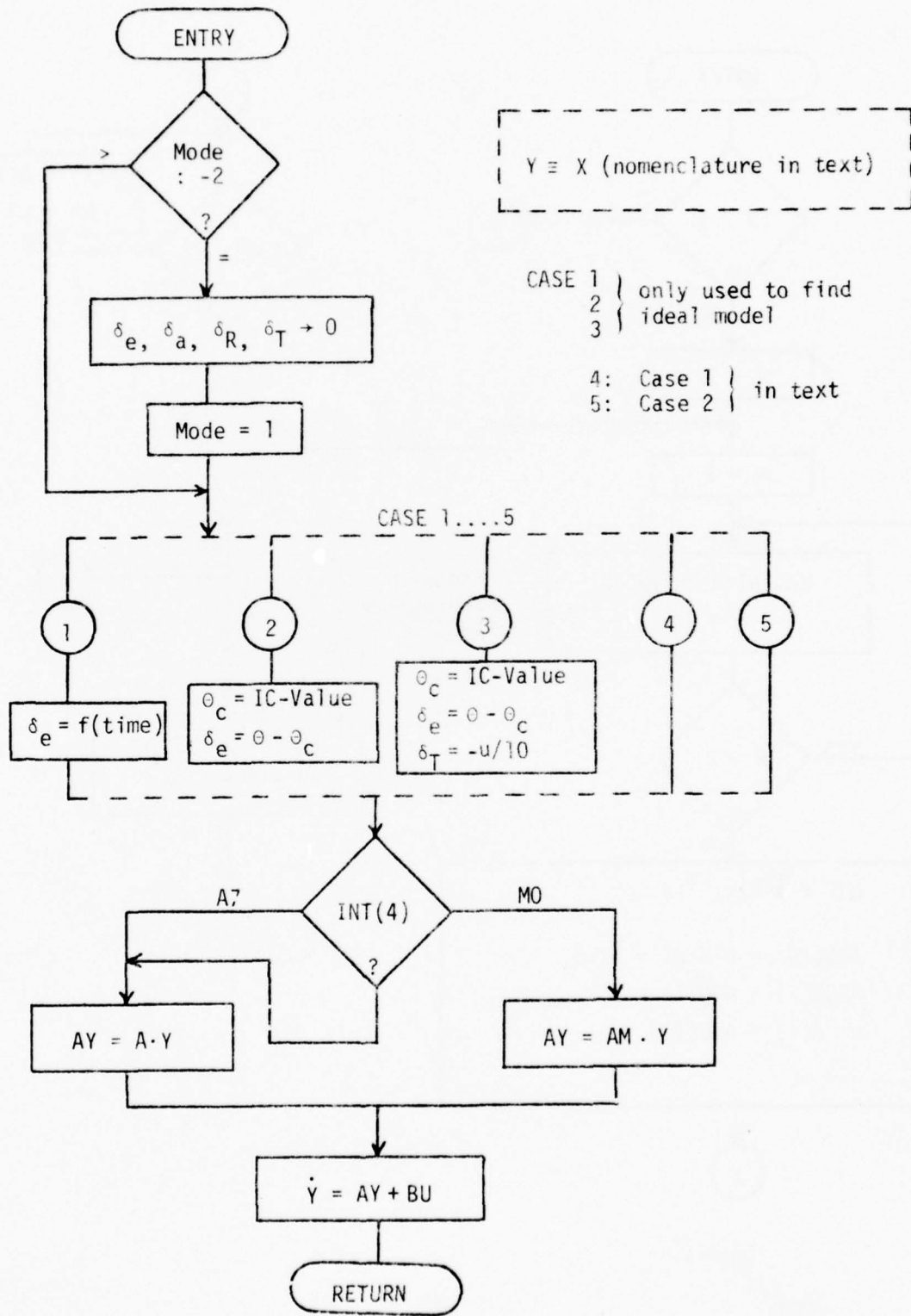
Subroutine
XHATSB



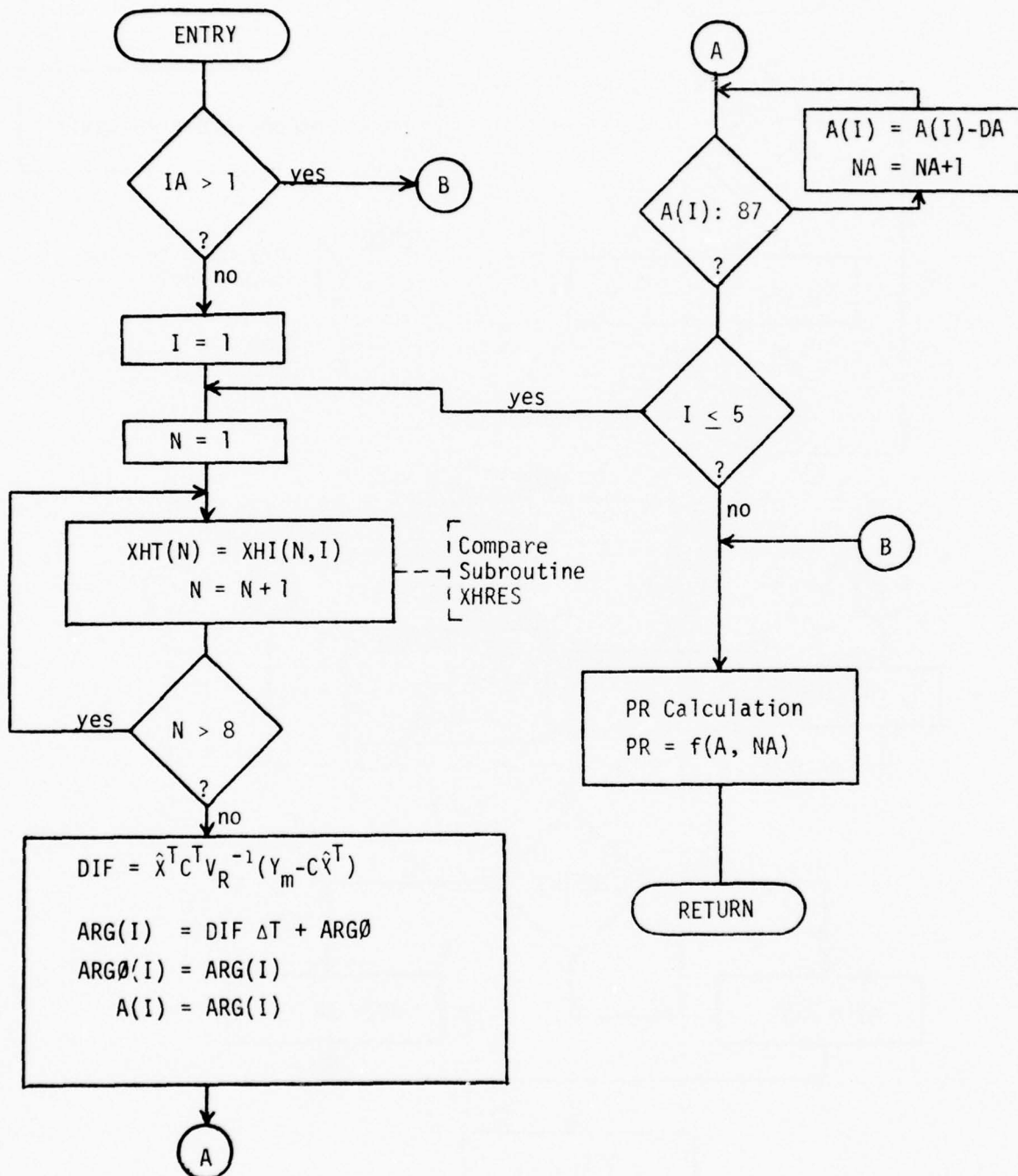
Subroutine
KGSUB



Subroutine
FCT



Subroutine
PROB



3. Program Comments and Source Listing

The variable names correspond closely to the notation in the text. The probability density function (compare Equation 16, page 8) was calculated (with $P(\mu_{2j}) = 1$) in the following way (short notation):

$$P_r(i) = \frac{P(i)e^{A(i)}}{\sum_{j=1}^5 P(j)e^{A(j)}}$$

with

$$\begin{aligned} A &= \int_{t_0}^t \hat{X}^T C^T V_R^{-1} Y_m dt - \int_{t_0}^t ||C\hat{X}||_{V_R^{-1}}^2 dt \\ &= \int_{t_0}^t \hat{X}^T C^T V_R^{-1} Y_m dt - \int_{t_0}^t \hat{X}^T C^T V_R^{-1} C \hat{X} dt \\ &= \int_{t_0}^t \hat{X}^T C^T V_R^{-1} (Y_m - C\hat{X}) dt \end{aligned}$$

To prevent overflow for longer simulation time

$$A(i) = A_n(i) + n(i) D_a$$

$$D_a = \text{const}, A_n < 87.0$$

was introduced. Therefore,

$$P_r(i) = \frac{1}{\text{sum}(i)}$$

with

$$\text{sum}(i) = \sum_{j=1}^5 \frac{P(j)}{P(i)} e^{A_i}$$

and

$$A_i = A_n(j) - A_n(i) + (n(j) - n(i)) D_a$$

The program writes the results on the disk.

File name:

FOR001.DAT for A7

FOR002.DAT for Ideal Model (M0)

FOR003.DAT for Model Following (MF)

The following pages show a typical dialog.

```

TERMINAL OUTPUT (Y/N):N
T1:.02
DELTA1:..002
T2:20.
DELTA2:..5

```

NO=1
NO=3
NO=3
NO=3
NO=1

AM-MATRIX OK? (Y/N): N

CHANGE OF VALUE IN ROW (R) AND COLUMN (C)

IF R=0 NEXT COLUMN,
IF C=0 END OF CHANGING

C: 2
R: 5
AM(5,2)= 0.
R: 6
AM(6,2)= 0.
R: 4
C: 5
R: 5
AM(5,4)= -1.
R: 6
AM(6,4)= 1.5
R: 7
C: 5
R: 5
AM(5,7)= -.5
R: 6
C: 7

C: -0.0634-22.6800 0.0000 0.0000 -5.7660 0.0000 0.0000 3.1870-32.0240
-0.0009 -0.3230 1.0000 0.0000 0.0000 -0.0995 0.0000 0.0000 0.0000
0.0000 -3.5770 -0.3860 -0.0000 -0.0000 -0.0982 0.0025 0.0000 0.0000
0.0000 0.0122 0.0000 -0.1862 0.0000 0.3216 -0.9469 0.1166 0.0129
0.0000 0.0000 0.0000 -1.0000 -0.8490 -0.3323 -0.5000 0.0000 0.0000
0.0000 0.0000 0.0000 1.5000 0.0193 -0.1276 0.0000 0.0000 0.0000
0.0000 0.0000 0.0000 0.0000 1.0000 0.3397 0.0000 -0.0116
0.0000 0.0000 1.0000 0.0000 0.0000 0.0000 0.0104 0.0000
AM-MATRIX OK? (Y/N): Y
RUN TIME: 5.
CASE: 4

FORTAN IV V018-02 THU 26-MAY-77 14:37:06 PAGE 001
 CORE=08K. UIC=[123.1] .LP/LI:1=A7MAN

C MODEL FOLLOWING PROGRAM

 C PART 1

C INPUT AND CALCULATION

C VERSION 5/26

C IN THIS PART OF THE PROGRAM MANY SUBROUTINES
 C OF THE SCIENTIFIC SUBROUTINE PACKAGE (SEE
 C RSX-11M MANUAL) WERE USED.

```

0001 COMMON /A7/DAT(20),INT(10),R(5)
0002 COMMON /A7R/SA(64),SB(32),SAM(64),UM(4)
0003 COMMON /KAL/XHATK(8),RKG(64),C(8),CT(8),VRIN(8)
0004 COMMON /NOISE/SIGMA(8),YM(8)
0005 COMMON /OUT/YHOUT(3),SVDOUT(8,5)
0006 DIMENSION PRMT(5),Y(8),DERV(8),AUX(8,8)
0007 DIMENSION CY(8),IG(8),VN(8)
0008 EXTERNAL FCT-OUTP
0009 EQUIVALENCE(MODE,INT(3)),(R(2),DELTIM)
0010 EQUIVALENCE(NF,INT(4)),(U,R(3))
0011 DATA IDEV2/'TT'/,JES/'Y'/,IDEV1/'DP'/
0012 DATA C/8K1./

```

C WEIGHTING OF CONTROL
 C W=1. MODEL REFERENCE CONTROL ONLY. NO PILOT INPUT
 C W=0. PILOT INPUT ONLY
 C W=1.

C STANDARD DEVIATION OF THE NORMAL DISTRIBUTION
 C OF NOISE

```

0014 SIGMA(1)=2.5
0015 SIGMA(2)=.01
0016 SIGMA(3)=.02
0017 SIGMA(4)=.01
0018 SIGMA(5)=.02
0019 SIGMA(6)=.02
0020 SIGMA(7)=.01
0021 SIGMA(8)=.01
0022 DO 111 I=1,8
0023 SIGMA(I)=SIGMA(I)/2.
0024 111 CONTINUE

```

C DIFFERENT INPUTS FOR SPECIAL RUN

```

0025 WRITE(5,10)
0026 10 FORMAT('STERMINAL OUTPUT (Y,N):')
0027 READ(5,11) IOUT
0028 11 FORMAT(A1)
0029 IF(IOUT.NE.JES) GO TO 500
0031 WRITE(5,250)

```

CORE=08K. UIC=C123.11 .LP/LJ:1=A7MAN

```

0032 250 FORMAT('NUMBER OF TERMINAL:')
0033 READ(5,201) NT
0034 CALL ASNLUN(6,IDEV2,NT)
0035 980 CONTINUE
0036 WRITE(5,13)
0037 13 FORMAT('ST1:')
C
C DELTA IS THE TIME BETWEEN TWO CONTROL CALCULATIONS
0038 READ(5,14) DELTA
0039 14 FORMAT(F10.6)
0040 WRITE(5,15)
0041 15 FORMAT('DELTA1:')
C
C PRINT(3) IS THE INITIAL TIME INCREMENT IN THE
C RUNGE-KUTTA SUBROUTINE TO SOLVE THE FLIGHT
C DYNAMICS EQUATIONS
0042 READ(5,14) PRINT(3)
0043 WRITE(5,16)
0044 16 FORMAT('ST2:')
C
C R(4) IS THE INTEGRATION TIME TO SOLVE
C THE RICCATI EQUATIONS
0045 READ(5,14) R(4)
0046 WRITE(5,17)
0047 17 FORMAT('DELTA2:')
C
C R(5) IS THE INITIAL TIME INCREMENT IN THE RUNGE-
C KUTTA SUBROUTINE TO SOLVE THE RICCATI EQUATIONS
0048 READ(5,14) R(5)
0049 WRITE(5,12)
0050 12 FORMAT('OSIMULATION OF')
1 17. MO. MF : NO=1//
2 17. MO. MF : NO=2//
3 17. MF : NO=3//
4 17. NO=//
0051 READ(5,201) INT(4)
0052 NUMF=INT(4)
0053 DO 600 I=NUMF,4
0054 CALL FDBSET(I,'UNKNOWN')
0055 600 CONTINUE
0056 201 FORMAT(11)
0057 401 IF(INT(6).EQ.0) CALL INPUT
0059 970 CONTINUE
0060 WRITE(5,210)
0061 210 FORMAT('ERUN TIME: ')
0062 READ(5,211) R(1)
0063 211 FORMAT(F4.0)
0064 WRITE(5,220)
0065 220 FORMAT('SCASE: ')
0066 READ(5,201) INT(2)

```

FORTRAN IV V018-02 THU 26-MAY-77 14:37:06 PAGE 003
 CORE=08K. UIC=C123.1J .LP/LI:1=A7MAN

```

0067 CALL CLOSE(5)
0068 CALL ASNLUN(5,IDEV1,0)
0069 CALL ASSIGN(5,'ERROR.DAT')
0070 PRMT(4)=.001
0071 NDIR=8
0072 CONTINUE
0073 DO 2 I=1.20
0074   DAT(I)=0.
0075   DO 5 I=1.8
0076     Y(I)=0.
0077     DO 6 I=1.4
0078       UM(I)=0.
0079   C
0080   IC-VALUES: YAW RATE IF CASE 4
0081   PITCH RATE IF CASE 5
0082   IF(INT(2).EQ.4) Y(6)=-10./57.295
0083   IF(INT(2).EQ.5) Y(3)=9./57.295
0084   MODE=-2
0085   C
0086   BEGIN OF MAIN LOOP OF PROGRAM
0087   -----
0088   C
0089   CONTINUE
0090   PRMT(1)=DAT(20)
0091   PRMT(2)=PRMT(1)+DELTIM
0092   C
0093   DERY: INPUT VECTOR OF ERROR WEIGHTS (DESTROYED)
0094   LATEROX DERY IS THE VECTOR OF DERIVATIVES
0095   DO 100 I=1.8
0096     DEPY(I)=.125
0097   CONTINUE
0098   IF(INT(4).NE.3) GOTO 150
0099   C
0100   SIMULATION OF MEASUREMENTS USING Y
0101   CALL MRD(C,Y,CY:8.8,2.0,1)
0102   DO 40 I=1.8
0103     CALL GAUSS(15(I),SIGNA(I),AM,VN(I))
0104     YN(I)=CY(I)+VN(I)
0105   CONTINUE
0106   C
0107   OUTPUT OF MEASUREMENTS
0108   YHOUT(1)=YN(1)
0109   DO 45 I=2.8
0110     YHOUT(I)=57.295*YN(I)
0111   CONTINUE
0112   C
0113   MODEL REFERENCE CONTROL CALCULATION
0114   CALL CTRL
0115   C
0116   FLIGHT SIMULATION

```

PAGE 004
LP/L1:1=ATTN

THU 26-MAY-77 14:37:06

FORTRAN IV V01B-02
CORE=08K, UIC=1123.13

```

0102 150 CALL PKGS(PRMT,Y,DERV,NDIM,IHLF,FCT,OUTP,AUX)
0103 IF(IHLF.GT.10) GOTO 900
0105 IF(MODE) 50,50,999
C
C END OF MAIN LOOP
C -----
0106 900 CONTINUE
0107 WRITE(6,901) IHLF,INT(4)
0108 901 FORMAT('**** ERROR *** IHLF: ',I2,' AT INT(4): ',I1)
0109 999 CONTINUE
0110 CALL END
0111 ENDFILE NF
0112 CALL CLOSE(NF)
0113 INT(4)=INT(4)+1
0114 IF(INT(4).LE.3) GOTO 60
0115 WRITE(6,950) INT(7)
0117 950 FORMAT('MAX IHLF IN XHAT-CALCULATION:',I2)
0118 ENDFILE 4
0119 STOP
0120 500 CONTINUE
C
C LOG UNIT ASSIGN. IF NO TERMINAL OUTPUT
0121 CALL ASHLUN(6,IDEV1,0)
0122 CALL ASSIGN(6,'DP,DAT')
0123 GOTO 900
0124 END

```

FORTRAN IV V01B-02 THU 26-MAY-77 14:37:10 PAGE 001
 CORE=08K, UIC=C123.1J ,LP/LI:1=A7MAN

```

0001      SUBROUTINE INPUT
0002      C
0003      C
0004      COMMON /A7/DAT(20),INT(10),R(5)
0005      COMMON /MATR/SA(64),SB(32),SPM(64),UM(4)
0006      COMMON /K/RBAR(16),S(32),SBT(32),RKBAR(32),P(64)
0007      COMMON /CON/TA,A13,A14,A37
0008      DIMENSION DA(8,8),DB(8,4),DAM(8,8)
0009      DATA JES,'Y','N','N'
0010
0011      DA IS THE A-MATRIX OF A7 IN DOUBLE DIMENSIONED STORAGE
0012      DATA DA/-.0634,-.00087,-.6*0.,
0013      1 -22.68,-.323,-3.577,.0122,3.09,-1.486,2*0.,
0014      2 0.,1.,-.386,4*0.,1.,
0015      3 -5.766,0.,-.0000009,-.1062,-4.45,-.1885,2*0.,
0016      4 0.,-.0995,-.00818,.3216,-.849,.0193,1.,0.,
0017      5 0.,-.0338,.0025,-.9459,.3323,-.1276,.3397,0.,
0018      6 3.187,2*0.,.1166,3*0.,.0104,
0019      7 -32.024,2*0.,.0129,2*0.,-.0116,0./
0020      DATA DB/-.1025,-.057,-2.92,-.0037,-.292,.1095,2*0.,
0021      1 4*0.,.431,.031,2*0.,
0022      2 .698,2*0.,.0255,1.4,-.998,2*0.,
0023      3 10.,7*0./
0024      MOD=2
0025
0026      SA IS THE A-MATRIX OF A7 IN SINGLE DIMENSIONED STORAGE
0027      CALL ARRAY(MOD,8,8,8,SA,DA)
0028      A13=SA(13)
0029      A14=SA(14)
0030      A37=SA(37)
0031
0032      DAM IS THE A-MATRIX OF THE IDEAL MODEL
0033      C
0034      C
0035      C
0036      CALL COPY(DA,DAM,8,8,0)
0037      CONTINUE
0038      N=8
0039      M=8
0040      WRITE(5,101) ((DAM(IR,IC),IC=1,M),IR=1,N)
0041      101 FORMAT(' ',8F8.4)
0042      WRITE(5,110)
0043      110 FORMAT('SAM-MATRIX OK? (Y/N): ')
0044      111 READ(5,111) KN
0045      111 FORMAT(A1)
0046      IF (KN.EQ.JES.AND.KN.NE.NO) GOTD 100
0047      IF (KN.EQ.JES) GOTD 200
0048      WRITE(5,201)
0049      201 FORMAT('CHANGE OF VALUE IN ROW (R) AND COLUMN (C):')
0050      1 '01F R=0 NEXT COLUMN.'/
0051      2 ' IF C=0 END OF CHANGING'
0052      250 WRITE(5,251)

```

FORTRAN IV V018-02 THU 26-MAY-77 14:37:10 PAGE 002
 CORE=08K. UID=L123.11 .LP/L1:1=ATTN

```

0032 251 FORMAT('C: ')
0033      READ(5,252) IC
0034 252 FORMAT(I1)
0035      IF(IC.EQ.0) GOTO 100
0037      IF(IC.GT.8.OR.IC.LT.0) GOTO 250
0039      CONTINUE
0040      WRITE(5,261)
0041 261 FORMAT('R: ')
0042      READ(5,252) IR
0043      IF(IR.EQ.0) GOTO 250
0045      IF(IR.GT.8.OR.IR.LT.0) GOTO 260
0047      WRITE(5,271) IR,IC
0048 271 FORMAT('SAM',I1,'.',I1,'.')
0049      READ(5,272) DAM(IR,IC)
0050 272 FORMAT(F10.0)
0051      GOTO 260
0052 280 CONTINUE
C
C      SAM IS THE A-MATRIX OF THE IDEAL MODEL
C      IN SINGLE DIMENSIONED STORAGE
C      CALL ARRAY(MOD.N,M,N,M,SAM,DAM)
0053      CONTINUE
0054 300 N=8
0055      M=4
0056      CALL ARRAY(MOD.N,M,N,M,SAM,DB)
0057 600 RETURN
0058      END
0059

```


FORTRAN IV V01B-02 THU 26-MAY-77 14:37:13 PAGE 001
 CORE=00K. UIC=1123.11 ,LP/LI:1=07MM

```

0001 SUBROUTINE FCT(X,Y,DERY)
-----
0002 COMMON /AT/DAT(20),INT(10),R(5)
0003 COMMON /NTR/SA(64),SB(32),SAM(64)
0004 COMMON /CON/IA,A13,A14,A37,SUMP(4)
0005 DIMENSION Y(8),DERY(8),AY(8),BU(8),U(4)
0006 DIMENSION UPILOT(4)
0007 EQUIVALENCE (UPILOT(1),DE), (UPILOT(2),DA),
1      (UPILOT(3),DR), (UPILOT(4),DT)
0008 EQUIVALENCE(MODE,INT(3)),(W,R(3))
0009 IF(MODE.GT.-2) GO TO 10

C THE NEXT TWO STATEMENTS ARE ONLY IMPORTANT,
C IF FUNCTIONS FOR DERIVATIVES ARE USED
C
0011 ALD=19.
0012 BETAO=6.
0013 DO 11 I=1.4
0014 SUMP(I)=0.
11  U(I)=0.
0015 DE=0.
0016 DA=0.
0017 DR=0.
0018 DT=0.
0019 MODE=-1
0020
10  CONTINUE
C
C INT(2) : 'CASE'
C CASE 1,2,3 WERE USED TO FIND AND TEST IDEAL MODEL
C CASE 4 IS FLIGHT WITH IC-VALUE OF YAU RATE
C CASE 5 IS FLIGHT WITH IC-VALUE OF PITCH RATE
C GO TO (1,2,3,4,5),INT(2)
0022 1 CONTINUE
0023 IF(X.GT.1.) DE=-.05*(X-1.)
0024 GO TO 100
0025 2 CONTINUE
0026 THC=.01
0027 DE=Y(8)-THC
0028 GO TO 100
0029 3 CONTINUE
0030 THC=.01
0031 DE=Y(8)-THC
0032 DT=-Y(1)/10.
0033 4 CONTINUE
0034 5 CONTINUE
0035 100 CONTINUE
0036 DO 150 I=1.4
0037 150
C
C W: WEIGHTING (SEE MAIN PROGRAM)
0039 U(I)=WA*SUMP(I)+(1.-W)*UPILOT(I)

```


FORTRAN IV V018-02 THU 26-MAY-77 14:37:16 PAGE 001
 CORE=08K, UIC=[123.1] ,LP/LI:1=A7MAN

```

0001 SUBROUTINE OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      C-----
      C
0002 COMMON /A7/DAT(20),INT(10),R(5)
0003 COMMON /KAL/XHATK(8),PKG(64),C(8),CT(8),VRIN(8)
0004 COMMON /NOISE/SIGMA(8),YM(8)
0005 COMMON /OUT/YMOUT(8),SVOUT(8.5)
0006 DIMENSION Y(8),DERY(8),PRMT(5),CY(8),IG(8),VN(8)
0007 EQUIVALENCE (MODE,INT(3)),(NF,INT(4))
0008 IF (IHLF.EQ.IHLF1) GOTO 300

      C TO COMPILE STATEMENTS WITH A D IN THE FIRST
      C COLUMN THE SWITCH /DE MUST BE USED IN THE
      C COMMAND LINE FOR THE FORTRAN COMPILER
      C OTHERWISE THE STATEMENTS ARE TREATED AS
      C COMMENT LINES
      C
0010 D 451 FORMAT('0IHLF:',12,' AT FLIGHT NO:',11)
0011 IHLF1=IHLF
      300 CONTINUE

      C
      C DATA TRANSFER
      C
      C NF MEANS FILE NUMBER
      C
      C NF=1 : VALUES FOR A7 FILE NAME: FOR001.DAT
      C 2 : VALUES FOR M0 FOR002.DAT
      C 3 : VALUES FOR NF FOR003.DAT
      C
0012 IF (MODE.GT.-1) GOTO 310
0014 REMIND NF
0015 N=0
0016 DELT=0.
0017 XM=R(1)
0018 WRITE(NF) XM
0019 DELTA=XM/500.
0020 MODE=0
      310 CONTINUE
0021
0022 IX=X
0023 IF (IX.EQ.IX1) GOTO 450
      WRITE(6,400) IX,INT(4)
0025 D 400 FORMAT('0FLIGHT TIME:',12,' SEC AT FLIGHT NO:',11)
      IX1=IX
0026 450 CONTINUE
0027 IF (X.LT.DELT) RETURN
      C
      C NORMAL DATA TRANSFER
      C
0029 DAT(20)=X
0030 DAT(1)=Y(1)
0031 DO 302 I=2,8

```

PAGE 002
LP/LI:1=ATMAN

THU 26-MAY-77 14:37:16

FORTRAN IV V018-02
CORE=08K. UFC=[123.13]

```
0032 302 DAT(1)=57.295*Y(1)
0033 WRITE(NF) DAT
0034 N=N+1
0035 DELT=DELTA*N
0036 IF(X.LT.R(1)) RETURN
0038 MODE=1
0039 RETURN
0040 END
```

PAGE 001
LP/LI:1=A7MAN

THU 26-MAY-77 14:37:17

FORTRAN IV V018-02
CORE=08K. UIC=[123.1]

```

0001 SUBROUTINE END
      C
      C
      C
      C
      THIS SUBROUTINE WAS USED TO SAVE THE
      TIME OF END OF CALCULATION
      COMMON /A7/DAT(20),INT(10)
      EQUIVALENCE (NF,INT(4))
      DIMENSION ITIME(4)
      CALL TIME(ITIME)
      WRITE(NF) ITIME
      RETURN
      END
0002
0003
0004
0005
0006
0007
0008

```

FORTRAN IV V018-02 THU 26-MAY-77 14:37:18 PAGE 001
CORE=08K. UIC=[123,1] ,LP/LI:1=A716N

SUBROUTINE CONTRL

```

0001 C
0002 C
0003 COMMON /A7/DAT(20),INT(10),R(5)
0004 COMMON /MATR/SA(64),SB(32),SAM(64),UM(4)
0005 COMMON /KAL/XHATK(8),PKG(64),C(8),CT(8),VRIN(8)
0006 COMMON /K/RBAR(16),S(32),SBT(32),RKBAR(32)
0007 COMMON /CON/IA,A13,A14,A37,SUMP(4),XHI(8.5)
0008 COMMON /OUT/YMOUT(8),SVDOUT(8.5)
0009 DIMENSION UMP(4),UAR(4.5),PROUT(5)
0010 EQUIVALENCE(KEN,INT(6))
0011 IA=1
0012 DO 10 I=1,4
0013 SUMP(I)=0.
0014 10 CONTINUE
0015 C
0016 C
0017 C INTRODUCING POSSIBLE PARAMETER VECTORS FOR THE
0018 C DERIVATIVES EXPECTED TO BE UNCERTAIN.
0019 C (COMPARE TEXT PAGE 12)
0020 SA(13)=A13*(1.-.45*(IA-3))
0021 SA(14)=A14*(1.-.45*(IA-3))
0022 SA(37)=A37*(1.-.45*(IA-3))
0023 CALL SUBCON
0024 DO 50 I=1,4
0025 UAR(I,IA)=UM(I)
0026 50 CONTINUE
0027 IA=IA+1
0028 IF(IA.LE.5) GOTO 1
0029 DO 200 I=1,5
0030 200 I=1
0031 C
0032 C CALCULATION OF PROBABILITY
0033 CALL PROB(PR)
0034 C
0035 C PREPARING THE OUTPUT
0036 PROUT(1)=PR
0037 DO 300 N=1,4
0038 UM(N)=UAR(N,IA)
0039 300 CONTINUE
0040 CALL SMPY(UM,PR,UMP,4,1.0)
0041 DO 200 N=1,4
0042 200 N=1
0043 C
0044 C THE CONTROL TERM SUMP(N) IS THE SOLUTION OF
0045 EQUATION 10 PAGE 6 AND IS USED IN SUBROUTINE FCT
0046 SUMP(N)=SUMP(N)+UMP(N)
0047 200 CONTINUE
0048 C
0049 C PROVIDING THE OUTPUT OF TIME, MEASUREMENTS, PROBABILITIES
0050 C AND STANDARD DEVIATIONS (FILE: FOR004.DAT)

```


PAGE 002
.LP/LI:1=A7MAN

THU 26-MAY-77 14:37:18

FORTRAN IV V01B-02
CORE=08K. UID=C123.11

0036 T=DAT(20)
0037 WRITE(4) T.YMOUT,PROUT.SVDOUT
0038 KEN=1
0039 RETURN
0040 END

FORTRAN IV V01B-02 THU 26-MAY-77 14:37:20 PAGE 001
 CORE=03K, UIC=F123.17 ,LP/LI:1=A711AH

```

0001 SUBROUTINE PROB(PR)
-----
0002 COMMON /A7/ DAT(20),INT(10),R(5)
0003 COMMON /CON/IA,A13,A14,A37,SUMP(4),XHI(8,5)
0004 COMMON /KAL/XHATK(8),RKG(64),C(8),CT(8),VRIN(8)
0005 COMMON /NOISE/SIGNA(8),YIK(8)
0006 DIMENSION XHT(8),XHTT(8),TEMP1(8),TEMP2(8),TEMP3(8)
0007 DIMENSION P(5),ARG(5),ARGO(5)
0008 DIMENSION A(5),NA(5)
0009 EQUIVALENCE(KEN,INT(6))

C
C P(I) ARE THE A PRIORI PROBABILITIES
0010 DATA P/5*.2/,HALF/.5/,ARGO/5*.0/,DA/5./
0011 IF(IA.GT.1) GOTO 200
0013 SUM=0.
0014 DO 960 I=1,5
0015 NA(I)=0
0016 CONTINUE
0017 DO 100 I=1,5
0018 DO 25 N=1,8
0019 XHT(N)=XHI(N,I)
0020 CONTINUE
25
C
C CALCULATION OF PROBABILITIES (PR) AS
C MENTIONED IN THE TEXT OF THIS APPENDIX
C
0021 CALL GHTRA(XHT,XHTT,8,1)
0022 CALL MPD(XHTT,CT,TEMP1,1,8,0,2,8)
0023 CALL MPD(TEMP1,VRIN,TEMP2,1,8,0,2,8)
0024 CALL MPD(C,XHT,TEMP1,8,8,2,0,1)
0025 CALL SMPY(TEMP1,HALF,TEMP1,8,1,0)
0026 CALL GHSUB(YN,TEMP1,TEMP3,8,1)
0027 CALL GMPD(TEMP2,TEMP3,DIF,1,8,1)
0028 ARG(I)=DIF*(2)+ARGO(I)
0029 ARGO(I)=ARG(I)
0030 A(I)=ARG(I)
900 CONTINUE
0031 IF(A(I).GE.87.) GOTO 901
100 CONTINUE
0034 CONTINUE
200 CONTINUE
0036 SUM=0.
0037 DO 300 J=1,5
0038 AI=A(J)-A(IA)+(NA(J)-NA(IA))*DA
0039 IF(AI.GT.95) GOTO 902
0041 SUM=SUM+P(J)/P(IA)*EXP(AI)
0042 CONTINUE
300
C
C PR IS THE SOLUTION OF EQUATION 16 PAGE 8
C IMPORTANT: PR=PR(IA) COMPARE SUBROUTINE PROB
C PR=1./SUM
0043

```

PAGE 002
.LP/LI:1=A7HAN

THU 26-MAY-77 14:37:20

FORTRAN IV V01B-02
CORE=08K. UIC=C123.11

```
0044 400 CONTINUE
0045 RETURN
0046 901 CONTINUE
0047 A(I)=A(I)-DA
0048 NA(I)=NA(I)+1
0049 GOTO 900
0050 902 CONTINUE
0051 PR=0.
0052 GOTO 400
0053 END
```

FORTRAN IV V01B-02 THU 26-MAY-77 14:37:22 PAGE 001
 CORE=08K, UTC=[123,11] ,LP/1:1=A7MAN

```

0001      SUBROUTINE SUBCON
0002      C
0003      C
0004      COMMON /A7/DAT(20),INT(10),R(5)
0005      COMMON/NSTR/SA(64),SB(32),SAR(64),UM(4)
0006      COMMON/CONTR/MODEC
0007      COMMON/K/RBAR(15),S(32),SBT(32),RKBAR(32),P
0008      DIMENSION PAR(5),P(64),PDOT(64),STOR(8,64)
0009      DIMENSION XHAT(8)
0010      EQUIVALENCE (KEH,INT(6))
0011      EXTERNAL RICCAT,RESULT
0012      IF (KEH.EQ.1) GOTO 400
0013
0014      C
0015      C
0016      C
0017      C
0018      C
0019      C
0020      C
0021      C
0022      C
0023      C
0024      C
0025      C
0026      C
0027      C
0028      C
0029      C
0030      C
0031      C
0032      C
0033      C
0034      C

```

THE MAIN GOAL OF THIS SUBROUTINE IS TO SOLVE
 THE RICCATI EQUATIONS TO GET P FROM EQUATION 12
 PAGE 7 (USING SUBROUTINE RKGS AND SUBROUTINE RICCAT)

```

MODEC=-1
NO=64
PAR(1)=R(4)
PAR(2)=0.
PAR(3)=-R(5)
PAR(4)=-.0001
DO 300 I=1,64
  P(I)=0.
  PDOT(I)=1./64.
300 CONTINUE
CALL RKGS(PAR,P.PDOT,NO,1HALF,RICCAT,RESULT,STOR)
IF (1HALF.GT.10) GOTO 900
GOTO 999
900 WRITE(5,901) 1HALF
901 FORMAT('0000 ERROR *** 1HALF: ',12)
999 CONTINUE
CALL KBARUP
400 CONTINUE
CALL KALMAN(XHAT)
CALL GMFD(RKBAR,XHAT,UM,4,8,1)
RETURN
END

```

THU 26-MAY-77 14:37:23

SUBROUTINE PICCAT(TR,P,PDOT)

```

      COMMON INTR SA(64),SB(32),SAM(64)
      COMMON CONTR/NODEC
      COMMON /K/REAR,S,SBT,PKBAR(32)
      DIMENSION OP(8),RP(4),CO(8,8),SC0(64)
      DIMENSION SBT(32),SC0T(64),SR1(32),SR2(32)
      DIMENSION SR3(32),SR4(32),RPAR(16)
      DIMENSION SC0A(64),SAMC0(64),CANCAM(64),S(32)
      DIMENSION CANT(64),O1(64),OBAR(64)
      DIMENSION LSTOPB(4),NSTOREB(4)
      A P(64),PDOT(64)
      B BR(32),BRS(64),AMRS(64),ABRS(64)
      C PDOT1(64),PDOT2(64),PB(32)
      D PBR(32),PBRST(64),PDOT3(64)
      E ST(32),STR(32),PDOT4(64),PDOT5(64)

```

TASK OF THIS SUBROUTINE: SEE CALLING

SUBROUTINE SUBCON

DATA CO/1.,7*0.,

2 1*0.,1.,6*0.,

3 2*0.,1.,5*0.,

4 3*0.,1.,4*0.,

5 4*0.,1.,3*0.,

6 5*0.,1.,2*0.,

7 6*0.,1.,1*0.,

8 7*0.,1.,

DATA C/-1.,

IF(MODEC) 1,2,3

CONTINUE

OP(1)=1.

OP(2)=1.

OP(3)=1.

OP(4)=1.

OP(5)=1.

OP(6)=1.

OP(7)=1.

OP(8)=1.

RP(1)=1.

RP(2)=1.

RP(3)=1.

RP(4)=0.

CALL ARRAY(2,8,8,8,8,SC0,CO)

CALL GMTRA(SB,SBT,8,4)

CALL GMTRA(SC0,SC0T,8,8)

CALL GMTRA(SBT,SC0T,SR1,4,8,8)

CALL MPRD(SR1,OP,SR2,4,8,8,2,3)

CALL GMTRA(SR2,SC0,SR3,4,8,8)

CALL GMTRA(SR3,SB,SR4,4,8,4)

0011

0012

0013

0014

0015

0016

0017

0018

0019

0020

0021

0022

0023

0024

0025

0026

0027

0028

0029

0030

0031

0032

0033

FORTRAN IV V01B-02 THU 26-MAY-77 14:37:23 PAGE 002
 CORE=08K, UIC=0123,11 .LPVLI:1=A7MAN

```

0034 CALL GNADD(SR4,PP,REAR,4,4,0,2)
0035 CALL GNPRD(SCO,SA,SCOR,8,8,8)
0036 CALL GNPRD(SAM,SCO,SAMCO,8,8,8)
0037 CALL GMSUB(SCOR,SAMCO,CAMCAM,8,8)
0038 CALL GNPRD(SR2,CAMCAM,5,4,8,8)
0039 CALL GNTRA(CAMCAM,CANT,8,8)
0040 CALL GNPRD(CANT,OP,01,8,8,0,2,8)
0041 CALL GNPRD(O1,CAMCAM,OBAR,8,8,8)
0042 CALL MINV(REAR,4,DETS,LSSTO,FB,MSSTO,FB)
0043 CALL GNPRD(SB,REAR,BR,8,4,4)
0044 CALL GNPRD(BR,5,BPS,8,4,8)
0045 CALL GMSUB(SA,BPS,AMERS,8,8)
0046 CALL GNTRA(S,ST,4,8)
0047 CALL GNPRD(ST,REAR,STR,8,4,4)
0048 CALL GNPRD(STR,5,PDOT4,8,4,8)
0049 MODEC=0
0050 2 CONTINUE
0051 CALL GNPRD(P,AMERS,PDOT1,8,8,8)
0052 CALL GNTRA(AMERS,ABRST,8,8)
0053 CALL GNPRD(ABRST,P,PDOT2,8,8,8)
0054 CALL GNPRD(P,SB,PB,8,8,4)
0055 CALL GNPRD(PB,REAR,PBR,8,4,4)
0056 CALL GNPRD(PBR,SBT,PBRBT,8,4,8)
0057 CALL GNPRD(PBRBT,P,PDOT3,8,8,8)
0058 CALL GNADD(PDOT1,PDOT2,PDOT5,8,8)
0059 CALL GMSUB(PDOT5,PDOT3,PDOT1,8,8)
0060 CALL GNADD(PDOT1,OBAR,PDOT5,8,8)
0061 CALL GMSUB(PDOT5,PDOT4,PDOT,8,8)
0062 CALL SHFY(PDOT,C,PDOT,8,8,0)
0063 3 CONTINUE
0064 RETURN
0065 END

```


FORTRAN IV V01B-02 THU 26-MAY-77 14:37:29 PAGE 001
 CORE=09K. UID=C123.17 ,LP/L1:1=ATTN

SUBROUTINE KBARUP

0001 C

C

REAL K1

0002

COMMON/K/REAR(16),S(32),SBT(32),PKBAR(32),P(64)

0003

DIMENSION SBT(32),K1(32)

0004

C

THIS SUBROUTINE SOLVES EQUATION 11 PAGE 7

DATA C/-1./

0005

CALL GMPD(SBT,P,SBTP,4,8,8)

0006

CALL GMADD(S,SBTP,K1,4,8)

0007

CALL GMPD(REAR,K1,PKBAR,4,4,8)

0008

CALL SMPY(PKBAR,C,PKBAR,4,8,8)

0009

RETURN

0010

END

0011

PAGE 001
LP/LI:1=A7MAN

FORTRAN IV V01B-02 THU 26-MAY-77 14:37:29
CORE=08K, UIC=1123.1J

```

0001 SUBROUTINE KALMAN(XHAT)
0002 C
0003 C
0004 COMMON /A7/DAT(20),INT(10),R(5)
0005 COMMON /MATR/ SA(64),SB(32)
0006 COMMON /KAL/ XHATK(8),RKG(64),C(8),CT(8),VRIN(8)
0007 COMMON /NOISE/SIGMA(8),YM(8)
0008 COMMON /K/ RBAR(16),S(32),SBT(32),RKBAR(32)
0009 COMMON /OUT/YMOUT(8),SYDOUT(8.5)
0010 COMMON /CON/IA
0011 DIMENSION XHAT(8)
0012 EQUIVALENCE(KEN,INT(6))
0013 C
0014 C THE TASK OF THIS SUBROUTINE IS TO SOLVE THE
0015 C FILTER EQUATIONS (EQUATION 13 PAGE 7)
0016 C TO GET XHAT
0017 DATA AM/0./
0018 IF(KEN.EQ.1) GOTO 400
0019 IF(LKEN.EQ.1) GOTO 400
0020 LKEN=1
0021 REWIND 4
0022 WRITE(4) R(2)
0023 DO 200 I=1,8
0024 VRIN(I)=1./((SIGMA(I)*SIGMA(I))
0025 200 CONTINUE
0026 CALL NTRA(C,CT,8.8.2)
0027 400 CONTINUE
0028 C
0029 C CALCULATE KG (EQUATION 14 PAGE 7)
0030 CALL KGSUB
0031 C
0032 C CALCULATE XHAT (EQUATION 13 PAGE 7)
0033 CALL XHATSB
0034 DO 300 I=1,8
0035 XHAT(I)=XHATK(I)
0036 300 CONTINUE
0037 RETURN
0038 END

```

FORTRAN IV V018-02 THU 26-MAY-77 14:37:30 PAGE 001
 CORE=08K. UIC=1123.1J .LP/LI:1=A7HAN

```

0001 SUBROUTINE KGSUB
0002 C
0003 C
0004 COMMON /A7/DAT(20),INT(10),R(5)
0005 COMMON /KAL/ XHATK(8),RKG(64),C(8),CT(8),VRIN(8)
0006 COMMON /CON/IA
0007 COMMON /OUT/YNOUT(8),SVDOUT(8,5)
0008 DIMENSION V(64),STORV(8,64),PT(5),VDOT(64),VCT(64),SIG(8)
0009 DIMENSION VLOCAL(64,5)
0010 EXTERNAL VLOCAL,VRES
0011 EQUIVALENCE(KEN,INT(6))
0012 DATA NO/64/

0011 C
0012 C WITH SUBROUTINE RKGS AND SUBROUTINE VLOCAL
0013 C EQUATION 15 PAGE 8 IS SOLVED TO GET V
0014 IF(KEN.EQ.1) GOTO 400
0015 PT(3)=R(2)/2.
0016 PT(4)=.001
0017 SIG(1)=10.
0018 SIG(2)=.1
0019 SIG(3)=.2
0020 SIG(4)=.1
0021 SIG(5)=.2
0022 SIG(6)=.2
0023 SIG(7)=.1
0024 SIG(8)=.1
0025 DO 100 I=1,64
0026 VLOCAL(I,IA)=0.
0027 100 CONTINUE
0028 DO 200 I=1,8
0029 VLOCAL(8*(I-1)+I,IA)=SIG(I)**2
0030 200 CONTINUE
0031 IF(IA.EQ.1) PT(1)=PT(2)
0032 PT(2)=PT(1)+R(2)
0033 DO 400 I=1,64
0034 VDOT(I)=1./64.
0035 V(I)=VLOCAL(I,IA)
0036 400 CONTINUE
0037 SVDOUT(1,IA)=SORT(V(1))
0038 DO 300 I=2,8
0039 SVDOUT(I,IA)=57.295*SORT(V(8*(I-1)+I))
0040 300 CONTINUE
0041 CALL RKGS(PT,V,VDOT,NO,INFF,VLOCAL,VRES,STORV)
0042 DO 500 I=1,64
0043 VLOCAL(I,IA)=V(I)
0044 500 CONTINUE
0045 C
0046 C CALCULATION OF KG
0047 CALL MPRD(V,CT,VCT,8,8,0,2,8)
0048 CALL MPRD(VCT,VRIN,RKG,8,8,0,2,8)

```

PAGE 002
LP/LI:1-A7MAN

THU 26-MAY-77 14:37:30

FORTTRAN IV V01B-02
CORE=08K. UIC=[123.1]

0046 RETURN
0047 END

FORTRAN IV V01B-02 THU 26-MAY-77 14:37:32 PAGE 001
 CORE=08K. UIC=123.1J .LP/LI:1=A7MAN

```

0001 SUBROUTINE VCAL(X,V,VDOT)
      C
      C
0002 COMMON /MATR/ SA(64)
0003 COMMON /KAL/ XHATK(8),RKG(64),C(8),CT(8),VRIN(8)
0004 DIMENSION V(64),VDOT(64),TEMP1(64),TEMP2(64),TEMP3(64)
0005 DIMENSION AT(64)
      C
      C
0006 SOLVING EQUATION 15 PAGE 8 TO GET V
0007 DO 100 I=1,8
0008   IDIAG=8*(I-1)+1
0009   IF(V(IDIAG).LT.0.) V(IDIAG)=0.
0010   100 CONTINUE
0011   CALL GINTR(SA,AT,8,8)
0012   CALL GMPRD(SA,V,TEMP1,8,8,8)
0013   CALL GMPRD(V,AT,TEMP2,8,8,8)
0014   CALL GMADD(TEMP1,TEMP2,TEMP3,8,8)
0015   CALL MPRD(V,CT,TEMP1,8,8,0.2,8)
0016   CALL MPRD(TEMP1,VRIN,TEMP2,8,8,0.2,8)
0017   CALL MPRD(TEMP2,C,TEMP1,8,8,0.2,8)
0018   CALL GMPRD(TEMP1,V,TEMP2,8,8,8)
0019   CALL GMSUB(TEMP3,TEMP2,VDOT,8,8)
0020   RETURN
0021   END

```


PAGE 001
 .LP/LI:1=ATMAN

FORTRAN IV V01B-02 THU 26-MAY-77 14:37:33
 CORE=08K. UIC=C123.17

```

0001 SUBROUTINE VRES(X,V,VDOT,IHFF,PT)
      C -----
      C
0002 DIMENSION V(64),VDOT(64),PT(5)
      C
      C CONCERNING D IN FIRST COLUMN SEE
      C COMMENT IN SUBROUTINE OUTP
0003 IF(IHFF.EQ.IHFF1) GOTO 1
      D WRITE(6,100) IHFF
      D 100 FORMAT('0IHFF IN KALMAN (V):',I2)
      IHFF1=IHFF
0005 1 CONTINUE
0006 RETURN
0007 END
0008

```

FORTRAN IV V01B-02 THU 26-MAY-77 14:37:34 PAGE 001
 CORE=08K, UTC=C123.11 ,LP/LI:1=A7MAN

```

0001 SUBROUTINE XHATSB
-----
0002 COMMON /A7/ DAT(20),INT(10),R(5)
0003 COMMON /KAL/ XHATK(8),RKG(64),C(8),CT(8),VRIN(8)
0004 COMMON /K/RBAR(16),S(32),SBT(32),RKBAR(32)
0005 COMMON /CON/IA
0006 DIMENSION XH(8),XHDOT(8),STORXH(8,8),PXH(5)
0007 DIMENSION XHLOC(8,5)
0008 EXTERNAL XHCAL,XHRES
0009 EQUIVALENCE (KEN,INT(6))

C
C WITH SUBROUTINE RKG AND SUBROUTINE XHCAL
C THE FILTER EQUATIONS ARE SOLVED
C
0010 DATA NOX/8/
0011 IF (KEN.EQ.1) GOTO 400
0013 PXH(3)=R(2)/2.
0014 PXH(4)=.001
0015 DO 100 I=1,8
0016 XHLOC(I,1A)=0.
100 CONTINUE
0017 IF (INT(2).EQ.4) XHLOC(5,1A)=-10./57.295
0018
0020 CONTINUE
0021 IF (IA.EQ.1) PXH(1)=PXH(2)
0023 PXH(2)=PXH(1)+R(2)
0024 DO 450 I=1,8
0025 XHDOT(1)=1./8.
0026 XH(1)=XHLOC(1,1A)
0027
450 CONTINUE
0028 CALL RKG(PXH,XH,XHDOT,NOX,IXH,XHCAL,XHRES,STORXH)
0029 DO 200 I=1,8
0030 XHATK(I)=XH(I)
0031 XHLOC(I,1A)=XH(I)
0032
200 CONTINUE
0033 RETURN
0034 END

```

FORTRAN IV V01B-02 THU 26-MAY-77 14:37:35 PAGE 001
 CORE=00K. UIC=C123.1J ,LP/LI:1=ATMAN

```

0001 SUBROUTINE XHCAL(X,XH,XHDOT)
      C
      C
0002 COMMON /A7/DAT(20),INT(10),R(5)
0003 COMMON /MATR/SA(64),SB(32)
0004 COMMON /KAL/ XHATK(8),RKG(64),C(8),CT(8),VRIN(8)
0005 COMMON /NOISE/SIGNA(8),YIN(8)
0006 COMMON/K/PBAR(16),S(32),SBT(32),RKBAR(32)
0007 DIMENSION XH(8),XHDOT(8),TEMP1(8),TEMP2(64)
0008 DIMENSION TEMP3(8),TEMP4(8),TEMP5(8),YML(8)
      C
      C
      EQUATION 13 PAGE 7 IS USED TO GET XHAT
      CALL GMPRD(SA,XH,TEMP1,8,8,1)
      CALL GMPRD(SB,RKBAR,TEMP2,8,4,8)
      CALL GMPRD(TEMP2,XH,TEMP3,8,8,1)
      CALL NPRD(C,XH,TEMP4,8,2,8,1)
      CALL GMSUB(YIN,TEMP4,TEMP5,8,1)
      CALL GMPRD(RKG,TEMP5,TEMP4,8,8,1)
      CALL GMADD(TEMP1,TEMP3,TEMP5,8,1)
      CALL GMADD(TEMP5,TEMP4,XHDOT,8,1)
      RETURN
      END
0009
0010
0011
0012
0013
0014
0015
0016
0017
0018

```

PAGE 001
LP/LI:1=07MAN

THU 26-MAY-77 14:37:37

FORTRAN IV V018-02
CORE=08K, UTC=[123.1]

SUBROUTINE XHRES(X,XH,XHDOT,IHXH,NXH,PXH)

0001 C
0002 C

0003 COMMON/AT/ DAT(20),INT(10),R(5)
0004 COMMON /CON/IA,A13,A14,A37,SUMP(4),XHI(8,5)
0005 DIMENSION XH(8),XHDOT(8),PXH(5)
EQUIVALENCE (INT(7),IHMAX)

0006 C THIS SUBROUTINE PROVIDES THE MAXIMUM NUMBER
0007 C OF BISECTIONS OF THE INITIAL TIME INCREMENT
0008 C IN THE XHAT-CALCULATION

IHMAX=IHAXD(IHMAX,IHXH)

DO 200 I=1,8

XHI(I,IA)=XH(I)

200 CONTINUE

RETURN

END

0006
0007
0008
0009
0010
0011

APPENDIX C
Model Following Program

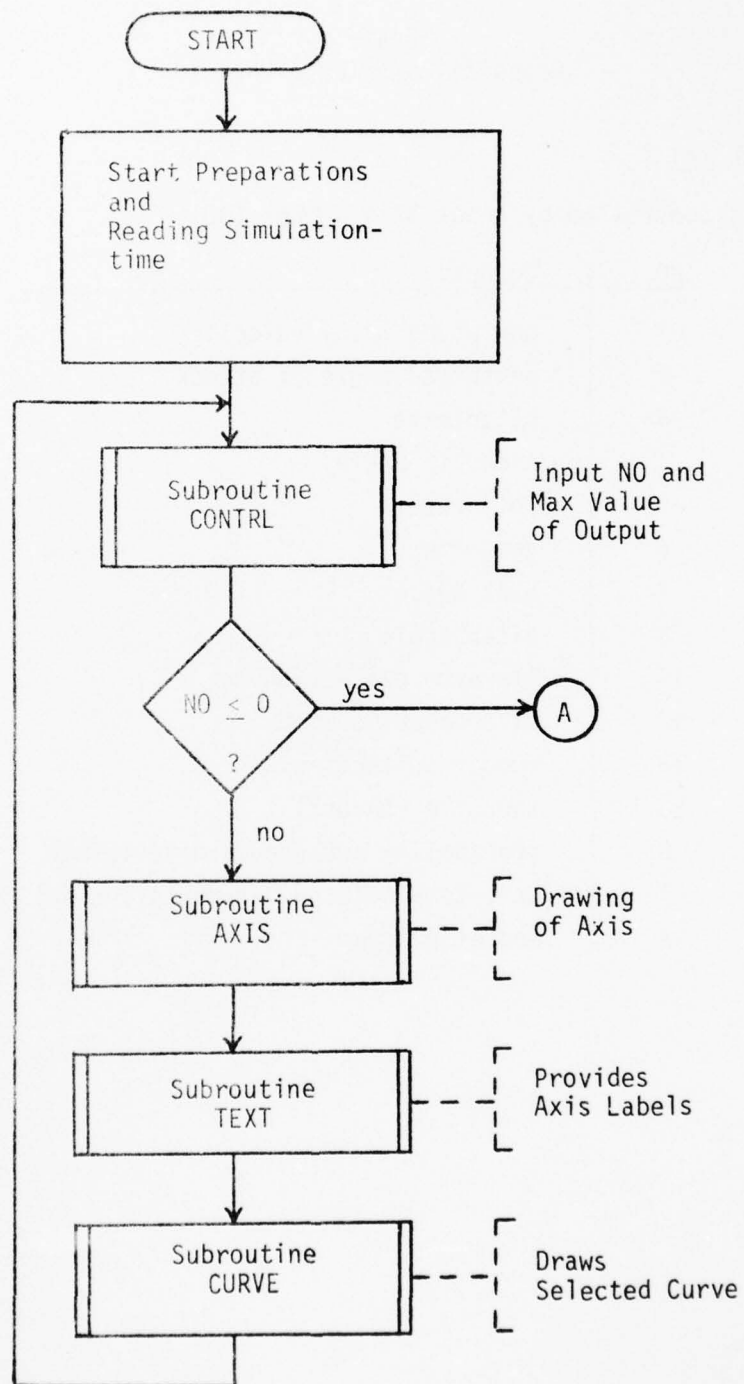
-Part 2-
Graphical Display of Results

1. Program Control

The program was controlled by input of a number (NO):

NO	Output
1	perturbed total velocity
2	perturbed angle of attack
3	pitch rate
4	sideslip angle
5	roll rate
6	yaw rate
7	bank angle
8	pitch angle
11	elevator deflection
12	aileron deflection
13	rudder deflection
14	throttle (thrust)
0	probability and standard deviation
-1	as 1 to 8 with measurements (noise)
-2	end of program

2. Flow Chart Program, Part 2



AD-A041 436

FRANK J SEILER RESEARCH LAB UNITED STATES AIR FORCE --ETC F/G 19/5
HIGH ANGLE OF ATTACK FLIGHT CONTROL USING STOCHASTIC MODEL REFE--ETC(U)
MAY 77 R B ASHER, D GOEBEL

UNCLASSIFIED

FJSRL-TR-77-0010

NL

2 OF 2

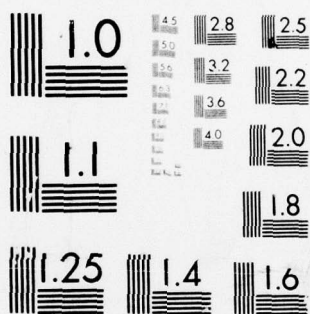
AD
A041436



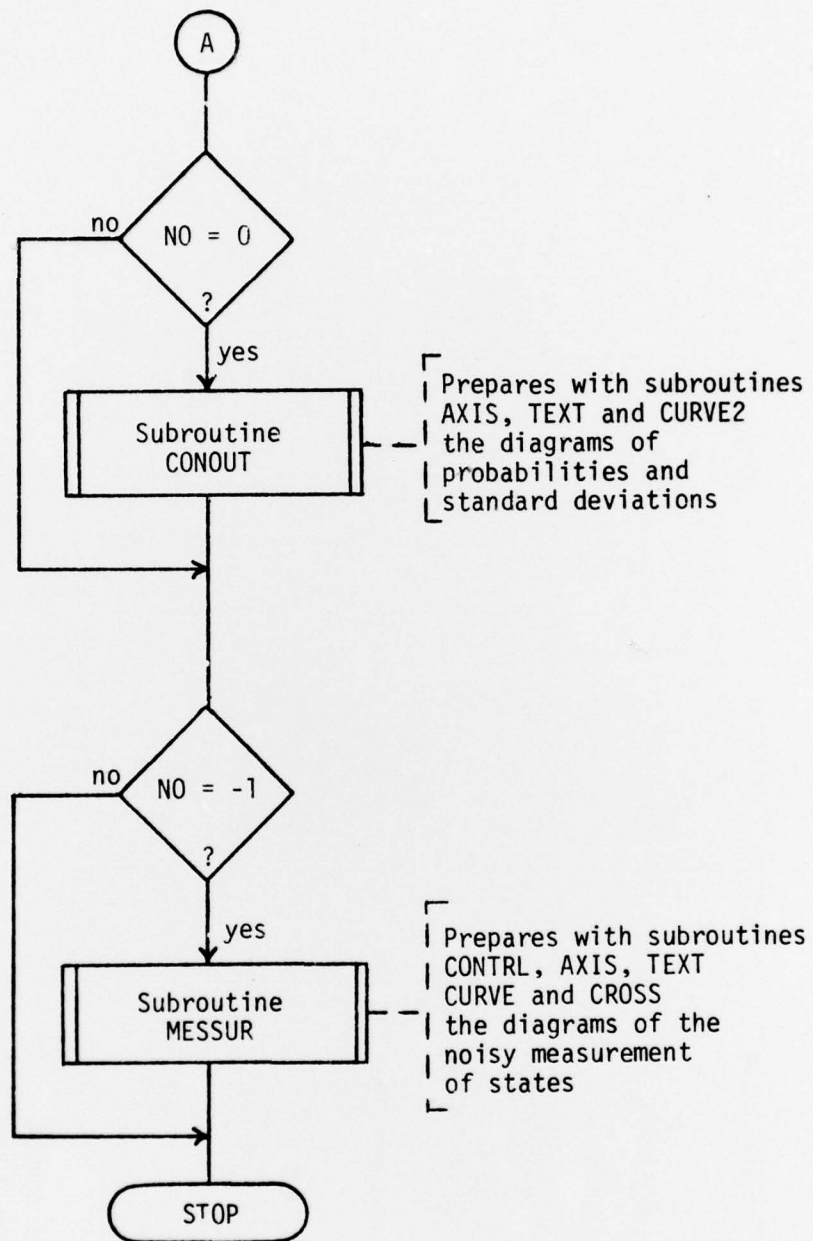
END

DATE
FILMED

7-77



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A



FORTRAN IV V01B-02 THU 26-MAY-77 14:41:31 PAGE 001
 CORE=08K. UIC=[123,1] ,LP:/LI:1=A7IEN

MODEL FOLLOWING PROGRAM

PART 2

GRAPHICAL DISPLAY
 VERSION 5/26

IN THIS PART OF THE PROGRAM MANY SUBROUTINES
 OF THE GRAPHICAL DISPLAY PACKAGE (SEE MANUAL)
 WERE USED

COMMON /DISP/41,NO,EMAX,M,IB

DIMENSION ITIME(4)

DATA IDEV/TT/

1

WRITE(5,10)

10 FORMAT('NUMBER OF DIALOG TERMINAL: '))

READ(5,11) NUMT

IF(NUMT.NE.3.AND.NUMT.NE.4) GOTO 1

11

FORMAT(11)

CALL ASNLUN(6,IDEV,NUMT)

CALL INITT(400)

CALL TERM(NUMT-2,1024)

IF(NUMT.EQ.4) CALL CHRSLZ(3)

REWIND 1

READ(1) XM

CALL TWINDO(400,900,300,700)

350

CONTINUE

CALL CONTRL

IF(NO.LE.0) GOTO 900

YB=-1.

CALL AXIS

CALL TEXT

CALL CURVE

CALL TSEND

READ(6,550) DUM

550 FORMAT(F4.0)

GOTO 350

900 CONTINUE

IF(NO.EQ.0) CALL CONOUT

IF(NO.EQ.-1) CALL MESSUR

CALL HOME

CALL NEWFAG

READ(3) ITIME

WRITE(5,950) ITIME

950 FORMAT('END OF CALCULATION AT: ',4A2)

CALL FINITT(0,720)

STOP

END

0001

0002

0003

0004

0005

0006

0007

0008

0009

0010

0011

0012

0013

0015

0016

0017

0018

0019

0020

0022

0023

0024

0025

0026

0027

0028

0029

0030

0031

0033

0035

0036

0037

0038

0039

0040

0041

0042

FORTRAN IV V01B-02 THU 26-MAY-77 14:42:02 PAGE 001
 CORE=09K, UIC=123.11 ,LP:1:1=A7MBN

```

0001 SUBROUTINE AXIS
0002 COMMON /DISP/XM,NO,RMAX,M,YB
0003 CALL DWINDO(0.,XM,YB,1.)
0004 CALL NEWFAG
0005 IDELTA=100
0006 IF(XM.LE.1000.) IDELTA=10
0008 IF(XM.LE.100.) IDELTA=1
0010 IXM=XM+IDELTA
0011 DO 351 IX=IDELTA,IXM, IDELTA
0012 XPLOT=IX-IDELTA
0013 CALL MOVEA(XPLOT,YB)
0014 YL=YB+.02
0015 IF(IX.EQ.IDELTA) YL=1.
0017 IF(IX.EQ.IXM) YL=1.
0019 CALL DRAWA(XPLOT,YL)
0020 IXP=400+500*XPLOT/XM
0021 CALL MOVABS(IXP,175)
0022 CALL TSEND
0023 XOUT=XM*(IX-1)/(IXM-1)
0024 WRITE(6,500) XOUT
0025 FORMAT('+',F2.0)
0026 500 CONTINUE
0027 NY=1
0028 IF(YB.GE.0.) NY=11
0030 DO 352 IY=NY,21
0031 YPLOT=(IY-11)/10.
0032 CALL MOVEA(0.,YPLOT)
0033 XL=.01*XM
0034 IF(IY.EQ.11) XL=XM
0036 IF(IY.EQ.1) XL=XM
0038 IF(IY.EQ.21) XL=XM
0040 CALL DRAWA(XL,YPLOT)
0041 IF(MOD(IY,2).EQ.0) GOTO 352
0043 IYP=200+25*(IY-1)
0044 IF(YB.GE.0) IYP=200+50*(IY-11)
0046 CALL MOVABS(300,IYP)
0047 CALL TSEND
0048 YOUT=-RMAX+2*RMAX*(IY-1)/20
0049 WRITE(6,501) YOUT
0050 501 FORMAT('+',F6.1)
0051 352 CONTINUE
0052 RETURN
0053 END

```

FORTRAN IV V01B-02 THU 26-MAY-77 14:42:10 PAGE 001
CORE=08K. UIC=[123.1] .LP:21:1=ATMEN

```

0001 SUBROUTINE TEXT
0002 COMMON /DISP/XT,NO,RMAX,M,YB
0003 DIMENSION ITEXT(10,14)
0004 DATA ITEXT/
0005 1'VE','LO','CI','TY',,,'A','TT','AC','K',,
0006 2'AN','GL','E','OF','A',,,'TE',,,'R',,
0007 3'PI','TC','H',,,'PA','TE',,,'E',,
0008 4'SI','DE',,,'S','LI','P',,,'AN','GL','E',,
0009 5'RO','LL',,,'R',,,'AT','E',,,'E',,
0010 6'YA',,,'W',,,'PA','TE',,,'E',,
0011 7'BA',,,'NK',,,'A',,,'NG',,,'LE',,,'E',,
0012 8'PI',,,'TC',,,'H',,,'AN',,,'GL',,,'E',,
0013 9',,,'',,,'',,,'',,,'',,,'',,,'',,,'',,
0014 A',,,'',,,'',,,'',,,'',,,'',,,'',,,'',,
0015 1'EL','EV',,,'AT',,,'OR',,,'D',,,'EF',,,'LE',,,'CT',,,'IO',,,'N',,
0016 2'AI',,,'LE',,,'RO',,,'N',,,'DE',,,'FL',,,'EC',,,'TI',,,'ON',,
0017 3'PU',,,'DD',,,'ER',,,'D',,,'EF',,,'LE',,,'CT',,,'IO',,,'N',,
0018 4'TH',,,'RO',,,'TT',,,'LE',,,'E',,
0019 CALL MOVABS(680,150)
0020 CALL TSEND
0021 WRITE(6,100)
0022 100 FORMAT(' + TIME [SEC]')
0023 IXT=330
0024 IF (YB,GE,0) IXT=510
0025 CALL MOVABS(IXT,760)
0026 CALL TSEND
0027 WRITE(6,200) (ITEXT(I,NO),I=1,10)
0028 200 FORMAT(' + ',10A2)
0029 CALL MOVABS(330,730)
0030 CALL TSEND
0031 GO TO (1,2,3,2,3,3,2,2,5,5,2,2,2,4),NO
0032 1 WRITE(6,201)
0033 GOTO 5
0034 2 WRITE(6,202)
0035 GOTO 5
0036 3 WRITE(6,203)
0037 GOTO 5
0038 4 WRITE(6,204)
0039 FORMAT(' + [FPS]')
0040 201 FORMAT(' + [FPS]')
0041 202 FORMAT(' + [DEG]')
0042 203 FORMAT(' + [DEG/SEC]')
0043 204 FORMAT(' + [ - ]')
0044 5 RETURN
0045 END

```


PAGE 001
LP: 11:1=A71EN

THU 26-MAY-77 14:42:18

FORTRAN IV V01B-02
CORE=08K, UIC=113.11

```

0001 SUBROUTINE CONTRL
0002 COMMON /DISP/MI.NO,RMAX,M.YB
0003 IYU=700
0004 500 CONTINUE
0005 IYU=IYU-100
0006 CALL MOVABS(0,IYU)
0007 CALL TSEND
0008 WRITE(6,501)
0009 501 FORMAT('END: ')
0010 READ(6,502) NO
0011 502 FORMAT(I2)
0012 IF (NO.GT.20) GOTO 500
0013 IF (NO.LE.0) GOTO 600
0014 WRITE(6,511)
0015 511 FORMAT('MAX VALUE: ')
0016 READ(6,512) RMAX
0017 512 FORMAT(F8.0)
0018 600 RETURN
0019 600
0020 END
0021

```

PAGE 001
LP:ALI:1-A718N

FORTRAN IV V01B-02 THU 26-MAY-77 14:42:23
COPE=08K, UIC=C123.1J

```

0001 SUBROUTINE CURVE
0002 COMMON /DISP/MI,NO,RMAX,M,YB
0003 DIMENSION DAT(20),NUM(3),ITIME(4)
0004 DATA NUM,'A7','NO','HE' /
0005 DY0=.04
0006 DX=.02*MI
0007 NFILE=1
0008 IF(MI.EQ.1) NFILE=3
0009
100 REWIND NFILE
0010 READ(NFILE) XI
0011
400 CONTINUE
0012 READ(NFILE) DAT
0013 YPLOT=DAT(NO)*RMAX
0014 XPLOT=DAT(20)
0015 IF(XPLOT.GT.0.) GOTO 401
0016 CALL MOVEA(0.,YPLOT)
0017 XNUM=XI*(.5+NFILE/10.)
0018 GOTO 400
0019
401 CONTINUE
0020 CALL DRAWA(XPLOT,YPLOT)
0021 IF(XPLOT.GT.XNUM) GOTO 450
0022 YPLOT1=YPLOT
0023 IF(XPLOT.GE.XM) GOTO 800
0024
450 CONTINUE
0025 DY=DY0
0026 IF(YPLOT.GT.YPLOT1) DY=-DY
0027 CALL DRAWA(XPLOT+DX,YPLOT+DY)
0028 IF(DY.LT.0.) CALL MOVEA(XPLOT+DX,YPLOT+3.*DY)
0029 CALL ROUTST(2,NUM(NFILE))
0030 CALL MOVEA(XPLOT,YPLOT)
0031 XNUM=2.*XI
0032 GOTO 400
0033
200 CONTINUE
0034 NFILE=NFILE+1
0035 IF(NFILE.LE.3) GOTO 100
0036 RETURN
0037 END
0038
0039
0040
0041
0042
0043
0044
0045

```

FORTRAN IV V018-02 THU 26-MAY-77 14:42:37 PAGE 001
 CORE=08K. UTC=C123.1J ,LP:LI:1=A7MEN

```

0001 SUBROUTINE CONOUT
0002 COMMON /DISP/XTI.NO,RMAX,M.YB
0003 DIMENSION YHOUT(8),PROUT(5),SVDO(8,5)
0004 YB=0.
0005 XMAX=XI
0006 RMAX=1.
0007 YMAX=RMAX
0008 CALL AXIS
0009 CALL MOVABS(600,150)
0010 CALL TSEND
0011 WRITE(6,50)
0012 FORMAT('TIME [SEC]')
0013 CALL MOVABS(330,730)
0014 CALL TSEND
0015 WRITE(6,60)
0016 FORMAT('PROBABILITY')
0017 DO 100 I=1,5
0018 REWIND 4
0019 READ(4) DELTIM
0020 CONTINUE
0021 READ(4) T,YHOUT,PROUT,SVDO
0022 CALL CURVE2(T,PROUT(I),I,XMAX,YMAX)
0023 IF(T.LT.XM-DELTIM) GOTO 500
0024 CALL TSEND
0025
0026 FORMAT(F4.0)
0027 CONTINUE
0028 READ(6,550) DUM
0029 RMAX=10.
0030 YMAX=RMAX
0031 DO 200 I=1,8
0032 CALL AXIS
0033 CALL MOVABS(330,760)
0034 CALL TSEND
0035 WRITE(6,70)
0036 FORMAT('STD. DEV. OF ')
0037 NO=1
0038 CALL TEXT
0039 DO 300 N=1,5
0040 REWIND 4
0041 READ(4) DELTIM
0042 CONTINUE
0043 READ(4) T,YHOUT,PROUT,SVDO
0044 CALL CURVE2(T,SVDO(I,N),N,XMAX,YMAX)
0045 IF(T.LT.XM-DELTIM) GOTO 400
0046 CONTINUE
0047 CALL TSEND
0048 READ(6,550) DUM
0049 CONTINUE
0050 RETURN
0051 END
0052

```

```

FORTRAN IV      V018-02      THU 26-MAY-77 14:42:47      PAGE 001
CORE=08K.  UTC=[123.11]      .LP: 1:1=H718N

0001      SUBROUTINE CURVE2(X,Y,N,XMAX,YMAX)
0002      Y=Y/YMAX
0003      IF (X.GT.0.) GOTO 401
0005      CALL MOVEA(0.,Y)
0006      XNUM=XMAX*(.3+N/.15.)
0007      DY0=.04
0008      DX=.02*XMAX
0009      RETURN
0010      401 CONTINUE
0011      CALL DRAWA(X,Y)
0012      IF (X.GT.XNUM) GOTO 450
0014      Y1=Y
0015      IF (X.GE.XMAX) CALL TSEND
0017      RETURN
0018      450 CONTINUE
0019      DY=DY0
0020      IF (Y.GT.Y1) DY=-DY
0022      CALL DRAWA(X+DX,Y+DY)
0023      IF (DY.LT.0.) CALL MOVEA(X+DX,Y+2.*DY)
0025      CALL TSEND
0026      WRITE(6,1) N
0027      FORMAT('+.11)
0028      CALL MOVEA(X,Y)
0029      XNUM=2.*XMAX
0030      RETURN
0031      END

```

PAGE 001
.LP: /LI: 1=A7MEN

THU 26-MAY-77 14:42:52

FORTRAN IV V01B-02
CORE=08K. UID=C123.1J

```

0001 SUBROUTINE MESSUR
0002 COTTON /DISP/CM,NO,RMAX,M,YB
0003 YB=-1.
0004 N=1
0005 1 CONTINUE
0006 READ(6.2) DUM
0007 2 FORMAT(F4.0)
0008 CALL NEWFAG
0009 CALL CONTRL
0010 IF(N0.LE.0) RETURN
0011 CALL AXIS
0012 CALL TEXT
0013 CALL CURVE
0014 CALL CROSS
0015 CALL TSEND
0016 GOTO 1
0017 RETURN
0018 END
0019

```


PAGE 001
LP:ALI:1=ATMEN

THU 26-MAY-77 14:42:59

FORTRAN IV V01B-02
CORE=00K, UID=C123.11

```

0001 SUBROUTINE CROSS
0002 COMMON /DISP/XM,NO,RMAX,M
0003 DIMENSION YMOUT(8),PROUT(5),SVOUT(8,5)
0004 DY=.01
0005 DX=XM/200.
0006 REWIND 4
0007 READ(4) DELTIM
0008 READ(4) T,YMOUT,PROUT,SVOUT
0009 YPLOT=YMOUT(NO)/RMAX
0010 XPLOT0=T-DX
0011 YPLOT0=YPLOT-DY
0012 YPLOT1=T+DX
0013 YPLOT1=YPLOT+DY
0014 CALL MOVEA(XPLOT0,YPLOT0)
0015 CALL DRAWA(XPLOT1,YPLOT1)
0016 CALL MOVEA(XPLOT0,YPLOT1)
0017 CALL DRAWA(XPLOT1,YPLOT0)
0018 IF(T.LT.(XM-DELTIM)) GOTO 1
0020 RETURN
0021 END

```