NASA CR-145004

# DESIGN OF AN ALL-ATTITUDE FLIGHT CONTROL SYSTEM TO EXECUTE COMMANDED BANK ANGLES AND ANGLES OF ATTACK

By

George H. Burgin

and

David M. Eggleston

**NASA**

National Aeronautics and
Space Administration

# ABSTRACT

A flight control system for use in air-to-air combat simulation has been designed. The input to the flight control system are commanded bank angle and angle of attack, the output are commands to the control surface actuators such that the commanded values will be achieved in near minimum time and sideslip is controlled to remain small.

For the longitudinal direction, a conventional linear control system with gains scheduled as a function of dynamic pressure is employed. For the lateral direction, a novel control system, consisting of a linear portion for small bank angle errors and a bang-bang control system for large errors and error rates is employed.

## PREFACE

The work reported herein was performed under Contract NAS1-13773 between January 1975 and January 1976.  The study was conducted under the direction of W. Hankins, Simulation and Human Factors Branch, Langley Research Center, NASA.

This study was performed at Decision Science, Inc. with George H. Burgin acting as responsible project manager and David M. Eggleston as principal investigator.

The purpose of this contract was twofold.  The primary goal was to design and to demonstrate a control system that would accept, from the "Adaptive Maneuvering Logic" (AML) program commands for angle of attack and for bank angle and translate these commands into control surface deflections.

The secondary goal of this contract was to make available to the Simulation and Human Factors Branch at LRC a package of computer programs which can be used in the design of such a control system for other aircraft than F-4.

Both these goals have been achieved.  The aircraft driven by the AML program and controlled by the control system described in this report not only performs as well in simulated air combat than the previous performance model, but in some instances performs superior to the performance model.

The computer programs described in this report demonstrate that the secondary goal has also been achieved.  Enough documentation of these programs is provided in this report so that prospective users of these programs should have no problems in using them to design similar control systems for different aircraft.

CONTENTS

v

# DESIGN OF AN ALL-ATTITUDE FLIGHT CONTROL SYSTEM TO EXECUTE COMMANDED BANK ANGLES AND ANGLES OF ATTACK

By George H. Burgin and David M. Eggleston
Decision Science, Inc.

## SUMMARY

This final report on Contract NAS1-13773 is organized in three different major parts plus an Appendix.

In the first part, the general concepts and the philosophy that were used in developing a flight control system for the F-4, as represented by the equations of motion programmed on the DMS, are described. The flight control system accepts as input commanded angle of attack and bank angle and determines from these and the current state variables of the controlled aircraft the required stabilator, aileron and rudder deflections. The general six degree of freedom equations of motion are presented, as well as the linearized equations for the longitudinal and the lateral axes. Discussed are also difficulties of general nature associated with the roll maneuver.

The second part specializes in the above concepts for an F-4, with special emphasis on the adverse yaw and inertial coupling. An attempt is made to formulate a set of performance specifications for such a control system, and the associated difficulties are discussed. Finally, the control system as designed for longitudinal and lateral axis is given and the method of determining its gains is explained and some typical response time histories are presented.

The third part of the report is devoted to a description of the computer programs developed under this contract. Detailed flow charts of the major programs and subroutines are presented, the input requirements for each individual program are listed and instructions for the use of these programs are presented. Two

major programs and a set of five supporting programs were developed. The program ATDYN is a versatile, general purpose program for linear control system transfer function analysis and the program AML75 employs the developed control system to control an F-4 driven by the AML program in an air-to-air combat simulation. The supporting five programs are used for pole and zero determination in closed-loop root locus studies for determining reasonable values for control system gains.

# INTRODUCTION

It is a well known fact that piloting an aircraft in close-in air-to-air combat is one of the most demanding tasks for a pilot. A thorough understanding of the complex dynamic relationships governing the flight in air-to-air combat is required for pilots to be trained effectively for this difficult task. This same insight into flight dynamics is a prerequisite for the design of new fighter aircraft, associated weapons systems and flight control systems.

Considerable progress has been made during the last years in providing the aircraft design engineers and the pilots with air-to-air combat simulations. These range from relatively unsophisticated flight simulators to air-combat simulation, where real aircraft, equipped with range instrumentation, fly over an air-combat maneuvering range. Flight instructors can monitor the engagement from ground consoles; the engagement may also be replayed for debriefing purposes.

The Differential Maneuvering Simulator (DMS) at the NASA, Langley Research Center (LRC) is a good example of a flight simulator which can be used for air-to-air combat simulation and which provides a very high degree of realism. It is used in LRC's Advanced Fighter Technology program and is described in Reference 1. Among the research tools available on that simulator is a computer driven, interactive opponent, selecting its maneuvers according to an Adaptive Maneuvering Logic (AML) (References 2, 3 and 4). This AML driven opponent, often called the "iron pilot", is indeed a worthy opponent and usually "beats" its human adversary.

However, the mathematical model used in the AML program is a performance model. That means, thrust, lift, drag and turning capabilities are represented at their maximum, steady state values. The AML driven aircraft therefore does not adequately represent the

transient behavior and the associated handling qualities of the aircraft. As a consequence, the AML driven aircraft never becomes uncontrollable or unstable. This gives the "iron pilot" a definite advantage, most pronounced in the low speed, low dynamic pressure flight regimes.

The objective of this study was to design a flight control system which can be used to replace the performance model of the present AML program by equations of motion identical to those representing the simulated aircraft flown by the pilot. Thus, the AML driven aircraft will have identical handling qualities and performance as the piloted aircraft. It will no longer be a perfectly stable aircraft, but it will have the same spin and departure characteristics as the aircraft flown by the human pilot.

The three primary control variables in air combat are bank angle, load factor and magnitude of thrust. Thrust is simulated by assuming either idle, military or afterburner thrust. Bank angle and load factor have to be controlled by commanding input to the actuators of the control surfaces of the aircraft, typically to stabilator, aileron, spoiler and rudder actuators.

Whenever the AML program executes a tactical decision, typically once every second, it determines a desired maneuverplane rotation angle and a desired load factor. These commands can be translated into a desired bank angle and desired angle of attack. The problem addressed in this study can be summarized as follows:

> For commanded bank angles ($\phi_c$) and commanded
> angles of attack ($\alpha_c$), which are given at one
> second intervals, find a control law that
> will make the aircraft follow these commands
> in some "optimal" manner.

Keeping in mind that such an attitude control system has to operate over practically the entire flight envelope of a given high performance fighter aircraft, this seems like a formidable task. However, one must realize that the problem is not quite so

4

horrendous as it might seem at first glance for the following
reasons:

(a) The dynamics of the plant (here the aircraft) are accurately
known. Unlike in the design of a control system for a real
aircraft, where stability and control derivatives are only
estimated and have uncertainties associated with them, in the
case considered here all these coefficients are exactly known.
Also known is the exact form of the differential equations
describing the motion of the aircraft to be controlled.

(b) The aircraft is represented as a rigid body.

(c) All aircraft state variables are accessible without noise,
without measurement errors and without sensor dynamics.

(d) The number of nonlinearities in the system is relatively
small.

This report is organized in such a way that it proceeds from
the general topics, applicable for the control system design for
any fighter aircraft to results which are specific for the F-4 and
finally to the detailed descriptions of the computer programs de-
veloped in this study.

To design an efficient flight control system, the basic, open
loop aircraft dynamics must first be understood thoroughly. The
open loop characteristics of the aircraft suggest certain require-
ments of the control system. For instance, strong adverse yaw may
suggest certain strategies for aileron-rudder control for rolling
under high angles of attack.

The first part of the report explains the basic philosophy
and the procedure used in the design of the flight control system.
The results presented in this part are aircraft independent and
they can be used for the design of a similar system for some other
advanced fighter aircraft.

The second part presents results specifically for the F-4.
First, the linearized open and closed-loop behavior of the F-4

aircraft is explained, followed by results for the full set of the nonlinear equations of motion.

Also included in this second part is a discussion of a preliminary set of performance specifications. In the course of the work performed under this contract, it became quite clear that formulating a set of performance specifications for an aircraft/control system for such a complex mission as air-to-air combat becomes a formidable task which in itself would require more resources than were allocated to the entire contract.

The present tendency in aircraft specifications is still to describe the aircraft as an equivalent second order system. It is left to the airframe designer and to the control system designer to develop an aircraft which resembles a second order systems. To the authors of this report, such an approach appears unacceptable for air-to-air combat maneuvering. Not even the most gifted and ingenious designer will ever build an aircraft, which, when performing the drastic maneuvers required in air-to-air combat, will behave like a second order system!

The third part of the report is devoted to the description of the computer programs developed under this contract. Flow charts and detailed input and output descriptions are provided. In addition to the two major programs, AML75 and ATDYN, a set of five small, individual programs for linear control system design are described.

Of the two major programs, AML75 may be considered as the end product of this contract. It allows exercise of the developed control system in any one of the three following modes:

(1)  Single aircraft, driven by an inputted command sequence of angle of attack and bank angle at one second intervals.

(2)  AML driven target aircraft against canned trajectory attacker.

6

(3) AML driven target aircraft against AML driven
     attacker using the old performance math model.

The program ATDYN is a tool for linear transfer function an-
alysis.  It incorporates the subroutine TRANS, a versatile and very
powerful transfer function analysis program.  This program was de-
veloped and used for many years at General Dynamics/Astronautics
and later General Dynamics/Convair for control system design and
analysis of the Atlas and Centaur missiles.  It determines poles
and zeros of closed-loop systems without requiring that the cor-
responding equations be set up in first order form.

# SYMBOLS, SIGN CONVENTIONS AND TERMINOLOGY

Since the equations of motion as used in this study are identical to the equations of motion as documented in Reference 5, the symbols used in this report are, wherever possible, compatible with the ones used in Reference 5. For the sake of completeness, most of these symbols are listed again in the following list:

| | |
|---|---|
| $A_{long}$ | Coefficient matrix for linearized longitudinal equations. |
| $A_{lat}$ | Coefficient matrix for linearized lateral equations. |
| $b$ | Wing span. |
| $B_{long}$ | Vector of forcing function coefficients for longitudinal equations ($\delta_s$). |
| $B_{lat, 1}$ | Vector of forcing function coefficients for lateral equations ($\delta_a$). |
| $B_{lat, 2}$ | Vector of forcing function coefficients for lateral equation ($\delta_R$). |
| $\bar{c}$ | Mean aerodynamic cord. |
| $C_D$ | Drag coefficient. |
| $C_L$ | Lift coefficient. |
| $C_\ell$, $C_m$, $C_n$ | Rolling-, pitching-, and yawing-moment coefficients, respectively. |
| $C_x$, $C_y$, $C_z$ | Longitudinal-, lateral-, and vertical-force coefficients, respectively. |
| $\vec{F}$ | Total external force vector. |
| $F_x$, $F_y$, $F_z$ | x, y and z components in body axis system of the vector $\vec{F}$. |

8

| | |
|---|---|
| $g$ | Acceleration due to gravity. |
| $\vec{G}$ | External moment vector. |
| $\vec{h}$ | Angular momentum vector. |
| $h_x$, $h_y$, $h_z$ | Components of vector $\vec{h}$. |
| $h$ | Altitude. |
| $I_{xx}$, $I_{yy}$, $I_{zz}$ | Moments of inertia about body x, y and z axes, respectively. |
| $I_{xz}$ | Product of inertia about body x and z axes. |
| $K_{I\alpha}$ | Gain for integral $\alpha$-feedback. |
| $K_{I\phi}$ | Gain for integral $\phi$-feedback. |
| $K_{P\alpha}$ | Gain for proportional $\alpha$-feedback. |
| $K_{P\beta}$ | Gain for proportional $\beta$-feedback. |
| $K_{P\phi}$ | Gain for proportional $\phi$-feedback. |
| $K_{R\alpha}$ | Gain for rate $\alpha$-feedback. |
| $K_{R\beta}$ | Gain for rate $\beta$-feedback. |
| $K_{R\phi}$ | Gain for rate $\phi$-feedback. |
| $K_1$ thru $K_{51}$ | Coefficients in linearized equations of motion (1 thru 19 longitudinal, 30 to 51 for lateral). |
| $L$ | Lift. |
| $L$, $M$, $N$ | Rolling, pitching and yawing moment, respectively. |
| $\ell_T$ | Distance between center of gravity and aerodynamic center of tail. |
| $M$ | Mach number. |
| $m$ | Total aircraft mass. |

| | |
|---|---|
| p, q, r | Rotational rates about x-body, y-body and z-body axes, respectively. |
| $\bar{q}$ | Dynamic pressure. |
| S | Wing area. |
| s | Complex Laplace transform variable. |
| T | Thrust. |
| u, v, w | Components of velocity along x-body, y-body and z-body axes, respectively. (In sections about linear analysis, u, v and w denote small changes of these variables from a steady-state reference value.) |
| $\vec{V}$ | Velocity vector of aircraft center of gravity. |
| X, Y, Z | Aerodynamic forces along the x-body, y-body and z-body axes, respectively. |
| $X_s$, $Y_s$, $Z_s$ | Aerodynamic forces along the x, y and z axes in the stability axes coordinate system. |
| $x_e$, $y_e$ | Aircraft x and y coordinates in earth fixed coordinate system. |
| $\alpha$ | Body angle of attack. |
| $\alpha_c$ | Commanded body angle of attack. |
| $\alpha_w$ | Wing angle of attack. |
| $\alpha_E$ | Angle of attack error. |
| $\beta$ | Sideslip angle. |
| $\delta_a$ | Aileron deflection. |
| $\delta_R$ | Rudder deflection. |
| $\delta_S$ | Stabilator deflection |
| $\delta_{SB}$ | Speed brake deflection. |
| $\delta_{SP}$ | Spoiler deflection. |

10

| | |
|---|---|
| $\rho$ | Air density. |
| $\psi$, $\theta$, $\phi$ | Euler yaw, pitch and roll angle, respectively. |
| $\phi_C$ | Commanded Euler roll angle |
| $\phi_E$ | Error between actual and commanded angle $\phi$. |
| $\tau_{lon}$ | Stabilator actuator lag. |
| $\tau_{lat}$ | Aileron actuator lag. |
| W | Aircraft weight. |

As in Reference 5, the equations in this report use dimensional control derivatives and static force and moment derivatives. For example,

$$C_{\ell_\beta} = \frac{\delta C_\ell}{\delta \beta} .$$

Nondimensional dynamic derivatives are used, such as

$$C_{\ell_p} = \frac{2V}{b} \frac{\delta C_\ell}{\delta_p} .$$

Sign convention for control surface deflections.-

Aileron:     $\delta_a$ positive for trailing edge of left aileron down.

Spoiler:     $\delta_{sp}$ positive for right spoiler up.

Rudder:     $\delta_R$ positive for trailing edge left.

Stabilator: $\delta_S$ positive for trailing edge down.

A note on terminology.- Throughtout this report, the expression "old equations of motion" refers to the equations of motion as they are used in the present (1975) version of the AML program on the DMS.  Characteristic for these equations of motion is the fact, that no moments of the aircraft are calculated.  The attitude of the aircraft is obtained in the following way:  Forces acting on the center of gravity are calculated under the assumption that the aircraft has a given angle of attack and zero sideslip angle.  By integration of these forces, the velocity vector at the end of an integration step is obtained.  The requirement of the net forces lying in a given plane, a given angle of attack and a vanishing sideslip angle then define uniquely the attitude that the aircraft should have at the end of this integration cycle.  Taking the difference between the actual and the desired aircraft attitude defines the desired attitude change.  Postulating that this attitude change can be accomplished by three rotations about the aircraft body axes during one integration stepsize defines the desired body rotational rates p, q and r.  These rotational rates are then processed through a digital filter which provides the actual body rotational rates.  These rates are integrated to provide the actual attitude at the end of an integration step.

The expression "new equations of motion" refers to the combination of the nonlinear, six degrees of freedom equations of motion as used in the aircraft simulation on the DMS and the differential equations of the flight control system as developed in this study.   The new equations of motion can be considered as describing the (nonlinear) transfer function between the input commands (angle of attack, bank angle and thrust level) and the aircraft motion.

12

# PART I:  GENERAL CONCEPTS

# CONTROL SYSTEM DESIGN PHILOSOPHY AND PROCESS

The task of executing AML commanded maneuvers is clearly a
problem of an automatic control system synthesis. Several syn-
thesis methods have evolved over the last years, one of the best
summaries of these methods can be found in a paper by Whitaker
and Potter (Ref. 6). Although this paper is almost ten years old,
it is the opinion of the authors of this report that most of the
facts and opinions expressed in this paper are still true today.

The synthesis methods may be classified into two broad cate-
gories. The older procedure, which may be considered to be an
outgrowth of classical feedback theory, is an iterative trial and
error process. It operates mainly in the frequency domain, ad-
justs gains, time constants, etc. until the location of the roots
of the transfer function of the closed-loop system indicate proper
frequencies and dampings. It is interesting to note that the MIL-
SPECS MIL-F-8785, "Military Specifications - Flying Qualities of
Piloted Airplanes" rely to a great extent on these same concepts
of natural frequency and damping. This trial and error procedure
may, today, seem somewhat unscientific, but when executed by an
experienced design engineer, it generally takes only a few iter-
ations to obtain desirable root locations.

The newer approach to control system design is based on what
is generally called modern control theory. Here, the design of a
control system is started by defining a performance index and the
control system is then designed such that the performance index
is minimized. For linear, second order systems, analytical methods
are known to minimize certain performance indices. But even for
this simple design task, the computational effort may become
uneconomically large. In addition, an aircraft, especial-
ly when flown in air-to-air combat, cannot be represented as a
linear second order system or as several, independent second order
systems. The nonlinear effects of the cross coupling terms in the
equations become very significant. In a later section of this

14

report, the problems arising from inertial cross coupling are explained in more detail.

The fact that the actual equations of motion of the aircraft are nonlinear, of course, would not exclude the use of the principle of minimizing a certain performance index; it would only exclude the possibility of finding an analytical solution to the problem.  The almost unsurmountable difficulty appears to lie in the definition of meaningful performance indices.  All the clasical methods of linearization are based upon perturbations about a nominal flight condition.  Quite often, this nominal flight condition is trimmed, level, unaccelerated flight.  Such a flight condition may prevail just prior to the beginning of an engagement, but thereafter, during the dogfight, it is unlikely that the straight and level flight is ever resumed.  Typical air combat maneuvers may find the aircraft initially in a steep dive, banked, say, 90 degrees to the left, and a relatively low angle of attack. The required maneuver may then be a 90-degree right roll combined with a pitch to an angle of attack which corresponds to a maximum lift coefficient.  It is conceivable that for this particular maneuver, one might find a meaningful and realistic performance index and then iterate on the design of a control system for that particular flight situation.  The next maneuver, which may be commanded, may be a change in bank angle without a change in angle of attack.  It is unlikely that the same performance criterion, as in the previous maneuver, would be desirable in this situation. This illustrates the difficulties associated with specifying performance criteria.  Since these should be made a function of flight condition, may be dynamic pressure, due to the nonlinearity of the equations of motion, they would also have to be a function of the command input, single axis commands versus multiple axes command and small changes versus large changes.

Adaptive Control Systems.- Adaptive Control Systems are characterized by incorporating a mechanism which automatically adjusts the control system parameters (usually gains) to adapt the control

system to either changes in the external environment or the flight conditon or to changes in the aircraft dynamics. Ever since the first adaptive control systems were used with the X-15 aircraft, they have been quite popular with experimental aircraft. The main advantage of an Adaptive Control System is that the aircraft dynamics (that is, its stability and control derivatives) do not have to be known before the control system is actually used in the aircraft. In the particular task here, however, the dynamics of the aircraft is exactly known, since it is given by the DMS equations of motion. Consequently, there is no need to have an Adaptive Control System.

Design Process.- The fastest way to obtain a working control system capable of executing command sequences generated by the AML program appeared to be to use classical, linear control systems resembling classical pitch and yaw dampers in stability augmentation systems. Figure 1 shows, in the form of a flow chart, the design process. The design process starts with selecting representative flight conditions. Three conditions were selected, all at 15,000 feet straight and level flight; a very low dynamic pressure (q=150 psi), a medium dynamic pressure (q=300 psi), and a high dynamic pressure ($\overline{q}$=840 psi). One might also consider additional flight conditions with low, medium and high dynamic pressure, one at close to sea level and another one at real high altitude, say, 35,000 feet. To familiarize one's self with the basic properties of the open loop airframe at these flight conditions, it is recommended that one obtain the transfer functions $\alpha(s)/\delta s(s)$ and $\phi(s)/\delta a(s)$ as well as the location of the roots of these transfer functions. This provides a first indication of what one might reasonably expect in terms of pitch and roll response.

Next, a basic longitudinal and lateral control system is selected and initial gain values are determined by engineering judgment, by comparison with similar control systems and by inspection of the open loop response. Ideally, then, root loci plots are developed by varying one gain at a time. If it is possible to

PROGRAM ATDYN

START

(A)

SELECT REPRESENTATIVE FLIGHT CONDITIONS

DETERMINE ALL COEFFICIENTS REQUIRED FOR LINEAR ANALYSIS

OBTAIN TRANSFER FUNCTIONS FOR OPEN LOOP LINEARIZED AIRCRAFT $\frac{\alpha(s)}{\delta_s(s)}$; $\frac{\phi(s)}{\delta_a(s)}$

SELECT FORM OF CONTROL SYSTEM BOTH FOR LONGITUDINAL AXIS AND LATERAL AXIS AND SELECT TENTATIVE GAINS

OBTAIN ROOT LOCATIONS FOR THE SELECTED CONTROL SYSTEM

ARE ROOTS SATISFACTORILY LOCATED

YES → (C)

NO

HAVE WE TRIED HARD ENOUGH WITH DIFFERENT VALUES OF GAINS AND TIME CONSTANTS?

NO

YES

MODIFY CONTROL SYSTEM PARAMETERS

ADD MORE FEEDBACK VARIABLES AND/OR SHAPING NETWORKS TO CONTROL SYSTEM

Figure 1.- Flow Diagram for Control System Design Process.

17

Figure 1 (Continued)

18

Figure 1 (Continued)

19

COMPARE PERFORMANCE BETWEEN OLD AND NEW

FLY AML-DRIVEN NONLINEAR EQUATION OF MOTION AGAINST A/C DRIVEN BY OLD EQUATIONS OF MOTION

G
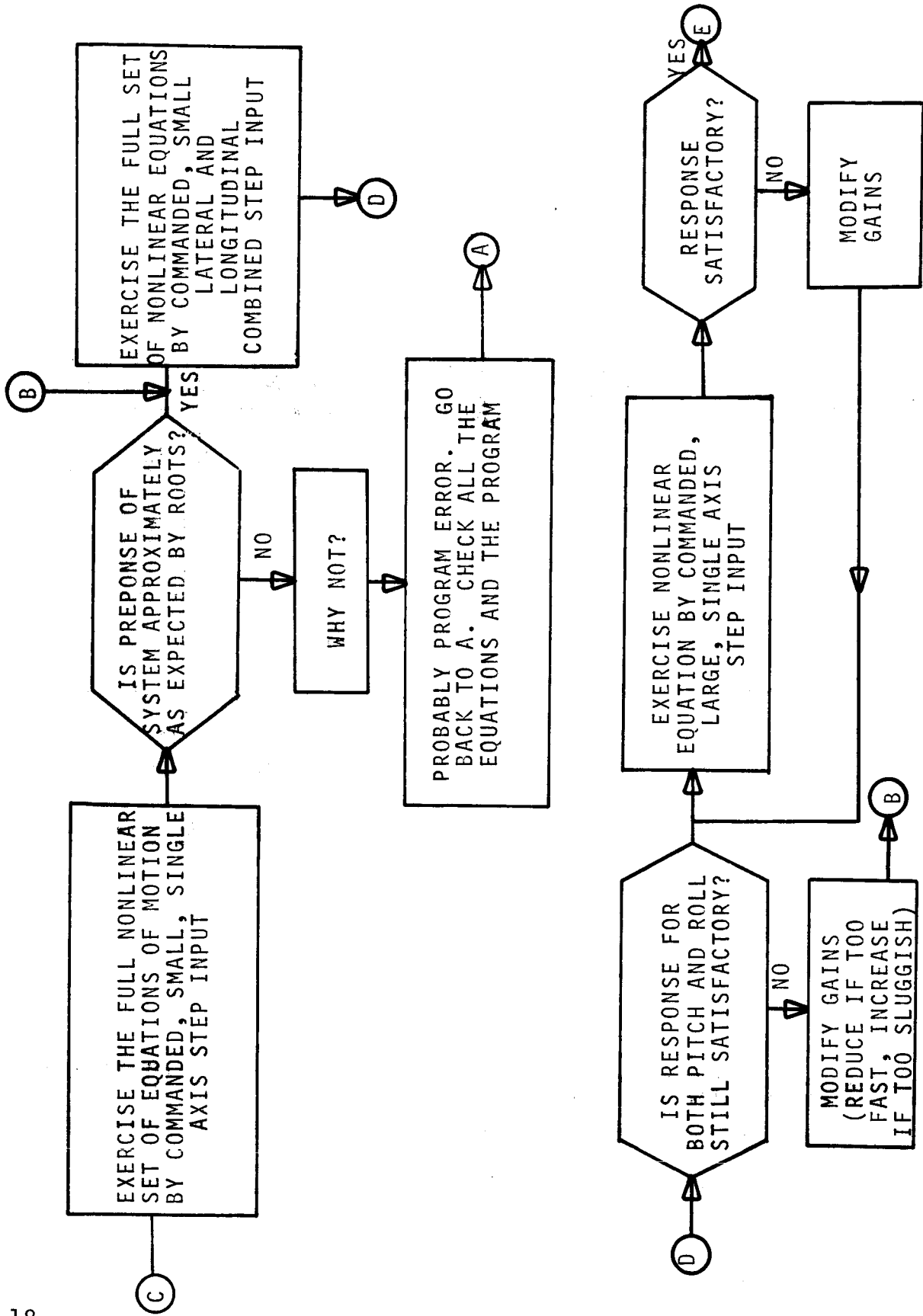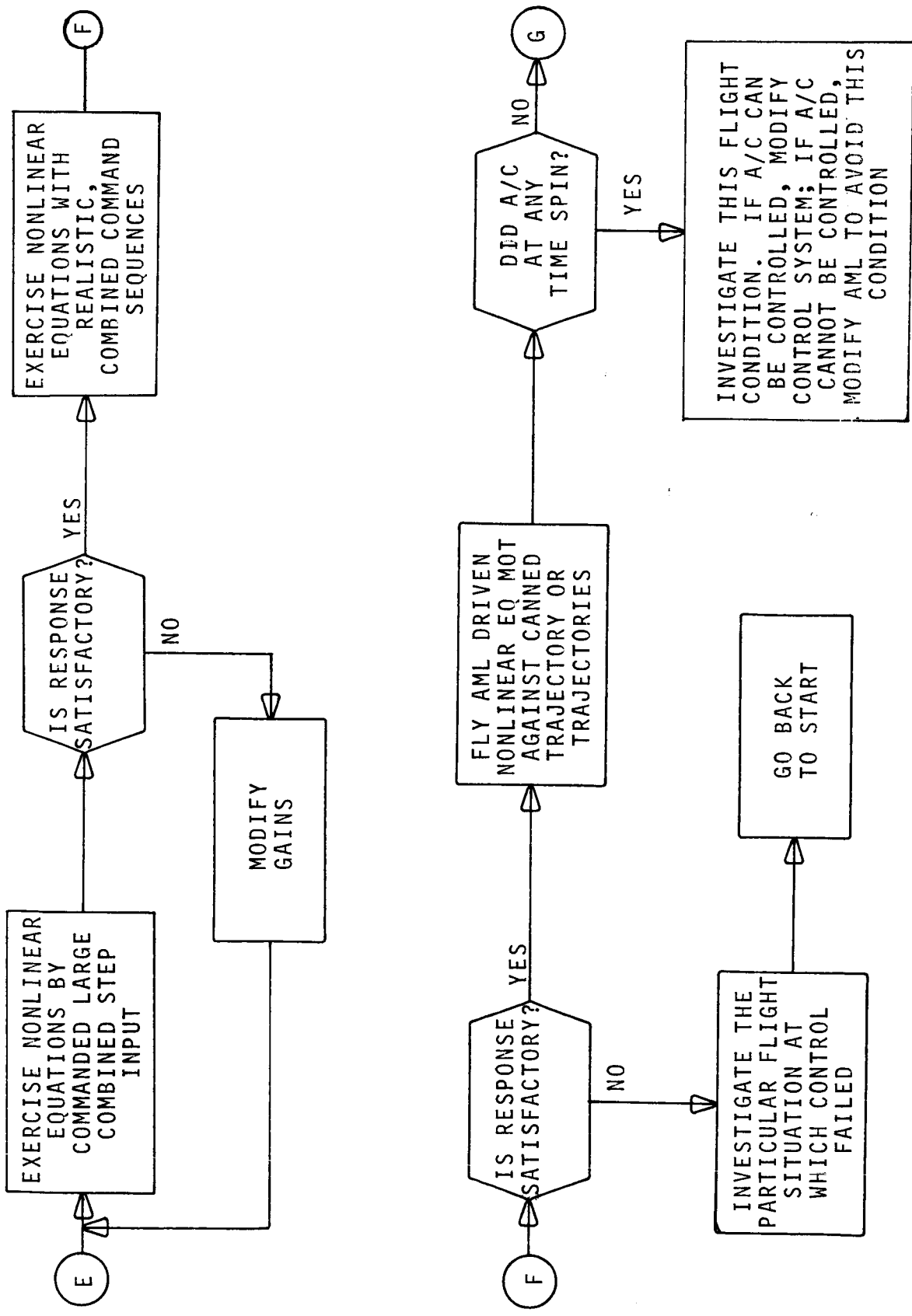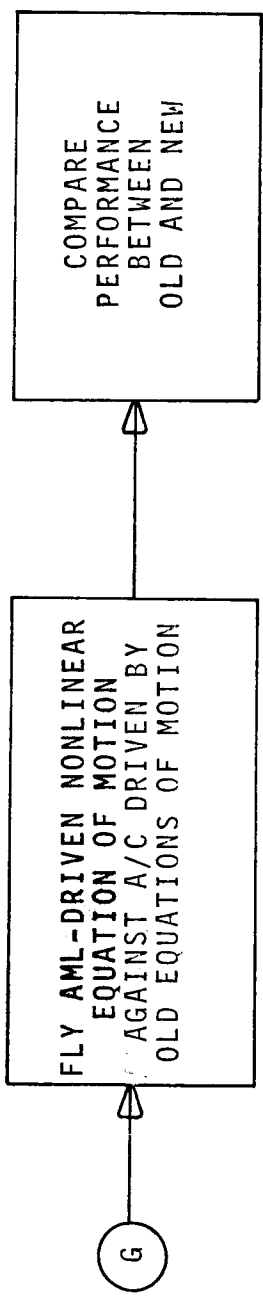
Figure 1   (Concluded)

find gains which will result in a satisfactory distribution of the roots, the design proceeds to the next step. If, however, the roots cannot be moved to desirable locations, the control system itself, rather than just its gains, need modification. Either more variables should be fed back or additional elements, like lead/lag networks should be added and new gains determined.

After a feedback system has been found whose roots appear to satisfy the desired response characteristic, a crucial step follows, and that is to determine how well the nonlinear equations of motion can be controlled with the control system obtained by the linear analysis. For small perturbations, the nonlinear equations of motion should behave very closely to the linear case unless certain types of nonlinearities, such as deadbands and hysteresis are present. If such nonlinearities exist, it is recommended to first remove those from the nonlinear program to obtain a meaningful comparison between linear and nonlinear analysis.

If, for small amplitude, single axis disturbances, the response of the nonlinear system is drastically different from the reponse of the linear system, the most probable cause is an error in one or both of the two programs. Careful comparison of the computer program against the original equations and comparison of all the coefficients between the two programs may reveal the error. Other potential problem areas exist in the mathematical formulation of the differential equations of the control system and in an inadequate method of integration or an unsuitable integration stepsize. On the other hand, since the programs for the linear analysis contain several critical portions, the following checks should be made: Are the matrices, from which the eigenvalues are determined, set up properly? Are the signs of all the feedback paths correct? Do the eigenvalues found by the computer program really satisfy the equation $[A] - \lambda[I] = 0$? Note that the comparison between linear and nonlinear equations offers a valuable tool for program checkout and debugging.

Once the response of the linear and nonlinear systems agree
well enough so that it seems likely that the two programs are error
free, one may proceed to the next step which basically investigates
how much are linear and nonlinear systems different for:

- small amplitude longitudinal and lateral
  command input combined.
- large amplitude single axis command input.
- large amplitude longitudinal and lateral
  command input combined.

Each one of these three tests may require slight modifications of
the gains or, in extreme cases, even the control system.

After a satisfactory response for single command input (lat-
eral and longitudinal simultaneously and individually) has been
obtained, the program is exercised by command input sequences. The
objective here is to investigate how the system behaves if it ob-
tains a new command input before the response to the previous command
input has settled to steady state. During this process, the air-
craft is also operated at flight conditions different from the ones
assumed at the beginning of the design process. Thus, unsatisfac-
tory response characteristics may be revealed at certain flight
conditions. The closed loop response at these critical flight
conditions may then be investigated by repeating the entire pro-
cess outlined so far, starting with determining all the coeffic-
ients of the aircraft by the program ATDYN.

If the control system has passed all the tests so far, it is
reasonable to assume that it has now the capability to "fly" the
aircraft with a command sequence actually generated by the AML. A
canned trajectory is read in for the attacker aircraft, the AML
program controls the target. If a reasonable attacker trajectory
is selected, the target aircraft has to fly under flight conditions
typical for air combat maneuvering. By this time, the main ob-
jective as far as the control system is concerned, should be to
prevent the aircraft from departing. Note that this is now the

22

combined responsibility of the AML program and the control system. The AML program should command maneuvers such that the aircraft will not enter flight conditions with extremely low dynamic pressure in which the aircraft cannot be controlled. The control system should at all times execute the AML commanded maneuvers, if physically possible, in such a way as to keep the aircraft under control. Specifically, simultaneous roll and pitch maneuvers must be executed so that good use is made of the specific aerodynamic properties of a particular aircraft.

The final step in the design process is to use the AML program with the new equations of motion against an AML program with the old equations of motion. The primary objective of this mode is to compare performance between the new and old equations of motion, while secondary objectives are to obtain estimates about computer time required for implementation of the new equations of motion in real time on the DMS and to obtain additional information concerning the performance of the AML program with the new equations of motion. Most important here is again the question of stability. Any time the AML-driven aircraft should become unstable, an investigation about its cause must be made. The crucial question obviously is: Should the AML program have avoided getting into an undesirable flight condition or was the control system not capabile of performing a legitimate AML maneuver in a stable manner?

Figure 2 illustrates the individual programs which support the linear analysis and gain selection.

During the work performed under this contract, the situation that the AML-driven aircraft, running either against canned trajectories or against another AML-driven aircraft, became unstable, did never occur. However, only a very limited number of such runs were made, and it is possible that in a more extensive checkout, occasional departures of the AML-driven aircraft might occur.

Figure 2.- USE OF LINEAR ANALYSIS AND CONTROL PROGRAMS
STEPS IN ANALYSIS AND GAIN SELECTION FOR
A SPECIFIC FLIGHT CONDITION

1.

ATDYN
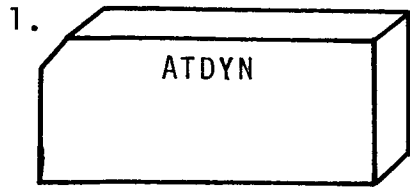
Linearized dynamics open-loop transfer functions printout.

K-array deck for use with other programs.

K-array

2.

LONLOP

6x6 matrix
$\alpha_c$ forcing function
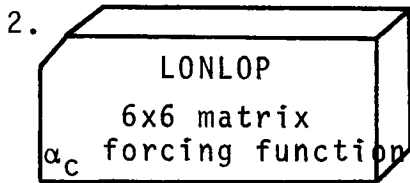
Closed-loop roots of longitudinal axis. Change $K_{P_\alpha}$, and $K_{R_\alpha}$ until roots are satisfactory.

3.

LABEZR

3x3 matrix
$\delta_r$ forcing function
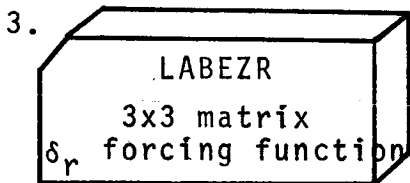
Find zeros and poles of open-loop transfer function $\dfrac{\beta(s)}{\delta_r(s)}$

4.

LATDYN

3x3 matrix
$\delta_a$ forcing function
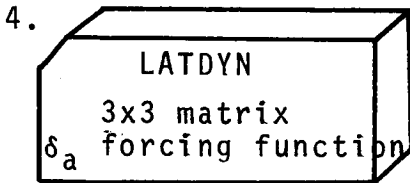
Close $\beta$-loop on rudder to obtain desired frequency and damping of Dutch roll. Vary $K_{P_\beta}$ and $K_{R_\beta}$ until satisfactory.

5.

LATLOP

7x7 matrix
$\phi_c$ forcing function
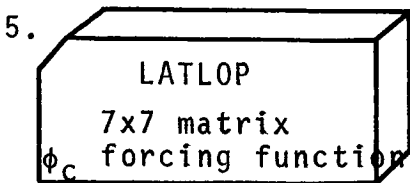
Find roots of closed-loop for lateral axis. Includes actuator lag and simulated rate limit.

6.

LALINI
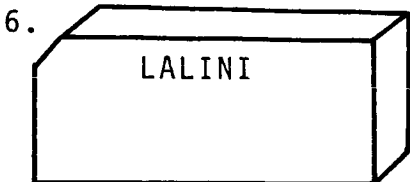
Find time response of lateral axis equations for desired initial conditions, e.g., all zero I.C.s except $\beta(0)=+5$ degrees.

24

# EQUATIONS OF MOTION

### The general, six degrees of freedom, equations
### of motion for a rigid body

The most general form of the rigid body equations, expressed in a space-fixed coordinate system, are the well known two vector equations

$$\vec{F} = \frac{d}{dt}(m\vec{v})$$

$$\vec{G} = \frac{d\vec{h}}{dt}$$

where  $\vec{F}$   resultant external force vector acting on center of mass

$\vec{G}$   resultant external moment vector acting about center of mass

$\vec{v}$   velocity vector of center of mass

$\vec{h}$   angular momentum vector

m   mass

Transforming these equations from a space-fixed reference frame to one rotating with the aircraft, the classical Euler equations of motion are obtained, in vector form:

$$\vec{F} = m\frac{\delta\vec{v}}{\delta t} + m\vec{\omega} \times \vec{v}$$

$$G = \frac{\delta\vec{h}}{\delta t} + \vec{\omega} \times \vec{h}$$

or, expressed in the six corresponding scalar equations;

$$F_x = m(\dot{u} + qw - rv)$$

$$F_y = m(\dot{v} + ru - pw)$$

$$F_z = m(\dot{w} + pv - qu)$$

$$L = \dot{h}_x + qh_z - rh_y$$

$$M = \dot{h}_y + rh_x - ph_z$$

$$N = \dot{h}_z + ph_y - qh_x$$

Expressing the angular momentum components and their derivatives in terms of moments of inertia, we obtain the standard body-axis rigid-body equations of motion as used on the DMS (and on most other flight simulators):

$$\dot{u} = \frac{F_x}{m} + rv - qw$$

$$\dot{v} = \frac{F_y}{m} + pw - ru$$

$$\dot{w} = \frac{F_z}{m} + qu - pv$$

$$\dot{p} = \frac{L}{I_{xx}} + (\dot{r} + pq)\frac{I_{xz}}{I_{xx}} + qr\frac{(I_{yy} - I_{zz})}{I_{xx}}$$

$$\dot{q} = \frac{M}{I_{yy}} + (r^2 - p^2)\frac{I_{xz}}{I_{yy}} - pr\frac{(I_{xx} - I_{zz})}{I_{yy}}$$

$$\dot{r} = \frac{N}{I_{zz}} + (\dot{p} - qr)\frac{I_{xz}}{I_{zz}} - pq\frac{(I_{yy} - I_{xx})}{I_{zz}}$$

The above form of the equations of motion is most useful for analysis; for programming on a digital computer, they have to be rearranged because the equation for $\dot{p}$ contains $\dot{r}$ and the equation for $\dot{r}$ contains $\dot{p}$. This computational difficulty can be overcome by substituting the $\dot{r}$ equation into the $\dot{p}$ equation and then solving for $\dot{p}$, which yields:

$$\dot{p} = \left[ L - qr(I_{zz}-I_{yy}) + I_{xz}\, pq + \left\{ N - pq(I_{yy}-I_{xx}) - I_{xz}\, qr \right\} \right.$$

$$\left. \cdot \frac{I_{xz}}{I_{zz}} \right] \Big/ \left( I_{xx} - \frac{I_{xz}^{2}}{I_{zz}} \right)$$

The equation for $\dot{r}$ needs no reformulation because, when $\dot{r}$ is calculated, $\dot{p}$ is already known.

### Linearized Equations for Longitudinal Direction

The linearized equations of aircraft motion used in this study are derived in the book by Blakelock, Ref. 7. The equations describing longitudinal (pitch) motions given on page 21 of Ref. 7 may be written in the form (for definition of axes, see Figure 3)

$$K_1{}'\ddot{U} + K_2{}'\dot{U} + K_3{}'\dot{\alpha} + K_4{}'\alpha\; K_5\dot{\theta} + K_6\theta = K_7\delta_s \quad \text{(x-axis force)}$$

$$K_8{}'\dot{U} + K_9{}'\dot{\alpha} + K_{10}{}'\alpha + K_{11}\dot{\theta} + K_{12}\theta = K_{13}\delta_s \quad \text{(z-axis force)}$$

$$K_{14}{}'\dot{U} + K_{15}{}'\ddot{\alpha} + K_{16}{}'\alpha + K_{17}\ddot{\theta} + K_{18}\dot{\theta} = K_{19}\delta_s \quad \text{(pitch moment)}$$

with the coefficients $K_1$ through $K_{19}$ being defined as:

$$K_1 = \frac{mV_T}{s\bar{q}} \qquad\qquad K_3 = -\frac{\bar{c}}{2V_T}\, C_{x_{\dot{\alpha}}} \quad \text{(neglected)}$$

$$K_2 + -C_{x_u} = -\frac{V_T}{s\bar{q}}\frac{\alpha F_x}{\alpha u} \qquad K_4 = C_{x_\alpha}$$
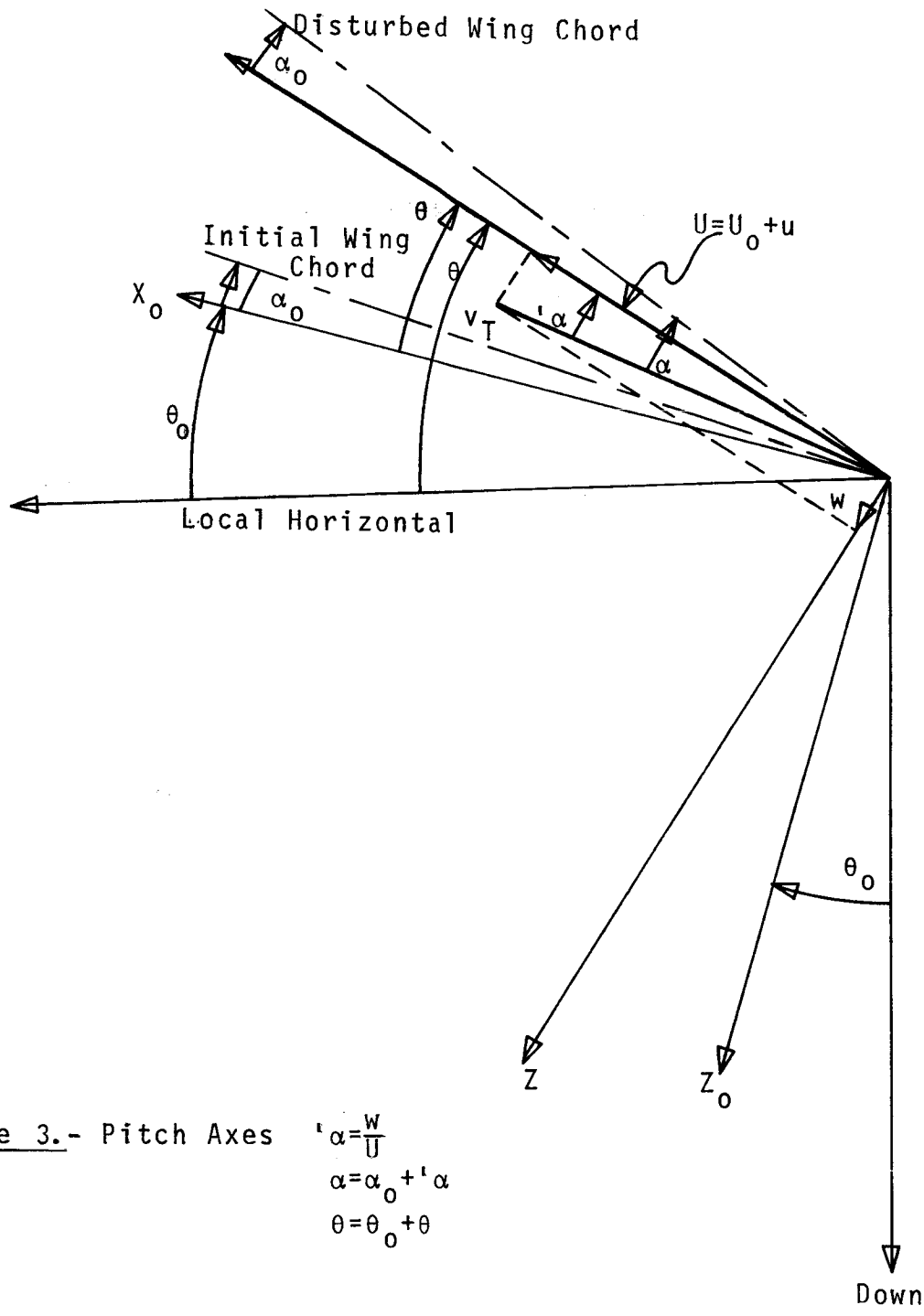
Figure 3.- Pitch Axes  $'\alpha = \dfrac{w}{U}$

$\alpha = \alpha_0 + '\alpha$

$\theta = \theta_0 + \theta$

$$K_5 = -\frac{\overline{c}}{2V_T} C_{x_{\overline{q}}} \quad \text{(neglected)}$$

$$K_{13} = \frac{\overline{c}}{\ell_T} C_{m_{\delta_s}} = C_{z_{\delta_s}}$$

$$K_6 = \frac{mg}{S\overline{q}} \cos\theta = -C_w \cos\theta$$

$$K_{14} = -C_{m_u}$$

$$K_7 = C_{x_{\delta_s}} \quad \text{(neglected)}$$

$$K_{15} = -\frac{\overline{c}}{2V_T} C_{m_{\dot{\alpha}}}$$

$$K_8 = -C_{z_u} = \frac{C_z(u+\Delta u) - C_z(u)}{\Delta u}$$

$$K_{16} = -C_{m_{\alpha}}$$

$$K_9 = K_1 - \frac{\overline{c}}{V_T} C_{z_{\dot{\alpha}}}, \quad C_{z_{\dot{\alpha}}} \cong \frac{\overline{c}}{\ell T} C_{m_{\dot{\alpha}}}$$

$$K_{17} = \frac{I_{yy}}{S\overline{q}c}$$

$$K_{10} = C_{z_{\alpha}} = C_{L_{\alpha}} - C_D$$

$$K_{18} = -\frac{\overline{c}}{2V_T} C_{m_q}$$

$$K_{11} = -K_1 - \frac{\overline{c}}{2V_T} C_{z_q}$$

$$K_{19} = C_{m_{\dot{\delta}_s}}$$

$$K_{12} = \frac{mg}{S\overline{q}} \sin\theta$$

where

$$'u = \frac{u}{U}$$

$u$ = x-axis velocity change

$U_0$ = x-axis velocity at steady flight condition

$$'\dot{u} = \frac{d}{dt}('u)$$

$'\alpha$ = angle of attack perturbation

$\theta$ = pitch angle perturbation

$\delta_s$ = stabilator deflection

$m$ = total aircraft mass

$V_T$ = total velocity

$S$ = wing planform reference area

$$\bar{q} = \text{dynamic pressure} = \frac{1}{2}\rho V_T^2$$

x = aircraft forward body axis

y = aircraft right-wing body axis

z = aircraft downward body axis

$\bar{c}$ = mean aerodynamic chord

v = Y-axis velocity change

w = x-axis velocity change

$\ell_T$ = length between c.g. and aerodynamic center of tail

The coefficients in the K array are defined in terms of the nondimensional stability derivatives such as, for example, $C_{x_\alpha} = \frac{1}{S\bar{q}}\frac{\partial F_x}{\partial \alpha}$. Since an adequate discussion of stability derivatives for aircraft is beyond the scope of this report it is assumed that readers lacking familiarity with them can refer to Ref. 7.

The term $K_2$ involves the partial derivative $\frac{\partial F_x}{\partial u}$ which depends upon both drag and thrust variations with forward speed. Since air combat usually requires either full afterburner or idle thrust, the corresponding variation in thrust with forward speed was evaluated and included in $\frac{\partial F_x}{\partial u}$ for either afterburner or idle thrust in AML75. In CMDSEQ the value of $\frac{\partial F_x}{\partial u}$ is calculated based upon afterburner thrust.

### Linearized Equations for Lateral Direction

The linearized equations for the lateral-directional axes (roll-yaw) are given on page 116 of Blakelock's book as

$$K_{30}\ddot{\phi} + K_{31}\dot{\phi} + K_{32}\ddot{\psi} + K_{33}\dot{\psi} + K_{34}\beta = K_{35}\delta_a + K_{36}\delta_R \quad (\text{roll})$$

$$K_{37}\ddot{\phi} + K_{38}\dot{\phi} + K_{39}\ddot{\psi} + K_{40}\dot{\psi} + K_{41}\beta = K_{42}\delta_a + K_{43}\delta_R \quad (\text{yaw})$$

$$K_{44}\dot{\phi} + K_{45}\phi + K_{46}\dot{\psi} + K_{47}\psi + K_{48}\dot{\beta} + K_{49}\beta = K_{50}\delta_a + K_{51}\delta_R \quad (\text{y-force})$$

where

$$K_{30} = \frac{I_{xx}}{S\bar{q}b}$$

$$K_{31} = \frac{b}{2V_T} C_{\ell_p}$$

$$K_{32} = -\frac{I_{xz}}{S\bar{q}b}$$

$$K_{33} = -\frac{b}{2V_T} C_{\ell_r}$$

$$K_{34} = C_{\ell_\beta}$$

$$K_{35} = C_{\ell_{\delta_a}}$$

$$K_{36} = C_{\ell_{\delta_R}}$$

$$K_{37} = K_{32}$$

$$K_{38} = -\frac{b}{2V_T} C_{n_p}$$

$$K_{39} = \frac{I_{zz}}{S\bar{q}b}$$

$$K_{40} = \frac{b}{2V_T} C_{n_r}$$

$$K_{41} = -C_{n_\beta}$$

$$K_{42} = C_{n_{\delta_a}}$$

$$K_{43} = C_{n_{\delta_R}}$$

$$K_{44} = \frac{b}{2V_T} C_{y_p}$$

$$K_{45} = C_{y_\phi} = -\frac{mg}{S\bar{q}} \cos\phi \cos\phi$$
$$= -K_6 \cos\phi$$

$$K_{46} = K_1 - \frac{b}{2V_T} C_{y_r}$$

$$K_{47} = -C_{y_\psi} = -K_{12}$$

$$K_{48} = K_1$$

$$K_{49} = C_{y_\beta}$$

$$K_{50} = C_{y_{\delta_a}}$$

$$K_{51} = C_{y_{\delta_r}}$$

$I_{xx}$ = x-axis moment of inertia

b = wing span

$\delta_a$ = aileron deflection

$\delta_R$ = rudder deflection

$I_{zz}$ = z-axis moment of inertia

g = standard gravitational ac-
celeration

$I_{yz}$ = yz product of inertia =
$\int yz\,dm$

31

# THE ROLL MANEUVER

## Maneuver Limiting Factors

To roll the aircraft fast and accurately is one of the key tasks in air combat maneuvering. The roll maneuver is conventionally defined as a rotation of the aircraft about its longitudinal axis with an angular velocity p. As W. H. Phillips has shown (ref. 11), even small disturbances about a condition of rapid, steady rolling may cause longitudinal and lateral instabilities. If there is a significant angle or angular rate between the velocity vector and the aircraft longitudinal axis, the situation gets even more complicated. The following two situations are quite important for the design of a control system and are, therefore, discussed in some detail in the following: 1) Initiation of a roll from a flight condition with a large angle of attack, and 2) Simultaneous rotation about the x-body and the y-body axis (simultaneous roll and pitch).

## Roll Maneuver Under High Angle of Attack

If an aircraft performed a pure rotation about its longitudinal axis only, after a rotation of ninety degrees the initial angle of attack would be pure sideslip and after 180 degrees, the initially positive angle of attack would be a negative angle of attack with the same magnitude as the initial angle of attack. To verify the above statement, consider an aircraft flying initially with wings level, pointing north, and an angle of attack of $\bar{\alpha}$. Its direction cosine matrix D then is

$$[D] = \begin{bmatrix} \cos\bar{\alpha} & 0 & -\sin\bar{\alpha} \\ 0 & 1 & 0 \\ \sin\bar{\alpha} & 0 & \cos\bar{\alpha} \end{bmatrix}$$

The velocity vector in inertial coordinates is:

32

$$\vec{V} = \begin{pmatrix} \dot{x}_e \\ 0 \\ 0 \end{pmatrix}$$

Since

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = [D] \cdot \vec{V} = \begin{pmatrix} \dot{x}_e \cos\bar{\alpha} \\ 0 \\ \dot{x}_e \sin\bar{\alpha} \end{pmatrix}$$

and by definition

$$\alpha = \text{atan}\left(\frac{w}{u}\right)$$

$$\beta = \text{asin}\left(\frac{v}{V_{Total}}\right)$$

it follows that

$$\alpha = \bar{\alpha}$$
$$\beta = 0$$

Let now the aircraft perform a rotation of 90 degrees about its x-body axis, then

$$[D] = \begin{bmatrix} \cos\bar{\alpha} & 0 & -\sin\bar{\alpha} \\ \sin\bar{\alpha} & 0 & \cos\bar{\alpha} \\ 0 & -1 & 0 \end{bmatrix}$$

and consequently

$$\alpha = 0$$
$$\beta = \bar{\alpha}$$

The importance of this phenomenon, as far as control system design is concerned, lies in the fact that rolling under angle of attack causes sideslip even if no side forces were generated, simply by the geometrical relationships. Rolling under angles of attack therefore requires simultaneous yawing. To keep a roll

33

maneuver, which is initiated with no angle of attack, coordinated, a yawing moment must be generated to compensate for the yaw moment due to $C_{np}$ (yaw due to roll) and due to $C_{n\ A}$ and $C_{n\ sP}$ (adverse yaw due to aileron and yaw due to spoilers). Small rudder deflections will be sufficient to keep the net yaw moment close to zero.

For large angles of attack, the situation is quite different. Not only is the adverse yaw stronger (requiring more rudder deflection), but as it was shown above, additional yawing moment must be generated to yaw the aircraft in such a fashion that the angle of attack remains in the aircraft symmetry plane and does not degenerate into sideslip.

In Section II of this report these relationships are illustrated by some numerical examples for the F-4.

### Simultaneous Roll and Pitch Maneuver, Cross Coupling

The above discussion assumed that during the roll maneuver, no change of the angle of attack was desired, that is, that the angular rate about the body y-axis remained essentially zero. Often, in air-to-air combat, the optimum maneuver consists of changing both the bank angle and the angle of attack. During such a maneuver, both roll rate and pitch rate are nonzero and, in order to keep the maneuver coordinated, the yaw rate will also be nonzero. Inspection of the equations of motion shows that under this condition, the terms $qr\ \dfrac{(I_{yy} - I_{zz})}{I_{xx}}$ (etc) become effective. This effect is generally called inertial cross-coupling. Special attention has been paid in the past to the $\dot{q}$ equation. The inertial cross coupling term there is

$$-pr\ \frac{(I_{xx} - I_{zz})}{I_{yy}}$$

Since in most modern fighters, $I_{xx}$ is considerably smaller than $I_{zz}$, high roll rates couple strongly into the longitudinal motion

34

and some aircraft become actually unstable at high roll rates.

But even the cross coupling term in the $\dot{p}$ equation, that is

$$qr \frac{(I_{yy} - I_{zz})}{I_{xx}}$$

can become significant in the design of a control system for air combat maneuvering. Even when $I_{yy}$ and $I_{zz}$ are of the same order, since they both are large, the difference between the two may be quite significant, and since it is divided by the small moment of inertia about the longitudinal axis, its effect is amplified. Once the existence of a significant cross-coupling term in the foll acceleration equation for a given aircraft has been established, the control system designer has to take some corrective action.

For instance, for an F-4, where $I_{zz}$ is greater than $I_{yy}$, the cross coupling term for the roll acceleration equation is of the form

$$-K \, qr$$

where -K represents the negative quantity

$$\frac{I_{yy} - I_{zz}}{I_{xx}}$$

The sign of the cross-coupling term obviously changes with the sign of q. If an increase in angle of attack is desired, q will be positive, for a decrease of the angle of attack, q will be negative.

It seems reasonable to assume that, for a desired roll moment, depending on the sign of q, the cross-coupling term might add to or oppose the desired roll moment.

One might therefore consider to incorporate some logic into the lateral and longitudinal control systems which would cause the aircraft to roll and pitch simultaneously if the cross-coupling term assists the roll moment but perform roll and pitch sequentially if the cross-coupling term is of opposite sign to the desired roll moment.

# PART II: SPECIFIC CHARACTERISTICS

## FOR THE F-4

# OPEN LOOP CHARACTERISTICS OF THE F-4

## General Characteristics

To effectively synthesize an attitude control system, it is necessary to know and understand the dynamic and aerodynamic properties of the basic aircraft. Of prime importance are those affecting fast pitch and roll maneuvering over the entire flight envelope encountered in air-to-air combat.

The original plan was to present time histories for control surface step input, first stabilator, aileron and rudder steps alone and then combined step input. While the open loop poles of the linearized transfer functions indicate what type of response may be expected, it would be interesting to compare the response of the full set of nonlinear equations of motion with the results expected from the poles. Several such comparisons were made, but, as pointed out in the preface, the limited resources did not permit adequate documentation of these results. While they were important and useful to arrive at the final control system, once the control system is designed, it is no longer necessary to have access to these intermediate results.

As just an example, Figures 4 and 5 show typical open loop responses for an aircraft with an initial sideslip angle and all control surfaces in their neutral trim position, both responses at 15,000 feet altitude, the first with M=0.32 and the second with M=0.8. Additional open loop characteristics can be obtained from Reference 5.

## Adverse Yaw

At the beginning of this study, it was believed that due to the extremely high adverse yaw of the F-4 roll commands at high angles of attack could not be initiated by aileron control. However, inspection of the pertinent control derivatives at Mach numbers below 1.1 shows that the following is true:
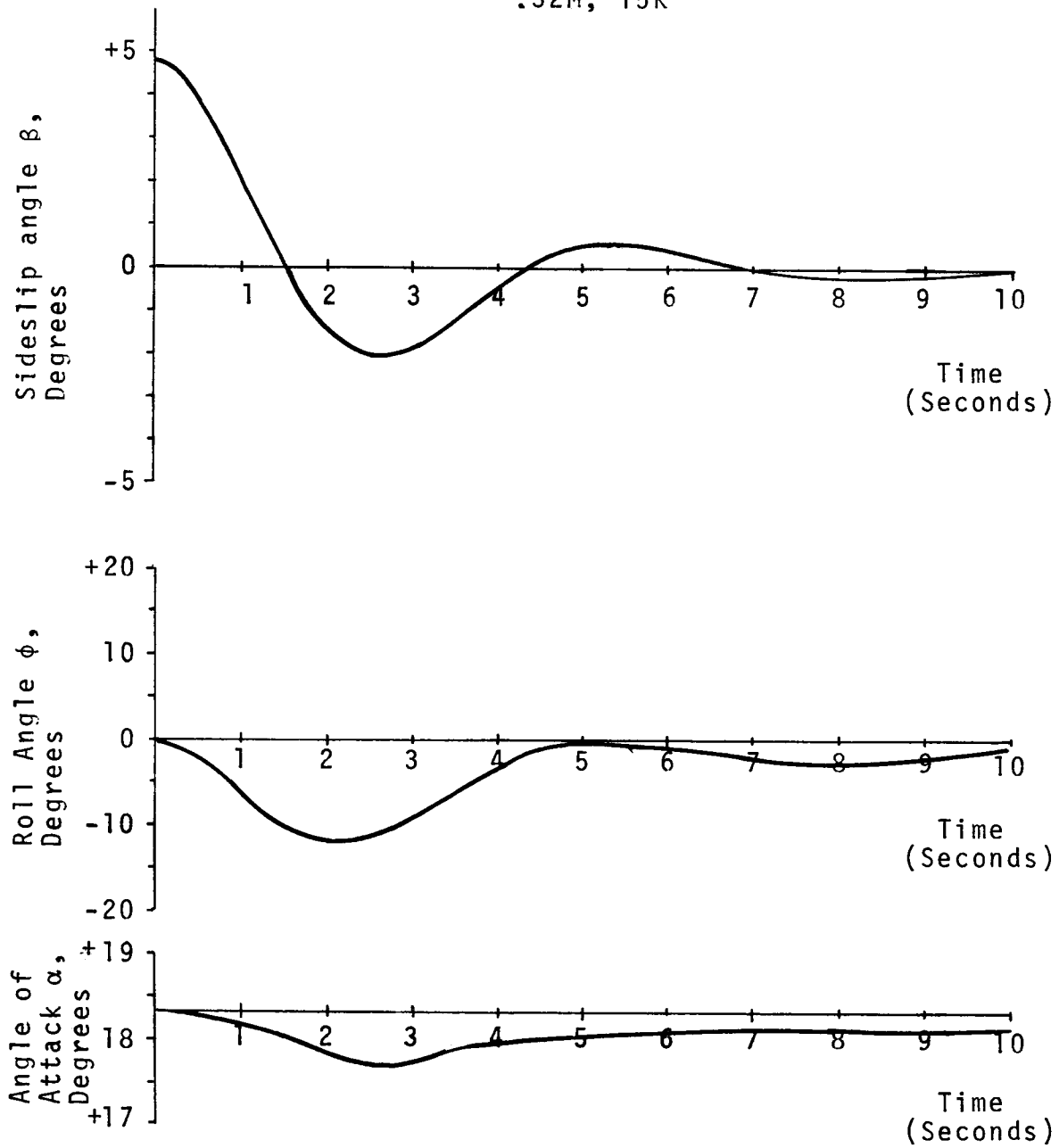
Figure 4.- Open Loop Response at M=.32 at 15,000 feet.
Sideslip angle = + 4.75 degrees at t=0.

.8M, 15K
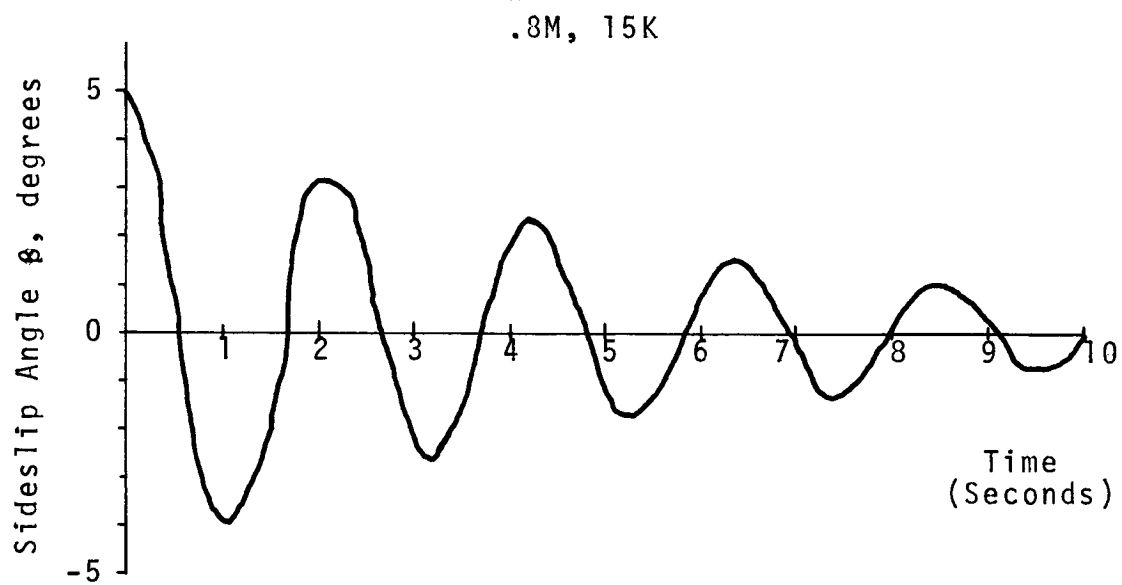


Figure 5.- Open Loop Response at M=0.8 at 15,000 Feet
Sideslip Angle = 5° at t=0.

(a) $C_{\ell_{\delta a}}$ is positive for all flight regimes.

(b) $C_{\ell_{\delta a}}$ is almost independent of the angle of attack.

(c) $C_{\ell_{\delta sp}}$ is positive for all flight regimes.

(d) $C_{\ell_{\delta sp}}$ decreases strongly with increasing angle of attack.

As a consequence, even under high angles of attack, aileron (and the associated spoiler) deflection will create a rolling moment in the desired direction. Note that the rudder deflection occurs automatically due to the β-feedback loop into the rudder. Fortunately, the control loops as designed are capable of properly compensating the adverse yaw effect.

# SOME PERFORMANCE SPECIFICATIONS

## Problems Associated with Performance Specifications

Specifying performance characteristics for an aircraft/control system configuration may be compared to specifying handling qualities of an aircraft. The well known specification, MIL-F-8785 "Military Specification - Flying Qualities of Piloted Airplanes" (see, for instance, Ref. 8), illustrates a possible approach to a set of specifications.

The main difficulty in defining specifications for aircraft performance during air combat maneuvering is the fact that the aircraft does not behave like a second order linear system and that, therefore, response characteristics are dependent on the amplitude of the maneuver under consideration. Not only are, of course, the basic equations of motion nonlinear, but severe nonlinearities exist in the limits of control surface deflections and possibly in the rates of these deflections.

While, for instance, in a linear system, the response for a commanded one-degree change of angle of attack would be the identical time function as for a two-degree change in angle of attack, with simply a different scale on the ordinate axis, this is not the case for a nonlinear system.

A first approach would, therefore, be to set up performance specifications for realtively small commanded changes. As long as certain types of nonlinearities, such as dead zones and hysterisis, are absent, these "small command" specifications certainly give some indication about the aircraft performance.

In air-to-air combat, however, the commanded changes are not small, but they may be extremely large, like from trim angle of attack to maximum angle of attack. It might, therefore, be more meaningful to specify certain constraints which should be satisfied in such extreme maneuvers. Even if such specifications were set up, say maximum rise time, etc., to demonstrate that these

specifications are satisfied would require substantial effort (max. angle of attack for each flight condition has to be established first).

For the lateral direction, "small commands" would be of the order of a few degrees, while again the maneuvers commanded in air-to-air combat often much larger. Time to roll from level flight to say 90 degrees might be a reasonable specification here.

Note that in all the above discussion, single axis command inputs were assumed. Clearly, the performance criteria set up in this single axis mode have to be relaxed if simultaneous command input are considered. The amount of individual specifications required becomes now quite large, because already in the single axis mode, the performance criteria are at least a function of dynamic pressure and of angle of attack.

It is suggested that the entire subject of performance specification be studied in a separate effort, not necessarily associated with the AML program. Special emphasis should be given to performance specifications at high angles of attack, where roll reversal, reduced roll control power and, particularly on F-4 aircraft, spin susceptibility may limit the maneuvering capabilities.

For an attitude control system responding to commands from the AML program, the most important requirement is fast response to achieve the commanded maneuvers. The system must also be stable for continuous flow of input in all combinations of input magnitude and frequency.

It is quite difficult to define performance specifications for an aircraft control system over the entire flight envelope of air combat. The aircraft dynamics are complex and the responses are nonlinear. Even at the same speed and altitude, the dynamics and control characteristics still depend upon angle of attack, pitch angle and roll angle. The time required for a roll maneuver is a nonlinear function of the magnitude and sign of the desired roll maneuver.

At very low speeds,which can occur as the result of air com-
bat maneuvering, roll control of the F-4 becomes very difficult.
The adverse yaw of the ailerons at low speed and high angles of
attack is so powerful that even hard rudder is insufficient to
maintain control during attitude maneuvers.

To completely characterize and specify the controlled respon-
ses of the aircraft over this entire flight envelope in a restric-
tive but achievable manner is a task worthy of the entire resources
of this study.

In order to reduce the specifications to manageable form the
following restrictions were imposed:

(1)  **Speed range** - from Mach 0.3 to Mach 1.1.

(2)  Specifications are given only for the aircraft in
     trimmed, level flight.

(3)  Specifications are given only for a limited set
     of flight conditions.

Comments on restriction 1.- While the control systems devel-
oped will successfully fly an F-4 slower than a velocity corres-
ponding to $\bar{q} = 150$ lb/ft$^2$ a control system optimized for fast,
effective attitude control at very low $\bar{q}$ would require a different
approach to aileron and rudder control.  This would be an inter-
esting study but is beyond the scope of the present resources.

Comments on restrictions 2 and 3.- The aircraft dynamics vary
significantly with angle of attack and even change somewhat with
pitch and roll angle.  The computer time available under this con-
tract was not sufficient to allow simulation of the controlled
responses for a large number of flight conditions after the best
performing control system and gains had been found.  This simula-
tion would be necessary to evaluate the performance achieved for
other flight conditions.

The flight conditions chosen are representative of the entire
flight envelope.  Rise time and overshoot for step input commands

44

for other flight conditions nearby is expected to be similar.

## Longitudinal Rise Time for Step Input

The most important specification for an air combat attitude control system is rise time for a commanded step input. Figure 6 shows the specification and the achieved values of rise time as a function of dynamic pressure at an altitude of 15,000 feet. Also shown for comparison is the computed period of the short period mode of the basic aircraft. Sequences of pitch command step input were given which included a range of sizes of steps from a few degrees up to the maximum change from the trim angle of attack possible without overstressing the aircraft. The points enclosed by a square represent the average rise time while the highest and lowest recorded rise times are shown using horizontal bars above and below the squares.

The observed rise times were usually less than the time of one period of the short period mode. The fastest rise times are a-chieved near Mach=.8. The control system should never exceed the specification line which is formed of straight line segments be-tween the points;

$$\overline{q} = 150 \text{ lb/ft}^2 \qquad t_r = 3.0 \text{ sec}$$

$$\overline{q} = 850 \text{ lb/ft}^2 \qquad t_r = 1.6 \text{ sec}$$

$$\overline{q} = 1500 \text{ lb/ft}^2 \qquad t_r = 1.4 \text{ sec}$$

Figure 7 shows the level flight trim angle of attack and the maximum allowed angle of attack for flight of the F-4 at 15,000 feet. This information is necessary to find the magnitude of al-lowed change in angle of attack.

## Lateral Rise Time for Step Input

The rise time in roll for flight at 15,000 feet is shown as a function of dynamic pressure in Figure 8. Some points obtained in simulation runs with the bang-bang roll control system are also

45

Figure 6.   Allowable Rise Time for
Angle of Attack Response

Legend

1.  Based on trimmed level flight as
    initial conditions

2.  Step input commanded in alpha

3.  H = 15,000 feet

4.  ⊢─□─⊣   = slowest observed rise time
              = average observed rise time
              = fastest observed rise time

spec line

one period of short period mode

M=1.0

m=.8

M=.6

$\overline{q}$, lb/ft$^2$

Rise Time $t_r$, sec

4.0   3.0   2.0   1.0   0

0  100  200  300  400  500  600  700  800  900  1000  1100  1200  1300  1400  1500
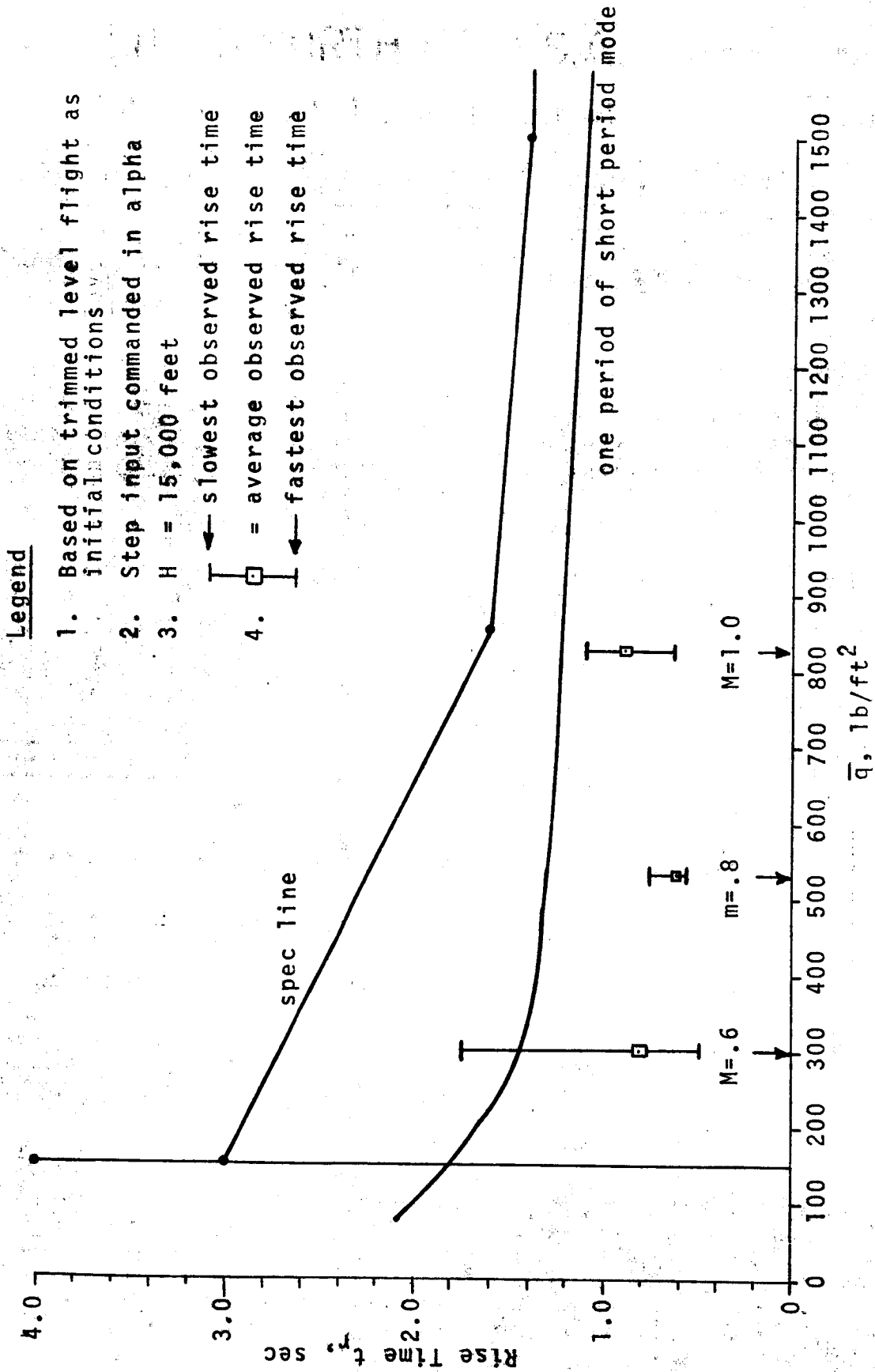
Figure 7.- Wing Trim Angle of Attack for Level Flight F-4 Model at H=15,000 ft.

| M | V | $\bar{q}$ | $C_L$ | $\alpha_W$ |
|---|---|---|---|---|
| .317 | 335 | 83.9 | .919 | 18.0 |
| .6 | 635 | 301 | .256 | 4.36 |
| .8 | 846 | 535 | .144 | 2.25 |
| 1.0 | 1057 | 836 | .0922 | 1.32 |

NOTE:

$\alpha_W = +1$

$\alpha_W$ = wing angle of attack

$\alpha$ = body x-axis angle of attack

Max allowable $\alpha_W$ (corresponding to $C_L$ max)

flat after reaching $\alpha_W = 30$

| $C_L$ | .77 | .26 | .15 | .11 | .091 | .077 |

M=.32   M=.6   M=.8   M=1

Dynamic Pressure $\bar{q}$, lb/ft$^2$

Speed V, ft/sec

Wing Angle of Attack, $\alpha_W$ degrees

47
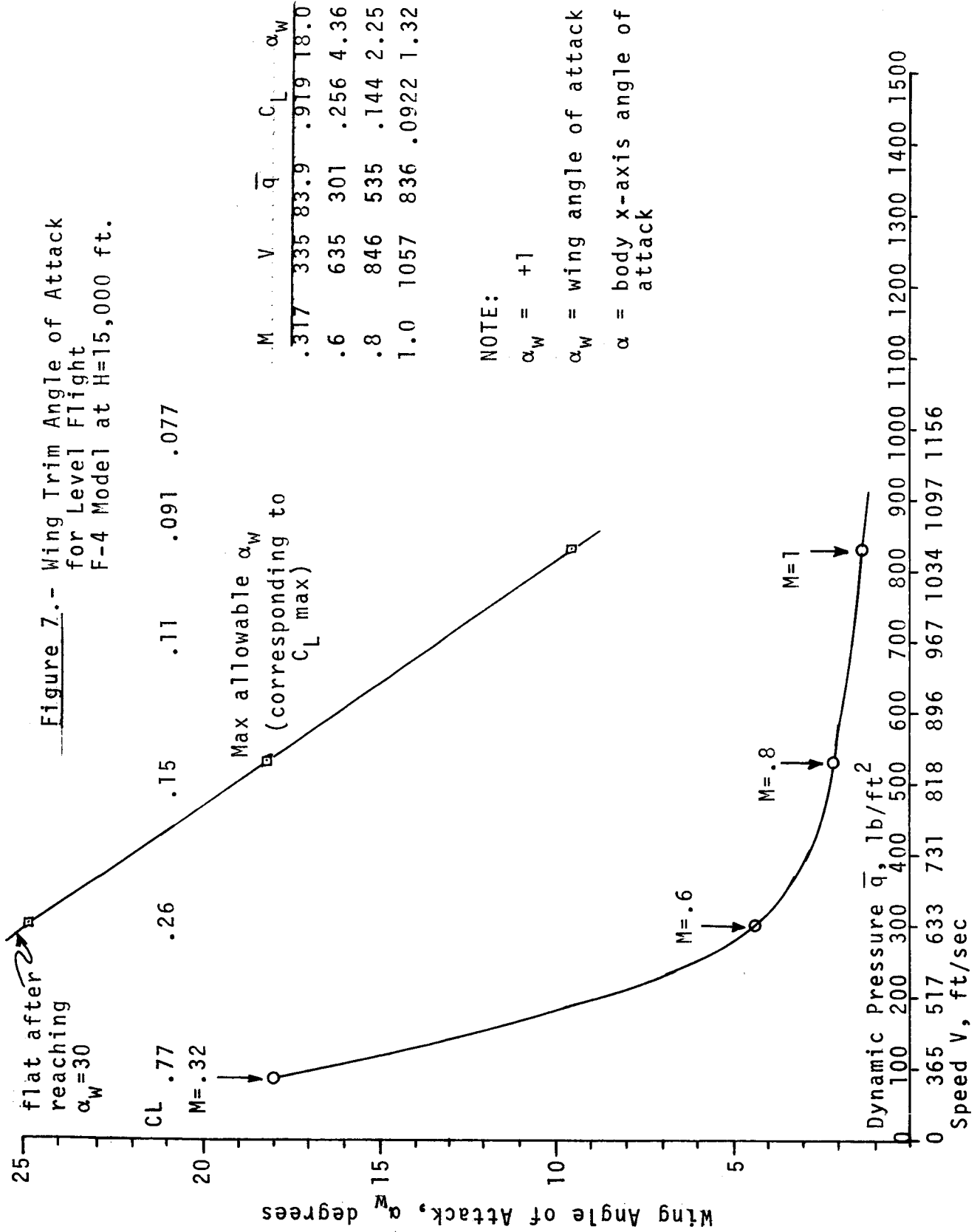
Figure 8.- Rise Time for Step Command in Roll

Legend
1. Based on trimmed level flight initial conditions
2. Up to 170° step command in roll
3. H = 15,000 feet
4. Symbols: largest rise time, average rise time, smallest rise time
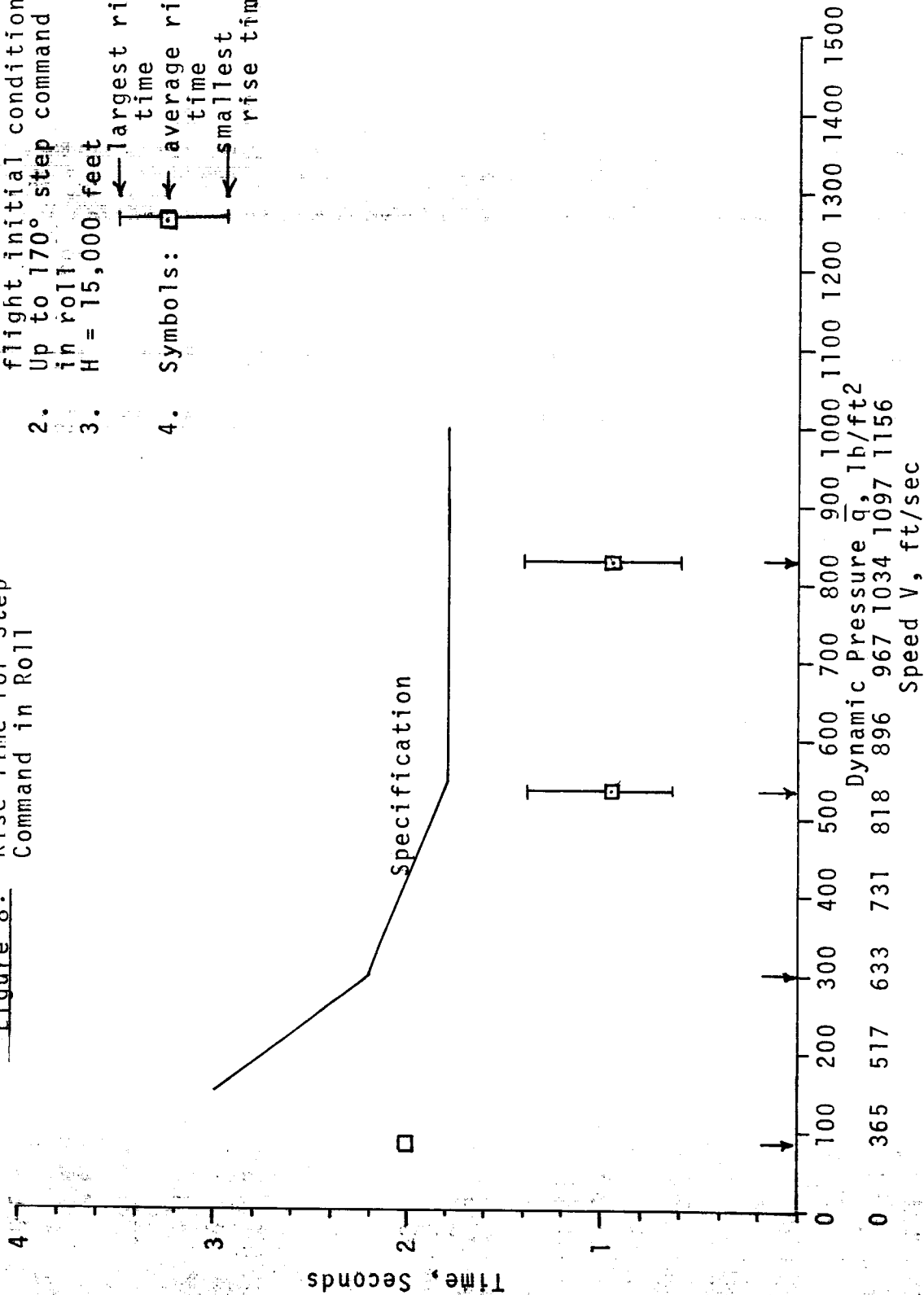
shown. The bang-bang system gives extremely fast and effective roll control.

In Figure 9 is shown the observed rise time for the roll control system at Mach=1.0 at 15,000 feet as a function of the commanded roll angle in degrees. The control system chooses the proper direction in which to roll so that the roll maneuver executed in one second never exceeds 180 degrees. The rise time appears nearly constant out to about 50 degrees which is due partly to the effect of the linear control system for small roll errors. With the bang-bang roll control system in effect the rise time increases nonlinearly as a function of commanded roll angle.

Although specifications were not developed for flight at 1000 feet and 35,000 feet the curves of wing trim angle of attack for the F-4 are shown in Figures 10 and 11 versus Mach number. These would be useful in establishing and verifying rise time specifications for pitch and roll step input at these flight conditions.

### Other Specifications for Controlled Response

Although the rise time is critical in specifying control system response for air combat attitude maneuvering, other performance measures may be necessary or desirable. Primary among these are the overshoot, the settling time, and the steady-state error.

The amount of overshoot is controlled mainly by the natural damping of the aircraft and the artificial damping due to pitch rate $\dot{q}$, roll rate $\dot{\phi}$, and sideslip angle rate $\dot{\beta}$. Since we had essentially perfect, noise-free values of these rate signals, overshoot in responses to step input could be controlled with no difficulties whatever by raising the rate gain coefficients. Note, however, that raising rate gain in a loop results in increasing the rise time.

With good rate signals, as in this study, the overshoot resulting from single-axis step input commands was usually less than

Figure 9 .- Rise Times for Roll Versus
Commanded Roll Change
F-4 at M-1.0 at 15,000 ft.
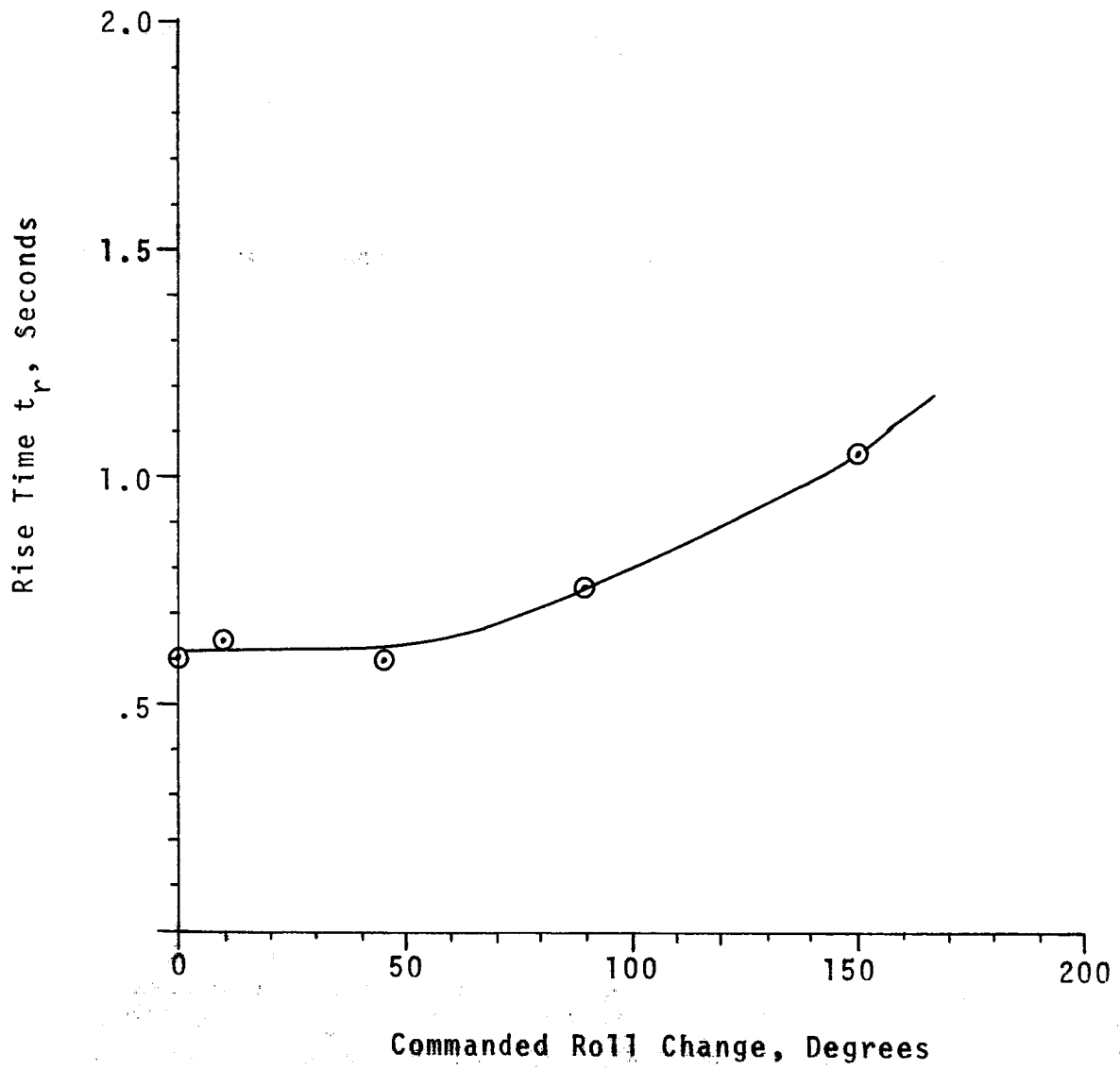
Figure 10 .- Wing Trim Angle of Attack
for Level Flight
F-4 at H=1,000 ft.

| M | V | $\bar{q}$ | $C_L$ |
|---|---|---|---|
| .235 | 261.1 | 78.7 | .980 |
| .3 | 333.9 | 128.7 | .599 |
| .6 | 667.8 | 514.7 | .150 |
| .8 | 890.4 | 914.9 | .0843 |

| $\alpha_w$ | Max L/w | $C_L$ | Max L/w $\alpha_w$ |
|---|---|---|---|
| 30.0 | 1.0 | .98 | 30.0 |
| 10.62 | 2.07 | .98 | 30.0 |
| 2.57 | 5.13 | .77 | 14.9 |
| 1.33 | 7.0 | .59 | 10.3 |

$\rho = .0023081$

$C_s = 1113.05$

Max Load Factor

Level Flight

M=.23  .3     .6      .8

Wing Angle of Attack, $\alpha_w$ Deg.

Dynamic Pressure $\bar{q}$, lb/ft$^2$

0   100   200   300   400   500   600   700   800   900   1000   1100

0   294   416   510   589   658   721   779   833   883   931   976

Speed V, ft/sec

51
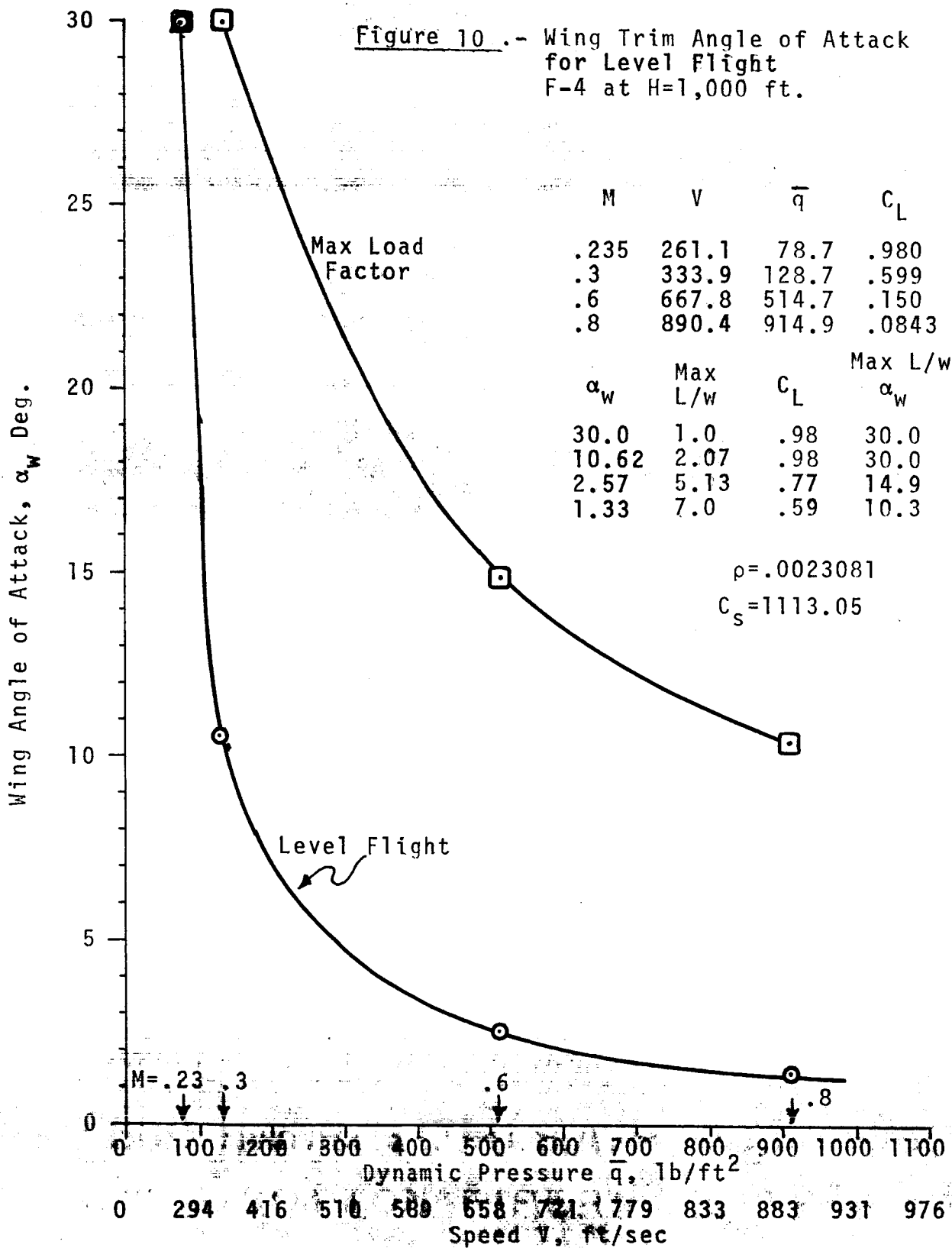
Figure 11.- Wing Trim Angle of Attack for Level Flight F-4 at H=35,000 ft.

| M | V | $\bar{q}$ | $C_L$ |
|---|---|---|---|
| .6 | 584.0 | 125.6 | .614 |
| .8 | 778.6 | 223.3 | .345 |
| 1.0 | 973.3 | 348.9 | .221 |
| 1.1 | 1070.6 | 422.1 | .183 |

| $\alpha_w$ | Max L/w | Max L/w $C_L$ | Max L/w $\alpha_w$ |
|---|---|---|---|
| 10.5 | 1.0 | | 25.0 |
| 5.2 | 2.8 | .97 | 24.9 |
| 3.0 | 4.5 | .99 | 22.2 |
| 2.5 | 5.3 | .97 | 18.1 |

$\rho = .00073654$

$C_s = 973.276$
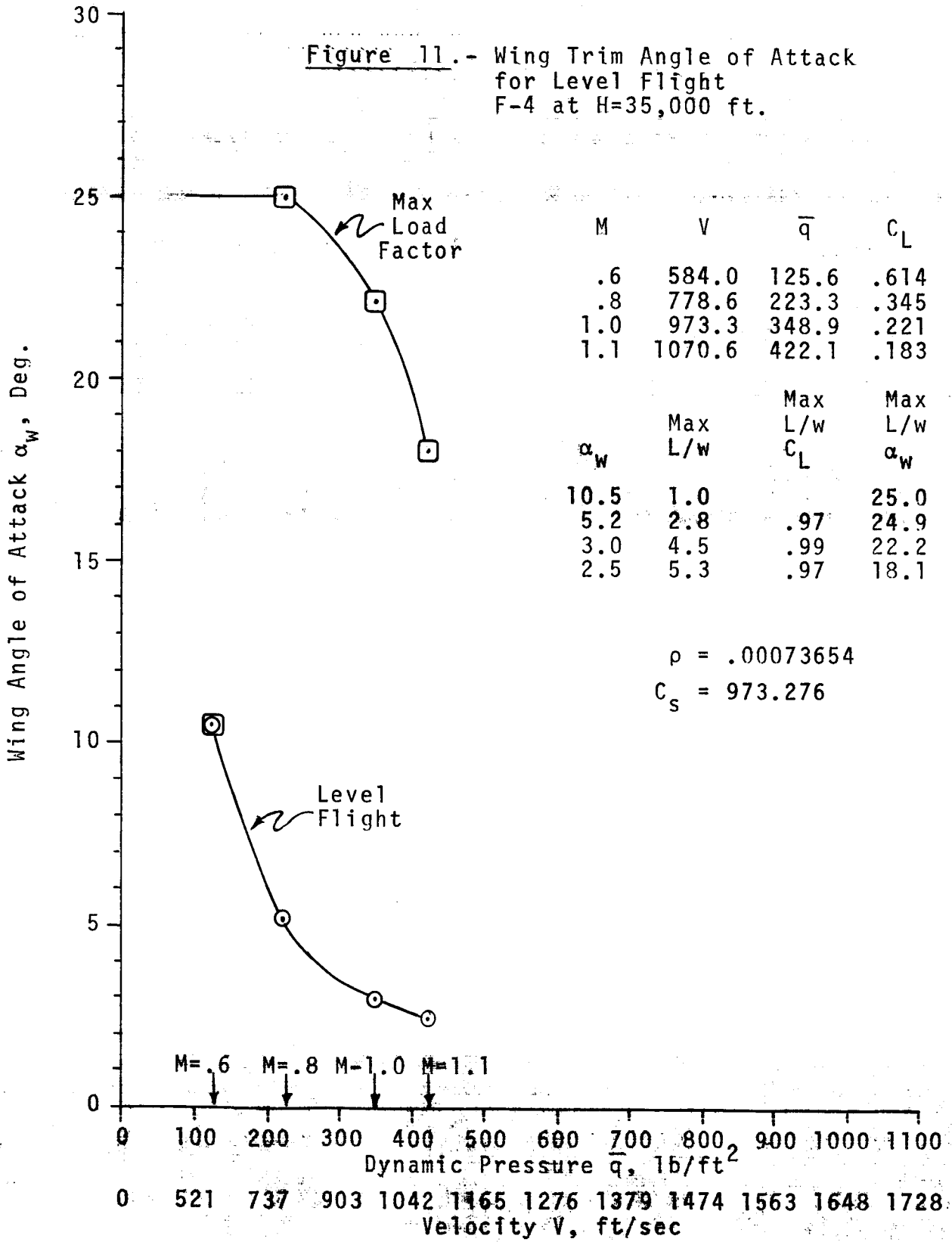
two degrees in pitch and five degrees in roll, regardless of the magnitude of the input step. Thus, specification values given below should always be satisfied when the control gains are properly adjusted.

Specifications for Single-Axis Step Input Response

For step commands $\Delta\alpha_c$ in pitch:

Overshoot must be less than five degrees in magnitude for step input command $\Delta\alpha_c$ less than 15 degrees to final values of $\alpha$ in the range of $0° \leq \alpha \leq 20°$ starting from trimmed level flight.

For step commands $\Delta\phi_c$ in roll:

Overshoot must be less than 20 degrees in magnitude for step input $\Delta\phi_c$ up to 180 degrees starting from a condition of trimmed level flight.

For combined maneuvers simultaneously in pitch and roll the control system incorporates logic to avoid large roll errors due to rigid body coupling. Before this logic was implemented, roll overshoots as high as 360 degrees were observed during extreme roll rate maneuvers with combined pitch commanded changes. With the addition of the control logic and the development of the bang-bang control system for roll, the observed overshoots for combined maneuvers were dramatically reduced. Due to the limited resources of this study, no specifications were developed and verified for combined pitch-roll maneuvers.

The settling time in response to a step input is usually defined as the time elapsed before the response enters and remains inside a region $\pm 5$ percent of the input step magnitude centered on the steady state output. For the purposes of air-to-air combat it would be preferable to make the settling time within $\pm 10$ percent of the steady state value or to give the settling time in terms of absolute errors in pitch or roll. A pitch error of $\pm 1.5$ degrees and a roll error of $\pm 3$ degrees could be recommended.

53

# THE LONGITUDINAL CONTROL SYSTEM

## Longitudinal Control for Linearized Equations

Since one of the ground rules for the control system design was the unlimited accessability to all state variables for feedback purposes, the most logical choice for selecting a feedback path was to feed back the very variable which was to be controlled, that is, the angle of attack. Note that the linearized equations for the longitudinal direction have been written in such a form that $\alpha$ is one of the state variables.

The control system used for the longitudinal or pitch axis is shown in Figure 12. The proportional gain $K_{P\alpha}$ represents the number of degrees the stabilator should be deflected per degree of angle of attack error $\alpha_E$. At low speeds and low dynamic pressure $\bar{q}$ a larger stabilator deflection is needed to exert fast control, while at high speed and large dynamic pressure, a considerably lower $K_{P\alpha}$ must be used to avoid overstressing the aircraft. From a speed of response point of view $K_{P\alpha}$ should be as high as the other contraints will allow.

The integral gain $K_{I\alpha}$ is necessary to remove steady-state error because $K_{P\alpha}$ cannot be raised high enough to do so. If $K_{I\alpha}$ is too low, steady state errors will persist indefinitely, while if it is too high, loop instability may result.

Rate feedback gain $K_{R\alpha}$ provides an error signal proportional to $\frac{d\alpha}{dt}$ which damps the pitch motion and provides lead compensation to the longitudinal control loop. If $K_{R\alpha}$ is too high, pitch response will become very slow and sluggish. This gain should be set only as high as necessary to provide damping of pitch axis motions.

The program LONLOP (Longitudinal Loop) calculates the dynamics of the pitch control loop for any set of gains.

The transfer function is negative since positive stabilator is defined as down elevator (stick forward) which results in a reduced
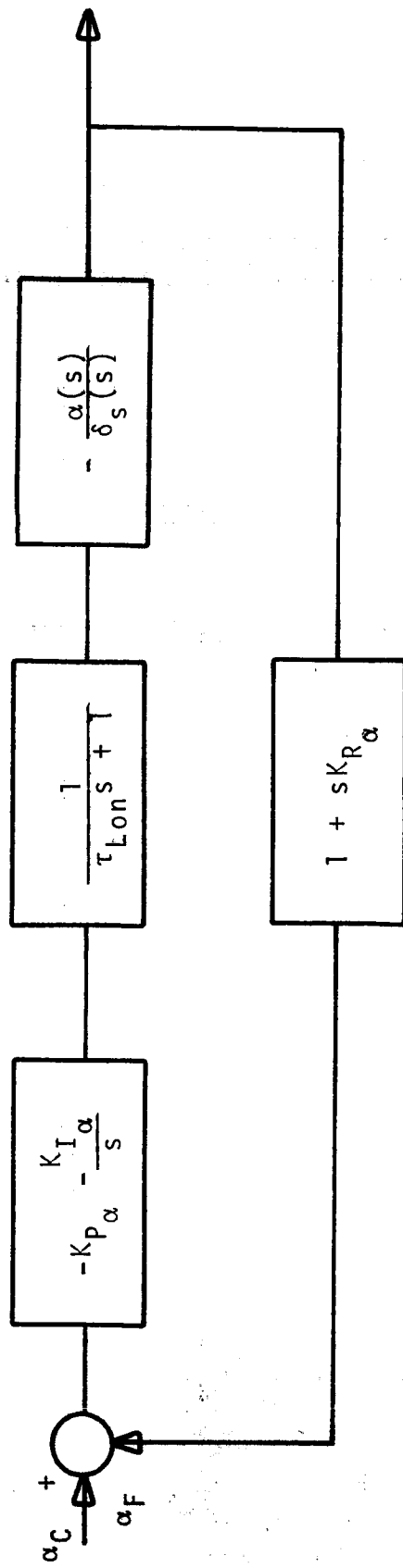
Figure 12.- Longitudinal Control Loop
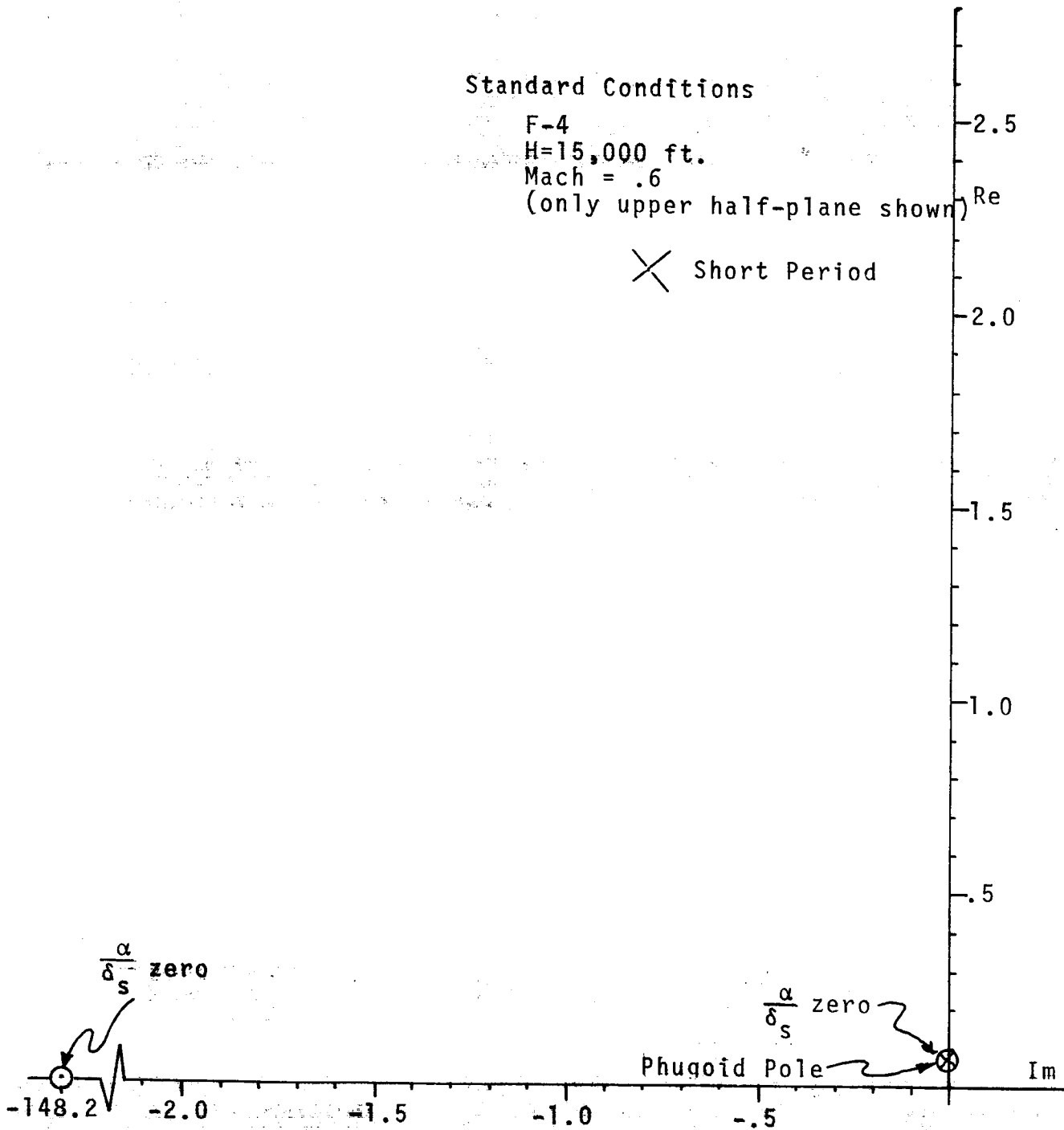
Figure 13.- Longitudinal Dynamics-Open Loop

angle of attack.

This transfer function shows a set of zeros that very nearly cancels the phugoid poles, indicating that the phugoid mode is not very important in angle of attack response to stabilator deflection. The phugoid mode is usually of very low frequency, in this case with a period of $\frac{2\pi}{\omega} = \frac{2\pi}{.069} = 91$ seconds. The period of the short period mode, on the other hand, is $\frac{2\pi}{2.113} = 3$ seconds. If these roots are plotted in the complex plane, they appear as in Figure 13.

A typical response of a system to a step input is shown below:
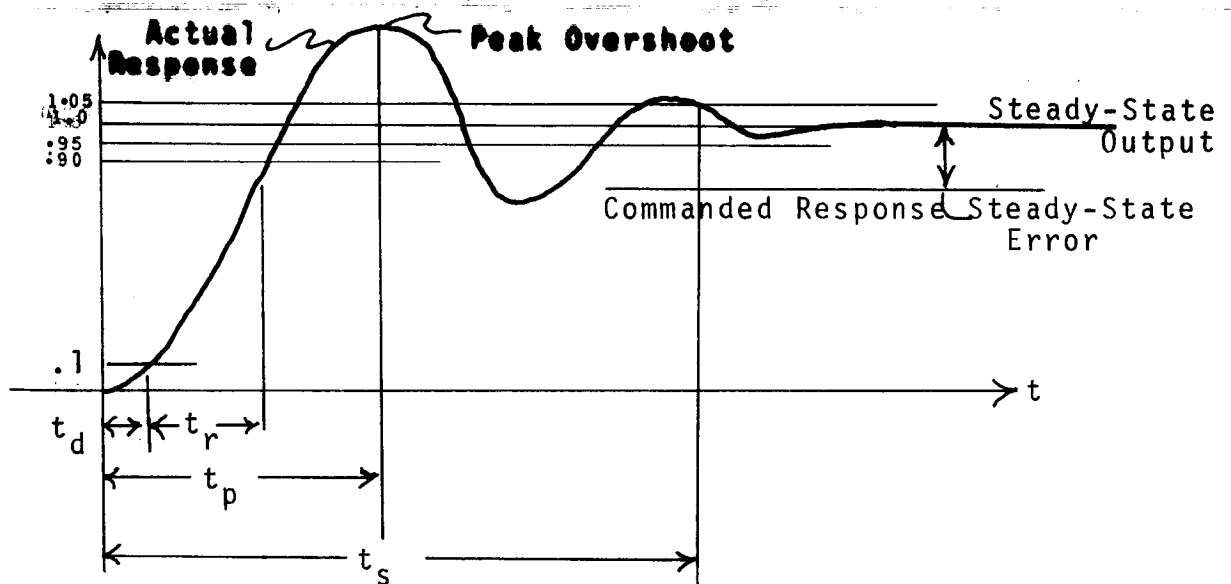


**Figure 14.- Typical Response to Step Command Input**

Here, $t_d$ is the delay time, $t_r$ is the rise time, and $t_s$ is the settling time.

For the fastest possible attitude control the delay plus rise time must be as short as is physically possible and the overshoot must still be kept reasonably small. The settling time is of lesser

57

importance. Small variations in $\alpha$ about the proper mean value will have very little effect on the aircraft trajectory in air combat.

### Procedure for Determining Longitudinal Gains

The first step in determining desirable gains is to run ATDYN for the flight condition of interest and obtain a data deck of the K array and the printout of roots. This output includes both longitudinal and lateral data.

The transfer function for $\frac{\alpha(s)}{\delta_s(s)}$ between stabilator and angle of attack is given in the printout section entitled LONG. WITH ALPHA/DELS ZEROS as the ratio of the third order polynomial to the fourth order polynomial. For the standard conditions (V=634.7 ft/sec for Mach .6, H=15,000 ft, $\alpha=\theta=3.3375°$) there results

$$\frac{\alpha(s)}{\delta_s(s)} = \frac{(-.0773 - .04426s - 14.18s^2 - .0957s^3)}{(.0306 + .0224s + 6.446s^2 + 2.0039s^3 + 1.2634s^4)}$$

which in factored form using corresponding roots becomes

$$\frac{\alpha(s)}{\delta_s(s)} = \frac{-.0957 \; [(s+.00154)^2 + .0719^2] \; (s+148.2)}{1.2634 \; \underbrace{[(s+.00100)^2 + .069^2]}_{\text{phugoid poles}} \; \underbrace{[(s+.7920)^2 + 2.113^2]}_{\text{short-period poles}}}.$$

Steady-state errors are important and cannot be allowed. They would significantly compromise the ability of the aircraft to follow AML commanded maneuvers.

Once the $\frac{\alpha}{\delta_s}$ transfer function is understood the next step is to solve for the roots of the combined (control system + aircraft), or closed-loop, as portrayed in Figure 13. This computation is done by LONLOP. Input data for LONLOP consists of the K-array deck plus a set of longitudinal loop gains ($K_{P\alpha}$, $K_{I\alpha}$, $K_{R\alpha}$). The first run of LONLOP can be done with these gains set to zero. The resulting poles (denominator roots) should be the same as those obtained in

58

ATDYN, and this serves as a check.

To obtain very fast response relatively high values of $K_{P\alpha}$ and $K_{I\alpha}$ are necessary. $K_{P\alpha}$ as high as four can be used at very low speed, while $K_{P\alpha}=2$ may be quite high enough at high speed. Values of $K_{I\alpha}$ usually run between 25% and 150% of the $K_{P\alpha}$ used. The rate gain $K_{R\alpha}$ should be as low as possible to result in keeping the damping of the closed loop short period poles at least as high as $\zeta=.25$.

It is quite helpful to plot the location of the closed-loop poles for ranges of one gain with the others held either constant or in a constant ratio. In the present case, since the controller transfer function is

$$-K_{P\alpha} - \frac{K_{I\alpha}}{s} = -\left(\frac{K_{P\alpha}s+K_{I\alpha}}{s}\right) = \frac{-K_{P\alpha}\left(s+\frac{K_I}{K_{P\alpha}}\right)}{s}$$

it is convenient to keep the ratio $\dfrac{K_{I\alpha}}{K_{P\alpha}}$ constant.

This fixes the zero location due to this term at the point $s = \dfrac{-K_{I\alpha}}{K_{P\alpha}}$. A plot of closed-loop root locations with this ratio fixed and a series of values of $K_{P\alpha}$ then gives the root locus for proportional gain changes. Although not necessary for the use of these programs a knowledge of the procedures and meaning of the root locus design technique is definitely useful. Readers lacking this background can obtain a brief introduction from Appendix C of Reference 7 (Blakelock).

Once a set of suitable gains is found they should be tested using the program version of NONLIN to find the response to a step change in $\alpha$. Usually the damping obtained in these nonlinear model runs is higher than the linear program results indicate.

Gains suitable at high speed will not be appropriate at low speed, so that for the complete flight envelope at least three and possibly more sets of gains, each applicable over some portion of the flight envelope, will be necessary for longitudinal control of a given aircraft.

## Longitudinal Control for Nonlinear Equations

Basically the same control system as described under the linear longitudinal control is implemented in the AML75 program. Since angle of attack in the nonlinear equations is not one of the state variables, its rate is computed as

$$\dot{\alpha} = \frac{uq - vp + g\left\{t_z + \bar{q}S[C_x \sin\alpha_p + C_{z'} \cos\alpha_p]/_W + CD33\right\} - \dot{u}w/_u}{u + \bar{q} \, Sg \, \frac{\bar{c}}{2V_T} \, C_{L_{\dot{\alpha}}} \cos\alpha_p/_W}$$

The variable $\dot{\phi} = p + (q \sin\phi + r \cos\phi) \tan\theta$ is also computed for later use.

A control logic was incorporated to avoid commanding extreme pitch maneuvers while relatively violent roll maneuvers were in progress. This logic is one possible way to avoid the pitch-roll coupling problem.

The logic implemented first calculates the magnitude of the present roll error $|\phi_c - \phi|$ and the magnitude of the roll rate error $|\dot{\phi}_c - \dot{\phi}| = |\dot{\phi}|$. If the conditions $|\phi_c - \phi| > 30$ degrees and $|\dot{\phi}| > 50$ deg/sec are simultaneously satisfied LOCNTR goes into a "pitch slowly" mode such that

$$\alpha_{cMD} = \alpha + \frac{1}{2} (\alpha_c - \alpha).$$

This means that the pitch control aims for a point halfway between the existing angle of attack and the desired angle of attack $\alpha_c$.

An additional logic was added which ensures that during such a relatively violent roll maneuver no angles of attack greater than 10 degrees are commanded. This logic momentarily brings the aircraft to a low angle of attack region where roll control is most effective.

60

When the error and error integral terms have been computed the stabilator deflection is calculated:

$$\delta_s = -K_{P\alpha} \, \alpha_E - K_{I\alpha} \int_0^t \alpha_E \, dt$$

Finally, the limits of stabilator motion are imposed, namely -23 deg $< \delta_s < + 7$ deg.

Scheduling of the gains as a function of dynamic pressure was implemented to provide fast pitch control over widely different flight conditions. For dynamic pressures $150 < \bar{q} < 520$ lb/ft$^2$ the longitudinal gains read in as VAR(3), VAR(6), and VAR(10) in the input data deck.are used. For convenience the longitudinal gains are written in the vector form $(K_{P\alpha}, K_{I\alpha}, K_{R\alpha})$.

The gains for $\bar{q} < 150$ lb/ft$^2$ are set at (3, 3, .6) and for $\bar{q} > 520$ lb/ft$^2$ are set at (1, 2, .25) in LONCTR and can be changed only by altering the corresponding statements in the program.

# THE LATERAL CONTROL SYSTEMS

## Linear Lateral Control

The aircraft dynamics and consequently the control systems necessary for the lateral axes are more complex than for the longitudinal axis. Since in this study all state variables are known, control of roll was based on knowing roll angle $\phi$ and roll rate $\dot{\phi}$. Control of rudder was based on knowing sideslip angle $\beta$ and its rate $\dot{\beta}$.

With the rather complex nonlinear model of an aircraft the first approach was to determine what performance could be achieved with simple linear control and more particularly to find those conditions under which these basic linear controllers became marginal or exhibited poor performance or instability.

The linear control systems developed for rudder and aileron control are shown in Figure 15. The $\beta$ loop was set up originally without rate feedback. Large error excursions during combined maneuvers resulted, and the addition of $\beta$ rate feedback brought almost direct control of Dutch roll damping and greatly improved performance.

Roll control was obtained using proportional, integral, and rate feedback. Integral control was required to reduce roll steady state error to zero.

Quite early in this study it was found that a linear controller with proper gains would give very good roll control for small step inputs, yet for large inputs with the same gains gave slow response, large overshoots, and generally poor control. Negative rate feedback is also inappropriate for large roll maneuvers because it results in reducing the maximum roll rate and thus slowing down roll maneuvering. For these reasons, and because of the need to maneuver as fast as possible in air combat, a bang-bang control system was developed for aileron control.
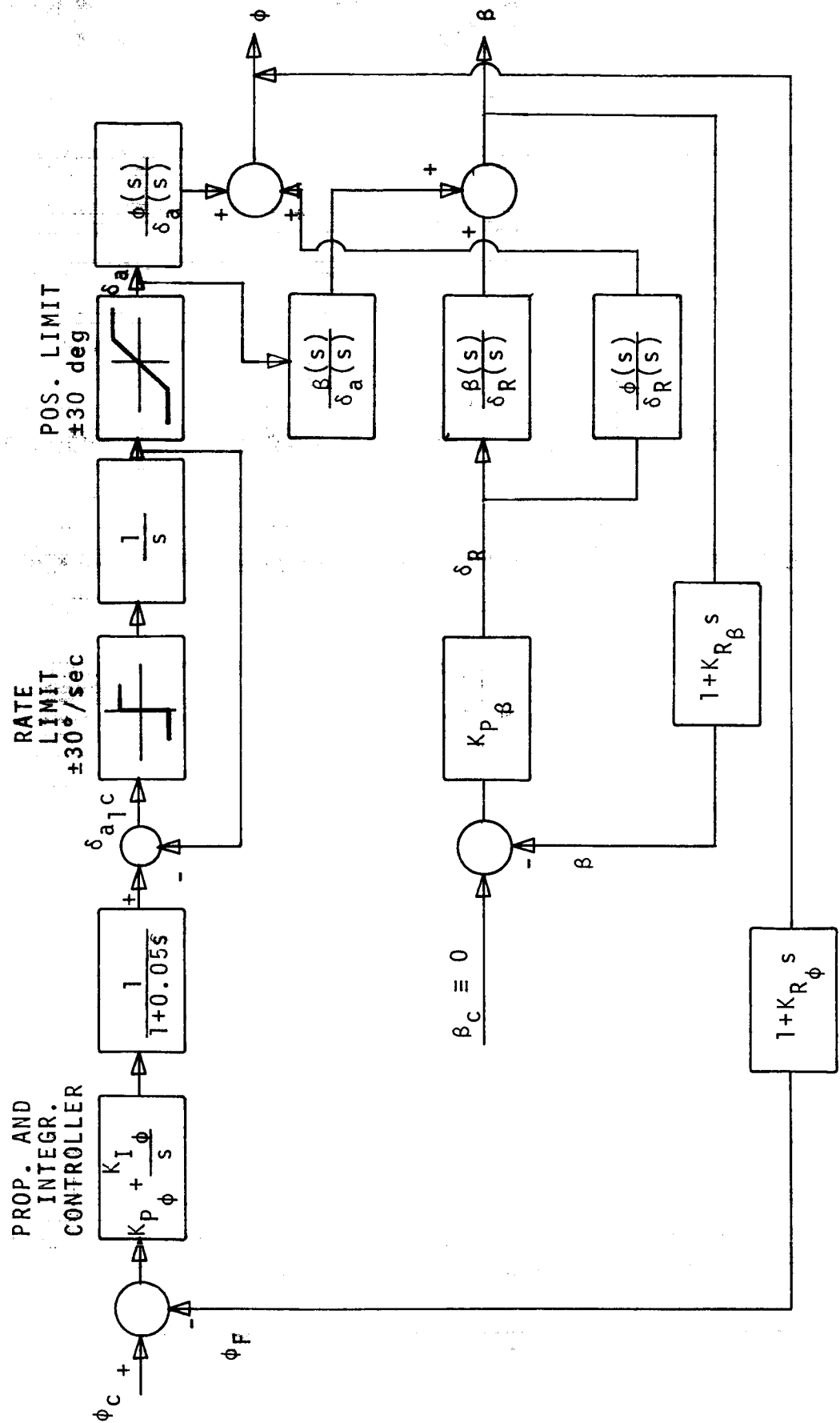
Figure 15.- Linear Lateral Control Loops

## The Bang-Bang Control System
## for Aileron Control

It is easily proven that for time-optimal control of a system with limited control authority the controls should always be deflected to their stops.   Thus, achieving a new roll orientation would involve applying full aileron in the direction to reduce roll error, followed by switching to full aileron in the opposite direction at the correct moment to result in zero error simultaneously with zero error rate.

The general problem of time-optimal attitude maneuvering of an aircraft is quite complex, and will not be treated here.  It is apparent that a time-optimal attitude control system would have to give the fastest possible response in single-axis maneuvers as well as in combined pitch and roll maneuvers.  Thus, time-optimal roll control would be an important segment of the problem.

As a first approximation in developing the bang-bang control system, the assumption was made that the aircraft in its roll behavior may be approximated as a simple double-integral plant.  Note that by definition, the error is:

$$\phi_e = \phi_c - \phi$$

and, consequently, the ordinate in the phase plane is

$$\dot{\phi}_e = -\dot{\phi}$$

Figure 16 a shows the phase plane plot for such a plant, for any given initial condition for roll error and error rate, these curves show the response to full negative aileron (solid lines) and to full positive aileron (broken lines).  Note that the curve $\gamma_+$ contains the loci of all points for which, if full positive aileron is applied, the system will simultaneously reach zero error rate.  Similarly, $\gamma_-$ is the location of all points with that same property for full negative aileron.

64
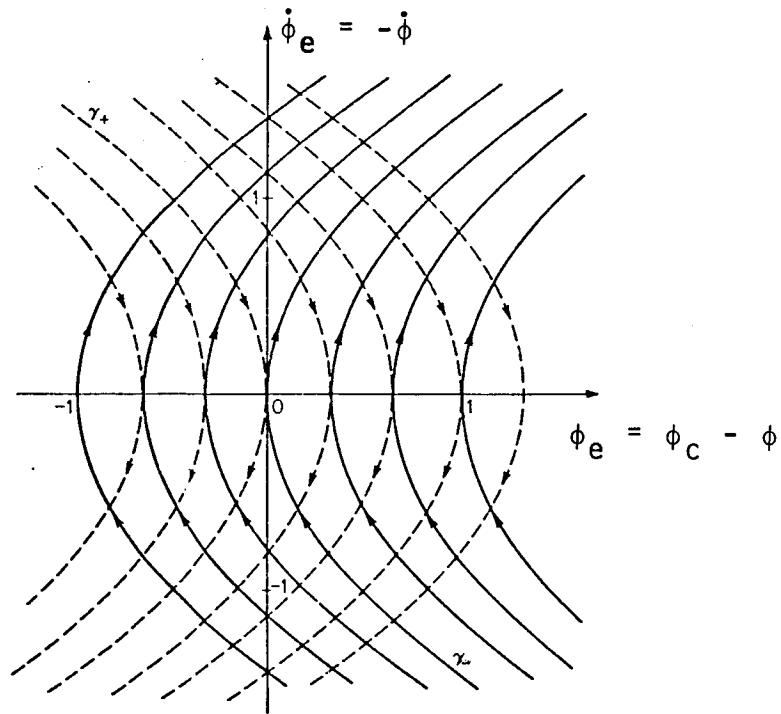
Figure 16a.- Phase Plane Curves for
Pure Double Integral Plant
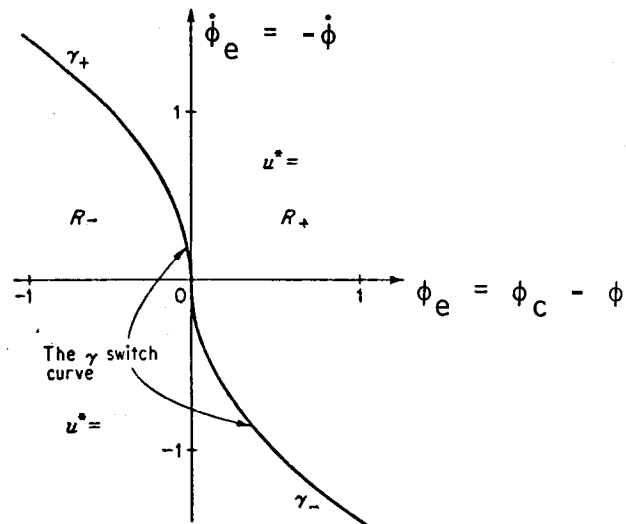


Figure 16b.- Aileron Switching Curve for
Pure Double Integral Plant

Assume that the system initially does not find itself on either the $\gamma_+$ or the $\gamma_-$ curve, but anywhere within the region labeled $R_-$ in Figure 16b. The basic idea of bang-bang control then, of course, is to apply full negative aileron until the solid line through our initial point has brought the system onto the curve $\gamma_+$, the switching curve. At this moment, full positive aileron is applied until both error and error rate vanish simultaneously at which time aileron is neutralized. If the system finds itself at time zero somewhere in the region labelled $R_+$, the same process with reversed signs is applied.

Still under the assumption of a pure double integral plant, we can easily calculate the switching curves. This is done by back-tracking the switching curve from the origin; note that in the following, time is reversed (see Reference 9):

$$\ddot{\phi}_e = \frac{L_{max}}{I_{xx}}$$

where $L_{max}$ is the assumed rolling moment, constant during the control maneuver. Since the "initial" condition is $\phi_e = 0$ and $\dot{\phi}_e = 0$, it follows that:

$$\dot{\phi}_e(t) = \frac{L_{max}}{I_{xx}} t$$

and

$$\phi_e(t) = \frac{L_{max}}{I_{xx}} \frac{t^2}{2}$$

Eliminating t, we obtain

$$\dot{\phi}_e = \sqrt{2 \frac{L_{max}}{I_{xx}} \phi_e}$$

66

Note that, if the three following assumptions were true, we would have a perfect time-optimal control system:

1. Roll dynamics is pure double integration plant.
2. Roll moment remains constant during the maneuver under investigation.
3. Aileron can be "switched" from full positive to full negative and reverse instantaneously.

Since the three assumptions listed above are, of course, not satisfied in reality, we have to investigate how this affects the control system. The main effect of assumptions one and two not being satisfied is a change in the shape of the phase plane curves.

The response of a fighter aircraft to hard right aileron exhibits high roll acceleration as long as the roll rate is small, but soon approaches a maximum roll rate as a result of natural roll damping from the wing and limited aileron effectiveness. This effect is shown in the phase plane plot of Figure 17.

If the aileron is instantaneously reversed when the aircraft is rolling at maximum roll rate, the roll deceleration is very large because both the aileron and the natural roll damping are then acting in the same direction. This is shown in the trajectory ABC, where at point B the aileron is switched from $\delta_a = + 30$ degrees to $\delta_a = -30$ degrees.

Theoretically, to achieve time optimal roll control, the aileron would have to be reversed at exactly the right moment. The point of switching in the $\phi_e$, $\dot{\phi}_e$ phase plane depends on the roll acceleration that will be attained after switching. Under the ground rules of this study (unlimited access to all state variables and complete knowledge of all the stability and control derivatives), one can compute the exact roll acceleration (deceleration) obtained from full opposite aileron at any instant. The switching logic implemented here assumes this roll acceleration to remain constant during the time of switching to the time where the phase plane trajectory would pass through the origin, Section B-C
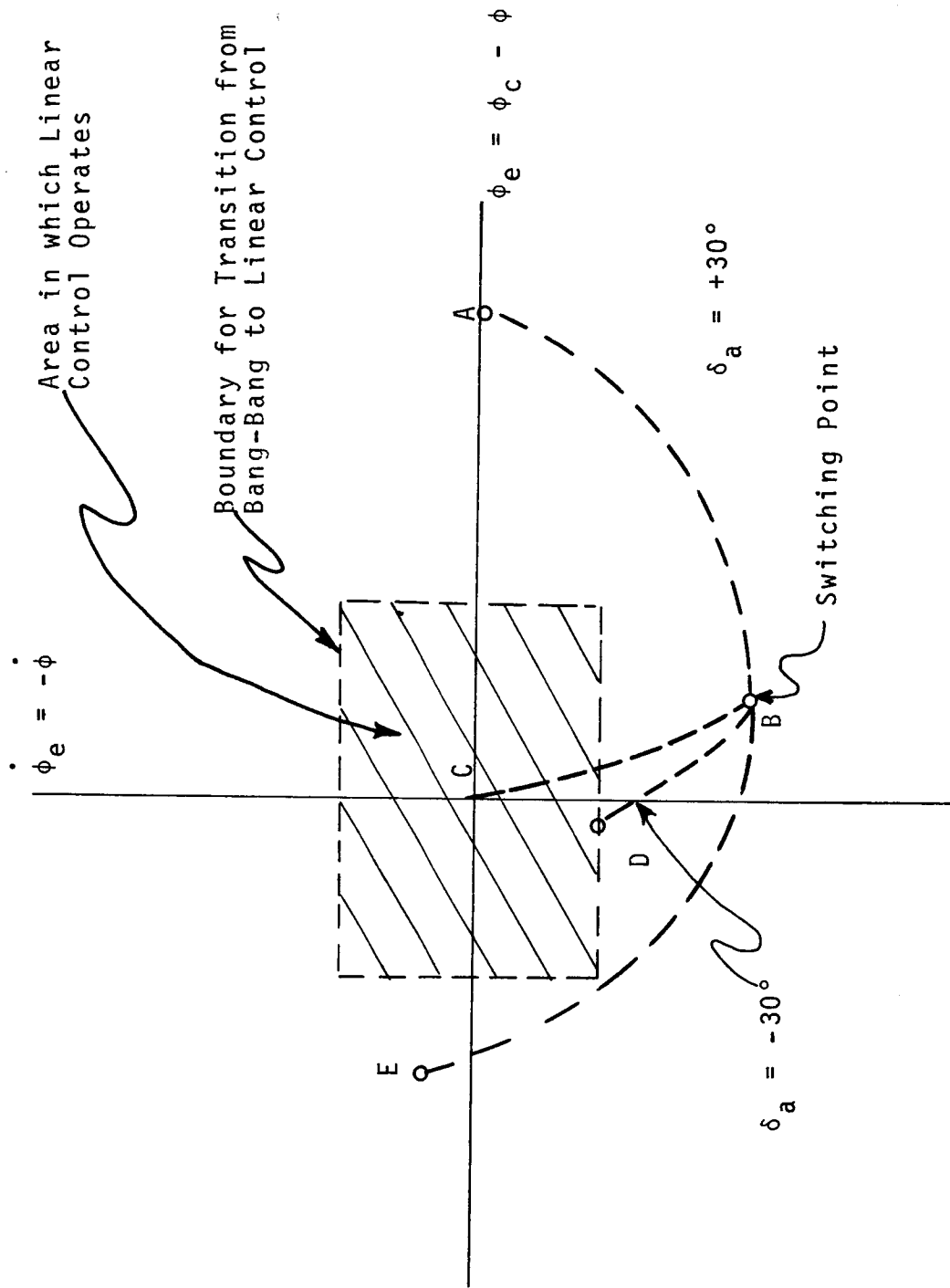
Figure 17.- Phase Plane for Aileron Control

of trajectory in Figure 17.

Note that in reality, the roll moment does not remain constant after the switching, but the roll deceleration will become smaller as the roll rate decreases. That means, the trajectory will not actually follow BC, but, say, BD. Fortunately, this is of little consequence, because at point D, the control system quits using bang-bang control and the linear control system takes over.

One might ask the question, what happens if the trajectory, after the switching point B, misses the linear region. This could, of course, happen if the limits for the linear region are set by the user of the program to rather small values. The trajectory BE illustrates this situation. Should this happen, the trajectory will, of course, eventually reach the switching line in the quadrant $\dot{\phi}_e$ pos and $\phi_e$ negative, where the program would switch $\delta_a$ again back to positive 30 degrees. The system would remain stable and eventually again reach the linear control boundary. Of course, this would no longer constitute a time-optimal control. By choosing the linear boundaries reasonably large, this situation can be avoided.

# TYPICAL RESPONSES TO COMMAND SEQUENCES

It was pointed out earlier in this report that an important step in the control system design is the examination of the closed loop system under a realistic command sequence. Extensive runs were made using the program AML75 in Mode I to obtain such response time histories. Plotting routines were written to plot the command sequence and the associated response. Also plotted, were generally the three control surface deflections $\delta_s$, $\delta_a$ and $\delta_R$.

One typical such response is shown in Figures 18 and 19. The run started at level flight at an altitude of 15,000 feet and a Mach number of .6. The commanded angles of attack given in one-second intervals, reading from left to right, was

ALPCOM

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 8.54 | 8.53 | 10.70 | 8.53 | 8.52 | 8.52 | 12.35 | 10.08 | 6.56 | 5.84 |
| 8.51 | 4.74 | 16.28 | 16.33 | 16.63 | 8.51 | 10.58 | 12.19 | 13.51 | 8.52 |
| 10.75 | 12.47 | 8.53 | 8.51 | 8.50 | 8.49 | 8.49 | 10.40 | 8.48 | -3.86 |
| 15.94 | 8.45 | 10.15 | 11.55 | 12.7- | 8.48 | 8.48 | 10.43 | 11.89 | -3.07 |

while the commanded roll angle followed the sequence

PHICOM

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 61.15 | 61.19 | 70.95 | 71.72 | 72.01 | 72.45 | 85.48 | 91.95 | 91.11 | -105.3 |
| -117.8 | 0 | -64.82 | -74.63 | -84.58 | -57.54 | -62.99 | -65.28 | -65.51 | -49.50 |
| -56.59 | -59.22 | -56.81 | -47.80 | -49.24 | -50.53 | -51.69 | -57.34 | -53.36 | 180.00 |
| 97.84 | 77.34 | 80.54 | 80.33 | 78.63 | 6.09 | -165.9 | -166.7 | -167.1 | 180.00 |

Note the transition between nine seconds ($\phi_c$ = 91.11) and 10 seconds ($\phi_c$ = -105). In the plot of the commanded angles, these two points are connected directly, the control-system, however, executed the command using a right roll, going with $\phi$ through 180 degrees.

70

Figure 18.- PITCH COMMAND SEQUENCE RESPONSE
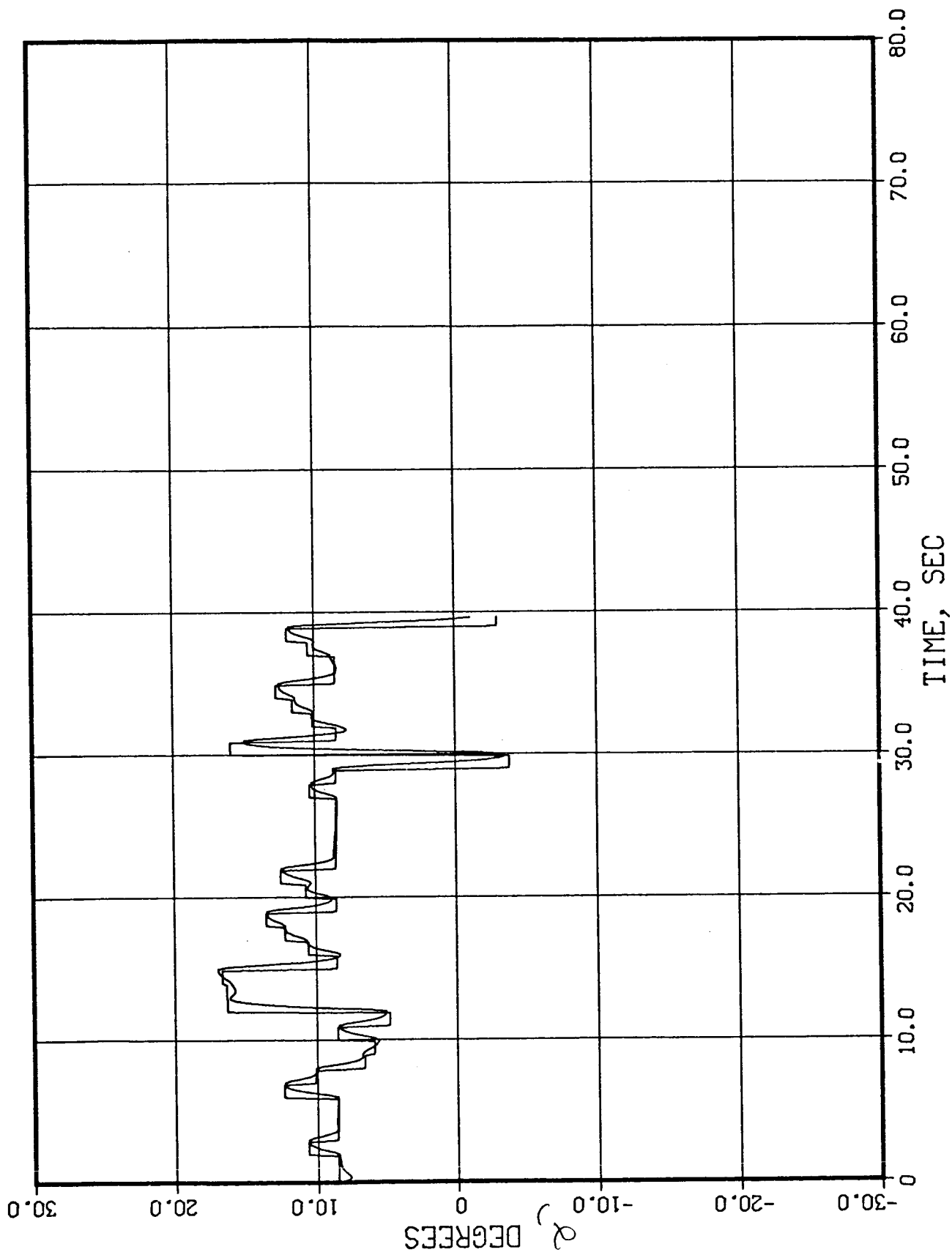
Figure 19.- ROLL COMMAND SEQUENCE RESPONSE
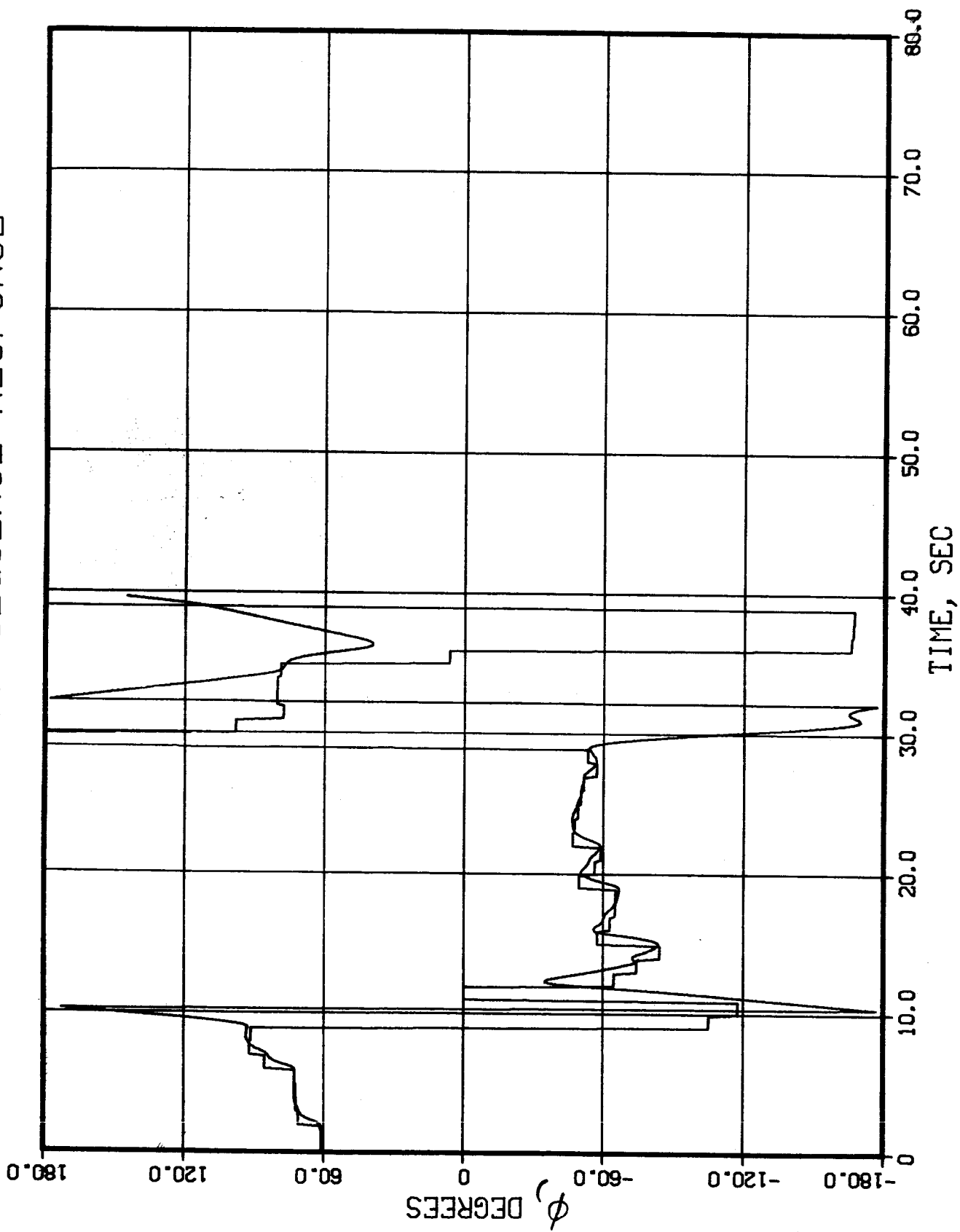
PART III:  DESCRIPTION OF THE
COMPUTER PROGRAMS

# PROGRAM AML75

## General Description

This program represents the main end product of this study. It combines the adaptive maneuvering logic with the flight control system described in the previous sections of this report. It has three basic modes of operation which are summarized below:

1. <u>Exercising the program with inputted command sequences</u>.- In this mode, the program accepts as input initial conditions for the target aircraft and a sequence of commands for roll angles and angles of attack. These commands are interpreted by the program to be equally spaced in time one second apart. Running the program in this mode is equivalent to exercising the control system controlling the nonlinear equations of motion, and, therefore, is most suitable for checking out modifications and improvements of the control system.

2. <u>Exercising the program against canned trajectories of the attacker</u>.- The purpose of this mode is to have the AML program, with its equations of motion being governed by the new control system, react against a canned attacker trajectory which is read from a magnetic tape. Running in this mode gives the user an opportunity to compare the performance of the AML program with the new control system against the old program. This mode also provides a convenient tool for debugging the AML program and to check out changes in the logic of the tactical decision process. To make the program run fast enough so it may be used as a real time opponent on the DMS, this mode of running against a canned trajectory should also prove to be very valuable.

3. <u>Exercising the AML program with control system against the old AML program</u>.- Here, motion of the target is modeled by the same nonlinear equations of motion as on the DMS, while the equations of motion for the attacker are those used in the earlier version of the AML program. The purpose of this off-line program is

74

two-fold.  First, it demonstrates the capability of the control
system to "fly" the aircraft effectively under realistic air com-
bat conditions and to properly execute the AML commanded maneuvers.
Second, it serves as a stepping stone for later implementation on
the DMS.  In the design of this program, great care was taken to
retain as much as possible the logic of communication between the
driver program and the AML program as it is presently in use on
the DMS.  Specifically, the AML program is called in the same man-
ner as in the present DMS program, that is, with a single argument
which indicates the function which the AML program has to perform,
namely, initialization, resetting or (real time) operation.  Most
important, the calling program uses a stepsize comparable to the
one used on the DMS and calls, during the (real time) execution the
AML program once every integration step.  Internally, however, the
AML equations of motion use a reduced stepsize.  It was found that,
by using Euler integration, a stepsize of 1/64 of a second was
sufficiently small to keep the truncation error in the numerical
solution of the differential equations acceptably small.  The
driver program uses a stepsize of 1/16 of a second (this is the
stepsize generally used in the AML off-line program), thus, the
nonlinear equations of motion of the target aircraft and the flight
control system are integrated over four  steps before the AML pro-
gram returns control to the calling program.

## Structure of the Program AML75

Figure 20 represents graphically the subroutine structure of
the AML75 program.  Every line from left to right indicates that
the subroutine to the left of the line has one or several calls
to the subroutine to the right end of the line.  It may appear to
the casual observer that the structure of this program is unneces-
sarily complicated.  This is, indeed, the case, but it will be
excused if the history of the development of this program is under-
stood.
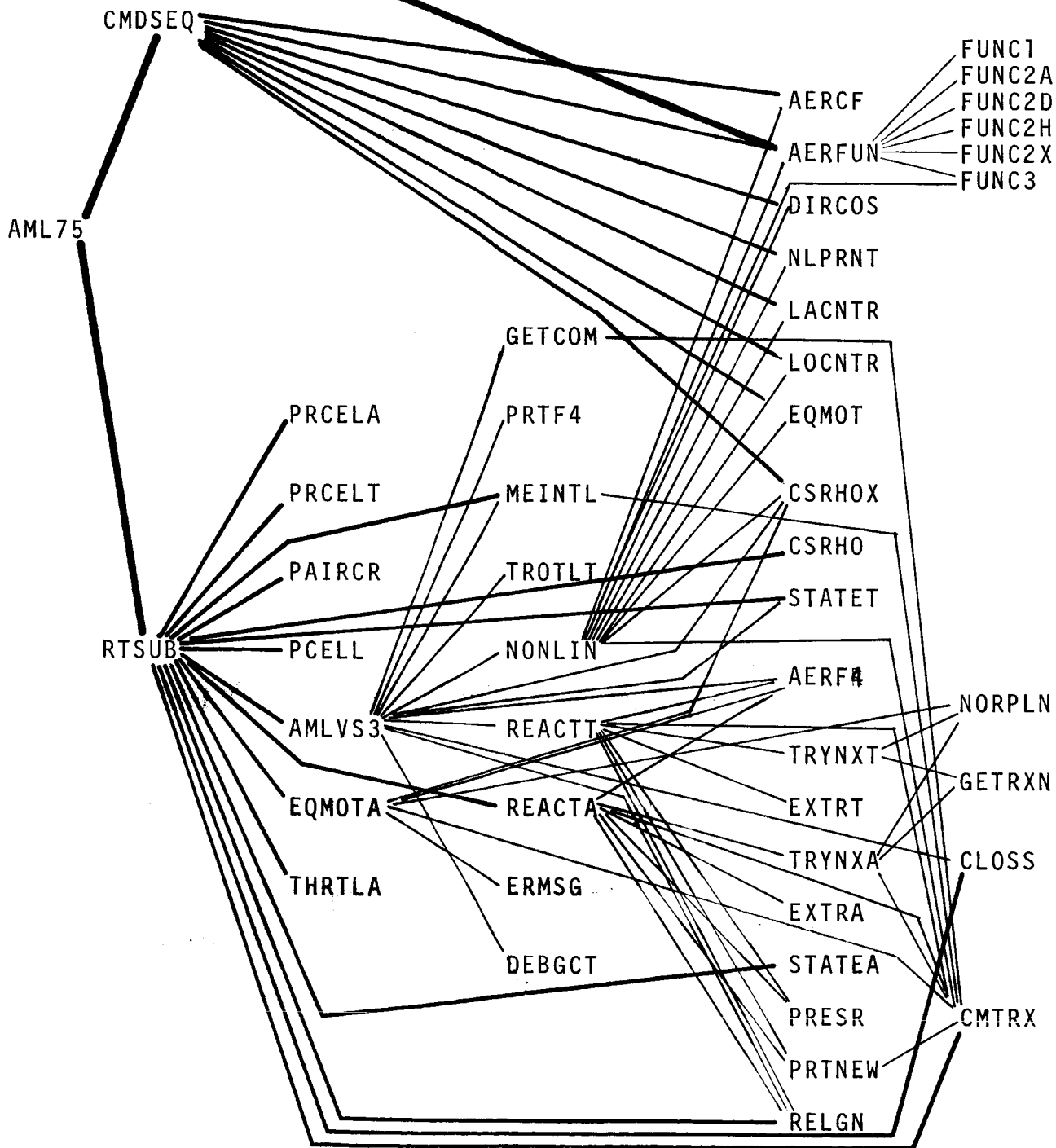
Figure 20.- Structure of the Program AML75

76

The main objective of this contract effort was to design and to develop a flight control system accepting as input commanded bank angles and commanded angles of attack and then to move the aircraft as efficiently as possible from its present attitude to the desired attitude.

The first step in such a design was to select some flight conditions which appeared to be fairly representative of air combat maneuvering. For a linearized analysis of these flight conditions and elementary control loops, stability and control derivatives as a function of Mach number, altitude and angle of attack were required. Thus, the first set of subroutines, whose primary purpose was to interpolate desired coefficients, were written and debugged. Most of this work was carried out on a small, inexpensive computer, a Honeywell 716. Program parts from this early phase are recognizable by their input-output unit assignments (3 and 7 for input, 4 for output). The subroutines AERFUN and the interpolation routines FUNCI, FUNC2A, FUNC2D, FUNC2H, FUNC2X and FUNC3 were developed during that first phase.

The next step consisted of developing the individual programs for the linear analysis. These programs are discussed in other sections of this report. Of importance here is only the fact that ATDYN uses some routines also used by AML75, namely: AERFUN, FUNCI, FUNC2A, FUNC2D, FUNC2H, FUNC2X, and FUNC3.

The logical next step was to verify that the control system based on the linearized equations would also be capable of adequately controlling the aircraft described by the complete set of nonlinear equations of motion. This was accomplished by exercising the AML75 program in Mode 1. The subroutine CMDSEO (Command Sequence) performs essentially this task. It reads all aerodynamic data and initial conditions for the aircraft (position and attitude and their derivatives) in the inertial reference frame and a string of angle of attack and bank angle commands and then solves the equations of motion for these initial conditions and the command sequence. For this program, all the necessary

subroutines to integrate the equations of motion were developed, such as:

AERCF: To calculate the coefficients $C_x$, $C_y$, $C_z$, $C_\ell$, $C_m$, $C_n$ .

EQMOT: To calculate forces and moments and to calculate $\dot{u}$, $\dot{v}$, $\dot{w}$, $\dot{p}$, $\dot{q}$, $\dot{r}$.

LAGNTR: To control aileron and rudder.

LOCNTR: To control stabilator.

DIRCOS: To calculate direction cosine matrix and Euler angles from quaternions.

NLPRNT: To print in condensed form aircraft data.

Note that so far in the development of the programs, only one aircraft was involved. Up to this point, there was therefore no need to identify the pertinent aircraft variables as pertaining to the attacker or to the target. The subroutines CMDSEQ and the six subroutines listed above identify all aircraft variables with neither a subscript nor an A or T appended to them.

After use of the program AML75, running in Mode 1, confidence in the proper working of the control system was obtained, the next step consisted of providing a command sequence actually generated by the AML program. The most convenient method to accomplish this was to exercise the AML driven program against a canned trajectory. This then required interfacing of the AML program and the aircraft driven by the control system. Basically, the formerly used sub-routine EQMOTT had to be replaced by the sequence of interpolating stability and control derivatives, determining rudder, aileron, and stabilator deflection, calculating forces and moments acting on the aircraft and then calculating $\dot{u}$, $\dot{v}$, $\dot{w}$, $\dot{p}$, $\dot{q}$ and $\dot{r}$, and, of course, of properly integrating these variables such that position and attitude of the aircraft in inertial space was obtained. But note, that this new sequence of computations for solving the equa-tions of motion had already been accomplished in the previous step,

except for one thing. When running AML75 in Mode 1, the thrust
was assumed to be full afterburner thrust at all times, for the
AML driven aircraft, this is no longer the case. Added to the
angle of attack and bank angle command was then a thrust command.
(To keep the program relatively simple, thrust commands of either
afterburner or idle only were introduced.) To accomplish this step
three major subroutines were added: RTMAIN3, AMLVS3 and NONLIN.
The intent was to structure the program so that it could be trans-
ferred with a minimum of changes to the DMS. The program RTMAIN3
performs essentially the same function as the program RTMAIN on
the DMS. Specifically, as RTMAIN on the DMS calls AMLVS2(K) with
K=1 for initialization, K=2 for resetting and K=3 for real time
operation, the routine RTMAIN3 calls the subroutine AMLVS3 the
these three arguments for performing the same three functions (the
operate function is, of course, not a real time function here).
The trajectory of the attacker aircraft is read from magnetic tape
every iteration cycle (1/16 of a second) of RTMAIN3.

In December 1975, the final step of the program development
was initiated. Its purpose was to replace the attacker's read-in
trajectory by a trajectory of an attacker aircraft driven and com-
puted by the old equations of motion and the AML program. This
required the addition of such routines as EQMOTA, THRTLA, REACTA,
TRYNXA, STATEA and EXTRA.

# DESCRIPTION OF THE INDIVIDUAL SUBROUTINES
## OF THE AML75 PROGRAM

### Program AML75 (Main Program)

The program AML75 has three different modes of operation.  It may simulate the aircraft/control system configuration under a sequence of commands input by data cards (Mode 1).  In Mode 2, it simulates the AML-driven target aircraft, equipped with the control system, maneuvering against a canned trajectory of an attacker.  In its third mode, it simulates a dogfight between an attacker described by the old equations of motion against a target described by the new equations of motion.

The main program AML75 serves simply as driver for selecting the mode.  A data card containing the mode is read and then, if it is Mode 1, the subroutine CMDSEQ is called,  if it is Mode 2 or 3 the subroutine RTSUB is called. (See Figure 21)

### Subroutine AERCF (Aerodynamic Coefficients)

Purpose.- This subroutine calculates the nondimensional aerodynamic force and moment coefficients.

Input.- Stability and control derivatives, control surface deflections, weight, $C_{33}$.

Output.- The three force coefficients $C_x$, $C_y$, $C_z$ and the three moment coefficients $C_L$, $C_M$, $C_N$.

Method.- The following relationships are programmed:

$$C_x = -(C_{D_0} + C_D)$$

where $D_{D_0}$ was set to zero and $C_D$ is obtained by interpolation of function F210 ($C_D = f(C_L, M)$).

80
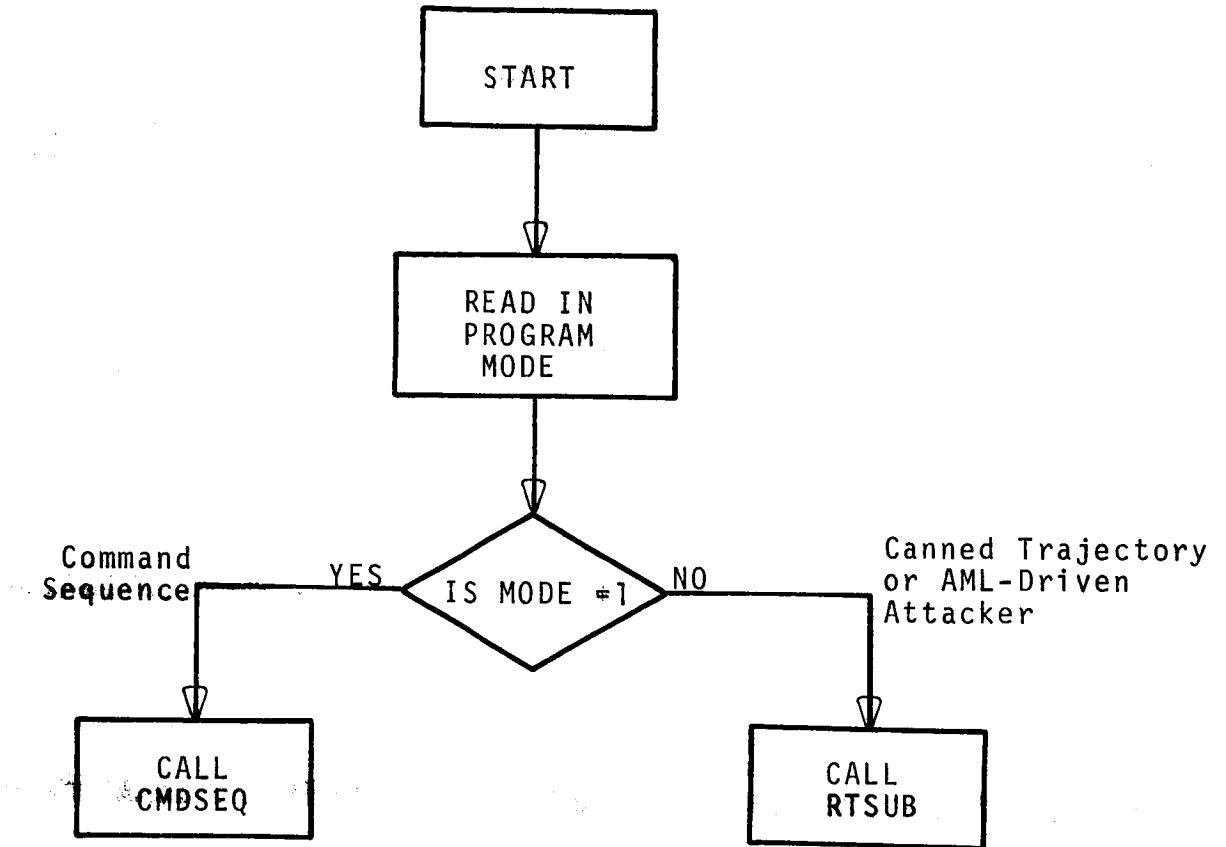
Figure 21.- Flowchart of Program AML75

$$C_y = C_{Y_\beta} \cdot \beta + \frac{b}{2V} (C_{Y_r} \cdot r + C_{Y_p} p)$$

$$+ C_{Y_{\delta a}} \cdot \delta a + C_{Y_{\delta sp}} \cdot \delta sp + C_{Y_{\delta R}} \cdot \delta R$$

$$C_z' = -(C_{L,b} + \frac{\bar{c}}{2V} C_{L_q} q + C_{L_{\delta a}} |\delta a|$$

$$+ C_{L_{\delta sp}} |\delta sp| + C_{L_{\delta s}} \delta s)$$

$$C_z = C_z' - \frac{\bar{c}}{2V} C_{L_{\dot{\alpha}}} \dot{\alpha}$$

with

$$\dot{\alpha} = uq - vp + \frac{g}{W} (T_z + \bar{q}S (C_x \sin\bar{\alpha} + C_z' \cos\bar{\alpha}) + gC_{33}$$

$$- \frac{\dot{u}w}{u} \quad / [u + \bar{q}Sg \frac{\bar{c}}{2V} C_{L_{\dot{\alpha}}} \cos\bar{\alpha} /W]$$

$$C_\ell = C_{\ell,b} + \frac{b}{2V} (C_{\ell_p} p + C_{\ell_r} r) + C_{\ell_{\delta a}} \delta a$$

$$+ C_{\ell_{\delta sp}} \delta sp + C_{\ell_{\delta R}} \delta R$$

$$C_m = C_{m,b} + \frac{\bar{c}}{2V} (C_{m_q} q + C_{m_{\dot{\alpha}}} \dot{\alpha} + C_{m_{\delta a}} |\delta a|$$

$$+ C_{m_{\delta sp}} |\delta sp| + C_{m_{\delta s}} \delta s$$

$$C_n = C_{n,b} + \frac{b}{2V} (C_{n_p} p + C_{n_r} r) + C_{n_{\delta a}} \delta a$$

$$+ C_{n_{\delta R}} \delta R + C_{n_{\delta sp}} \delta sp$$

82

## Subroutine AERFUN
### (Aerodynamic Function Interpolation)

Purpose.- To interpolate from the aerodynamic tables the stability and control derivatives and thrust.

Input.- Altitude, Mach number, angle of attack.

Output.- The following table summarizes the functions interpolated by the subroutine AERFUN.

| Function Symbol | Fortran Name | Independent Variables | Obtained from Tabular Data in Array | Interpolation Routine Used |
|---|---|---|---|---|
| $C_{L,b}$ | CLIFTB | M, $\alpha$, h | F101 | FUNC3 |
| $C_{L_{\delta sB}}$ | CLDLSB | M, $\alpha$ | F111 | FUNC2A |
| $C_{L_{\dot{\alpha}}}$ | CLADOT | M, h | F112 | FUNC2H |
| $C_{L_q}$ | CLQ | M, h | F113 | FUNC2H |
| $C_{L_{\delta s}}$ | CLDELS | M, $\alpha$, h | F114 | FUNC2H |
| $C_{L_{\delta a}}$ | CLDELA | M, h | F115 | FUNC2H |
| $C_{L_{\delta sp}}$ | CLDLSP | M | F116 | FUNC1 |
| $C_D$ | DCDLFT | M, $C_{L_{Trim}}$ | F210 | FUNC2D |
| $C_{D_{\delta SB}}$ | DLCDSB | M | F212 | FUNC1 |
| $C_{Y_\beta}$ | CYB | M, $\alpha$, h | F301 | FUNC3 |
| $C_{Y_p}$ | CYP | M, $\alpha$, h | F311 | FUNC3 |
| $C_{Y_r}$ | CYR | M, $\alpha$, h | F312 | FUNC3 |
| $C_{Y_{\delta a}}$ | CYDELA | M | F313 | FUNC1 |
| $C_{Y_{\delta sp}}$ | CYDLSP | M | F314 | FUNC1 |

| Function Symbol | Fortran Name | Independent Variables | Obtained from Tabular Data in Array | Interpolation Routine Used |
|---|---|---|---|---|
| $C_{Y_{\delta R}}$ | CYDELR | M, h | F315 | FUNC2H |
| $C_{\ell_\beta}$ | CLB | M, $\alpha$, h | F401 | FUNC3 |
| $C_{\ell_p}$ | CLP | M, $\alpha$, h | F411 | FUNC3 |
| $C_{\ell_r}$ | CLR | M, $\alpha$ | F412 | FUNC2A |
| $C_{\ell_{\delta a}}$ | CLRDLA | M, $\alpha$, h | F413 | FUNC3 |
| $C_{\ell_{\delta sp}}$ | CLRDSP | M, $\alpha$, h | F414 | FUNC3 |
| $C_{\ell_{\delta R}}$ | CLRDLR | M, $\alpha$, h | F415 | FUNC3 |
| $C_{m_{\delta sb}}$ | CMDLSB | M, $\alpha$ | F512 | FUNC2A |
| $C_{m_q}$ | CMQ | M | F513 | FUNC1 |
| $C_{m_{\dot\alpha}}$ | CMADOT | M, h | F514 | FUNC2H |
| $C_{m_{\delta s}}$ | CMDELS | M, $\alpha$, h | F515 | FUNC3 |
| $C_{m_{\delta a}}$ | CMDELA | M, $\alpha$, h | F516 | FUNC3 |
| $C_{m_{\delta sp}}$ | CMDLSP | M, $\alpha$ | F517 | FUNC2A |
| $C_{n_\beta}$ | CNB | M, $\alpha$, h | F601 | FUNC3 |
| $C_{n_p}$ | CNP | M, $\alpha$, h | F611 | FUNC3 |
| $C_{n_r}$ | CNR | M, $\alpha$, h | F612 | FUNC3 |
| $C_{n_{\delta sp}}$ | CNDLSP | M, $\alpha$ | F613 | FUNC2A |
| $C_{n_{\delta a}}$ | CNDELA | M, $\alpha$, h | F614 | FUNC3 |
| $C_{n_{\delta R}}$ | CNDELR | M, h | F615 | FUNC2H |

| Function<br>Symbol | Fortran<br>Name | Independent<br>Variables | Obtained from<br>Tabular Data<br>in Array | Interpolation<br>Routine<br>Used |
|---|---|---|---|---|
| T | THRUST | M, h | F811 or F831* | FUNC2H |

Method.- Tabular data are given for wing angle of attack. The subroutine obtains wing angle of attack by adding one degree to body angle of attack.

For the coefficient of drag, $C_{L_{Trim}}$ is

$$C_{L_{Trim}} = C_{L,b} + C_{L_{\delta s}} \delta s + C_{L_{\delta a}} |\delta a| + C_{L_{\delta sp}} |\delta sp|$$

## Subroutine AMLVS3 (AML Driver for DMS)

Purpose.- This subroutine will provide the replacement for AMLVS2 if and when the new equations of motion are programmed on the DMS. It is written in such a manner that the DMS subroutine RTMAIN should require no changes. RTMAIN will still call, as in the old version, AMLVS. With an argument 1 for initialization, 2 for resetting and 3 for performing one integration step during the operate cycle.

Input and Output.- The AMLVS3 program during initialization reads the aerodynamic tables used by the AML program and during operation transfers variables between the subroutine RTSUB (corresponds to RTMAIN on the DMS) and the AML subroutines.

Method.- A flow chart of the major operations performed by AMLVS3 is provided in Figure 22.

## Subroutine CMDSEQ (Command Sequence)

Purpose.- This is essentially the driver program used to exercise the nonlinear equations of motion and the control system if a sequence of maneuver commands is read in from data cards.

---

*Table 811 is used for calculation of afterburner thrust, Table F831 for idle thrust

85
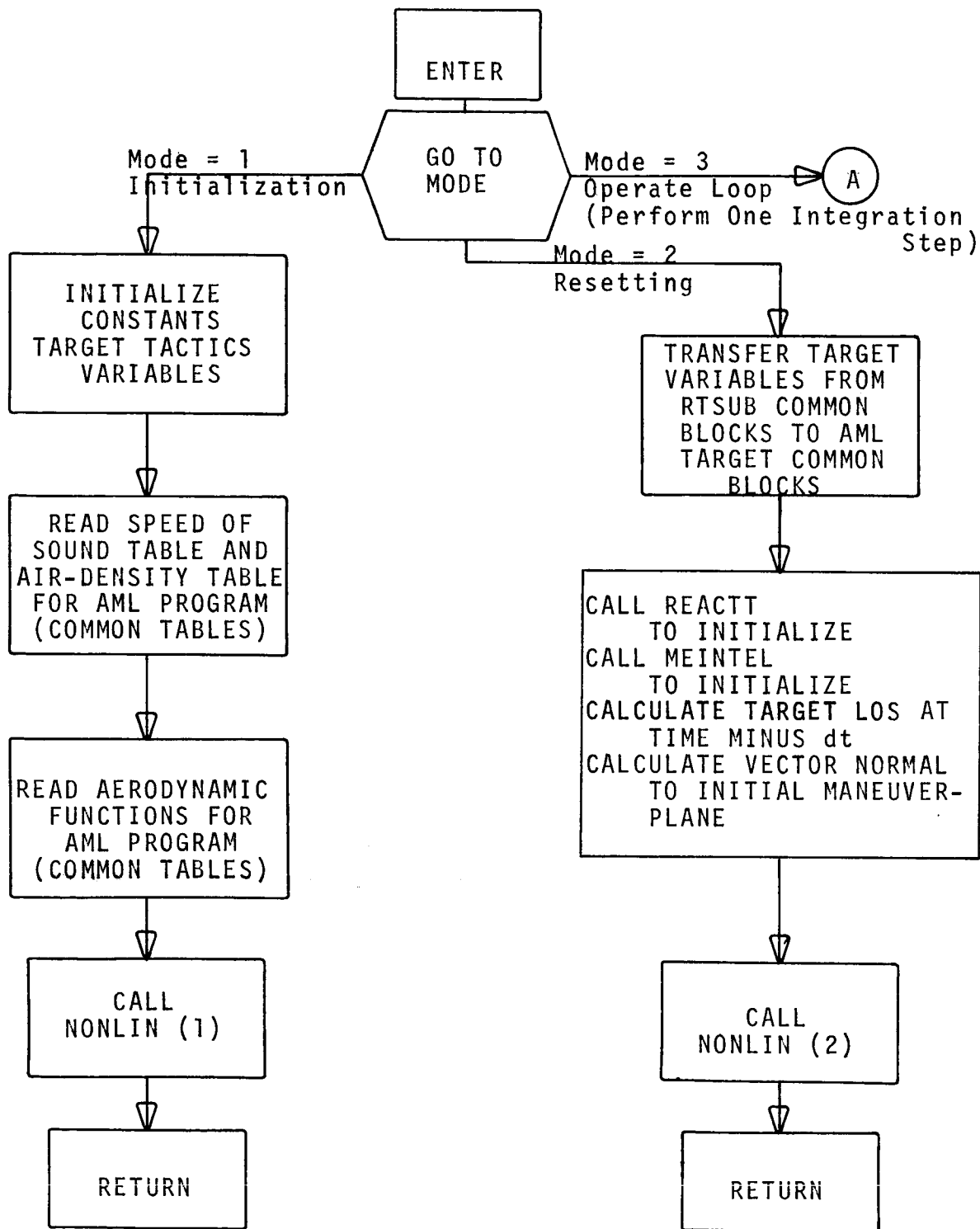
Figure 22.- Flowchart of Subroutine AMLVS3

```
                    ┌─────────────┐
                    │   OPERATE   │
                    │    LOOP     │
                    └─────────────┘
                           │
                           ▼
                          (A)
                           │
                           ▽
    ┌────────────────────────────────────────────────────┐
    │                                                    │
    │   TRANSFER ATTACKER VARIABLES FROM RTSUB COMMON     │
    │          BLOCKS TO AML COMMON BLOCKS                │
    │                                                    │
    └────────────────────────────────────────────────────┘
                           │
                           ▽
    ┌────────────────────────────────────────────────────┐
    │                                                    │
    │   TRANSFER TARGET VARIABLES FROM NONLIN COMMON      │
    │          BLOCKS TO AML COMMON BLOCKS                │
    │                                                    │
    └────────────────────────────────────────────────────┘
                           │
                           ▽
    ┌────────────────────────────────────────────────────┐
    │                                                    │
    │      CALCULATE PRESENT TARGET LOS ANGLE            │
    │      CALCULATE PRESENT RELATIVE GEOMETRY           │
    │                                                    │
    └────────────────────────────────────────────────────┘
                           │
                           ▽
```

| | |
|---|---|
| CALL NORPLN | TO OBTAIN NORMAL TO PRESENT MANEUVERPLANE |
| CALL REACTT | TO PERFORM TACTICAL MANEUVER SELECTION FOR TARGET |
| CALL GETCOM | TO TRANSLATE TARGET MANEUVER COMMAND |
| CALL RELGN | TO ESTABLISH PRESENT RELATIVE GEOMETRY |
| CALL TROTLT | TO OBTAIN TARGET THROTTLE POSITION |
| CALL STATET | TO GET TARGET TACTICAL STATE VECTOR |
| CALL MEINTL | TO UPDATE OFFENSIVE AND WEAPONS TIME |

```
                           │
                           ▽
                          (B)
```

Figure 22 (Continued)

To solve the nonlinear
equations of motion
for the target with
$\alpha$ and $\phi$ command and
thrust command as
set up above.

Ⓑ

```
┌───────────────────┐
│ ┌───────────────┐ │
│ │               │ │
│ │     CALL      │ │
│ │   NONLIN (3)  │ │
│ │               │ │
│ └───────────────┘ │
└───────────────────┘
```

```
┌───────────────────┐
│                   │
│      RETURN        │
│                   │
└───────────────────┘
```
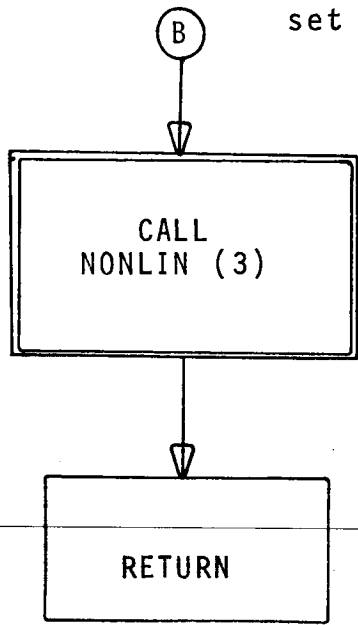
Figure 22. (Concluded)

Input and Output.- This subroutine reads all the necessary aerodynamic tables for the new equations of motion, the command sequence, initial conditions, run parameters and control system parameters. It prints trim conditions and, by calling NLPRNT, prints the time histories of the flight executing the commanded maneuvers. It also writes a magnetic tape containing state variables for subsequent plotting by the CALCOMP plotter.

Method.- Figure 23 shows in form of a flowchart the major operations performed by this subroutine.


## Subroutine CSRHOX
### (Speed of Sound, Air Density)

This is a replicate of the subroutine CSRHO of the AML program, the only difference lies in the storage allocation of the tables from which speed of sound and air density are interpolated. In the AML program, the two tables for these variables are in the common block "TABLES", where as in the new equations of motion routines there is a separate common block named "ATMOS". When the new equations are programmed to run on the DMS, the DMS functions for standard atmosphere should be used rather than reading separate tables for the AML program.


## Subroutine DIRCOS
### (Direction Cosines from Quaternions)

Purpose.- To calculate the direction cosine matrix and the Euler angles from the quaternions.

Input.- The current values of the target quaternions.

Output.- Target direction cosine matrix, Euler angles.

Method.- See write-up of subroutine Quatex and Oiler in Reference 3.

```
                    ┌─────────────┐
                    │    ENTRY    │
                    └─────────────┘
                           │
                           ▽
                 ┌──────────────────┐
                 │    INITIALIZE    │
                 │   CONSTRAINTS    │
                 │   AND AIRCRAFT   │
                 │    PARAMETERS    │
                 └──────────────────┘
                           │
                           ▽
              ┌──────────────────────────┐
              │  READ AERODYNAMIC TABLES │
              │    F-101 THROUGH F-831   │
              │     (COMMON TABLES)      │
              └──────────────────────────┘
                           │
                           ▽
              ┌──────────────────────────┐
              │   READ SPEED OF SOUND    │
              │     AND AIR DENSITY      │
              │  TABLES (COMMON ATMOS)   │
              └──────────────────────────┘
                           │
                           ▽
              ┌──────────────────────────┐
              │    READ AND PRINT        │
              │  COMMAND SEQUENCE        │
              │  FOR ALPHA AND PHI       │
              └──────────────────────────┘
```

```
   ( D )──────────────────────▷│
                               ▽
                    ┌──────────────────┐
                    │  READ AND PRINT  │
                    │    TITLE CARD    │
                    └──────────────────┘
                           │
                           ▽
                     ╱──────────╲                ┌──────────────────┐
                    ╱    DOES    ╲   YES          │  END FILE AND    │
                   ╱  TITLE CARD  ╲──────────▷    │     REWIND       │
                    ╲  SAY "END"? ╱               │  PLOTTING TAPES  │
                     ╲──────────╱                 └──────────────────┘
                           │ NO                           │
                           ▽                              ▽
              ┌──────────────────────────┐        ┌──────────────────┐
              │     READ AND PRINT       │        │      CALL        │
              │   INITIAL CONDITIONS     │        │      EXIT        │
              │  RUN CONTROL PARAMETER   │        └──────────────────┘
              │ CONTROL SYSTEM PARAMETER │
              └──────────────────────────┘
                           │
                           ▽
                         ( A )
```

Figure 23.- Flowchart of CMDSEQ

90
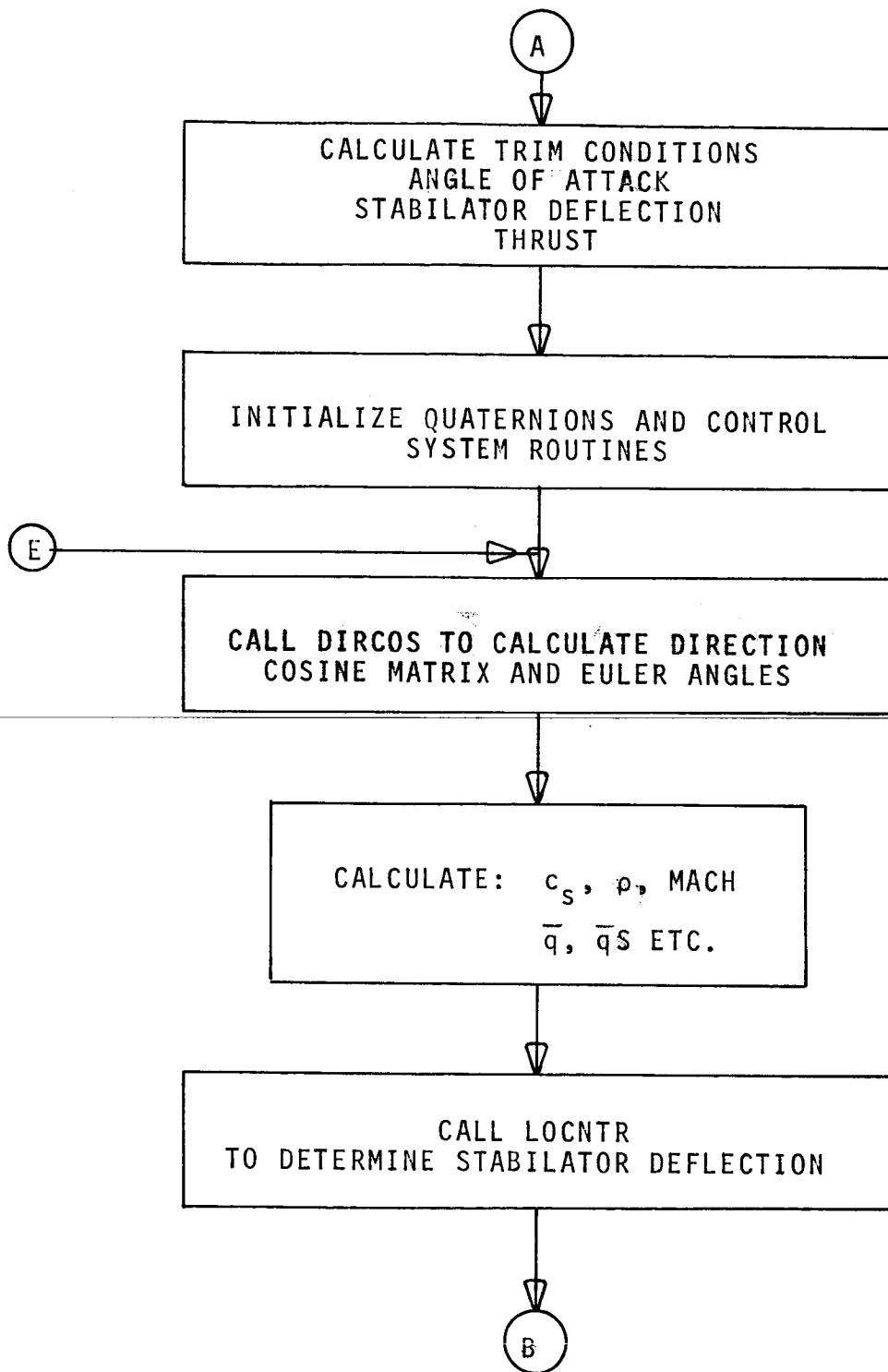
Figure 23.- (Continued)
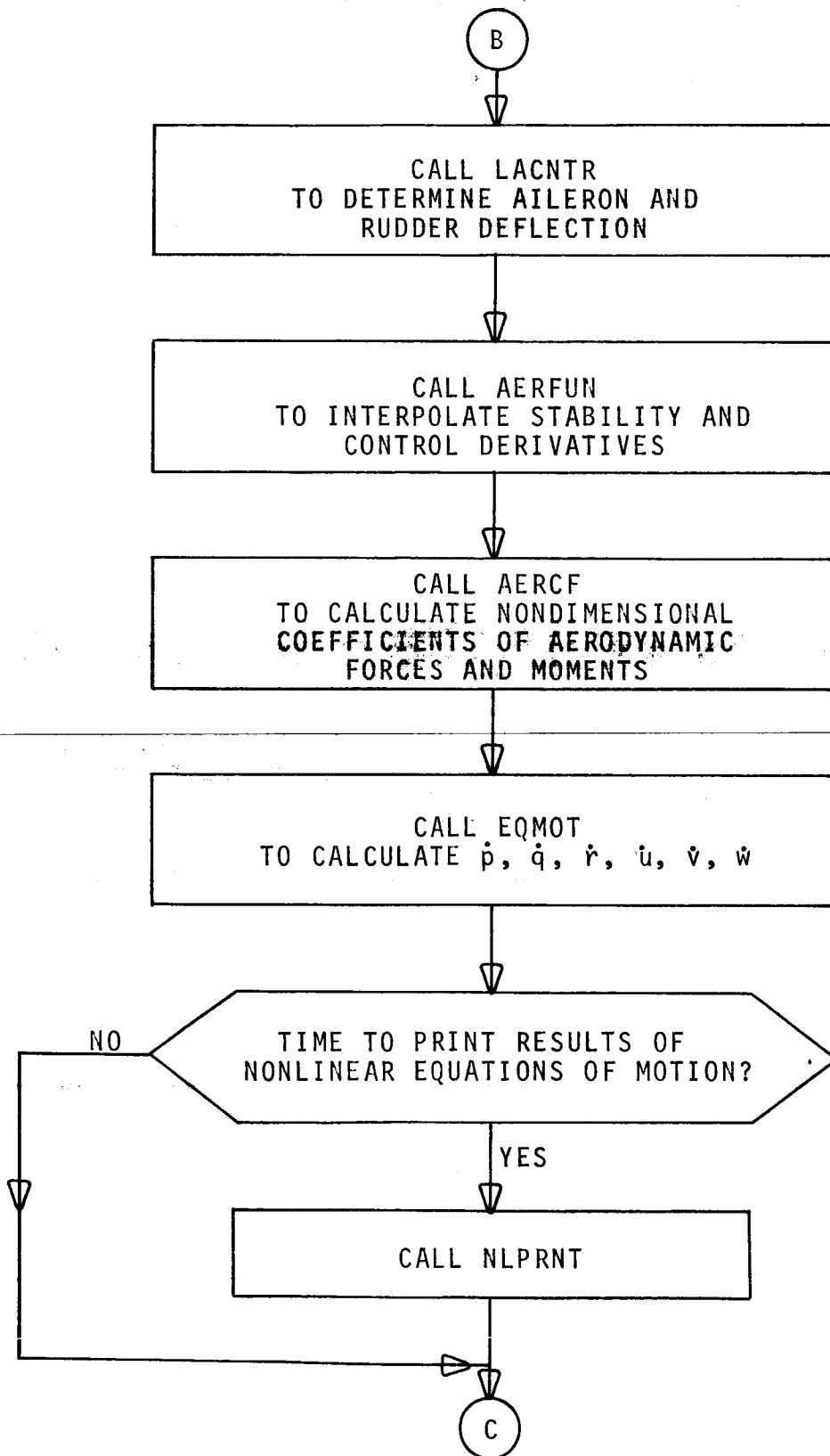
Figure 23  (Continued)
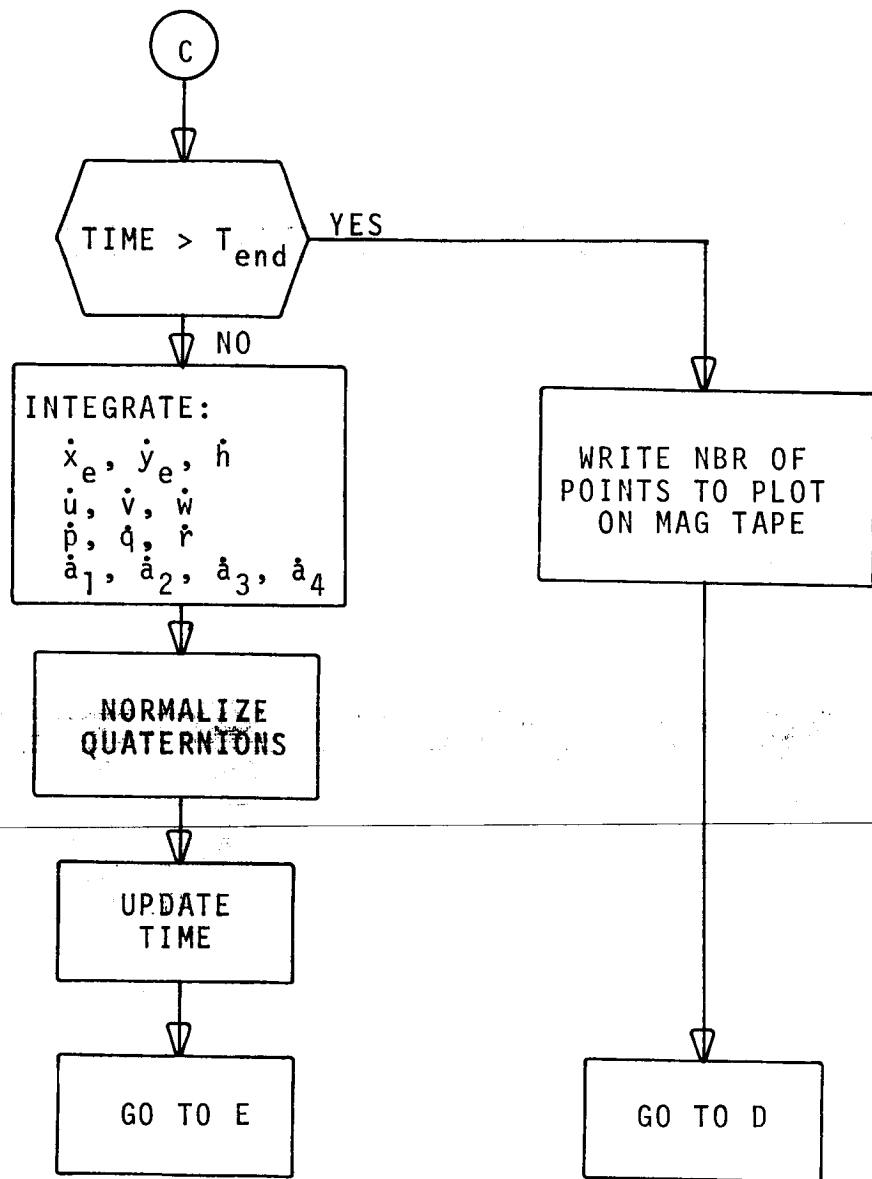
Figure 23 (Concluded)

Subroutine EQMOT
(New Equations of Motion)

Purpose.- To calculate the angular and linear accelerations.

Input.- Current values of u, v, w, p, q and r. $\bar{q}S$, the non-dimensional coefficients $C_X$, $C_Y$, $C_Z$, $C_L$ $C_M$, $C_N$.

Method.- Forces are first calculated in the stability axis system as

$$X_s = \bar{q} \ S \ C_X$$
$$Y_s = \bar{q} \ S \ C_Y$$
$$Z_s = \bar{q} \ S \ C_Z$$

and then transformed to the body axis system

$$X = X_s \cos_\alpha - Z_s \sin_\alpha$$
$$Y = Y_s$$
$$Z = X_s \sin_\alpha + Z_s \cos_\alpha$$

The moments are obtained directly as

$$L = \bar{q} \ S \ b \ C_\ell$$
$$M = \bar{q} \ S \ \bar{c} \ C_m$$
$$N = \bar{q} \ S \ b \ C_n$$

Next, $\dot{p}$, $\dot{q}$ and $\dot{r}$, as well as $\dot{u}$, $\dot{v}$ and $\dot{w}$, are calculated according to the equations derived in the section, "The General Six Degrees of Freedom Equations of Motion of a Rigid Body" in this report.

94

Subroutines FUNC..
(Interpolation Routines)

Purpose.- To interpolate aerodynamic functions.

Note.- In the above title, the two periods stand for the different number and letter combinations which make up the set of functions, FUNC1, FUNC2A, FUNC2D, FUNC2H, FUNC2X, FUNC3. The number indicates the number of independent variables. The letter, which is used in the subroutines with two independent variables, indicates what the two independent variables are:

A     Mach, Alpha
D     Mach, Coefficient of lift
H     Mach, Altitude
X     Mach, Altitude for thrust tables.

Input.- The values of the independent variables to be used in the interpolation. The function table to be used for interpolation.

Output.- The interpolated function value as a function of 1, 2 or 3 independent variables.

Method.- Linear interpolation is used in all routines. If the values of the independent variable fall outside the range for which the function is tabulated, the closest value in the table is used (no extrapolation). In FUNC2H, interpolation for altitude is only performed if h lies between 15,000 feet and 45,000 feet. For $h \leq 15{,}000$ feet, the function value at 15,000 feet is taken, for $h \geq 45{,}000$ feet, the function value at 45,000 feet is used.


Subroutine GETCOM (Get Command)

Purpose.- To translate the AML commands into commands for bank angle and angle of attack.

Input.- "New command" flag (ICMNWT), AML commanded load factor (as fraction of $g_{max}$ available), AML command maneuverplane rotation angle (ROTT).

Output.- Commanded bank angle (PHICMD), commanded angle of attack (ALFCMD).

Method.- First, the subroutine checks whether AML has commanded a new maneuver; if not, the old $\phi_{com}$ and $\alpha_{com}$ remain unchanged. If AML has issued a new command, the maximum available loadfactor (FLODMT) is calculated (by a call to AERF4). The actual loadfactor is then the product of desired load factor level and the maximum load factor available. The angle of attack may be calculated as follows:

$$L = W \cdot \text{load factor}$$

$$\alpha = \frac{L}{\overline{q}S} C_{L_\alpha}$$

The commanded bank angle is obtained from the AML commanded maneuverplane rotation angle by noting that the aircraft must be rotated by an angle $\phi*$ about the maneuverplane x-axis. This will create a component of the lift to compensate for the gravity force along the maneuverplane y-axis. The procedure for calculating $\phi*$ is explained in detail on page 19 and following of Reference 2.

Subroutine LACNTR (Lateral Control)

Purpose.- To determine aileron and rudder deflections.

Input.- Aircraft state variables, control system parameters.

Output.- Aileron deflection ($\delta a$) and rudder deflection ($\delta R$).

Method.- The functional relationships used are described in the earlier section, "The Lateral Control Systems".

## Subroutine LOCNTR
## (Longitudinal Control)

Purpose.- To determine stabilator deflection to achieve com-
manded angle of attack.

Input.- Aircraft state variables, control system parameters,
commanded angle of attack.

Output.- Stabilator deflection $\delta_s$.

Method.- Basically, the control system as described in the
section "The Longitudinal Control System" is implemented.

If $K_{P_\alpha}$ is zero, no longitudinal control is performed. Other-
wise, $\dot{\alpha}$ and $\dot{\phi}$ are computed first.

The following gains for $\overline{q} < 150$ lbs/sq. inch are set:

$$K_{P_\alpha} = 3 \qquad K_{I_\alpha} = 3 \qquad K_{R_\alpha} = 0.6$$

For dynamic pressure $\overline{q} > 520$ lbs/sq. inch, the following gains
are scheduled:

$$K_{P_\alpha} = 2 \qquad K_{I_\alpha} = 4 \qquad K_{R_\alpha} = 0.25$$

For dynamic pressures between 150 lbs/sq. inch and 520 lbs/sq.
inch, the gains as read in from the input card are used.

Next, the commanded angle of attack is reduced if the present
roll angle error is greater than 30 degrees and the roll rate ex-
ceeds 50 degrees/second.

Stabilator deflection according to Figure 12 is next calcu-
lated.

Finally, stabilator deflection is limited to -23 degrees and
+7 degrees.

## Subroutine NONLIN
### (Nonlinear Equations of Motion)

Purpose.- This subroutine is the driver for the routines per-
forming the solution of the nonlinear equations of motion in Mode 2
and Mode 3 of AML75.

Input and Output.- The aerodynamic data for the target air-
craft are read by this subroutine during initialization. During
the operate mode, the subroutine effects printout of some aircraft
key variables by calling NLPRNT, it also transfers target data
through the common block OPDAT1 to the calling program (AMLVS3).
It receives maneuver commands from AMLVS3 through common block
OPDAT2.

Method.- It is important to realize that the only communica-
tion between NONLIN and AMLVS3 is through the two common blocks
OPDAT1 and OPDAT2. Also note that NONLIN uses a different inte-
gration stepsize than the AML program. However, in both programs
the Fortran name for the integration step is DT; the mechanism, by
which these two different variables are kept separate is by having
one of them in common block K4, the other one in common block K5.
All the routines associated with the AML program contain common K4,
all routines associated with the solution of the nonlinear equa-
tions of motion contain common K5.

Otherwise, the subroutine NONLIN is very similar to the sub-
routine CMDSEQ, of which a flowchart is provided in Figure   .
The main differences between NONLIN and CMDSEQ are:  NONLIN is
called with one argument, which is 1 for initialization, 2 for re-
setting, and 3 for operation, whereas in CMDSEQ these three func-
tions are performed in sequence within CMDSEQ. NONLIN does not
read the command sequence data deck. NONLIN transfers aircraft
state variables to common OPDAT1. NONLIN picks up commands from
common OPDAT2.

98

## Subroutine NLPRNT
### (Print for Nonlinear Program)

This subroutine prints the following target variables whenever it is called by NONLIN or by CMDSEQ.

$$T \quad P \quad Q \quad R \quad V_{Tot} \quad \alpha \quad \beta \quad \delta a \quad \delta R \quad \delta S \quad \psi \quad \theta \quad \phi \quad g_{load} \quad h$$

All angles are printed in degrees, angular rates in degrees/second.

## Subroutine PAIRCR
### (Print Aircraft Data)

This subroutine corresponds to the subroutine PRNTAC of Reference 3 and produces printout as illustrated in Figure 11 of Reference 3. At the present time, not all variables printed by this subroutine may have been transferred to the corresponding common block from which PAIRCR selects the variables to print. (Example: P, Q and R for target, but these variables are printed in NLPRNT. The time interval between calls to PAIRCR is set to a fixed value of one second in subroutine RTSUB.

## Subroutine PCELL
### (Print Tactical (Cell) Data)

The printout generated by this routine allows for quick assessment of the relative tactical situation between the two aircraft. The printout is very similar to the one shown in Figure 12 of Reference 3, the annotations to questions 5 through 13 do not correspond to the actual questions analyzed in the routines STATEA and STATET.

## Subroutine RTSUB
### (Real Time Subroutine)

This subroutine simulates the functions performed on the DMS by the subroutine RTMAIN. For details on input, output and method, see flowchart of RTSUB, Figure 24.
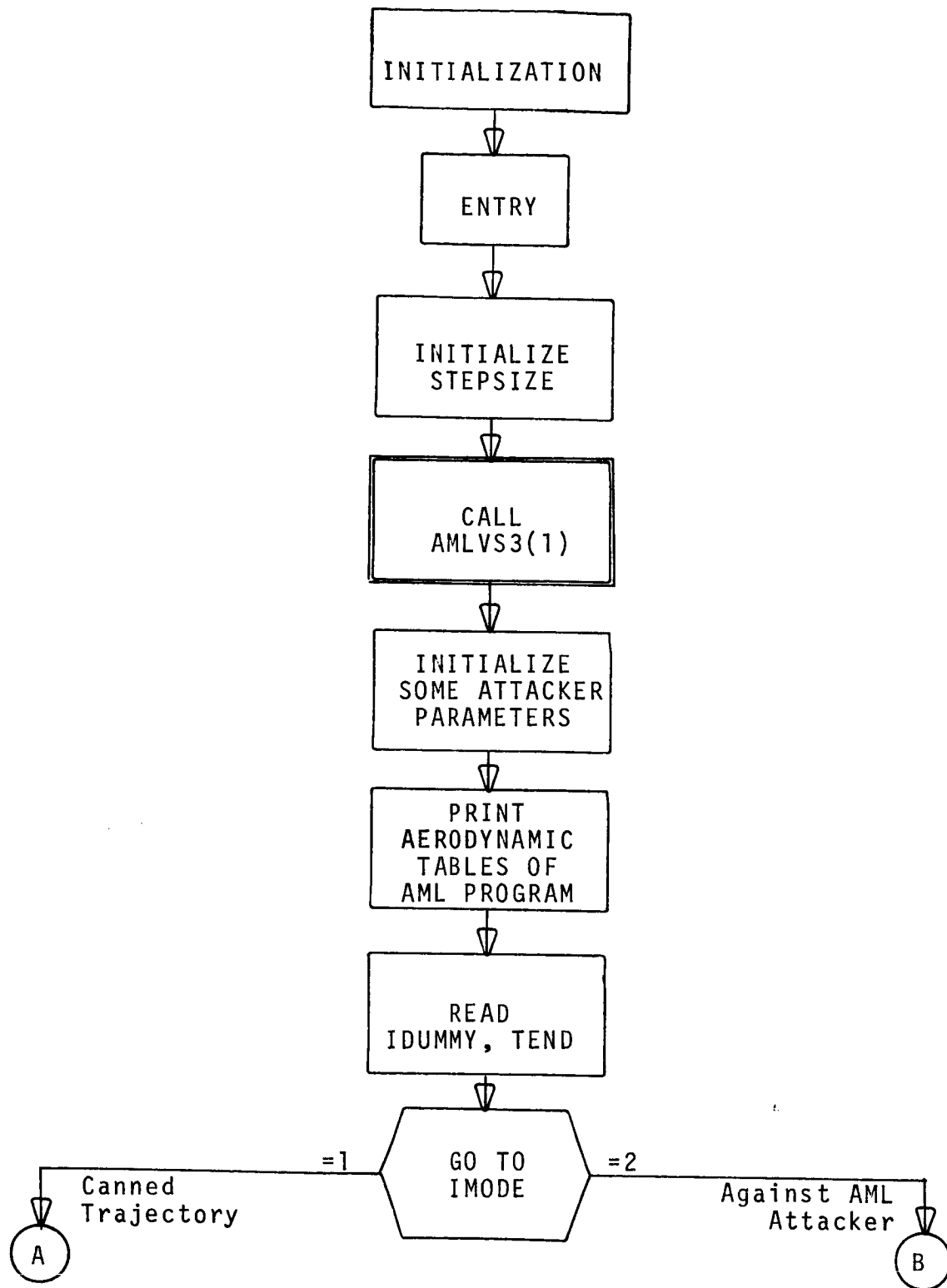
Figure 24.- Flowchart of RTSUB

```
         ┌A┐                ┌──────────┐              ┌B┐
         └─┘                │RESETTING │              └─┘
    ┌5┐   │ Canned          └──────────┘      ┌10┐     │ AML-Driven
    └─┘   ▽ Trajectory                        └──┘     ▽ Attacker
```

| | | |
|---|---|---|
| **READ ATTACKER TRAJECTORY TAPE FOR POSITIONING TO DESIRED START POINT** | | **READ CARD WITH TARGET INITIAL POSITION, VELOCITY AND ATTITUDE** |
| **READ CARD WITH TARGET INITIAL POSITION, VELOCITY AND ATTITUDE** | | **READ CARD WITH ATTACKER INITIAL POSITION, VELOCITY AND ATTITUDE** |
| **READ ONE RECORD FROM TAPE CONTAINING ATTACKER DATA** | | **CALCULATE ATTACKER DIRECTION COSINE MATRIX** |
| **CALCULATE ATTACKER INITIAL POSITION, VELOCITY AND ATTITUDE** | | **CALL AMLVS3(2)** |
| **BACKSPACE MAGTAPE BY ONE RECORD** | | **INITIALIZE EQMOTA AND REACTA** |

Go To Operate Loop

| | |
|---|---|
| **CALL AMLVS3(2)** | Go to Operate Loop → ◯ |

Figure 24 (Continued)

Figure 24 (Concluded)

Figure 24.- (Concluded)
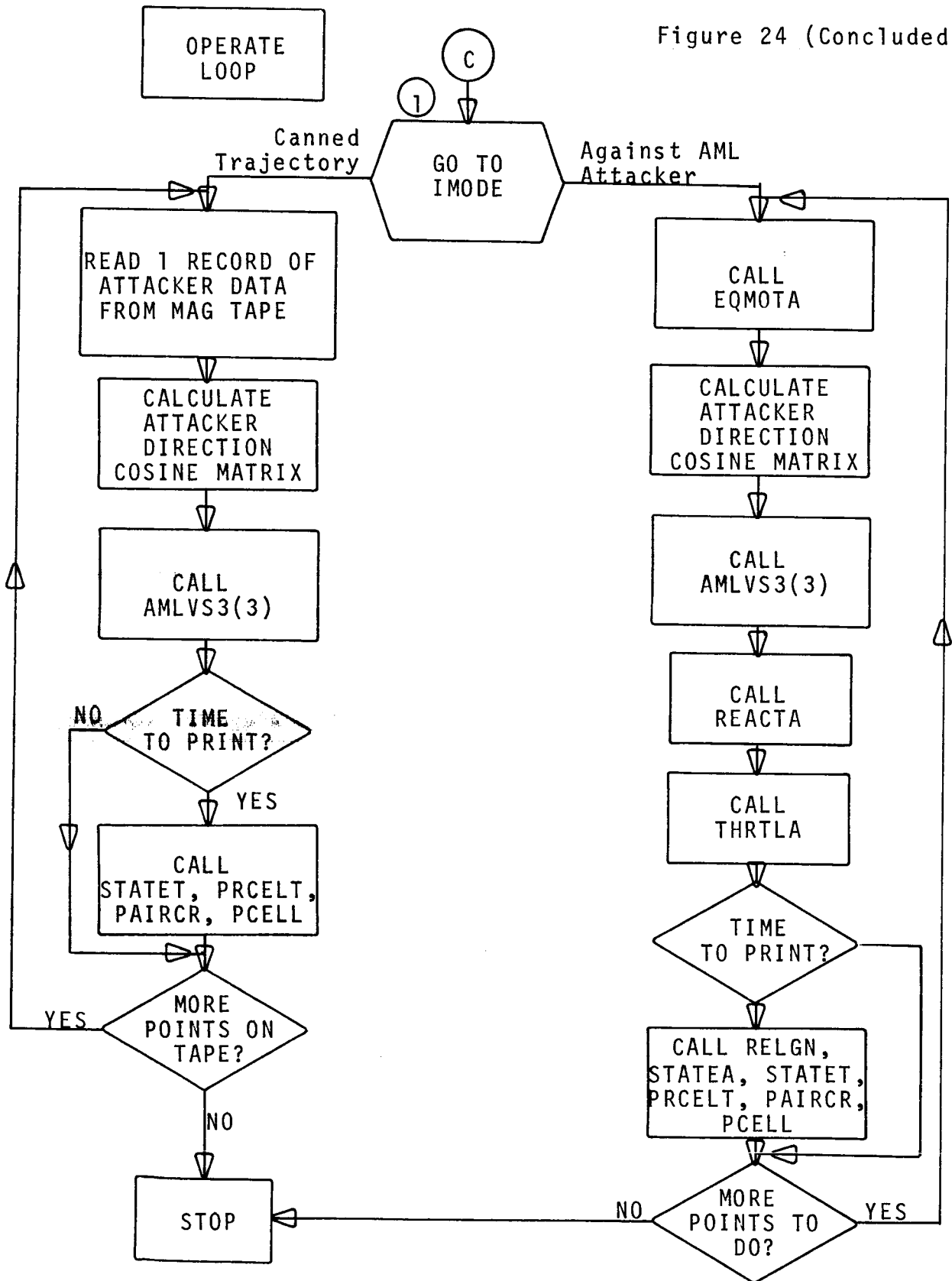
# INPUT DATA FOR AML75

The input data for this program depend on the mode in which the program is used.  For evaluation of command sequences (Mode 1) no input data associated with the AML program is required.  Figure 25 gives an overview of the input data deck for Mode 1.  In Modes 2 and 3, data for both the AML program and the new equations of motion are required.  The input data decks for Modes 2 and 3 are the same and are shown in Figure 26.

Certain blocks of data are used in different places and will therefore be described only once.

## Input Data for Mode 1 Operation

Program Mode.- First card of deck specifies program mode. This must be a 1 for Mode 1 operation.

<div align="center">

1 Card          FORMAT (I10)

</div>

Aerodynamic Functions for New Equations of Motion.- In all the following tables, unless specified otherwise, data listed as function of Mach number are given for the following 7 Mach numbers:

0.2, 0.4, 0.6, 0.8, 0.9, 1.0, and 1.1.

Functions tabulated as function of angle of attack are given for the following angle of attack values:

-5°, 0°, 5°, 10°, 15°, 20°, 25°, and 30°.

Functions tabulated as function of altitude are given for the two altitudes 15,000 feet and 45,000 feet.

(a) Basic coefficient of lift ($C_{L, basic}$, Function F101) is tabulated as function of Mach, angle of attack and altitude.

<div align="center">

16 Cards          FORMAT (7F10.7)

</div>

INPUT DATA TO AML75

Mode 1 (Command Sequence)



Figure 25.-Input Deck Setup for AML75 (Mode 1)

Read in
RTSUB

ATTACKER INITIAL
CONDITION      1 Card *

TARGET INITIAL
CONDITION      1 Card

TEND
1 Card

Read in
NONLIN (1)

LATERAL GAINS
1 Card

LONGITUDINAL GAINS
AND TEND
1 Card

26 Cards

SPEED OF SOUND AIR
DENSITY TABLES FOR
TARGET

356 Cards

AERODYNAMIC TABLES
FOR TARGET
(F-101 THROUGH F-831)

Read in
AMLVS3(1)

94 Cards

AERODYNAMIC DATA
FOR AML PROGRAM

*For Mode 2, this card
is not required

26 Cards

SPEED OF SOUND AIR
DENSITY TABLE FOR
AML PROGRAM

Read in
AML75

2 OR 3
PROGRAM MODE

1 Card

Figure 26.- Input Deck
Setup for AML75
(Modes 2 or 3)

(b) Coefficient of lift with respect to speedbrake deflection
($C_{L_{\delta sB}}$, Function F111) is listed as a function of Mach and altitude.

8 cards                    FORMAT (7F10.7)

(c) Coefficient of lift with respect to $\dot{\alpha}$ ($C_{L_{\dot{\alpha}}}$, Function F112) is listed as function of Mach and altitude.

2 cards                    FORMAT (7F10.7)

(d) Coefficient of lift with respect to q ($C_{L_q}$, Function F113) is listed as function of Mach and altitude.

2 cards                    FORMAT (7F10.7)

(e) Coefficient of lift with respect to stabilator deflection ($C_{L_{\delta s}}$, Function F114) is listed as function of Mach, angle of attack and altitude.

16 cards                   FORMAT (7F10.7)

(f) Coefficient of lift with respect to aileron deflection ($C_{L_{\delta a}}$, F115) is listed as function of Mach and altitude.

2 cards                    FORMAT (7F10.7)

(g) Coefficient of lift with respect to spoiler deflection ($C_{L_{\delta sp}}$, F116) is listed as function of Mach

1 card                     FORMAT (7F10.7)

(h) Coefficient of drag due to lift ($C_D$, F210) is listed as function of Mach, and coefficient of lift for the following 13 values of $C_L$:  0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 1.0, 1.2, 1.4, 1.6

13 cards                   FORMAT (7F10.7)

(i) Coefficient of drag due to speed brake deflection ($C_{D_{sb}}$, F212) is listed as function of Mach.

1 card                     FORMAT (7F10.7)

106

(j) Side-force derivative with respect to side slip angle ($C_{Y_\beta}$, F301) is listed as function of Mach, angle of attack and altitude.

    16 cards                    FORMAT (7F10.7)

(k) Side-force derivative with respect to roll rate ($C_{Y_p}$, F311) is listed as function of Mach, angle of attack, altitude.

    16 cards                    FORMAT (7F10.7)

(1) Side-force derivative with respect to yaw rate ($C_{Y_r}$, F312) is listed as function of Mach, angle of attack and altitude.

    16 cards                    FORMAT (7F10.7)

(m) Side-force derivative with respect to aileron deflection ($C_{Y_{\delta a}}$, F313) is listed as function of Mach.

    1 card                      FORMAT (7F10.7)

(n) Side-force derivative with respect to spoiler deflection ($C_{Y_{\delta sp}}$, F315) is listed as function of Mach and altitude.

    1 card                      FORMAT (7F10.7)

(o) Side-force derivative with respect to rudder deflection ($C_{Y_{\delta R}}$, F315) is listed as function of Mach and altitude.

    2 cards                     FORMAT (7F10.7)

(p) Rolling-moment derivative with respect to side-slip ($C_{\ell_\beta}$, F401) is listed as function of Mach, angle of attack and altitude.

    16 cards                    FORMAT (7F10.7)

(q) Rolling-moment derivative with respect to roll rate ($C_{\ell_p}$, F411) is listed as function of Mach, angle of attack and altitude.

    16 cards                    FORMAT (7F10.7)

(r)  Rolling-moment derivative with respect to yaw rate ($C_{\ell_r}$, F412) is listed as function of Mach and angle of attack.

        8 cards              FORMAT (7F10.7)

(s)  Rolling-moment derivative with respect to aileron deflection ($C_{\ell_{\delta a}}$, F413) is listed as function of Mach, angle of attack and altitude.

       16 cards             FORMAT (7F10.7)

(t)  Rolling-moment derivative with respect to spoiler deflection ($C_{\ell_{\delta sp}}$, F414) is listed as function of Mach, angle of attack and altitude.

       16 cards             FORMAT (7F10.7)

(u)  Rolling-moment derivative with respect to rudder deflection ($C_{\ell_{\delta R}}$, F415) is listed as function of Mach, angle of attack and altitude.

       16 cards             FORMAT (7F10.7)

(v)  Pitching-moment coefficient ($C_{m,b}$, F501) is listed as function of Mach, angle of attack and altitude

       16 cards             FORMAT (7F10.7)

(w)  Pitching-moment derivative with respect to speed brake deflection ($C_{m_{\delta sB}}$, F512) is listed as function of Mach and angle of attack.

        8 cards               FORMAT (7F10.7)

(x)  Pitching-moment derivative with respect to pitching rate ($C_{m_q}$, F513) is listed as function of Mach number.

        1 card               FORMAT (7F10.7)

(y)  Pitching-moment derivative with respect to $\dot{\alpha}$ ($C_{m_{\dot{\alpha}}}$, F514) is listed as function of Mach and altitude.

        2 cards              FORMAT (7F10.7)

(z)   Pitching-moment derivative with respect to stabilator deflection ($C_{m_{\delta s}}$, F515) is listed as function of Mach, angle of attack and altitude.

    16 cards          FORMAT (7F10.7)

(aa)  Pitching-moment derivative with respect to aileron deflection ($C_{m_{\delta a}}$, F516) is listed as function of Mach, angle of attack and altitude.

    16 cards          FORMAT (7F10.7)

(bb)  Pitching-moment derivative with respect to spoiler deflection ($C_{m_{\delta sp}}$, F517) is listed as function of Mach and angle of attack.

    8 cards           FORMAT (7F10.7)

(cc)  Yawing-moment derivative with respect to sideslip angle ($C_{n_{\beta}}$, F601) is listed as function of Mach, angle of attack and altitude.

    16 cards          FORMAT (7F10.7)

(dd)  Yawing-moment derivative with respect to roll rate ($C_{n_{p}}$, F611) is listed as function of Mach, angle of attack and altitude.

    16 cards          FORMAT (7F10.7)

(ee)  Yawing-moment derivative with respect to yaw rate ($C_{n_{r}}$, F612) is listed as function of Mach, angle of attack and altitude.

    16 cards          FORMAT (7F10.7)

(ff)  Yawing-moment derivative with respect to spoiler deflection ($C_{n_{\delta sp}}$, F613) is listed as function of Mach and angle of attack.

    8 cards           FORMAT (7F10.7)

(gg) <u>Yawing-moment derivative with respect to aileron deflection</u>
($C_{n_{\delta a}}$, F614) is listed as function of Mach, angle of attack and altitude.

               16 cards                FORMAT (7F10.7)

(hh) <u>Yawing-moment derivative with respect to rudder deflection</u>
($C_{n_{\delta R}}$, F614) is listed as function of Mach and altitude.

               2 cards                FORMAT (7F10.7)

(ii) <u>Afterburner thrust</u> (F811) is listed as function of Mach and **altitude for the following altitudes (sea level, 10k, 20k, 30k, 40k, 50k, 60k feet)**

               7 cards                FORMAT (7F10.7)

(jj) <u>Idle thrust</u> (F831) is listed as function of Mach and altitude for the same values of altitude as in (ii) above

               7 cards                FORMAT (7F10.7)

The entire data deck for the aerodynamic and propulsion data for the new equations of motion is composed of <u>356 cards.</u>

<u>Atmospheric data</u>.- The next block of data is read into the tables in common block ATMOS. It consists of

(a) <u>Speed of sound</u> in feet/sec is tabulated as function of altitude from sea level to 60,000 feet at 500 feet increments.

               12 cards                FORMAT (10F8.7

(b) <u>Air density</u> in slugs/feet$^3$ is tabulated for the same altitude points as (a) above.

               13 cards                FORMAT (10F8.7

The entire atmospheric data block contains 26 cards.

<u>Command sequence</u>.- The next data block contains the commanded values of bank angle and angle of attack. Values are listed in pairs of ($\phi_{com}$, $\alpha_{com}$), $\phi$ and $\alpha$ expressed in degrees, the commands are interpreted by the program as being one second apart. The

first pair of commands is executed at time = 0, the second pair at time = 1 second, and so forth. The program expects command input for 199 seconds, however, only command input from time zero until $T_{end}$ are actually used. The input cards are prepared as follows:

$$\underbrace{\phi_c \quad \alpha_c}_{Time=0} \qquad \underbrace{\phi_c \quad \alpha_c}_{Time=1 \text{ sec}} \qquad \underbrace{\phi_c \quad \alpha_c}_{Time=2 \text{ sec}} \qquad \underbrace{\phi_c \quad \alpha_c}_{Time=3 \text{ sec}}$$

              50 cards                           FORMAT (8F10.2)

Title card.- The next card contains a run title, which is printed at the beginning of the printout of the program.

              1 card                             FORMAT (72A1)

Initial conditions.- The next card contains the information required to establish the initial conditions for the run. The following variables are listed on this card:

| | |
|---|---|
| $x_e$ | inertial x-coordinate |
| $y_e$ | inertial y-coordinate |
| h | inertial altitude |
| $V_{Tot}$ | magnitude of initial velocity vector |
| $\psi$ | body axis yaw Euler angle |
| $\theta$ | body axis pitch Euler angle |
| $\phi$ | body axis roll Euler angle |
| $\alpha$ | initial angle of attack |
| $\beta$ | initial sideslip angle |

Note that the initial angle of attack and the initial sideslip angle may be changed when the run requests certain trim options. (See section "Use of Options for CMDSEQ".)

              1 card                             FORMAT (7F10.2, 2F5.2)

Initial control surface deflections.- Initial aileron, rudder and stabilator deflection are listed on this card. Note again

that, depending on trim options, these values may be changed by the program. The initial spoiler deflection is set by the program to 1.4333...times the initial aileron deflection. Input are in degrees.

<div align="center">1 card               FORMAT (3F10.2)</div>

Run parameters.- This card contains the following variables:

DT

> Integration stepsize.
> This is the integration stepsize for the solution of the nonlinear equations of motion and it is usually set to 1/64" = 0.015625.

$T_{end}$

> End time.
> Generally, command sequences should be provided for time equals zero up to $T_{end}$ minus one second.

KPRINT

> A variable controlling the time interval between calls to subroutine NLPRNT. This time interval is equal to KPRINT times DT.

IVAR(1)

> Angle of attack trim control variable.
> If IVAR(1) is set to 1, the initial angle $\alpha$ is set to the trim angle of attack for trimmed, level flight. If the input initial conditions are such that trimmed level flight is not possible, the program will print this fact and will set the angle of attack to 30.9 degrees.

IVAR(3)

> Stabilator trim control variable.
> If this variable is set to one, the program will set the initial stabilator deflection to the trim value and thus override the previously inputted initial stabilator deflection.

IVAR(4)

> If this variable is set to 1, the program will keep the thrust during the entire run equal to the initial trimmed value.

<div align="center">1 card               FORMAT (2F10.6, 4 I 10)</div>

112

Longitudinal gains.- This card reads the following variables and parameters:

VAR(2)       Commanded angle of attack.
                      This is a remnant of an earlier version of this program (before entire command sequences were read in) and is no longer used. The input in this field is ignored by the program.

VAR(3)       Longitudinal gain $K_{P_\alpha}$.

VAR(6)       Longitudinal gain $K_{I_\alpha}$.

VAR(10)      Longitudinal gain $K_{R_\alpha}$ ($\alpha$- rate gain)

These three gains are used by the program during flight conditions of low dynamic pressure ($\overline{q} \leq 840$ psi). The high dynamic pressure gains are determined in the subroutine LOCNTR.

        1 card                         FORMAT (4F10.3)

Lateral gains and control parameters.- The following parameters associated with the lateral control system are listed on this card:

VAR(4)       Commanded bank angle.
                      The same comment as for VAR(2) in the preceding card applies here.

VAR(5)       Roll position gain $K_{P_\phi}$.

VAR(7)       Roll integral gain $K_{I_\phi}$.

VAR(9)       Roll rate gain $K_{R_\phi}$.

VAR(8)       Beta position gain $K_{P_\beta}$.

GN(1)        Beta rate gain $K_{R_\beta}$.

GN(2)        Bang-bang position threshold.

GN(3)        Bang-bang rate threshold.

The gains listed here are used for low $\bar{q}$ flight conditions ($\bar{q}$ < 840 lbs/square inch), gains for high dynamic pressure are determined in subroutine LACNTR.

1 card                          FORMAT (8F10.3)


Use of Options for Mode 1

By choosing combinations of input options a number of different types of controlled or uncontrolled responses can be obtained from CMDSEQ.

The flight condition is established by altitude, absolute (HAB) and total velocity (VTOTAL). With the body-axis Euler angles $\psi$, $\theta$, and $\phi$ any aircraft attitude initial condition can be input. Initial angle of attack $\alpha$ (ALPDEG) and initial sideslip ange $\beta_0$ (BETZER) establish the orientation of the relative wind with respect to the body axes.

The second card sets the initial deflection of aileron $\delta_a$ (DLADEG), rudder $\delta_R$ (DLRDEG), and stabilator $\delta_s$ (DLSDEG). Referring to card 4, if VAR(3) = $K_{p_\delta}$ is set to zero the stabilator deflection $\delta_s$ will be left at the initial value during the run. Likewise, if VAR(5) = $K_{p_\phi}$ is set to zero, the aileron deflection $\delta_a$ and the rudder deflection $\delta_R$ will both be left at their initial settings. This provides a convenient way of making the aircraft fly at fixed control surface deflections, be they zero or nonzero.

The trim control integers of card 3 cause the program to calculate and set the corresponding variable initially at the trimmed level flight value. Thus, if IVAR(1) = $I_{trim_\alpha}$ is set to 1, the program will calculate the trim angle of attack and set $\alpha$(ALPDEG) initially at this value. Likewise, IVAR(3) = $I_{trim_{stab}}$ = 1 will result in setting the stabilator initially at the trim value and

114

$IVAR(4) = I_{trim_{thrust}}$ will set the thrust level at the trimmed value. The only way to keep the stabilator at the trimmed value is to set $VAR(3) = K_{p_\alpha} = 0$.

For an example, the following settings would result in uncontrolled flight starting at a trimmed condition:

$$IVAR(1) = IVAR(3) = IVAR(4) = 1$$

$$VAR(3) = VAR(5) = 0$$

Input values of ALPDEG from card 1 and DLSDEG from card 2 would have no effect.

However, the following settings would result in uncontrolled but initially trimmed flight for the longitudinal axis and controlled flight in the lateral axes.

$$IVAR(1) = IVAR(3) = IVAR(4) = 1$$

$$VAR(3) = 0, \; VAR(5) \neq 0.$$

The desired gains for roll control would be input via card 5.

### Input Data for Modes 2 and 3 Operation

A complete data deck setup for Mode 3 is shown in Figure 26 and Appendix B lists a complete set of input data required for Mode 3.

Program mode.- The first card contains the specification for the desired program mode. Mode 2 requests exercising the AML program against a canned trajectory, Mode 3 against an AML program using the old equations of motion.

1 card                                      FORMAT (I10)

Atmospheric model definition for old equations of motion.- The speed of sound and air density tables as explained in the corresponding section of the Mode 1 input data description.

26 cards                                    FORMAT (10F8.7)

Mach number and altitude definitions.- In the following tables listing functions used for the old equations of the AML program, data given as function of Mach number are listed for the following 10 values: 0.2, 0.5, 0.8, 0.9, 1.0, 1.1, 1.2, 1.5, 1.8 and 2.0. Functions listed as function of altitude are given for the following 5 altitudes: sea level, 15k, 30k, 45k and 55k feet.

Maximum permissible load factor.- The maximum permissible load factor in g's is listed as function of Mach and altitude.

     5 cards                        FORMAT (10F8.7)

Minimum allowable Mach number.- This variable is listed as function of altitude.

     1 card                         FORMAT (5F7.3)

Maximum allowable Mach number.- This variable is listed as function of altitude.

     1 card                         FORMAT (5F7.3)

Lift coefficient derivative with respect to angle of attack $(C_{L_\alpha})$.- This variable is listed as function of Mach number and altitude.

     5 cards                        FORMAT (10F7.3)

Military thrust as function of Mach number and altitude.- Thrust for one engine is listed as function of Mach number and altitude for the following 14 values of the Mach number: 0.2, 0.4, 0.6, 0.8, 0.9, 1.0, 1.1, 1.2, 1.4, 1.6, 1.8, 2.0, 2.2 and 2.4, and for the following 7 values of altitude: Sea level, 10k, 20k, 30k, 40k, 50k, 60k feet.

     14 cards                       FORMAT (7F10.0)

Afterburner thrust.- Afterburner thrust is listed as function of Mach and altitude for the same Mach numbers and altitudes as military thrust.

     14 cards                       FORMAT (7F10.0)

Idle thrust.- Idle thrust is listed as function of Mach and altitude for the same data points as military thrust.

<div align="center">14 cards              FORMAT (7F10.0)</div>

Coefficient of drag.- The coefficient of drag is listed as function of coefficient of lift and of Mach number. Drag coefficients are given for the following 16 values of $C_L$: 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3, 1.4 and 1.5.

<div align="center">30 cards              FORMAT (6F8.4/6F8.4/4F8.4)</div>

Dive recovery angle.- Maximum dive angles, in degrees, from which the aircraft can pull out without hitting the ground, are listed as functions of Mach number and altitude for the following 5 values of altitude: Sea level, 3k, 6k, 9k and 12k feet.

<div align="center">5 cards              FORMAT (10F7.3)</div>

Sustained load factor.- The load factor at which the aircraft can perform a turn when in afterburner without losing energy is listed as function of Mach and altitude.

<div align="center">5 cards              FORMAT (10F7.3)</div>

The entire aerodynamic and propulsion deck for the old equations of motion consists of 94 cards.

Aerodynamic tables for target.- These are the tables F101 through F831 as explained in detail in the section "Input Data for Mode 1 Operation".

<div align="center">356 cards              FORMAT (7F10.7)</div>

Atmospheric data.- This data check is identical to the one on atmospheric data documented above.

<div align="center">26 cards              FORMAT (10F8.7)</div>

Longitudinal gains and end time.- This card contains the following information:

<blockquote>VAR(2)          Commanded angle of attack (not used in this program.</blockquote>

VAR(3)          Proportional $\alpha$-feedback gain $K_{P_\alpha}$.

VAR(6)          Integral $\alpha$-feedback gain $K_{I_\alpha}$.

VAR(10)         Alphadot feedback gain $K_{R_{\dot\alpha}}$.

TEND            End time.

Notes:  1) The gains are valid between 150 lbs/sq. inch $\leq \bar{q} \leq$ 530 lbs/sq. inch gains below and above these limits are determined in subroutine LOCNTR.

2) There are two values of $T_{end}$ read by this program (the one read here should be deleted from the program). The smaller value of $T_{end}$ will always terminate the run.

<div align="center">1 card                    FORMAT (8F10.3)</div>

Lateral gains and control parameters.- This card is identical to the corresponding card in Mode 1, see documentation "Input data for Mode 1 Operation".

<div align="center">1 card                    FORMAT (8F10.3)</div>

End time.- This card contains the end time Tend in seconds.

<div align="center">1 card                    FORMAT (10X, F10.2)</div>

Target aircraft initial conditions.- The target aircraft in Modes 2 and 3 is the one modeled by the new equations of motion. This card lists the initial conditions as follows:

$x_e$           Target initial x coordinate in inertial coordinate system.

$y_e$           Target initial y coordinate in inertial coordinate system.

h               Target initial altitude in inertial coordinate system.

$\dot{x}_e$     Target initial $\dot{x}_e$ velocity component.

118

$\dot{y}_e$     Target initial $y_e$ velocity component

$\dot{h}$     Target initial altitude dot component

$\psi$     Target initial body Euler yaw angle in degrees.

$\theta$     Target initial body Euler pitch angle in degrees.

$\phi$     Target initial body Euler roll angle in degrees.

Note:  The subroutine NONLIN resets the target aircraft to trimmed level flight so that the input of $\theta$ and $\phi$ are actually ignored.  It is also advisable to input $\dot{h}$ as zero to avoid conflict with the trim condition enforced in NONLIN.

        1 card                FORMAT (8F10.2)

Attacker aircraft initial conditions.- This card is required only in Mode 3, since in Mode 2, the attacker initial condition is read from the magnetic tape containing attacker data.  The list of variables on this card is the same as the one on the preceding card for the target.

        1 card                FORMAT (8F10.2)

Summarized, the entire input data deck for Mode 2 contains 507 cards, for Mode 3 it contains 508 cards.

# PROGRAM ATDYN

## General Description

The program ATDYN was originally envisioned as a self-contained program which would evaluate and print out complete information on the attitude dynamics of the aircraft for a specified flight condition. Stability derivatives calculated from tabular data were used to find the coefficients of the linearized longitudinal and lateral equations of motion. These equations were then to be used to find the transfer functions needed in the respective control loops of the longitudinal and lateral axes. It was also deemed desirable to find the coefficients for the first-order differential equation form of these equations for use in state variable analysis and synthesis techniques.

All closed-loop linear control analysis capability was incorporated into other programs than ATDYN. However, since a very limited pure proportional gain closed-loop root option exists in ATDYN, it is documented here for the sake of completeness.

The first approach used for attitude dynamics was to solve algebraically for the coefficients of the polynomial for the desired characteristic equation and use the subroutine POLFAC to factor the resulting polynomial. This approach is exemplified in the subroutine LATPOL which calculates the roots of the lateral characteristic equation. This approach is direct, reliable, and computationally fast but requires tedious algebraic manipulation for each polynomial needed. This algebraic approach was used again in the subroutine ALFZER to find the numerator roots (zeros) of the transfer functions $\alpha(s)/\delta_a(s)$ and $\alpha(s)/\delta_r(s)$ for aileron and rudder input, respectively. To find all the transfer functions needed using this approach would have been extremely tedious. Also, problems with the subroutine POLFAC indicated a need for an alternative root-finding routine. Furthermore, a procedure easily adaptable to both open and closed-loop analysis was needed.

120

After some experimentation with eigenvalue routines the rele-
vant subroutines from the EISPACK, the eigenvalue subroutine pack-
age developed by Argonne National Laboratory were applied. These
routines provided a trustworthy eigenvalue procedure but finding
numerator dynamics of transfer functions was not convenient.
Shortly thereafter the subroutine package TRANS was made opera-
tional. TRANS was developed specifically for control analysis and
was therefore more convenient than any previously tried technique.
Thereafter, all dynamics analysis was done using TRANS, but the
previously developed routines were kept for comparison.

## Structure of Program ATDYN

The program ATDYN consists of a main program and 22 individual
subroutines. Of these, the subroutine CMTRX is documented in Ref-
erence 3, the following eight subroutines are documented in the
section "Description of the Individual Subroutines of AML75" in
this report: CSRHOX, AERFUN, FUNC1, FUNC2A, FUNC2D, FUNC2H,
FUNC2X and FUNC3. Figure 27 illustrates the calling hierarchy of
the program ATDYN. The function of the remaining subroutines is
briefly explained below.

Subroutine KS.- This subroutine computes the 19 coefficients
$K_1$ through $K_{19}$ used for the linearized longitudinal equations and
the 22 coefficients $K_{30}$ through $K_{51}$ used for the lateral linear-
ized equations. It also punches on cards the entire array of the
coefficients $K_1$ through $K_{51}$. (This deck is used as input to a
number of the linearized analysis programs.)

Subroutine ALFZER.- This is a specialized routine to calcu-
late the numerator polynomial coefficients for the following
transfer functions

$$\alpha(s)/\delta_s(s) \qquad \phi(s)/\delta_a(s) \qquad \phi(s)/\delta_R(s).$$

It then finds the zeros of these polynomials.
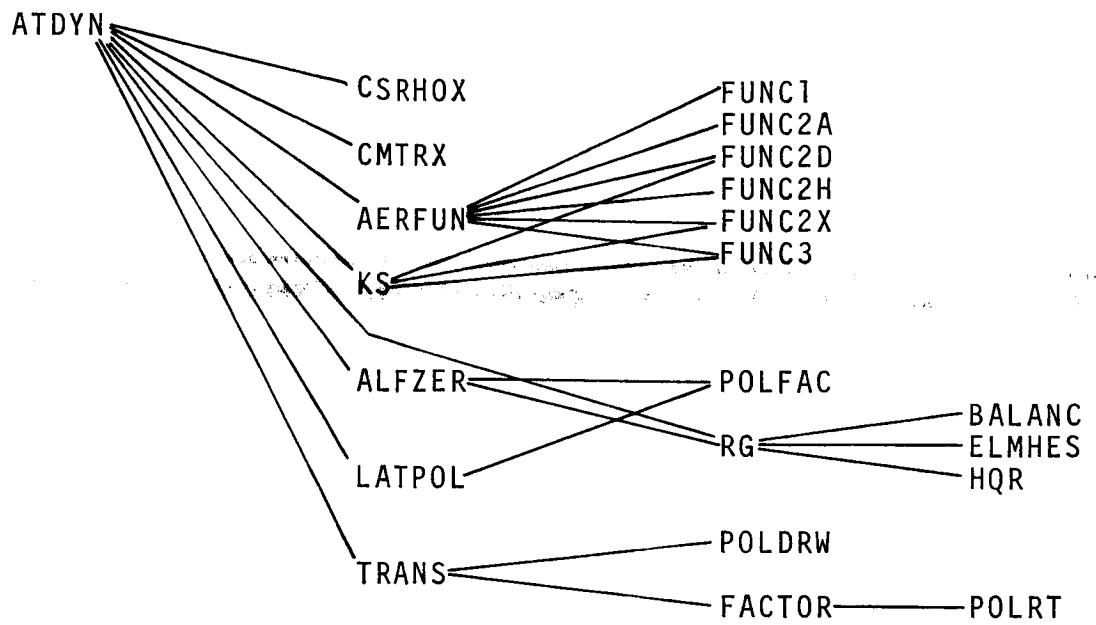
Figure 27.- Structure of Program ATDYN

Input to ALFZER consists of the K-array and a flag NCASE with the following function.

NCASE = 0       Calculates zeros of $\alpha(s)/\delta_s(s)$

NCASE = 1       Calculates zeros of $\phi(s)/\delta_a(s)$

NCASE = 2       Calculates zeros of $\phi(s)/\delta_R(s)$

These zeros are printed out in ALFZER and are not returned to the calling program.

Subroutine LATPOL.- This is a specialized subroutine which calculates the polynomial coefficients of the linearized lateral characteristic equation, and finds the corresponding roots. Input to LATPOL is the array of coefficients $K_i$, the roots are printed out by LATPOL but not returned to the calling program.

Subroutine POLFAC.- This is a general purpose polynomial factoring routine which sometimes fails to give the correct roots.

Subroutine FACTOR, POLDRW, POLRT.- These three routines perform the polynomial factorization for this subroutine TRANS.

Subroutines RG, BALANC, ELHES, HQR.- These are adaptations of subroutines from the Eigervalue package EISPACK, developed by Argonne National Labs and they are documented in detail in the writeup of EISPACK.

Subroutine TRANS.- This is a general-purpose subroutine for calculating transfer functions of linear systems directly from the Laplace transformed equations of motion of the system. The use of TRANS will be illustrated using the equations for the lateral axis as exemplified in LABEZR.

If the Laplace transforms of the lateral axis linearized equations are written in terms of the variables $\phi(s)$, $\psi(s)$, $\beta(s)$ with forcing functions $\delta_a(s)$ and $\delta_r(s)$ we obtain the following matrix:

| $\phi(s)$ | $\psi(s)$ | $\beta(s)$ | $\delta_a(s)$ | $\delta_r(s)$ |
|---|---|---|---|---|
| $(K_{30}S^2+K_{31}S)$ | $(K_{32}S^2+K_{33}S)$ | $K_{34}$ | $K_{35}$ | $K_{36}$ |
| $(K_{37}S^2+K_{38}S)$ | $(K_{39}S^2+K_{40}S)$ | $K_{41}$ | $K_{42}$ | $K_{43}$ |
| $(K_{44}S+K_{45})$ | $(K_{46}S+K_{47})$ | $(K_{48}S+K_{49})$ | $K_{50}$ | $K_{51}$ |

If these linear equations are solved according to Cramer's rule for the variable $\beta(s)$ for rudder input $\delta_r(s)$ we obtain

$$\beta(s) = \frac{D_\beta(s)\delta_r(s)}{D(s)} \quad \text{where}$$

$$D_\beta(s) = \begin{vmatrix} K_{30}S^2+K_{31}S & K_{32}S^2+K_{33}S & K_{36} \\ K_{37}S^2+K_{38}S & K_{39}S^2+K_{40}S & K_{43} \\ K_{44}S+K_{45} & K_{46}S+K_{47} & K_{51} \end{vmatrix}$$

i.e. the third column is replaced by the coefficients of desired forcing function and

$$D(s) = \begin{vmatrix} K_{30}S^2+K_{31}S & K_{32}S^2+K_{33}S & K_{34} \\ K_{37}S^2+K_{38}S & K_{39}S^2+K_{40}S & K_{41} \\ K_{44}S+K_{45} & K_{46}S+K_{47} & K_{48}S+K_{49} \end{vmatrix}$$

Thus, the transfer function $\beta(s)/\delta_r(s)$ would be given by the ratio of the polynomials $D_\beta(s)$ and $D(s)$.

To use TRANS to find these polynomials and their roots, we first must input the coefficients of the powers of s in the origi-

nal equations. This is done using the three-dimensional array C.
Thus, C(i, j, k) represents the element of the i-th row and j-th
column which is the coefficient of $s^{k-1}$. Starting with the zeroth
powers of s, we find:

$$C(1,3,1) = K_{34} \quad C(1,1,2) = K_{31} \quad C(1,1,3) = K_{30}$$

$$C(1,4,1) = K_{36} \quad C(1,2,2) = K_{33} \quad C(1,2,3) = K_{32}$$

$$C(2,3,1) = K_{41} \quad C(2,1,2) = K_{38} \quad C(2,1,3) = K_{37}$$

$$C(2,4,1) = K_{43} \quad C(2,2,2) = K_{40} \quad C(2,2,3) = K_{39}$$

$$C(3,1,1) = K_{45} \quad C(3,1,2) = K_{44}$$

$$C(3,2,1) = K_{47} \quad C(3,2,2) = K_{46}$$

$$C(3,3,1) = K_{49} \quad C(3,3,2) = K_{48}$$

$$C(3,4,1) = K_{51}$$

Since TRANS will use only one right-hand side, if we want $\delta_r(s)$
as the forcing function we must put the coefficients of $\delta_r(s)$ in
column 4 of the C matrix.

It would have been better if we could use integer zero to
represent $s^0$ but Fortran does not permit zero to be used as the
index of an array.

If the C array is correctly input the subroutine TRANS will
calculate properly the polynomial D(s).

To determine the numerator polynomial $D_\beta(s)$ we wish to replace
column 3 by column 4 ($\delta_r(s)$ coefficients) and evaluate the corres-
ponding determinant. This is done by means of the 5-element array
NMRTR. To obtain the numerator polynomial corresponding to the
third column we would let NMRTR = (3,0,0,0,0).

The call to TRANS would then be:
CALL TRANS(3,2,NMRTR)

where the first number, 3, is the size of the original matrix, the second number, 2, is the maximum power of s in the matrix, and NMRTR was described above.

## Input Data Deck Setup for Program ATDYN

Figure 27 illustrates an input data deck to the program ATDYN. The first two blocks of the data deck, consisting of the aerodynamic functions and the atmosphere data, are explained earlier in this report. For each individual case to be analyzed by ATDYN, a "RUN DATA" deck consisting of five cards is required. These five cards contain the following information:

<u>Card 1.</u>- Title card. The first 78 characters of this card will be printed at the beginning of the printout. If the title card contains END in columns 1 through 3, the computer run will be terminated.

<u>Card 2.</u>- Run options, FORMAT (4I.2). These option parameters are:

NOPT: Option number, may be an integer between 0 and 5, specifying the following options:

0 = longitudinal open loop roots only.

1 = longitudinal roots plus closed loop root with gain $K_{p_\alpha}$, specified on card 4.

2 = lateral open loop roots only.

3 = lateral open loop roots plus lateral closed loop root with gain $K_{p_\alpha}$, specified on card 5.

4 = open loop longitudinal and lateral roots.

5 = all options (0 through 4) combined.

IPTRAN: A parameter controlling polynomial printout in subroutine TRANS.

IPTRAN = 0: Part of polynomial printout will be suppressed.
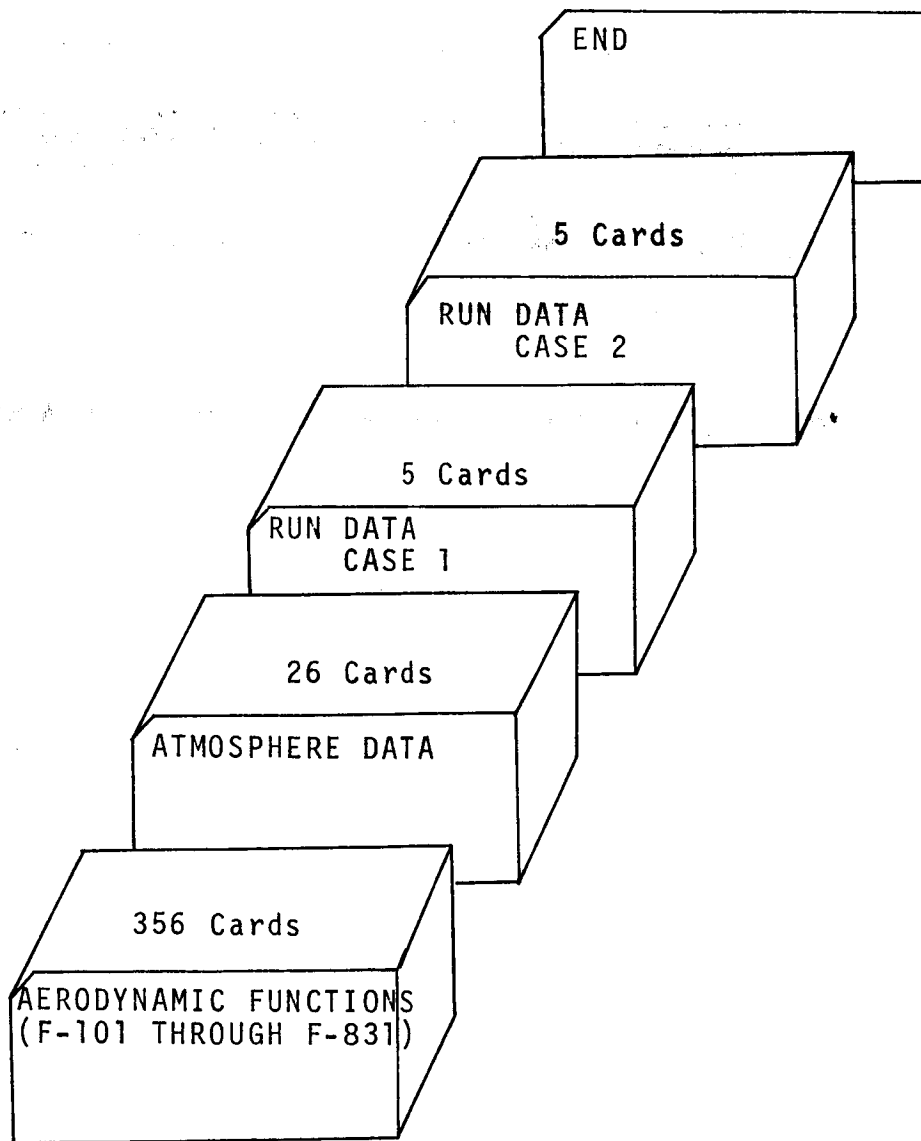
Figure 27- Input Deck Setup for Program ATDYN

IPTRAN = 1:   All polynomials will be printed.

JPTRAN:   A parameter controlling root printout in subroutine TRANS.

JPTRAN = 0:   Part of root printout will be suppressed.
JPTRAN = 1:   All roots will be printed.

KPTRAN:   A parameter controlling matrix printout in subroutine TRANS.

KPTRAN = 0:   printout of matrix elements will be suppressed.

KPTRAN = 1:   matrix elements will be printed.

The most frequently used options when running ATDYN for routine control analysis is:

NOPT=4      IPTRAN=0      JPTRAN=1      KPTRAN=0

Card 3.- Flight condition, FORMAT (8F10.2). The following variables are specified on this card.

| | |
|---|---|
| $x_e$ | Inertial x coordinate (not used by program) |
| $y_e$ | Inertial y coordinate (not used by program) |
| $h$ | Altitude |
| $V_{Tot}$ | Magnitude of velocity vector |
| $\psi$ | Euler yaw angle (not used by program) |
| $\theta$ | Euler pitch angle in degrees |
| $\phi$ | Euler roll angle in degrees |
| $\alpha$ | Body angle of attack in degrees |

Card 4.- $\alpha$-feedback gains. FORMAT (8F10.2). This card may be left blank unless option 1 or option 5 is used. In this case, eight values of gains $K_{p_\alpha}$ must be specified. The program will calculate the closed loop poles for these eight gains. However, since ATDYN has no provision for any other feedback loops, options 1 and 5 are of limited value.

Card 5.- $\phi$-feedback gains. FORMAT (6F10.4). This card may be left blank unless options 3 or 5 were selected. In this case,

128

six values of the gain $K_{p_\phi}$ must be specified for which the program will determine the closed loop poles. This option, because of the limit to one feedback loop, is of little value.

The user may analyze several flight conditions in a single computer run by stacking run data decks for several cases. Note that the run is terminated by the code word END in the title card.

## Output of Program ATDYN

The output of program ATDYN is best explained by using an example of an actual printout of ATDYN, as it is provided in Figure 28.

After printing out the initial conditions as specified by the input data to ATDYN, the aerodynamic coefficients as interpolated for this flight condition are printed.

Next, the array of the $K_i$ coefficients are printed. (See sections on linearized equation of motion for the definition of these coefficients.)

The options selected for running ATDYN are printed next (see description of input card 2).

The "longitudinal matrix" is the matrix $A_{long}$ as specified by the first order matrix differential equation

$$\dot{x} = A_{long} x + B_{long} \delta_s$$

with the x vector being ('u, $\theta$, $\dot{\theta}$, $\alpha$). This form assumes that the equations are known in first order form and Appendix A contains the formula for the coefficients of $A_{long}$ and $B_{long}$ as they are printed out here.

Next, the three zeros (roots of the numerator polynomial) of the transfer function $\alpha(s)/\delta_s(s)$ are printed. These zeros were obtained by use of the subroutine ALFZER and POLFAC which requires an explicit first-order expression for the transfer function. The lengthy algebraic calculations to obtain this transfer function in its desired form are not included in this report.

129

10 DECEMBER 1975    EXAMPLE RUN, STANDARD CONDS, H=15000, V=634.7

INITIAL CONDITIONS
SX =          0 SY =          0 ALT. =   15000.000 SPEED =    634.700
PSI =          0 THETA =    3.333 PHI =          .000000   0 ALPDEG =    3.33A
           13.402935  10601.477264    .000000               .028546    .015760

AERO COEFFICIENT BLOCK

CLIFTh =    .255905 CLDLSQ =              .0024 CLADOT =     .7100  CLD =          2.330000 CLDELS =    .006687
CLDELA =    .002279 CLDLSP =             -.000620 CLTRIM =    .2559 DCDLFT =      .0285 DLCDSH =       .023702
CYB =      -.011338 CYP =        .066992 CYR =   -.428556 CYDE-A =  -.000167 CYDSLP =        -.000060
CYDFLR =   -.001440 CLB =       -.001468 CLP =   -.327122 CLR =      .065303 CLRDLA =        .000461
CLRUSE =    .000142 CLRDLR =              .000197 CMBSIC =   -.007611 CMDLSB =   .0000 CMC =     -2.5600
CMADOT =  -1.049066 CMDELS =             -.009723 CMDELA =   -.000900 CMDLSP =   .000126 CAP =        .0018
CAP =      -.027112 CNR =      -.229263 CNDSLP =   -.00004A CNDELA =   -.000045 CNDELR =    -.000900
THRUST =  22304.794394

WEIGHT =  40872.000000 QBAR =    301.347522 RHO =          1057.360000
U =      633.623503 V =          0 W =           36.950601 HDOT =    .001495 CS =      634.700000
THETAW =          0 PSIW =          0                                .001495 CS =      634.700000
                                                                   -0 VMOR =

K ARRAY.

           LONGITUDINAL                    LATERAL

K( 1) =    5.048936            K(30) =    .004274
K( 2) =     .015760            K(31) =    .009895
K( 3) =          0             K(32) =   -.000494
K( 4) =    -.034842            K(33) =   -.001976
K( 5) =          0             K(34) =    .084103
K( 6) =     .255907            K(35) =    .036688
K( 7) =          0             K(36) =    .011262
K( 8) =     .514139            K(37) =    .000494
K( 9) =   -5.057908            K(38) =    .000820
K(10) =    3.327127            K(39) =    .023659
K(11) =   -5.019430            K(40) =    .006937
K(12) =          0             K(41) =   -.100663
K(13) =    -.383150            K(42) =    .001357
K(14) =     .000001            K(43) =   -.051566
K(15) =     .013522            K(44) =   -.002027
K(16) =     .232611            K(45) =    .255907
K(17) =     .049476            K(46) =   5.035969
K(18) =     .052352            K(47) =         -0
K(19) =    -.557079            K(48) =   5.048936
                              K(49) =   -.049641
                              K(50) =   -.014495
                              K(51) =    .082509

NOPT =    4    IPTRAN =    0    JPTRAN =    1    KPTRAN =    0

LONGITUDINAL MATRIX

130        -.00512144       -.05068535           0            .00690093
                   0             0       1.00000000              0
            .02776557             0        -.92511675      -4.52173613
           -.10166241             0         .99240439       -.65780690


Figure 28.- Sample Output of Program ATDYN

BLONG VECTOR

LONGITUDINAL ZEROS

| -146.169452 | 0 | -11.23892026 | -.07574863 |

| -.091543 | 0 |
| -.091543 | .071881 |
| | -.071881 |
| -0 | -0 |
| -0 | -0 |

LONGITUDINAL ROOTS

| 1 | -.792017 | 2.113374 |
| 2 | -.792017 | -2.113374 |
| 3 | -.001005 | .068956 |
| 4 | -.001005 | -.068956 |

IERR FROM EISPACK  0

LONG. WITH ALPHA/DELS ZEROS

POLYNOMIAL OF THE DETERMINANT STARTING WITH ZEROTH ORDER OF S.  COLUMN  2 REPLACED WITH COLUMN  4
-7.3004342E-02  -4.42641B0E-02  -1.4180970E 01  -9.5705790E-02

ROOTS OF POLYNOMIAL OF DEGREE N =  5

| REAL | IMAG | REAL | IMAG |
| -1.5432780E-03  -7.1881368E-02 | -1.5432780E-03  7.1981368E-02 | -1.4816945E 02 | 0 |

| FREQUENCY DAMPING | FREQ HZ | FREQUENCY DAMPING | FREQ HZ |
| .672  .0215 | -148.169  .0114 | -148.169 0 |

POLYNOMIAL OF THE DETERMINANT STARTING WITH ZEROTH ORDER OF S.  COLUMN  4 REPLACED WITH COLUMN  4
3.0007802E-02  2.2455418E-02  6.4456730E 00  2.0039135E 00  1.2634656E 00

ROOTS OF POLYNOMIAL OF DEGREE N =  4

| REAL | IMAG | REAL | IMAG |
| -1.0050980E-03  -6.8956383E-02 | -1.0050980E-03  6.8956383E-02 | -7.9201745E-01  -2.1133740E 00 |
| -7.9201745E-01  2.1133740E 00 | | |

| FREQUENCY DAMPING | FREQ HZ | FREQUENCY DAMPING | FREQ HZ |
| .729  .0140 | .0110 | 2.257  .3569  .3592 |

Figure 28 (Continued)

LATITUDINAL MATRIX

| 0 | 0 | 1.00000000 | 0 |
| 0 | 0 | -2.32545118 | .42943716 |
| 0 | 0 | 0 | 1.00000000 | 131 |
| 0 | -.08325242 | -.28422636 | -19.23379400 |
| .05068536 | .00040146 | -.99743179 | 3.85292069 | -.12866864 |

RLAT1 VECTOR

BLA12 VECTOR

|   |            |            | 5.0d067184 | 0 | .2470/695   | -.0028/084 |
|---|------------|------------|------------|---|-------------|------------|
|   |            | J          | 2.59349743 | 0 | -2.12055770 | .01634166  |

LATERAL ROOTS

| 1 | 0          |            |
|---|------------|------------|
| 2 | -2.5781504 | 0          |
| 3 | -.0167744  | 0          |
| 4 | -.0717963  | 2.1125110  |
| 5 | -.0717963  | -2.1125110 |

IERR FROM EISPACK   0

LAT. WITH PHI/DELA ZEROS

POLYNOMIAL OF THE DETERMINANT STARTING WITH ZEROTH ORDER OF S.   COLUMN   1 REPLACED WITH COLUMN   4
     0   2.0369918E-02   1.9917122E-03   4.6247846E-03

ROOTS OF POLYNOMIAL OF DEGREE N = 3
| REAL            | IMAG            | REAL | IMAG |
|-----------------|-----------------|------|------|
| -2.1533027E-01  | -2.0875652E 00  | 0    | 0    |
| -2.1533027E-01  | 2.0875652E 00   |      |      |

| FREQUENCY DAMPING | FREQ HZ | FREQUENCY DAMPING | FREQ HZ |
|-------------------|---------|-------------------|---------|
| 2.099   .1026     | .3340   | 0                 | 0       |

POLYNOMIAL OF THE DETERMINANT STARTING WITH ZEROTH ORDER OF S.   COLUMN   4 REPLACED WITH COLUMN   4
     0   9.8407377E-05   5.9078589E-03   2.4870321E-03   1.3946393E-03   5.0929982E-04

ROOTS OF POLYNOMIAL OF DEGREE N = 5
| REAL            | IMAG           | IMAG | REAL           | IMAG           |
|-----------------|----------------|------|----------------|----------------|
| -2.5781594E 00  | 0              | 0    | -7.1706306E-02 | -2.1125110E 00 |
| -7.1706306E-02  | 2.1125110E 00  | 0    |                |                |

| FREQUENCY DAMPING | FREQ HZ | FREQUENCY DAMPING | FREQ HZ |
|-------------------|---------|-------------------|---------|
| -2.578            | 0       | 2.114   .0339     | .3364   |
|                   | 0       |                   |         |
|                   | 0       |                   |         |

| PC 1 = | -.016774  | .006515 | 0        |
|--------|-----------|---------|----------|
| PC 2 = | -2.567792 | .001395 | 0        |
| PC 3 = | -.050047  | .002487 | 2.110127 |
| PC 4 = |           | .005904 |          |
| PC 5 = |           | .000024 |          |

LATERAL POLES FROM LATPOL

132

Figure 28 (Continued)

```
        -.068047              -2.110127
             -0                     -0


LATERAL ZEROS FOR AILERON INPUT        φ/δ_a

              0                      0

        -.215330              2.087565

        -.215330             -2.087565

             -0                    -0

             -0                    -0



LATERAL ZEROS FOR RUDDER INPUT        φ/δ_r

              0                      0

        -3.592133                    0

        3.692640                     0

             -0                    -0

             -0                    -0



END
```

Figure 28 (Concluded)

The longitudinal roots are printed next in the form:

Root Number      Real Part      Imaginary Part

are the eigenvalues of the longitudinal matrix printed out above. These roots were obtained by the subroutine EISPACK and the error message following these roots, when zero, indicates that there were no numerical difficulties encountered by the EISPACK routines in finding the eigenvalues of the matrix.

The remainder of the longitudinal dynamics printout was generated by the subroutine TRANS. A detailed description of how to interpret the printout of TRANS is provided in the section describing the program LATPOL. The matrices for subroutine TRANS were set up according to the linearized equations as listed in the section "Linearized Equations for the Longitudinal Direction" as follows:

| $\dot{u}$ | $\dot{\alpha}$ | $\theta$ | $\delta_s$ |
|---|---|---|---|
| $sK_1+K_2$ | $sK_3+K_4$ | $sK_5+K_6$ | $K_7$ |
| $K_8$ | $sK_9+K_{10}$ | $sK_{11}+K_{12}$ | $K_{13}$ |
| $K_{14}$ | $sK_{15}+K_{16}$ | $s^2K_{17}+sK_{18}$ | $K_{19}$ |

The numerator polynomial where column 2 is replaced by column 4 is therefore the numerator polynomial of the transfer function $\dot{\alpha}(s)/\delta_s(s)$.

The denominator polynomial (for the poles of the transfer functions) is obtained from the determinant of the matrix consisting of the three rows and the first three columns. (The printout annotates this with "Column 4 replaced by column 4").

Note that both the zeros and poles of the $\alpha(s)/\delta_s(s)$ transfer

134

function as calculated by the first order differential equations and by the subroutine TRANS are identical.

The program ATDYN then sets up the equations of motion for the lateral-directional axes in first order differential equation form

$$\dot{x} = A_{lat}\, x + B_{lat,1}\, \delta a + B_{lat,2}\, \delta R.$$

The x-vector is $(\phi,\ \dot{\phi},\ \psi,\ \dot{\psi},\ \beta)$. The derivation of the elements in the above matrices is too lengthy to be included in this report.

The matrix $A_{lat}$ and the vectors $B_{lat,1}$ and $B_{lat,2}$ are printed next, followed by the roots of the characteristic equation.

Next, the results of using subroutine TRANS are printed. The matrices were set up as follows (compare section "Linearized Equations for Lateral Direction"):

| $\phi$ | $\psi$ | $\beta$ | $\delta_a$ | $\delta_R$ |
|---|---|---|---|---|
| $s^2 K_{30} + s K_{31}$ | $s^2 K_{32} + s K_{33}$ | $K_{34}$ | $K_{35}$ | $K_{36}$ |
| $s^2 K_{37} + s K_{38}$ | $s^2 K_{39} + s K_{40}$ | $K_{41}$ | $K_{42}$ | $K_{43}$ |
| $s K_{44} + K_{45}$ | $s K_{46} + K_{47}$ | $s K_{48} + K_{49}$ | $K_{50}$ | $K_{51}$ |

The numerator zeros (column 1 replaced by column 4) are therefore the zeros of the transfer function $\phi(s)/\delta_a(s)$ while the roots of the characteristic equation are obtained by replacing column 4 with column 4.

Note again the perfect agreement in the values of the roots of the characteristic equations as obtained by EISPACK (as eigenvalues of the lateral matrix) and by TRANS (as zeros of the polynomial of the characteristic equation).

Finally, the program ATDYN prints out the results of the subroutine LATPOL. Remember, that LATPOL was written before the subroutine TRANS was available. First printed are the polynomial co-

135

efficients (polynomial of the characteristic equation). There is a slight difference in the coefficient of $s^5$ between TRANS and LATPOL which may be due to round-off errors. Also printed are the zeros of the transfer function $\phi(s)/\delta_a(s)$ (they agree exactly with those found by TRANS), and the zeros of the transfer function $\phi(s)/\delta_R(s)$.

In addition to the printed output of ATDYN, a deck of cards is punched containing the array of the coefficients $K_i$, i=1 to 51. The format for this punched output is 8F10.6, so the entire deck for the K-array consists of nine cards.

## Program LONLOP
## (Closed Loop Longitudinal Axis)

Purpose.- To obtain closed loop roots and zeros of the linearized longitudinal axis.

Input.- K-array Deck ($K_1$ through $K_{51}$)

       7 cards                    FORMAT (8F10.6)

      Title Card:

       1 card                     FORMAT (78A1)

      Longitudinal Gain Card:

This card contains one set of gains, that is

$$K_{P_\alpha}, \ K_{I_\alpha}, \ K_{R_\alpha}, \ \tau_{Lon}$$

       1 Card                    FORMAT (4F10.2)
      Lateral Gain Card:

This card is read by LONLOP, but any information on it is ignored. (Should be blank card.)

Run is terminated when title card contains END in columns 1, 2 and 3.

Output.- Printout of longitudinal closed loop roots and zeros as printed by subroutine TRANS and explained in writeup of ATDYN.

Method.- The program sets up the matrix of the closed loop longitudinal control loop as explained below:

For the purpose of stability analysis of the longitudinal control loop these equations were combined with the control equations portrayed in Figure 12 in a 6 x 7 matrix format suitable for analysis using the subroutine TRANS. This matrix takes the form

| | 1 $'u$ | 2 $\alpha$ | 3 $\theta$ | 4 $\delta_s$ | 5 $\delta_{sc}$ | 6 $\alpha_e$ | 7 $\alpha_c$ |
|---|---|---|---|---|---|---|---|
| 1 | $(K_1s+K_2)$ | $(K_3s+K_4)$ | $(K_5s+K_6)$ | $(-K_7)$ | $(0)$ | $(0)$ | $(0)$ |
| 2 | $(K_8)$ | $(K_9s+K_{10})$ | $(K_{11}s+K_{12})$ | $(-K_{13})$ | $(0)$ | $(0)$ | $(0)$ |
| 3 | $(K_{14})$ | $(K_{15}s+K_{16})$ | $(S_{17}s^2+K_{18}s)$ | $(-K_{19})$ | $(0)$ | $(0)$ | $(0)$ |
| 4 | $(0)$ | $(0)$ | $(0)$ | $(\tau_{Lon}s+1)$ | $(-1)$ | $(0)$ | $(0)$ |
| 5 | $(0)$ | $(0)$ | $(0)$ | $(0)$ | $(s)$ | $(K_{P_\alpha}s+K_{I_\alpha})$ | $(0)$ |
| 6 | $(0)$ | $(1+K_{P_\alpha}K_{R_\alpha}s)$ | $(0)$ | $(0)$ | $(0)$ | $(1)$ | $(1)$ |

The variable $\alpha_c$ is the commanded <u>perturbation</u> in angle of attack.

The first run should usually have blank cards for data. The characteristic equation with zero input gains should contain the aircraft longitudinal poles just as they were printed in ATDYN, and this serves as a check run.

## Program LABEZR
### (Open Loop Zeros and Poles $\beta(s)/\delta_R(s)$)

Purpose.- To calculate the open loop zeros and poles of $\beta(s)/\delta_R(s)$ for one specific flight condition.

<u>Input</u>.- K-array deck:  ($K_1$ through $K_{51}$)

        7 cards                           FORMAT (8F10.6)

      Title card:

        1 card                            FORMAT (78A1)

<u>Output</u>.- Open loop zeros and poles of $\beta(s)/\delta_R(s)$.

<u>Method</u>.- The matrix as given in the description of subroutine TRANS is set up and then TRANS is called.

<center>Program LATDYN (Closed $\beta$-Loop)</center>

<u>Purpose</u>.- To calculate poles and zeros of the lateral dynamics with only the $\beta$-loop closed.  This is used to vary $K_{P_\beta}$ and $K_{R_\beta}$ until satisfactory dutch roll characteristics are obtained.

<u>Input</u>.- K-array deck ($K_1$ through $K_{51}$):

        7 cards                           FORMAT (8F10.6)

      Title card:

        1 card                            FORMAT (8F10.6)

      Longitudinal gains

        1 card                            blank card

      Lateral gain $K_{P_\beta}$ and $K_{R_\beta}$

                                    FORMAT (30X, 2F10.2)

The runs are terminated if the title card contains END in columns 1 through 3.

    <u>Output</u>.- Closed loop poles and zeros of the transfer function $\phi(s)/\delta_a(s)$.

    <u>Method</u>.- The following matrix is set up for use by TRANS:

| $\phi$ | $\psi$ | $\beta$ | $\delta_a$ |
|---|---|---|---|
| $K_{30}s^2 + K_{31}s$ | $K_{32}s^2 + K_{33}s$ | $K_{34} + K_{36}K_{P_\beta} + K_{36}K_{P_\beta}K_{R_\beta}s$ | $K_{35}$ |
| $K_{37}s^2 + K_{38}s$ | $K_{39}s^2 + K_{40}s$ | $K_{43} + K_{43}K_{P_\beta} + K_{43}K_{P_\beta}K_{R_\beta}s$ | $K_{42}$ |
| $K_{44}s + K_{45}$ | $K_{46}s + K_{47}$ | $K_{49} + K_{51}K_{P_\beta} + (K_{51}K_{P_\beta}K_{R_\beta} + K_{48})s$ | $K_{50}$ |

The first run should use $K_{P_\beta} = K_{R_\beta} = 0$ for comparison of the poles with the poles generated by ATDYN.

## Program LATLOP (Closed Loop Lateral)

Purpose.- To calculate the closed loop poles and zeros of the transfer function $\phi(s)/\phi_c(s)$.

Input.- K-array deck.

|  |  |
|---|---|
| 7 cards | FORMAT (8F10.6) |
| Title card |  |
| 1 card | FORMAT (78A1) |
| Blank card |  |
| 1 card | blank |

Lateral gains $K_{P_\phi}$, $K_{I_\phi}$, $K_{R_\phi}$, $K_{P_\beta}$, $K_{R_\beta}$, $\tau_{Lat}$

|  |  |
|---|---|
| 1 card | FORMAT (6F10.2) |

The runs are terminated by END in columns 1, 2 and 3 of the title card.

Output.- Poles and zeros of the transfer function $\phi(s)/\phi_c(s)$.

Method.- The following matrix, reflecting the block diagram represented in Figure 29 is set up for analysis by the subroutine TRANS.

139

| $\phi$ | $\psi$ | $\beta$ | $\delta_a$ | $\delta_R$ | $\delta a_c$ | $\phi_E$ | $\phi_C$ |
|---|---|---|---|---|---|---|---|
| $K_{30}s^2+K_{31}s$ | $K_{32}s^2+K_{33}s$ | $K_{34}$ | $-K_{35}$ | $-K_{36}$ | $0$ | $0$ | $0$ |
| $K_{37}s^2+K_{38}s$ | $K_{39}s^2+K_{40}s$ | $K_{41}$ | $-K_{42}$ | $-K_{43}$ | $0$ | $0$ | $0$ |
| $K_{44}s+K_{45}$ | $K_{46}s+K_{47}$ | $K_{48}s+K_{49}$ | $-K_{50}$ | $-K_{51}$ | $0$ | $0$ | $0$ |
| $0$ | $0$ | $0$ | $.05\tau_{Lat}s^2+(0.05+\tau_{Lat})s+1$ | $0$ | $-1$ | $0$ | $0$ |
| $0$ | $0$ | $0$ | $0$ | $0$ | $-s$ | $K_{P_\phi}s+K_{I_\phi}$ | $0$ |
| $1+K_{P_\phi}K_{R_\phi}s$ | $0$ | $0$ | $0$ | $0$ | $0$ | $1$ | $1$ |
| $0$ | $0$ | $K_{P_\beta}+K_{P_\beta}K_{R_\beta}s$ | $0$ | $1$ | $0$ | $0$ | $0$ |

140
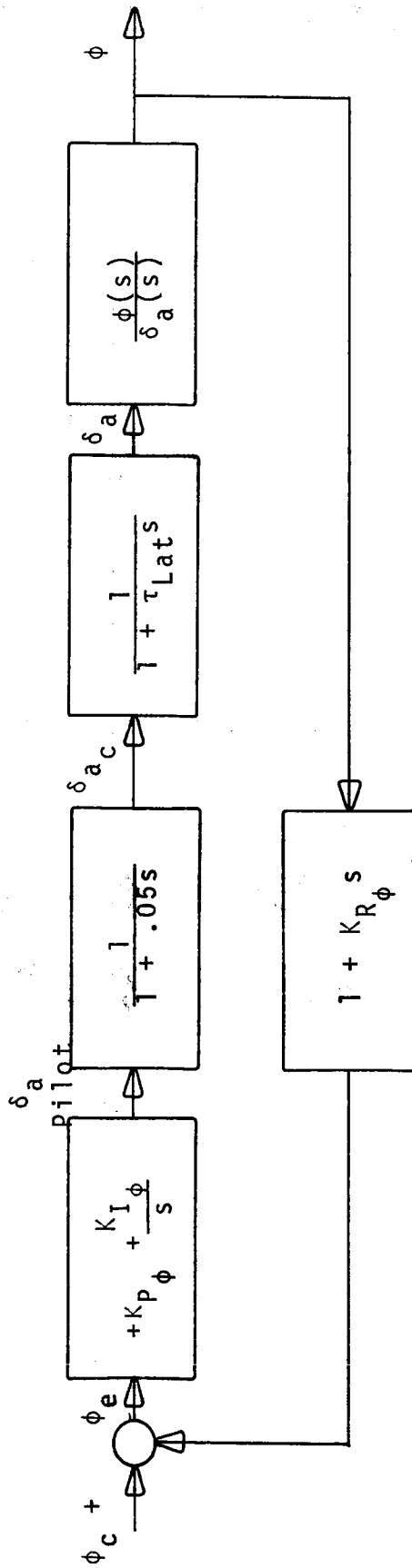
Figure 29.- Linear Roll Control Loop

141

Program LALINI
(Lateral Linear Integration)

Purpose.- To generate time histories of the linear, lateral, closed loop system. The aircraft dynamics are linear, but the nonlinear position and rate limit of the ailerons-spoilers are used.

Input.- The K-array deck.

$$7 \text{ cards} \qquad\qquad \text{FORMAT (8F10.6)}$$

Title card

$$1 \text{ card} \qquad\qquad \text{FORMAT (78A1)}$$

Blank card

$$1 \text{ card} \qquad\qquad \text{blank}$$

Lateral gains $K_{P_\phi}$, $K_{I_\phi}$, $K_{R_\phi}$, $K_{P_\beta}$, $K_{R_\beta}$, $\tau_{Lat}$

$$1 \text{ card} \qquad\qquad \text{FORMAT (6F10.2)}$$

Initial Conditions

$\psi$, $\phi$, $\dot{\psi}$, $\dot{\phi}$, $\beta$, $\phi_c$ (all in deg. and deg/sec)

$$1 \text{ card} \qquad\qquad \text{FORMAT (6F10.3)}$$

Run Parameters

$$DT, T_{end}, NPRNT$$

where

DT = integration stepsize

$T_{end}$ = end time

NPRNT= integration steps between printout

Output.- The following variables are printed:

Time $\dot{\phi}$ $\phi$ $\dot{\psi}$ $\psi$ $\phi_{err}$ $\delta_{a_p}$ $\delta_{a_c}$ $\delta_{a_{RL}}$ $\delta_a$ $\dot{\beta}$ $\beta$

(in degrees and degrees/second)

# INPUT-OUTPUT UNIT ASSIGNMENTS

The program AML75 requires the assignment of the following logical FORTRAN I/O units to physical units.

| FORTRAN I/O LOGICAL UNIT | PHYSICAL UNIT |
|---|---|
| 3 | Card Reader |
| 7 | Card Reader |
| 4 | Line Printer |
| 8 | Line Printer |
| 6 | Scratch Tape 1 |
| 9 | Scratch Tape 2 |

Fortran units 6 and 9 are used to create a data set used for subsequent plotting of command sequences and responses. These units may be bypassed at LRC. For running AML75 in mode 2 (AML versus canned trajectory), the additional I/O requirement exists:

| 11 | Magnetic Tape with recorded trajectory |
|---|---|

The program ATDYN requires the following assignments:

| 3 | Card Reader |
|---|---|
| 4 | Line Printer |
| 5 | Card Punch |

The individual small programs for the linear analysis require the following assignments:

| 3 | Card Reader |
|---|---|
| 4 | Line Printer |

# CONCLUSIONS

The work performed under this contract clearly indicates that it is possible to replace the performance model presently used in the AML program by an aircraft model of the same complexity as the one presently representing the aircraft flown by the human pilot on the DMS.

Limited experience with runs of the new equations of motion replacing the old performance model indicated, surprisingly, that the aircraft driven by the control system achieved higher cell values, that is, it was tactically superior to the old version.

The computational effort associated with the portion of the program performing the control system simulation is quite small compared to the computational effort to solve the nonlinear equations of motion with all its associated interpolation of aerodynamic coefficients.

While the longitudinal control system is of a relatively conventional design, some novel and interesting concepts have been incorporated into the lateral axis control system. Here, the combination of a bang-bang control system for large errors and/or error rates with a linear control system for small error and error rates represents a concept which deserves consideration for implementation in a real fighter aircraft.

144

# RECOMMENDATIONS FOR FURTHER STUDIES

## Implementation of the New
## Control System on the DMS

The following discussion points out some of the potential problems which might arise when the new AML program is implemented to run in real time on the DMS and suggests possible approaches for their solution.

Required memory storage may exceed available core storage if the AML program in its present form is combined with the DMS program. Required memory could be reduced drastically by the following modifications to the AML program: 1) Use tables defining the aircraft and atmosphere already existing in the DMS program. This would not only result in a reduction of core requirements, but it would also guarantee identical data for the DMS-driven and the AML-driven aircraft. 2) Use the same interpolation routines as the DMS program to obtain aerodynamic data.

A second and probably more serious problem might arise in the required amount of time to perform the calculations for the equations of motion for the target. Computational requirements for just one single pass through the full equations of motion are many times larger than they were for the old equations of motion. The calculation of all stability and control derivatives was not present in the old version. Furthermore, the calculation of the three moments L, M and N was not required by the old equations of motion and the three linear forces $F_x$, $F_y$ and $F_z$ were considerably simpler in the old version due to the assumption of no sideslip and the neglect of transient terms. To save computer time, the program is already set up to calculate certain stability and control derivatives only every fourth integration step. It is important to recognize that in order to run on the DMS in real time, those cycles which require the most computation time are the limiting factors. In the past, the limiting cycles were those during which

the AML program executed a tactical decision process. If now during those cycles already requiring the total available amount of computation time, the additional demand for the new equations of motion is made, not enough computer time will be available during those cycles. Several possible remedies for alleviating this difficulty can be considered. First, the programming of the AML tactical decision process could be improved by carefully sequencing the calculations in such a way that each variable is calculated only once during a cycle. Trigonometric functions might be calculated by table look up (many expressions involving trigonometric functions do not require the ten or better digit accuracy obtained by the standard library routines). Subroutine calls with very long argument lists could be replaced by more efficient calls, and several other programming "tricks" could be used, may be even replacing the FORTRAN coding of certain critical portions by more efficient routines written in assembly language.

Similar procedures could be used to speed up the solution of the equations of motion. The first step here should be to "stagger" the calculation of stability and control derivatives. Mach number and altitude within, say, one quarter of a second change not enough to warrant recalculation of all the terms which are functions of these two independent variables. Updating stability and control derivatives, a few of them during each integration cycle and may be none of them during the critical cycles where a tactical decision is made would spread out more evenly to requirements for computation time.

Should all the above proposed improvements still not make the AML program to execute fast enough, one might consider to use simplified equations of motion during the three or four time intervals during which the tactical decision is made. For instance, one might consider the aerodynamic moments as remaining constant over these few integration steps. Some analytical study could indicate how valid such an assumption would be.

All the above proposed measures for speeding up the execution time of the AML program did not assume any changes to the DMS program itself. But there is a good chance that additional time could be gained by improving the computational efficiency of the DMS program.

The final and ultimate step would be to use a second computer, working in parallel with the 6600 and to make this computer dedicated to solving the equations of motion for the AML program. A mini-computer of the size of a PDP-11 might be powerful enough to perform that task. Solving the equations of motion and simulating the control system on an analog computer interfaced to the CDC 6600 certainly would satisfy the computational requirements and relatively few signals would have to be transferred back and forth from the 6600. However, it is this author's personal opinion that such a hybrid solution should only be attempted if everything else fails.

Improved Performance Specifications

The authors of this report are convinced that the most significant progress in air-to-air combat in the coming years will not be made by developing new and improved tactics (we are talking still of only one-on-one engagements) but in developing better airframe control system combinations, especially for high angle of attack flight. A recent RFQ issued by the Air Force Systems Command (Ref. 10) entitled "Identification of Key Maneuver-Limiting Factors" clearly shows the trend to better understand the flight at high angles of attack. Associated with the factors degrading high angle of attack maneuverability is the problem of how to specify aircraft performance characteristics at high angle of attack. A key question here is, How valid is it to replace the aircraft by an equivalent second-order system under these conditions? The inherent strong nonlinearities affecting these flight conditions would appear to invalidate such an approach. How are handling qualities of an aircraft under these high angle of attack flight conditions best described? If it should turn out that fewer than six degrees

of freedom will not adequately describe high angle of attack flight, Mil Spec 8785 might simply no longer be applicable. Can a flight control system really alleviate these problems to such a degree that the aircraft then again could be described by a second order system?

It is quite obvious that a lot more questions exist than there are answers, and a new approach to study this important subject appears to be needed.

## Improved Control Strategies

The simultaneous change of angle of attack and bank angle received some limited consideration in this study and a relatively simple algorithm for angle of attack reduction during large roll maneuvers was developed, as described in the section of the longitudinal control for the nonlinear system. We have no proof that this algorithm is optimal, the only statement we can make is that it works. It even appears to work quite well, but considerably more effort than just engineering judgment should be put forth to refine this algorithm. It is certainly a key factor in executing simultaneous roll and pitch maneuvers in minimum time.

Since all the necessary computer programs to study this question are now available, it is strongly recommended that in the design of a similar control system for another aircraft than the F-4, this problem be studied thoroughly.

## Improvements to the AML Program

The AML program was developed specifically for an aircraft represented by a performance model. Certain restrictions on the aircraft motion were imposed, one of the most important ones that the aircraft fly in a maneuverplane. While the concept of the maneuverplane is without doubt of great value for setting up and for evaluating trial maneuvers; however, it seems that little benefit is obtained by postulating that the aircraft, between

148

tactical decisions, fly in a maneuverplane.

Other problems arising in the AML program show up drastically when flying with the new control system. The most pronounced of these difficulties occurs during pullups or dives through 90 degrees. Some ambiguities exist if the velocity vector has not yet made the transition through 90 degrees but the aircraft body axis, due to the angle of attack, already passed the singularity at 90 degrees.

It is quite likely that increased usage of the AML program with the new control system may reveal additional areas where the AML program could be improved.

Decision Science, Inc.
San Diego, California
5 February 1976

# REFERENCES

1.  Meintel, A. J., Jr. and Pennington, J. E.:  Differential Ma-
    neuvering Simulator.  Air-to-Air Combat Analysis and Sim-
    ulation Symposium, AFFDL-TR-72-57, Vol. I, U. S. Air Force,
    May 1972, pp. 187-210.

2.  Burgin, G. H.; Fogel, L. J.; and Phelps, J. P.:  An Adaptive
    Maneuvering Logic Computer Program for the Simulation of
    One-on-One Air-to-Air Combat.  Vol. I:  General Descrip-
    tion.  NASA CR-2582, September 1975.

3.  Burgin, G. H. and Owens, A. J.:  An Adaptive Maneuvering Logic
    Computer Program for the Simulation of One-on-One Air-to-
    Air Combat.  Vol. II:  Program Description.  NASA CR-2583,
    September 1975.

4.  Hankins, W. W.:  Interactive Computerized Air Combat Opponent.
    Proceedings of the AGARD FMP/GCP Joint Symposium on Flight
    Simulations/Guidance Systems Simulation.  The Hague,
    October 1975.

5.  Meintel, A. J.; Pennington, J. E.; and Hankins, W. W., III:
    Differential Maneuvering Simulator Validation.  NASA TM
    X-2827, October 1973.

6.  Whitaker, H. P. and Potter, J. E.:  Optimization of the Use of
    Automatic Flight Control Systems for Manned Aircraft.
    AGARD Conference Proceedings No. 17, September 1966.

7.  Blakelock, J. H.:  Automatic Control of Aircraft and Missiles.
    John Wiley and Sons, Inc., New York, 1965.

8.  Chalk, C. R.; et al:  Background Information and User Guide for
    MIL-F-8785(ASG), "Military Specifications - Flying Quali-
    ties of Piloted Airplanes."  AFFDL-TR 69-72, August 1969.

9.  Athans, M. and Falb, P. L.:  Optimal Control.  McGraw-Hill,
    New York, 1966. p. 507 ff.

10. Anonymous: Identification of Key Maneuver-Limiting Factors. Air Force Systems Command, RFQ F33615-76-R-3072, January 1976.

11. Phillips, W. H.: Effect of Steady Rolling on Longitudinal and Directional Stability. NASA TN 1627, 1948

## TABLE I
### LIST OF SUBROUTINES USED BY AML75

| | | | | |
|---|---|---|---|---|
| 1. | AERCF | | 25. | LACNTR |
| 2. | AERF4 | | 26. | LOCNTR |
| 3. | AERFUN | | 27. | MEINTL |
| 4. | AMLVS3 | | 28. | NLPRNT |
| 5. | CLOSS | | 29. | NONLIN |
| 6. | CMDSEQ | | 30. | NORPLN |
| 7. | CMTRX | | 31. | PAIRCR |
| 8. | CSRHO | | 32. | PCELL |
| 9. | CSRHOX | | 33. | PRCELA |
| 10. | DEBGCT | | 34. | PRCELT |
| 11. | DIRCOS | | 35. | PRESR |
| 12. | EQMOTA | | 36. | PRTF4 |
| 13. | EQMOT | | 37. | PRTNEW |
| 14. | ERMSG | | 38. | REACTA |
| 15. | EXTRA | | 39. | REACTT |
| 16. | EXTRT | | 40. | RELGN |
| 17. | FUNC1 | | 41. | RTSUB |
| 18. | FUNC2A | | 42. | STATEA |
| 19. | FUNC2D | | 43. | STATET |
| 20. | FUNC2H | | 44. | THRTLA |
| 21. | FUNC2X | | 45. | TROTLT |
| 22. | FUNC3 | | 46. | TRYNXA |
| 23. | GETCOM | | 47. | TRYNXT |
| 24. | GETRXN | | | |

## TABLE II
## LIST OF PROGRAMS AND SUBROUTINES FOR
## LINEAR ANALYSIS


1. Program ATDYN    uses the following subroutines not yet listed
under AML75

2. ALFZER
3. BALANC
4. ELMHES
5. FACTOR
6. HQR
7. KS
8. LATPOL
9. POLDRW
10. POLFAC
11. POLRT
12. RG
13. TRANS


The following five main programs require no subroutines not listed
earlier:

14. LONLOP
15. LABEZR
16. LATDYN
17. LATLOP
18. LALINI

## APPENDIX A

### Transformations of the Equations of Motion

Linear differential equations may be transformed in many different ways for the purposes of control analysis. In the cases that follow the transformations made computation more efficient and enabled finding the numerator dynamics or zeros of the transfer function of interest.

The differential equations of motion for the longitudinal or pitch axis, equations (1), contain first and second derivatives of the four variables involved. For numerical integration and state variable methods of analysis it is desirable to uncouple the derivatives resulting in one differential equation of first order for each variable present.

The first-order differential equation form of the equations pitch of motion may be obtained by algebraic elimination of coupled derivative terms from the linearized pitch dynamics, equations (1). If a state vector

$$x_1 = u$$

$$x_2 = \theta$$

$$x_3 = \dot{\theta}$$

$$x_4 = \alpha$$

is defined, equations (1) assume the form

$$K_1\dot{x}_1 + K_2x_1 + K_3\dot{x}_4 + K_4x_4 + K_5x_3 + K_6x_2 = K_7\delta_s \qquad \text{(x-axis)}$$

$$K_8x_1 + K_9\dot{x}_4 + K_{10}x_4 + K_{11}x_3 + K_{12}x_2 = K_{13}\delta_s \qquad \text{(z-axis)}$$

$$K_{14}x_1 + K_{15}\dot{x}_4 + K_{16}x_4 + K_{17}\dot{x}_3 + K_{18}x_3 = K_{19}\delta_s \qquad \text{(pitch)}$$

The z-axis equation may be solved for $\dot{x}_4$ and the result used to eliminate $\dot{x}_4$ from the x-axis and pitch equations. The resulting equations are:

$$\dot{x}_1 = \frac{1}{K_1}\left(-K_2+K_3\,\frac{K_8}{K_9}\right)x_1 + \frac{1}{K_1}\left(-K_6+K_3\,\frac{K_{12}}{K_9}\right)x_2 + \frac{1}{K_1}\left(-K_5+K_3\,\frac{K_{11}}{K_9}\right)x_3$$

$$+\frac{1}{K_1}\left(-K_4+K_3\,\frac{K_{10}}{K_9}\right)x_4 + \frac{1}{K_1}\left(K_7-K_3\,\frac{K_{13}}{K_9}\right)\delta_s$$

$$\dot{x}_2 = x_3$$

$$\dot{x}_3 = \frac{1}{K_{17}}\left(-K_{14}+K_{15}\,\frac{K_8}{K_9}\right)x_1 + \frac{K_{15}}{K_{17}}\frac{K_{12}}{K_9}x_2 + \frac{1}{K_{17}}\left(-K_{18}+K_{15}\,\frac{K_{11}}{K_9}\right)x_3$$

$$+\frac{1}{K_{17}}\left(-K_{16}+K_{15}\,\frac{K_{10}}{K_9}\right)x_4 + \frac{1}{K_{17}}\left(K_{19}-K_{15}\,\frac{K_{13}}{K_9}\right)\delta_s$$

$$\dot{x}_4 = \frac{K_8}{K_9}x_1 - \frac{K_{12}}{K_9}x_2 - \frac{K_{11}}{K_9}x_3 - \frac{K_{10}}{K_9}x_4 + \frac{K_{13}}{K_9}\delta_s$$

These equations define the coefficients of the matrix $A_{long}$ and $B_{long}$ in the equation

$$\dot{x} = A_{long}\,x + B_{long}\,\delta_s$$

used in ATDYN and LONLOP.