AIR FORCE INSTITUTE OF TECHNOLOGY

AIR UNIVERSITY
UNITED STATES AIR FORCE

SCHOOL OF ENGINEERING

WRIGHT-PATTERSON AIR FORCE BASE, OHIO

AFIT/GE/EE/77D-43

THE DESIGN AND SIMULATION OF A TAKEOFF STABILIZATION SYSTEM FOR AN AIRCRAFT WITH AN AIR CUSHION LANDING SYSTEM

THESIS

GE/EE/77D-43

Edward A. Kenney
Captain CAF

AFIT/GE/EE/77D-43

THE DESIGN AND SIMULATION OF A TAKEOFF STABILIZATION SYSTEM FOR AN AIRCRAFT WITH AN AIR CUSHION LANDING SYSTEM

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

by

Edward A. Kenney, B.Sc.

Captain CAF

Graduate Electrical Engineering

December 1977

Approved for Public Release; Distribution Unlimited.

Preface

When the Air Cushion System replaced the conventional takeoff and landing systems of the Jindivik remotely piloted vehicle, the possibility existed that instabilities in pitch, roll, and yaw could occur. As a result, this paper was intended as a design of a Takeoff Stabilization System for the Jindivik using existing autopilot sensors and incorporating an engine yaw thruster and vertical wing tip roll thrusters. When the design was completed, it was sufficiently general that the technique could be applied to any air cushion aircraft or VTOL aircraft. The Landing Stabilization System for the Jindivik using the same sensors and actuators is presently being designed by Captain Max Stafford as his thesis for the Air Force Institute of Technology (AFIT).

I wish to express my gratitude to my Thesis advisors, Dr. George Kurylowich of the Air Force Flight Dynamics Laboratory (AFFDL) and Major R. Potter of AFIT. Also, thanks are due to Captain James Negro of AFIT, Major Jack Randall and Mr. Jim Steiger of the Air Force Flight Dynamics Laboratory for their technical advice and assistance.

My wife, Jane, does not know how much she has contributed to this study, but her patience, understanding, and encouragement has definitely made the past eighteen months of work much easier.

Edward Kenney

Contents

List of Figures

List of Tables

List of Symbols

All symbols are in ft-lb-sec units unless indicated to the contrary.

Alphanumeric Symbols

Symbol		Definition
A		Area
AR	$= \frac{b^2}{S}$	Aspect Ratio
a		Horizontal distance between the inner and outer trunk attachment points
b		Wing span
c		Wing chord
C_D	$= \frac{D}{\bar{q} S}$	Drag coefficient
C_{D_0}		Drag coefficient for zero angle of attack and zero elevator angle
C_{D_α}	$= \frac{\partial C_D}{\partial \alpha}$	Variation of drag coefficient with angle of attack
CG		Centre of gravity (of aircraft)
C_L	$= \frac{L}{\bar{q} S}$	Lift coefficient

Symbol		Definition
C_{L_0}		Lift coefficient for zero angle of attack and zero elevator angle
C_{L_q}	$= \frac{2U}{c} \frac{\partial C_L}{\partial q}$	Variation of lift coefficient with pitch rate
C_{L_t}		Coefficient of lift of the tail
C_{L_α}	$= \frac{\partial C_L}{\partial \alpha}$	Aircraft lift curve slope
$C_{L_{\alpha_F}}$		Lift curve slope of the vertical stabilizer
$C_{L_{\alpha_t}}$		Lift curve slope of the horizontal stabilizer
$C_{L_{\alpha_{2-D}}}$		Theoretical two dimensional lift curve slope of an airfoil at 0° absolute
$C_{L_{\dot{\alpha}}}$	$= \frac{2U}{C} \frac{\partial C_L}{\partial \dot{\alpha}}$	Variation of lift coefficient with rate of change of angle of attack
C_ℓ	$= \frac{\mathcal{L}}{\bar{q} S b}$	Rolling moment coefficient
C_{ℓ_p}	$= \frac{2U}{b} \frac{\partial C_L}{\partial p}$	Variation of rolling moment coefficient with roll rate

Symbol		Definition
C_{l_r}	$= \frac{2U}{r} \frac{\partial C_L}{\partial r}$	Variation of rolling moment coefficient with yaw rate
C_{l_β}	$= \frac{\partial C_L}{\partial \beta}$	Variation of rolling moment coefficient with sideslip angle
C_m	$= \frac{M}{\bar{q} S c}$	Pitching moment coefficient
C_{m_q}	$= \frac{2U}{c} \frac{\partial C_m}{\partial q}$	Variation of pitching moment coefficient with pitch rate
$C_{m_{\dot\alpha}}$	$= \frac{2U}{c} \frac{\partial C_m}{\partial \dot\alpha}$	Variation of pitching moment coefficient with rate of change of angle of attack
C_n	$= \frac{N}{\bar{q} S b}$	Yawing moment coefficient
C_{n_p}	$= \frac{2U}{b} \frac{\partial C_n}{\partial p}$	Variation of yawing moment coefficient with roll rate
C_{n_β}	$= \frac{\partial C_n}{\partial \beta}$	Variation of yawing moment coefficient with sideslip angle
C_Y	$= \frac{F_Y}{\bar{q} S}$	Side force coefficient
C_{Y_F}		Side force coefficient for the vertical stabilizer

Symbol		Definition
C_{y_p}	$= \frac{2U}{b} \frac{\partial C_y}{\partial p}$	Variation of side force coefficient with roll rate
C_{y_r}	$= \frac{2U}{b} \frac{\partial C_y}{\partial r}$	Variation of side force coefficient with yaw rate
C_{y_β}	$= \frac{\partial C_y}{\partial \beta}$	Variation of side force coefficient with sideslip angle
D		Drag
d		Distance between trunk inner attachment points
F		Force
F_{Ax}		Aerodynamic force in the x direction
F_{Ay}		Aerodynamic force in the y direction
F_{Az}		Aerodynamic force in the z direction
F_{CT}		Trunk damping force
F_T		Force from the roll thrusters

Symbol	Definition
F_x	Force in the x direction
$F_{x_{EXT}}$	External force in the x direction
F_y	Force in the y direction
$F_{y_{EXT}}$	External force in the y direction
F_{yT}	Force from the yaw thrusters
F_z	Force in the z direction
$F_{z_{EXT}}$	External force in the z direction
g	Acceleration due to gravity
H_w	Distance between the ground tangent points on the sides of the trunk
H_y	Height of trunk cross section
i	Incidence angle of the horizontal stabilizer

Symbol	Definition
I_{xx}	Roll moment of inertia of the aircraft about the CG
I_{xz}	Product of inertia of the aircraft about the CG
I_{yy}	Pitch moment of inertia of the aircraft about the CG
I_{zz}	Yaw moment of inertia of the aircraft about the CG
k	C_L^{2} coefficient from drag polar
L	Lift
L_S	Length of the straight part of the trunk
$\mathcal{L}$	Rolling moment
$\mathcal{L}_A$	Aerodynamic rolling moment
$\mathcal{L}_{EXT}$	External rolling moment
$\mathcal{L}_{THRUSTERS}$, $\mathcal{L}_T$	Rolling moment of the roll thrusters

Symbol		Definition
ℓ_F		Distance from CG to mean aerodynamic chord of vertical stabilizer
ℓ_P		Peripheral distance from inner trunk attachment to first row of trunk orifices
ℓ_t		Distance from CG to mean aerodynamic chord of horizontal stabilizer
ℓ_W	$= \frac{b}{2}$	Length of one wing
M		Number of straight trunk segments in one quarter of trunk periphery
m		Mass
$\mathcal{M}$		Pitching moment
$\mathcal{M}_A$		Aerodynamic pitching moment
$\mathcal{M}_{EXT}$		External pitching moment
N		Number of curved trunk segments in one quarter of trunk periphery

Symbol	Definition
N_h	Number of trunk orifices per row
N_r	Number of rows of trunk orifices
$\mathcal{N}$	Yawing moment
$\mathcal{N}_A$	Aerodynamic yawing moment
$\mathcal{N}_{EXT}$	External yawing moment
$\mathcal{N}_{YT}$	Yawing moment produced by yaw thrusters
P, p	1. Roll rate (about x axis) 2. Pressure
P_A	Atmospheric pressure
P_{AV}	Average pressure
P_{CH}	Cushion pressure
P_T	Trunk pressure
Q, q	Pitch rate (about y axis)

Symbol	Definition
$\bar{q}$	Dynamic pressure
R, r	Yaw rate (about z axis)
S	Reference area (wing)
S_F	Vertical stabilizer reference area
S	Horizontal stabilizer reference area
SX	Roll thruster switching curve
SX_{YT}	Yaw thruster switching curve
T	Trunk loop tension force
t	Time
t_o	Initial time
t^*	Final time
U, u	Velocity of CG along x axis

Symbol	Definition
$u(t)$	Control variable
u_{MAX}	Maximum roll control
$u_{YT\,MAX}$	Maximum yaw thruster control
V, v	Velocity of CG along y axis (aircraft)
v_t	Trunk vertical velocity
W, w	Velocity of CG along z axis (aircraft)
X_{cx}	Distance of trunk segment centre from CG along vehicle x axis
y	Distance along wing
Z_{cx}	Distance of trunk segment centre from CG along vehicle y axis
z_F	Mean height of the vertical stabilizer

Greek Symbols

Symbol	Definition
α	Angle of attack of aircraft
α_F	Angle of attack of vertical stabilizer
β	1. Sideslip angle 2. Angle subtended by curved trunk segment from trunk centre of curvature
ϵ	Downwash angle
δ_e	Elevator angle
δ_f	Flap angle
$\delta(i)$	Angle of trunk curved segment from centre of curvature
δ_s	Side excursion of trunk
δ_x	Width of straight trunk segment
λ	Sweep angle of leading edge of wing (or horizontal stabilizer)

Symbol	Definition
Δ	Mathematical symbol meaning a small change
ρ	Air density
θ	Pitch attitude angle
θ_P	Inclination angle of lift and drag vectors
ϕ	Roll angle
ψ	Yaw angle

Abstract

The inherent instability in pitch and roll associated with an Air Cushion Landing System (ACLS) aircraft at low airspeeds was investigated, and a means to aid control in pitch and roll was developed. The control system required the use of vertical wing tip thrusters which provided thrust up or down depending on the control signal (similar to space vehicle thrusters). These thrusters could be activated alternately to control roll angle and roll rate with the use of a bang-bang optimal controller. As well, the thrusters would be set forward of the aircraft centre of gravity and could be activated in tandum to aid in pitch control.

The Jindivik Remotely Piloted Vehicle, an Australian target drone, was fitted with an ACLS and taxi tests showed the instability and need for a stabilization system. Subsequent use of Jindivik wind tunnel and taxi test data served as the basis for the development of the roll/pitch control system presented in this paper. Due to computational problems with the air cushion model of the computer program, the controller designs could not be completely verified; but expected trends in pitch, roll and yaw control were shown.

THE DESIGN OF A TAKEOFF STABILIZATION SYSTEM FOR AN AIRCRAFT WITH AN AIR CUSHION LANDING SYSTEM

Chapter I

Introduction

In the low speed range of a takeoff roll, the normal aircraft controls are not aerodynamically effective; hence, the pilot must control the heading by differential braking and wait until the ailerons become effective to control roll. During this time, the landing gear dampens most of the pitch and roll oscillations so that the pilot has few corrections to make in the latter part of the takeoff roll. However, when the conventional landing gear is replaced by an Air Cushion Takeoff System (ACTS) the pitch and roll damping is greatly reduced. This paper will use the Jindivik Remotely Piloted Vehicle as an example of an air cushion vehicle that can be controlled in the low speed range with the use of small jet thrusters on the wing tips and a thrust deflector on the tail section.

The Jindivik Remotely Piloted Vehicle (RPV) is an Australian target drone that can be launched and recovered on a runway. At present, the takeoff is accomplished with a takeoff dolly, as shown in Fig. 1, that provides a wing level attitude and directional control. At lift off the Jindivik separates from the dolly and the dolly brakes to a stop. Recovery of the Jindivik is done by landing on a single, four inch wide metal skid attached to the fuselage. Directional control during landing is maintained with the ailerons, the rolling moment thus produced

Fig. 1. Jindivik on a Takeoff Dolly

makes the drone ride up onto an edge of the skid and turn in the direction of the roll. For the last twelve years the Australian Air Force has used the Jindivik in this configuration with considerable success.

A joint project by the Australian Air Force and the United States Air Force was initiated in 1972 to incorporate an Air Cushion Landing System on the Jindivik. The objectives of this project were to convert the Jindivik to an all-terrain RPV and to advance air cushion technology. The drone and air cushion are shown in Fig. 2. Initial low speed taxi tests, completed in Australia, show that the aircraft fitted with the air cushion is unstable in roll, pitch, and directional control (yaw) (Ref. 13). Therefore, a Stability Augmentation System (SAS) will have

Fig. 2. Jindivik and Air Cushion Landing System

to be designed and incorporated into the autopilot before the RPV is airworthy.

Since the drone was designed to be launched from a directionally controlled dolly, it was not designed with a rudder. Implementation of a rudder during this project would require extensive structural changes and major changes to the autopilot and ground control units. Therefore, a yaw thruster was designed and fitted to the rear of the fuselage to direct the jet exhaust, thus providing a yawing moment. Roll and pitch control will be provided by a vertical roll thruster on the front tip of each wing pod. The roll thrusters will be activated alternately to control rolling moments and in tandum to control pitch. Since the roll thrusters are on the tip

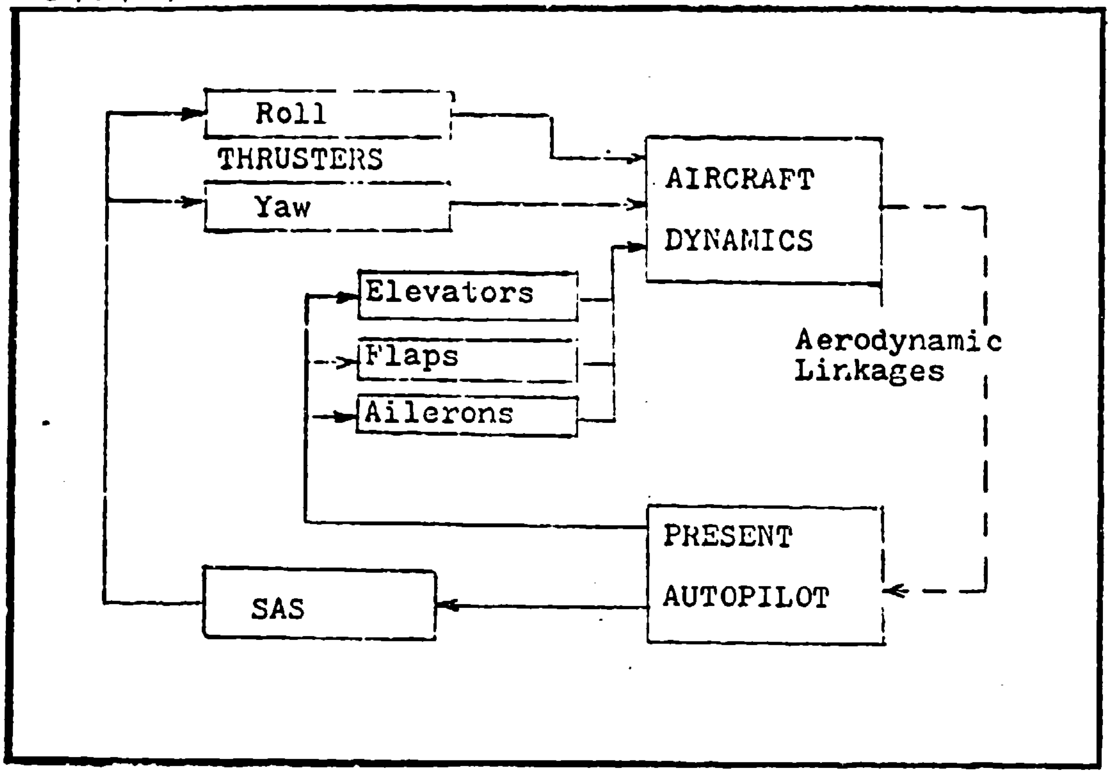


Fig. 3. Block Diagram of Autopilot and SAS Control Units

of the wing pods, they are approximately six feet ahead of the centre of gravity of the drone and hence can produce a moment to control the pitch attitude to some degree. As well, the roll thrusters can be directed up or down to counteract positive or negative rolling and pitching moments.

At velocities near fifty knots, the ailerons and flaps become aerodynamically effective and the roll thrusters are phased out to ensure that the vehicle is not overcontrolled. An additional advantage of turning off the roll thrusters is that a smaller gas supply is required for the thrusters. Hence, they can be used with a gas bottle rather than bleed air from the engine. This arrangement will greatly reduce the airframe and engine modifications required for implementation.

The stability augmentation system will be designed to use the existing sensors in the autopilot to control the roll and pitch thrusters. The SAS unit will be placed in the feedback control loop between the autopilot and actuators, as shown in Fig. 3.

This thesis is organized in the following manner: Chapter II develops the equations of motion and aerodynamic stability derivatives, Chapter III describes the air cushion model, Chapter IV discusses the controller design, Chapter V contains a description of the computer program and the simulation results, and Chapter VI the conclusions and recommendations.

Chapter II

The Determination and Solution of the Jindivik Equations of Motion

The following six simultaneous non-linear differential equations fully describe the motion of the Jindivik RPV. The positive sense of the variables is in the direction of the arrows in Fig. 4.

$$m(\dot{U} - VR + WQ) = -mg\sin\theta + F_{AX} + F_{X_{EXT}} \quad (1)$$

$$m(\dot{V} + UR - WP) = mg\sin\phi\cos\theta + F_{AY} + F_{Y_{EXT}} \quad (2)$$

$$m(\dot{W} - UQ + VP) = mg\cos\phi\cos\theta + F_{AZ} + F_{Z_{EXT}} \quad (3)$$

$$I_{xx}\dot{P} - I_{xz}\dot{R} - I_{xz}PQ + (I_{zz} - I_{yy})RQ = \mathcal{L}_A + \mathcal{L}_{EXT} \quad (4)$$

$$I_{yy}\dot{Q} + (I_{xx} - I_{zz})PR + I_{xz}(P^2 - R^2) = \mathcal{M}_A + \mathcal{M}_{EXT} \quad (5)$$

$$I_{zz}\dot{R} - I_{xz}\dot{P} + (I_{yy} - I_{xx})PQ + I_{xz}QR = \mathcal{N}_A + \mathcal{N}_{EXT} \quad (6)$$

The equations are first order in U, V, W., P, Q, and R with the added kinematic relationships.

$$P = \dot{\phi} - \dot{\psi}\sin\theta \quad (7)$$

$$Q = \dot{\theta}\cos\phi + \dot{\psi}\cos\theta\sin\phi \quad (8)$$

$$R = \dot{\psi}\cos\theta\cos\phi - \dot{\theta}\sin\phi \quad (9)$$

The assumptions used in the derivation of these equations were: (1) the aircraft is a rigid body, (2) the mass of the aircraft is constant for the duration of the analysis, (3) gravity is constant, (4) the earth is an inertial reference, (5) and there is body axis symmetry about the x-z plane (i.e., $I_{xy} = 0 = I_{yz}$).

Equations (1) to (9) are included in the subroutines of the "EASY Dynamic Analysis Computer Program to Aircraft Modeling"

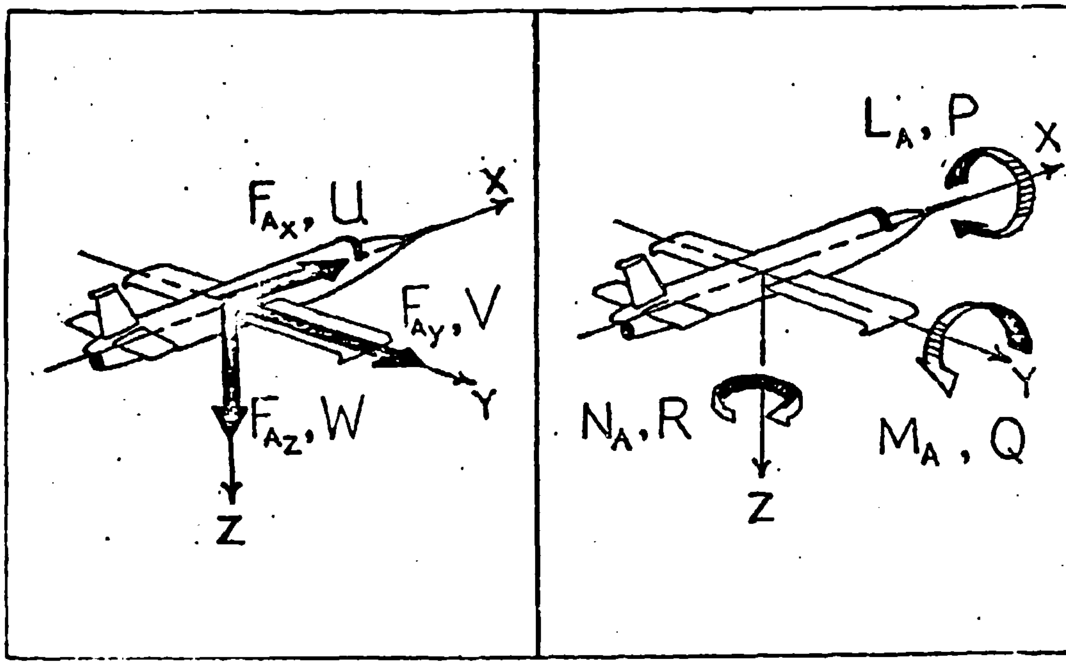


Fig. 4. Definitions of Vector Components in the Equations of Motion

(Ref. 5) which was used to simulate the takeoff motions of the Jindivik. The inputs required by EASY are the mass, inertias, geometry of the aircraft, and the aerodynamic and external forces acting on the drone. The air cushion system is considered to be a prime generator of the external forces and moments (aside from engine forces and moments) and is described in the next chapter. The mass, inertias and geometry are readily available from aircraft blue prints and reference manuals; and the aerodynamic forces and moments can be computed from wind tunnel model data and theoretical methods.

The aerodynamic forces and moments can be written as:

$$L = C_L \bar{q} S \qquad (10)$$

$$D = C_D \bar{q} S \qquad (11)$$

$$F_y = C_y \bar{q} S \tag{12}$$

$$\mathcal{L} = C_\ell \bar{q} S \tag{13}$$

$$\mathcal{N} = C_n \bar{q} S \tag{14}$$

$$\mathcal{M} = C_m \bar{q} S \tag{15}$$

where

$$C_L = f(u, \alpha, \dot{\alpha}, q, \delta_e, \delta_f) \tag{16}$$

$$C_D = f(u, \alpha, \dot{\alpha}, q, \delta_e, \delta_f) \tag{17}$$

$$C_y = f(\beta, \dot{\beta}, p, r, \delta_A) \tag{18}$$

$$C_\ell = f(\beta, \dot{\beta}, p, r, \delta_A) \tag{19}$$

$$C_n = f(\beta, \dot{\beta}, p, r, \delta_A) \tag{20}$$

$$C_m = f(u, \alpha, \dot{\alpha}, q, \delta_e, \delta_f) \tag{21}$$

The coefficients C_L, C_D, C_Y, C_ℓ, C_n, and C_m are non-dimensional. By determining the coefficients for every flight condition, the aerodynamic forces and moments can be calculated and added to the external forces and moments to produce the aircraft motions. These calculations are done by the EASY program but the program requires all the aerodynamic stability derivatives (all the functional relationships which determine the force and moment coefficients). The remainder of this chapter will deal with the derivation of the stability derivatives. These derivatives are derived in the stability axis system as defined by Blakelock (Ref. 2).

Reference 13 contains wind tunnel data for the Jindivik in various configurations, including one when fitted with the Air Cushion Recovery System (ACRS). The ACRS is the air cushion trunk with which the drone lands, but it also has an Air Cushion Takeoff System (ACTS) trunk with which it takes off. After takeoff, the ACTS is disengaged and drops to the ground. Both trunks are the same shape with the ACTS being about 21% larger in all dimensions. Since no wind tunnel data was available for the ACTS, the data for the ACRS was extrapolated by percentages and assumed to be fairly accurate for the ACTS. An example of the estimation technique is that the increase in the frontal area of the aircraft due to the replacement of the ACRS by the ACTS was 6%; therefore, the values of the coefficient of drag (C_D) were increased by 7%. Since the trunk does not generate lift, the coefficient of lift (C_L) was not affected, nor was the coefficient of side force (C_y); symmetry about the x-z axis meant that the coefficient of yawing moment (C_n) was not affected. The pitching moment coefficient (C_m) and rolling moment coefficient (C_ℓ) were affected by the percentage that the increased drag affected those moments. Thus, C_D , C_L , C_y , C_ℓ , C_m , and C_n can be empirically determined as functions of the angle of attack (α), the sideslip (β), and the elevator deflection (δ_e). In other words $\frac{\partial C_D}{\partial \alpha}$, $\frac{\partial C_D}{\partial \beta}$, $\frac{\partial C_D}{\partial \delta_e}$, $\frac{\partial C_L}{\partial \alpha}$, $\frac{\partial C_L}{\partial \delta_e}$, $\frac{\partial C_y}{\partial \beta}$, $\frac{\partial C_\ell}{\partial \beta}$, $\frac{\partial C_m}{\partial \alpha}$, $\frac{\partial C_m}{\partial \delta_e}$, and $\frac{\partial C_n}{\partial \beta}$ can be found from the wind tunnel data.

Non-dimensional derivatives were calculated because they provided a means of checking typical values and signs with

Roskam (Ref. 17) and Blakelock (Ref. 2). Before entering the derivatives into the EASY program, they were dimensionalized.

At low airspeeds the heave motion of the air cushion can create angles of attack beyond the stall limit, but at these speeds aerodynamic contributions to the aircraft dynamics are small.

Stability Derivative Derivation

C_L

From a curve of C_L vs. α of the wind tunnel data it can be shown that

$$C_L = C_{L_0} + \frac{\partial C_L}{\partial \alpha}\alpha \tag{22}$$

C_{L_0} and $C_{L_\alpha} \triangleq \frac{\partial C_L}{\partial \alpha}$ can be determined directly as the C_L intercept and slope of the curve.

C_D

From a drag polar of C_L vs. C_D it can be shown that

$$C_D = C_{D_0} + K C_L^2 \tag{23}$$

where C_{D_0} and K are determined by a curve fit of wind tunnel data of C_D and C_L. Substituting for C_L

$$\begin{aligned} C_D &= C_{D_0} + K(C_{L_0} + C_{L_\alpha}\alpha)^2 \\ &= C_{D_0} + K(C_{L_0}^2 + 2C_{L_0}C_{L_\alpha}\alpha + C_{L_\alpha}^2\alpha^2) \end{aligned} \tag{24}$$

differentiating

$$\begin{aligned} \frac{\partial C_D}{\partial \alpha} &= K(2C_{L_0}C_{L_\alpha} + 2C_{L_\alpha}^2\alpha) \\ &= 2KC_{L_\alpha}(C_{L_0} + C_{L_\alpha}\alpha) \end{aligned} \tag{25}$$

so

$$C_{D_\alpha} \triangleq \frac{\partial C_D}{\partial \alpha} = 2KC_{L_\alpha}C_L \tag{26}$$

Roskam (Ref. 17, pgs 4.12, 1.18, 4.25) shows that for velocities below 300 ft/sec the variation of lift, pitch moment, and drag with velocity is zero. Thus,

$$\frac{\partial C_D}{\partial u} = 0 = \frac{\partial C_L}{\partial u} = \frac{\partial C_m}{\partial u} \tag{27}$$

The quantities

$$\frac{\partial C_D}{\partial \delta_e},\ \frac{\partial C_D}{\partial \delta_f},\ \frac{\partial C_L}{\partial \delta_e},\ \frac{\partial C_L}{\partial \delta_f},\ \frac{\partial C_m}{\partial \alpha},\ \frac{\partial C_m}{\partial \delta_e},\ \frac{\partial C_m}{\partial \delta_f}$$

were all computer from curve-fits of the wind tunnel data.

$\underline{C_{L\dot{\alpha}}}$

From Roskam (Ref. 17) it can be shown that the angle of attack of the tail in downwash is

$$\alpha_t = \alpha - i - \epsilon \tag{28}$$

and for a particular angle of attack

$$\begin{aligned}\Delta\alpha_t &= -\Delta\epsilon \\ &= \frac{\partial \epsilon}{\partial \alpha}\Delta\alpha \\ &= \frac{\partial \epsilon}{\partial \alpha}\dot{\alpha}\Delta t \\ &= \frac{\partial \epsilon}{\partial \alpha}\dot{\alpha}\frac{l_t}{U}\end{aligned} \tag{29}$$

now the change in lift coefficient on the tail due to downwash is

$$\begin{aligned}\Delta C_{L_t} &= C_{L_{\alpha_t}}\Delta\alpha_t \\ &= C_{L_{\alpha_t}}\dot{\alpha}\frac{l_t}{U}\frac{\partial \epsilon}{\partial \alpha}\end{aligned} \tag{30}$$

The change in aircraft lift is

$$\Delta C_L = \Delta C_{L_t}\frac{S_t}{S} \tag{31}$$

$$\frac{\partial C_L}{\partial \dot{\alpha}} = C_{L_{\alpha_t}}\frac{l_t}{U}\frac{\partial \epsilon}{\partial \alpha}\frac{S_t}{S} \tag{32}$$

thus $(C_{L_{\dot{\alpha}}})_{TAIL} \triangleq \frac{\partial C_L}{\partial \dot{\alpha}}\frac{2U}{C} = \frac{2}{C}C_{L_{\alpha_t}} l_t \frac{S_t}{S}\frac{\partial \epsilon}{\partial \alpha}$ (33)

where $$C_{L_{\alpha_t}} = \frac{AR\cos\lambda\, C_{L_{\alpha_{2-D}}}}{AR\sqrt{1+\left(\frac{C_{L_{\alpha_{2-D}}}\cos\lambda}{\pi AR}\right)^2}+\frac{C_{L_{\alpha_{2-D}}}\cos\lambda}{\pi}} \tag{34}$$

$$C_{L_{\alpha_{2-D}}} = 5.73 \tag{35}$$

λ is sweep angle and AR is aspect ratio (Ref. 7)

The wing contribution to $C_{L\dot{\alpha}}$ is considerable but can not be estimated (using Roskam, DATCOM, etc.). So a "typical" value from Roskam of $C_{L\dot{\alpha}} \simeq 1.5\ \text{rad}^{-1}$ was used; fortunately this derivative is of minor importance (Ref. 17, p. 4·114).

$C_{m\dot{\alpha}}$

The contribution of the wing was neglected because it will be negligible with respect to the tail contribution. The correction to the pitching moment due to downwash on the tail is

$$(\Delta C_m)_{TAIL} = -\Delta C_{L_t} \frac{S_t}{S} \frac{\ell_t}{c}$$

$$= -C_{L_{\alpha t}} \frac{\partial \epsilon}{\partial \alpha} \frac{\dot{\alpha}}{c} \frac{\ell_t^2}{u} \frac{S_t}{S} \tag{36}$$

and $$\frac{\partial C_m}{\partial \dot{\alpha}} = -\frac{C_{L_{\alpha t}}}{c} \frac{\partial \epsilon}{\partial \alpha} \frac{\ell_t^2}{u} \frac{S_t}{S} \tag{37}$$

and $$C_{m\dot{\alpha}} \triangleq \frac{2u}{c} \frac{\partial C_m}{\partial \dot{\alpha}} = -\frac{2 C_{L_{\alpha t}} \ell_t^2 S_t}{c^2 S} \frac{\partial \epsilon}{\partial \alpha} \tag{38}$$

C_{Lq}

q changes the angle of attack on the tail by $\frac{q\ell_t}{u}$ radians (for quasistatic conditions)

so $$\Delta \alpha_t = \frac{q \ell_t}{u} \tag{39}$$

and $$\Delta C_L = \frac{S_t}{S} \Delta C_{L_t} = \frac{S_t}{S} C_{L_{\alpha t}} \frac{q \ell_t}{u} \tag{40}$$

differentiating $$\left(\frac{\partial C_L}{\partial q}\right)_{TAIL} = C_{L_{\alpha t}} \frac{S_t}{S} \frac{\ell_t}{u} \tag{41}$$

and the contribution of the wing body is negligible in comparison to the tail (Ref. 17, p. 153)

now $$C_{Lq} \triangleq \frac{2u}{c} \frac{\partial C_L}{\partial q} = \frac{2 C_{L_{\alpha t}} S_t \ell_t}{c S} \tag{42}$$

C_{m_q}

The moment on the tail is

$$\mathcal{M}_t = -\ell_t L_t = -\ell_t C_{L_t} \bar{q} S_t \qquad (43)$$

and the change in moment due to a change in angle of attack is

$$\Delta \mathcal{M}_t = -\bar{q} \ell_t S_t \Delta C_{L_t} = -\bar{q} \ell_t S_t C_{L_{\alpha_t}} \Delta\alpha_t = -\bar{q} \frac{\ell_t^2 S_t}{u} C_{L_{\alpha_t}} q \qquad (44)$$

and

$$\Delta C_{m_t} = -q \frac{\ell_t^2 C_{L_{\alpha_t}} S_t}{u c S} \qquad (45)$$

$$\left(\frac{\partial C_m}{\partial q}\right)_{TAIL} = -\frac{\ell_t^2 C_{L_{\alpha_t}} S_t}{u c S} \qquad (46)$$

since the wing contribution is negligible with respect to the tail (Ref. 17, p. 153)

$$C_{m_q} \triangleq \frac{2u}{c}\left(\frac{\partial C_m}{\partial q}\right)_{TAIL} = -\frac{2\ell_t^2 C_{L_{\alpha_t}} S_t}{c^2 S} \qquad (47)$$

β Derivatives

Wind tunnel data gave C_y, C_ℓ, and C_n vs. β for values of $|\beta| \leq 7°$, but in any takeoff with crosswind the sideslip will normally exceed 7°. The normal takeoff procedure will be to initially line the aircraft into the relative wind at the centerline and change the heading as the aircraft gains speed. This procedure should keep β within $\pm 30°$ and the present data can be curve fitted and extrapolated to this value. Consequently, expressions can be obtained for C_{y_β}, C_{ℓ_β}, and C_{n_β} from the data. The $\dot{\beta}$ derivatives have been assumed to be zero (Ref. 17). For the β, p, and r derivatives the effect of sidewash on the tail has been neglected.

p Derivatives

C_{n_p}

The change in C_n from the tail side force due to roll rate p is

$$(\Delta C_n)_{tail} = -\frac{\Delta C_{y_F} S_F \ell_F}{S b}$$

$$= -\frac{C_{L_{\alpha F}}\, p\, \mathfrak{z}_F S_F \ell_F}{u S b} \tag{48}$$

$$\left(\frac{\partial C_n}{\partial p}\right)_{tail} = -\frac{C_{L_{\alpha F}} \mathfrak{z}_F S_F \ell_F}{u S b} \tag{49}$$

where $\mathfrak{z}_F$ is the mean height of the fin, and the effect of sidewash has been neglected

so $$(C_{n_p})_{tail} \triangleq \frac{2u}{b}\left(\frac{\partial C_n}{\partial p}\right)_{tail} = -\frac{2 C_{L_{\alpha F}} \mathfrak{z}_F S_F \ell_F}{S b^2} \tag{50}$$

The wing contribution is in two parts due to lift and drag. For positive p the angle of attack is increased on the right wing and decreased on the left; thus inclining and changing the lift and drag vectors of each wing section. The inclination angle is $\Theta_p = \frac{py}{u}$ where y is the spanwise coordinate of the section. The change in lift is

$$\Delta L_{left} = \bar{q}\, C_{L_\alpha} \Delta\alpha\, c\, dy$$

$$= \bar{q}\, C_{L_\alpha} \frac{py}{u} c\, dy \tag{51}$$

and $$\Delta L_{right} = -\bar{q}\, C_{L_\alpha} \frac{py}{u} c\, dy \tag{52}$$

where $c\,dy$ is the area of the wing section.

So the change in the yawing moment due to lift is

$$\Delta N_{left} = -y\, \Delta L_{left} \frac{py}{u} \tag{53}$$

and
$$\Delta N_{right} = y \, \Delta L_{right} \frac{py}{U} \tag{54}$$

so
$$\Delta N_{sections} = -\frac{2\bar{q}\,c\,dy}{U^2} y \, C_{L\alpha} \, p^2 y^3 \tag{55}$$

and the total yawing moment of the wing is

$$\Delta N_{total} = \int_0^{\frac{b}{2}} \Delta N_{sections}$$

$$= -\frac{2\bar{q}\,c\,p^2}{U^2} C_{L\alpha} \int_0^{\frac{b}{2}} y^3 dy$$

$$= -\frac{\bar{q}\,c\,C_{L\alpha}\,p^2 b^4}{32\,U^2} \tag{56}$$

and
$$\frac{\partial N}{\partial p} = -\frac{\bar{q}\,c\,p\,b^4\,C_{L\alpha}}{16\,U^2} \tag{57}$$

$$(C_{n_p})_{wing\ lift} \triangleq \frac{2U}{S\bar{q}b^2}\frac{\partial N}{\partial p} = -\frac{C_{L\alpha}\,p\,b^2 c}{8\,S\,U} \tag{58}$$

the change in drag is

$$\Delta D_{left} = -\bar{q}\,C_{D\alpha}\,\Delta\alpha\,c\,dy$$
$$= -\bar{q}\,C_{D\alpha}\frac{py}{U}\,c\,dy \tag{59}$$

and
$$\Delta D_{right} = \bar{q}\,C_{D\alpha}\frac{py}{U}\,c\,dy \tag{60}$$

the corresponding change in yawing moment is

$$\Delta N_{left} = y\,\frac{\bar{q}\,c\,C_{D\alpha}}{U}\,p\,y\,dy \tag{61}$$

and
$$\Delta N_{right} = \bar{q}\,\frac{C_{D\alpha}\,p\,y^2}{U}\,c\,dy \tag{62}$$

so
$$\Delta N_{sections} = 2\frac{\bar{q}\,C_{D\alpha}\,p\,y^2\,c\,dy}{U} \tag{63}$$

and

$$\Delta N_{total} = 2\bar{q}\frac{c}{u}C_{D\alpha}\,p\int_0^{b/2} y^2\,dy$$

$$= \frac{\bar{q}\,c\,C_{D\alpha}\,p\,b^3}{12u} \tag{64}$$

now

$$\frac{\partial N}{\partial p} = \frac{\bar{q}\,c\,C_{D\alpha}\,b^3}{12u} \tag{65}$$

and

$$(C_{n_p})_{drag} \triangleq \frac{2u}{S\bar{q}b^2}\frac{\partial N}{\partial p} = \frac{C_{D\alpha}\,bc}{6S} \tag{66}$$

summing all effects

$$C_{n_p} = -\frac{2C_{L\alpha_F}\,z_F\,l_F\,S_F}{Sb^2} - \frac{C_{L\alpha}\,p\,b\,c}{8uS} + \frac{C_{D\alpha}\,b\,c}{6S}$$

$$= -\frac{2C_{L\alpha_F}\,z_F\,l_F\,S_F}{Sb^2} - \frac{C_{L\alpha}\,p\,b}{8u} + \frac{C_{D\alpha}}{6} \tag{67}$$

C_{y_p}

C_{y_p} is often negligible (Ref. 17, p. 170) and the tail is the major contributor. Let the mean change in angle of attack of the vertical stabilizer (fin) be

$$\Delta\alpha_F = -\frac{p\,z_F}{u} \tag{68}$$

where z_F is the mean height of the fin

now

$$\Delta C_{y_F} = C_{L\alpha_F}\,\Delta\alpha_F$$

$$= -C_{L\alpha_F}\frac{p\,z_F}{u} \tag{69}$$

so the change on the side force coefficient of the aircraft is (sidewash is neglected)

$$\Delta C_y = \frac{S_F}{S}\Delta C_{y_F} = -\frac{S_F\,p\,z_F}{S\,u}C_{L\alpha_F} \tag{70}$$

so

$$\frac{\partial C_y}{\partial p} = -\frac{S_F\,z_F\,C_{L\alpha_F}}{S\,u} \tag{71}$$

and

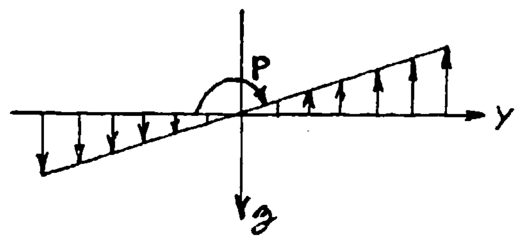
$$C_{y_p} = \frac{2u}{b}\frac{\partial C_y}{\partial p} = -\frac{2S_F\,z_F\,C_{L\alpha_F}}{Sb} \tag{72}$$

C_{l_p}

The wing is the only major contributor (Ref. 17, p. 170). Etkin (Ref. 9) shows that the rolling velocity, p , produces a change in the angle of attack of each wing section which is proportional to the span, i.e.,

$$\Delta\alpha = \frac{py}{u} \tag{73}$$

where y is the spanwise coordinate of the wing section. Then the lift distribution on the wing due to rolling is estimated to be



Etkin (Ref. 9) changes the triangular lift distribution to a sinusoidal distribution to account for the loss of lift at the wing tips due to spanwise flow around the wing tips. However, since the Jindivik has large tip tanks the lateral airflow will be minimized and the lift distribution will be closer to the triangular distribution shown.

The change in lift will be

$$\Delta L = 2\bar{q}\, C_{L_\alpha} \Delta\alpha\, c\, dy$$

$$= 2\bar{q}\, C_{L_\alpha} \frac{cpy}{u} dy \tag{74}$$

and the change in rolling moment due to this lift will be

$$\Delta \mathcal{L} = -y\,\bar{q}\, C_{L_\alpha} \frac{cpy}{u} dy \tag{75}$$

thus

$$\mathcal{L} = -\bar{q}\,\frac{c C_{L_\alpha} p}{U} \int_0^{\frac{b}{2}} y^2 dy$$

$$= -\bar{q}\,\frac{c C_{L_\alpha} p b^3}{24U} \tag{76}$$

so

$$\frac{\partial \mathcal{L}}{\partial p} = -\bar{q}\,\frac{c C_{L_\alpha} b^3}{24U} \tag{77}$$

and

$$\frac{\partial C_L}{\partial p} \triangleq \frac{1}{\bar{q} S b}\,\frac{\partial \mathcal{L}}{\partial p} = \frac{-C_{L_\alpha} b^{\not{2}} c}{\not{S}\, 24U} \tag{78}$$

so

$$C_{\ell_p} \triangleq \frac{2U}{b}\,\frac{\partial C_\ell}{\partial p} = -\frac{C_{L_\alpha} b c}{12 S} = -\frac{C_{L_\alpha}}{12} \tag{79}$$

r Derivatives

C_{y_r}

The tail is the prime contributor (Ref. 17, p. 174). The change in fin angle of attack due to a yaw rate is

$$\Delta\alpha_F = \frac{r\,\ell_F}{U} \tag{80}$$

so the change in side force due to the tail is

$$\Delta C_y = C_{L_{\alpha_F}}\,\frac{S_F\, r\,\ell_F}{S U} \tag{81}$$

$$\frac{\partial C_y}{\partial r} = C_{L_{\alpha_F}}\,\frac{S_F\,\ell_F}{S U} \tag{82}$$

$$C_{y_r} \triangleq \frac{2U}{b}\,\frac{\partial C_y}{\partial r} = \frac{2 C_{L_{\alpha_F}} S_F\,\ell_F}{S b} \tag{83}$$

C_{ℓ_r}

Contributions are from the wing and tail. The side force on the tail acts at $\mathfrak{z}_F$, the mean height of the fin

and

$$\Delta F_y = \Delta C_y\,\bar{q}\,S$$

$$= \bar{q}\,S\,C_{L_{\alpha_F}}\,\frac{S_F\, r\,\ell_F}{S U} \tag{84}$$

now

$$\Delta \mathcal{L} = \Delta F_y \, z_F$$

$$= \bar{q} S C'_{L\alpha_F} \frac{S_F r l_F z_F}{S U} \qquad (85)$$

differentiating

$$\frac{\partial \mathcal{L}}{\partial r} = \bar{q} S C_{L\alpha_F} \frac{S_F l_F z_F}{S U} \qquad (86)$$

and

$$(C_{l_r})_{tail} = \frac{2U}{b} \frac{1}{S\bar{q}b} \frac{\partial \mathcal{L}}{\partial r}$$

$$= \frac{2 C_{L\alpha_F} S_F l_F z_F}{S b^2} \qquad (87)$$

A positive r also increases lift on the left wing and decreases it on the right; the change in lift on each wing section is

$$\Delta L_{left\ section} = \Delta \bar{q} \, C_L c \, dy$$

$$= \frac{1}{2} \rho (\Delta u)^2 C_L c \, dy$$

$$= \frac{1}{2} \rho (r y)^2 C_L c \, dy \qquad (88)$$

and

$$\Delta L_{right\ section} = - \Delta \bar{q} \, C_L c \, dy$$

$$= -\frac{1}{2} \rho (r y)^2 C_L c \, dy \qquad (89)$$

now the change in rolling moment due to two sections at y is

$$\Delta \mathcal{L} = \rho r^2 y^3 C_L c \, dy \qquad (90)$$

and the total rolling moment change for the wing is

$$\Delta \mathcal{L} = \rho r^2 C_L c \int_0^{b/2} y^3 dy$$

$$= \frac{\rho r^2 C_L c b^4}{64}$$

$$= \frac{\bar{q} r^2 C_L c b^4}{32 U^2} \qquad (91)$$

now

$$\frac{\partial \mathcal{L}}{\partial r} = \frac{\bar{q}\, r\, C_L\, c\, b^4}{16\, U^2} \tag{92}$$

$$\left(\frac{\partial C_l}{\partial r}\right)_{wing} \triangleq \frac{1}{S\bar{q}b}\frac{\partial \mathcal{L}}{\partial r} = \frac{r\, C_L\, b^2}{16\, U^2} \tag{93}$$

and
$$\left(C_{l_r}\right)_{wing} \triangleq \frac{2U}{b}\frac{\partial C_l}{\partial r} = \frac{r\, C_L\, b}{8\, U} \tag{94}$$

summing the components

$$C_{l_r} = \frac{2 C_{L\alpha_F} S_F l_F \beta_F}{S b^2} + \frac{r\, C_L\, b}{8\, U} \tag{95}$$

$\underline{C_{n_r}}$

The tail and wing contribute to C_{n_r}. Knowing that the change in the fin angle of attack is

$$\Delta \alpha_F = \frac{-r\, l_F}{U} \tag{96}$$

and the moment arm of the tail is l_F then

$$(\Delta \mathcal{N})_{tail} = -\frac{l_F C_{L\alpha_F} S_F r l_F \bar{q} S}{S\, U} \tag{97}$$

so
$$\left(\frac{\partial \mathcal{N}}{\partial r}\right)_{tail} = -\bar{q} S C_{L\alpha_F} \frac{S_F\, l_F^2}{S\, U} \tag{98}$$

so
$$\left(C_{n_r}\right)_{tail} \triangleq -\frac{2U}{b}\,\frac{1}{S\bar{q}b}\,\frac{\partial \mathcal{N}}{\partial r}$$

$$= -2 C_{L\alpha_F} \frac{S_F\, l_F^2}{S\, b^2} \tag{99}$$

A positive r increases the drag on the left wing and decreases drag on the right wing. The change in drag on eac' wing section is

$$\Delta D_{\text{left section}} = \Delta\bar{q}\, C_D\, c\, dy = \frac{1}{2}\rho\, C_D\, (\Delta u)^2 c\, dy = \frac{1}{2}\rho\, C_D\, r^2 y^2 c\, dy \tag{100}$$

and
$$\Delta D_{\text{right section}} = -\frac{1}{2}\rho\, C_D\, r^2 y^2 c\, dy \tag{101}$$

so the change in yawing moment is

$$\Delta \mathcal{N} = -\rho\, C_D\, r^2 c \int_0^{b/2} y^3 dy = -\frac{\rho\, C_D\, r^2 c\, b^4}{64} \tag{102}$$

now
$$\frac{\partial \mathcal{N}}{\partial r} = -\frac{\rho\, C_D\, r\, c\, b^4}{32} = -\frac{\bar{q}\, C_D\, r\, c\, b^4}{16 u^2} \tag{103}$$

and
$$(C_{n_r})_{wing} = \frac{2u}{b}\,\frac{1}{S\bar{q}b}\,\frac{\partial \mathcal{N}}{\partial r} = -\frac{C_D\, r\, b}{8u} \tag{104}$$

summing the components

$$C_{n_r} = -\frac{2 C_{L_{\alpha F}} S_F \ell_F^2}{S b^2} - \frac{C_D\, r\, b}{8u} \tag{105}$$

Once all the derivatives had been determined or estimated, Ref. 15 was used to convert to dimentional bcdy-axis derivatives. The derivatives were the written into the EASY program.

Chapter III

The Air Cushion Model

The air cushion model used in this analysis is a truncated version of an ACLS model that was designed by Foster-Miller Associates Inc. of Waltham, Massachusetts for the National Aeronautics and Space Administration (NASA) (Ref. 4).

The basic ACTS configuration is shown in Fig. 5. The model includes four primary subsystems: (1) the fan, (2) the feeding system, (3) the trunk, and (5) the cushion. Air from the fan flows through the ducts and plenum (feeding system) and enters the trunk. The trunk has several rows of orifices that exhaust both to the cushion and the atmosphere. Thus, the airflow from the trunk has two components, one entering the cushion and the other leading directly to the atmosphere. The cushion flow exhausts to the atmosphere through the clearance gap formed between the trunk and ground. In addition to the basic flows described above, two other flows have been included in the model. These are the plenum bleed flow and the direct cushion flow. Plenum bleeding causes some of the air to flow directly from plenum to atmosphere, and has been used in some designs to improve the dynamic characteristics of the air supply system. Direct flow from the plenum to the cushion can also improve dynamic response. A pressure relief valve is also included in the basic configuration. It allows additional flow to vent from the plenum whenever the pressure exceeds a preset level, and thus improves stability by reducing fan stall.

The support force acting on the aircraft is made up of two components. The first occurs due to the cushion pressure acting over the cushion area. The second, which comes about only during ground contact, is given by the

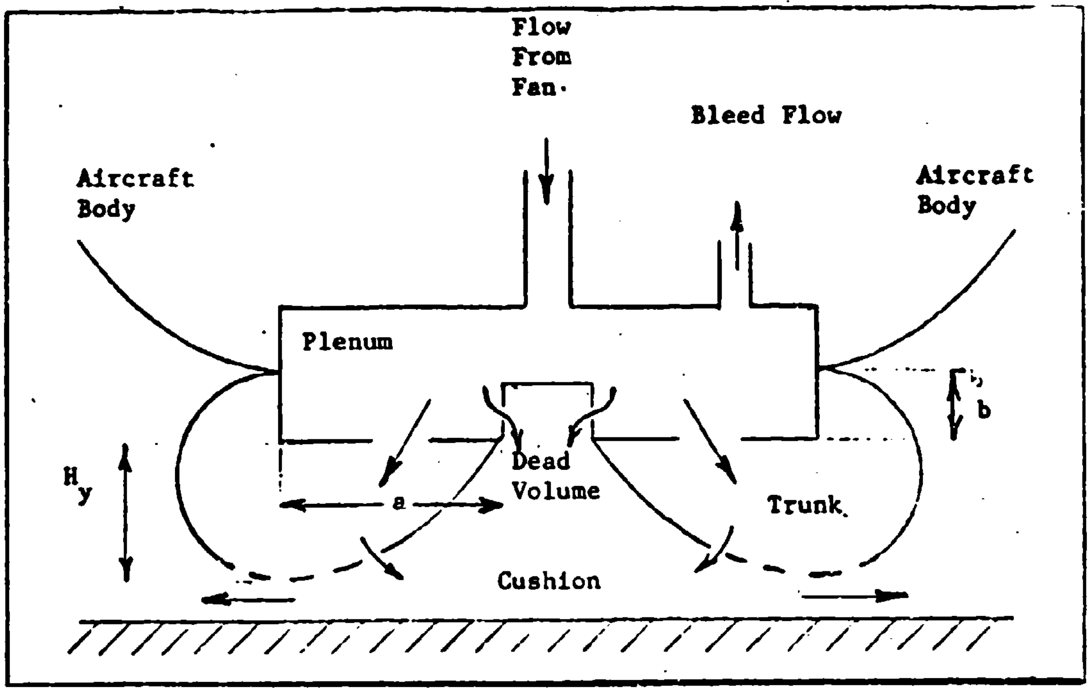


Fig. 5. Basic ACTS Configuration

contact pressure acting over the trunk contact area. The support force, in general, also gives rise to a moment, given by the product of the force and its distance from the CG of the aircraft.

In plan, the cushion has an oval shape, made up of a rectangular section with semicircular ends. The lengths a and b are the horizontal and vertical spacing between the points of attachment of the trunk to the aircraft body. The initial (undeformed) trunk shape is defined by the above two parameters and the perimeter ℓ_p and height H_y as shown. S_h is the (uniform) spacing between the rows of peripherally distributed orifices. The number of the orifices is selected independently by the number of orifice rows N_r and the number of orifices per row N_h. The cushion volume consists of two parts: an active (dynamically varying) region and a dead (static) region. The active volume depends on the trunk shape and ground profile. The dead volume, which is a design variable, includes recesses in the cushion cavity as shown.

The forces transferred to the aircraft act through the cushion and trunk. To help calculate these forces, the trunk and cushion are divided into segments as shown in Fig. 6. Each straight section of the cushion and trunk is divided into M rectangular segments, while each curved end is divided into N pie-shaped segments. Thus, the total number of segments is 2 (M + N). All cushion and trunk parameters are calculated first for each segment and then summed to give their total system values.

The dynamic analysis of the vehicle system is best derived with the help of two orthogonal coordinate frames of reference: a coordinate frame fixed in space (inertial frame), and a coordinate frame fixed to the vehicle (vehicle frame) with origin at the aircraft CG. The reason for two frames can be appreciated by recognizing that:

(a) Newton's law for translation motion requires that the CG acceleration be expressed relative to the inertial frame.

(b) The corresponding law for rotational motion, while valid in both inertial and vehicle frames, is applied more conveniently in the vehicle frame, because rotational inertia about any vehicle axis is constant, while the rotational inertia about any inertial (fixed) axis varies with aircraft position.

Accordingly, the two frames of reference have been defined as shown in Fig. 7. The vehicle frame with origin at the aircraft CG

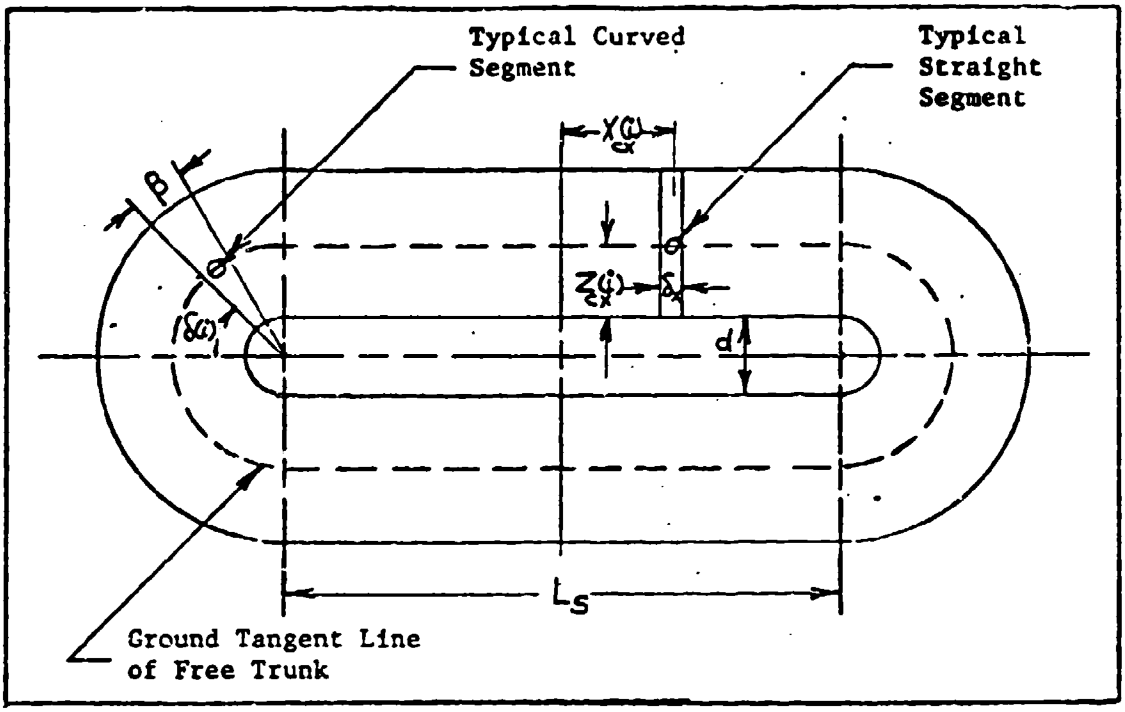


Fig. 6. Division of Trunk Into Segments

has roll, yaw and pitch axes x, y and z, respectively, fixed to the aircraft body as shown. The inertial frame has corresponding axes X, Y and Z fixed in space. The two frames coincide only when the aircraft has not undergone any rotation from equilibrium.

In the analysis, the actual runway profile underneath the ACTS is approximated by segments that coincide in plan with those of the trunk and are parallel to the cushion hard surface as shown in Fig. 7. With this model, all pressure forces act parallel to the vehicle yaw axis so that the segment torque components about the aircraft CG can be easily computed by multiplying the segment force by the fore-and-aft and/or lateral separation between the segment and the CG.

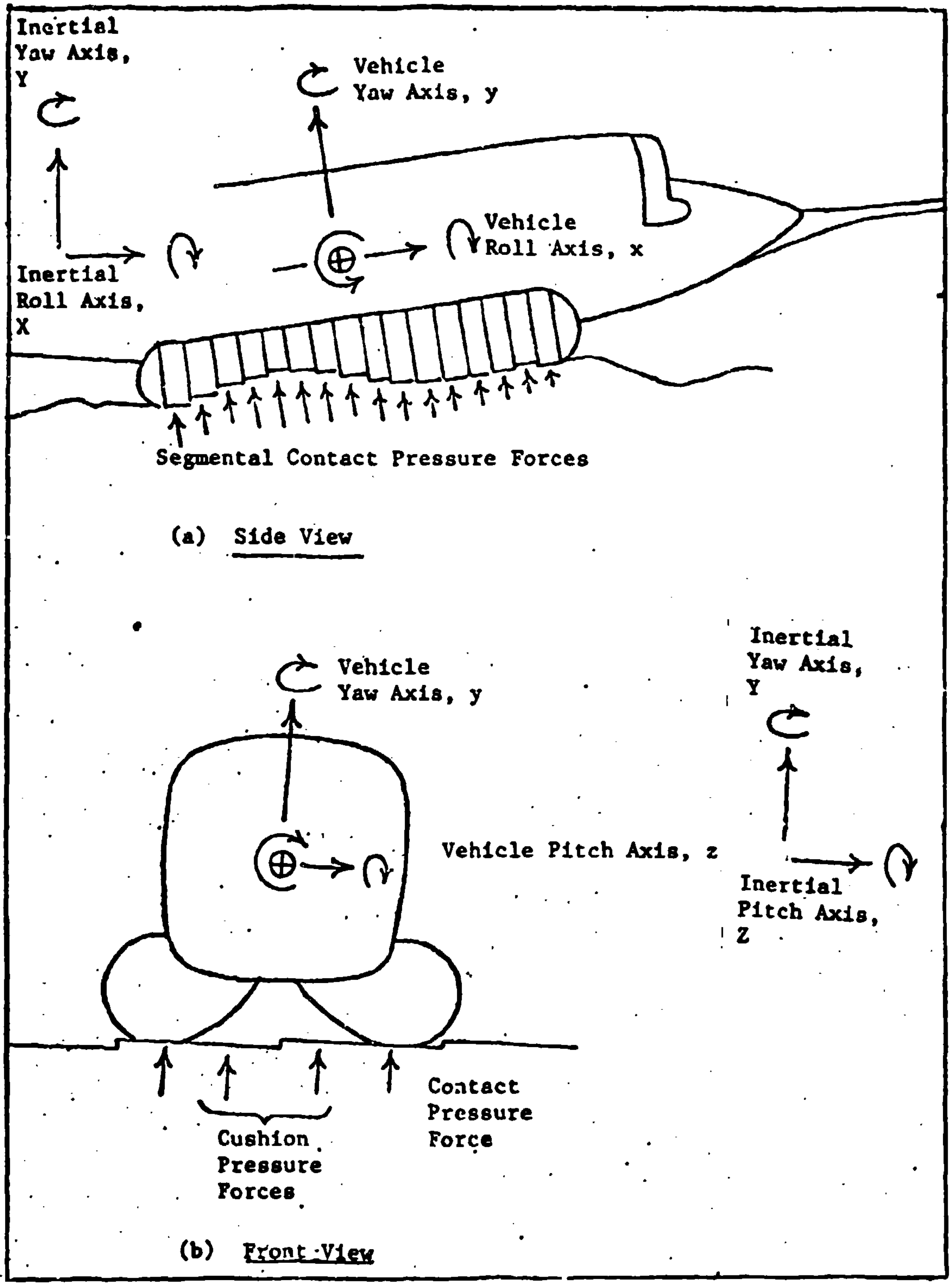


Fig. 7. Inertial and Vehicle Coordinate Frames

The analytical model of the ACTS consists of a set of equations which when solved determines the pressures, flows, forces, and motion of the system for various aircraft and runway parameters. The overall ACTS system is divided into two interrelated systems: the flow system and the force system. These systems are shown in Fig. 8 and Fig. 9. The flow system establishes the pressure-flow relationship for various subsystems of the ACTS. The force system establishes the corresponding force-motion relationships. The interdependence of the two systems comes about because the trunk deflection obtained from the force system changes the volumes and orifice areas that form part of the flow system. Similarly, the cushion and trunk pressures found from the flow system give rise to forces and moments that form inputs to the force system.

The Trunk Model

The major component of the ACTS model is the trunk model because it determines the trunk shape parameters (volume, and orifice and contact areas), contact pressure distribution and damping that form inputs to the ACLS flow and force systems.

Trunk Shape. In past work, two analytical models have been developed for the trunk shape: the Membrane Trunk Model (Ref. 8) and the Frozen Trunk Model (Ref. 6). The shortcoming of both these analyses was that they modeled the side and end segments of the trunk in the same way while test data now confirm that the shorter curved end segments (front and rear) behave very differently from the longer, straight side segments. Fig. 10 shows the trunk cross section measured at the center of the side and end segments as the load on the cushion is increased. The entire side segment tends to bow outward and avoid ground contact, while the end segment remains virtually fixed, except for a flattening

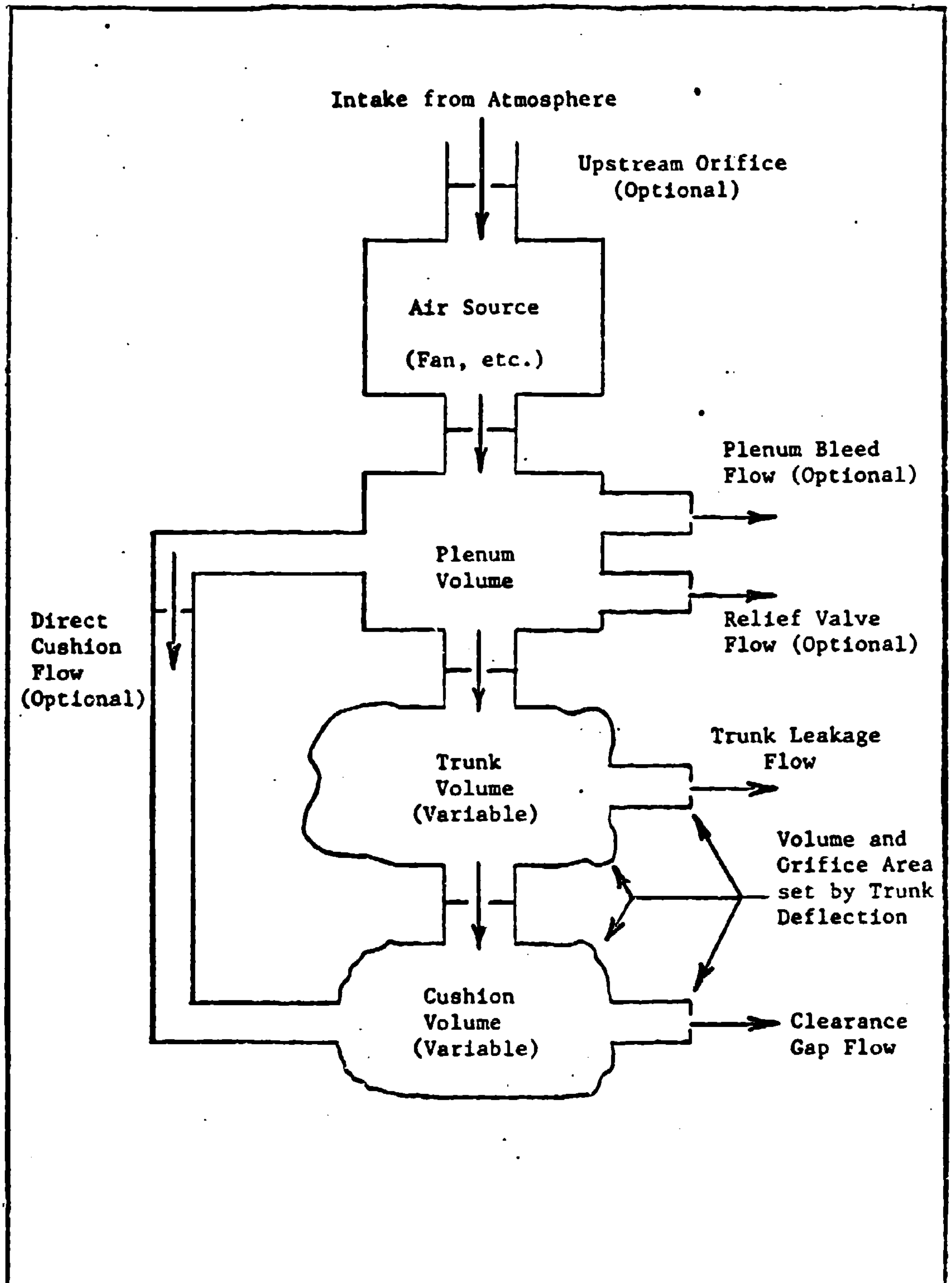


Fig. 8. ACTS Flow System

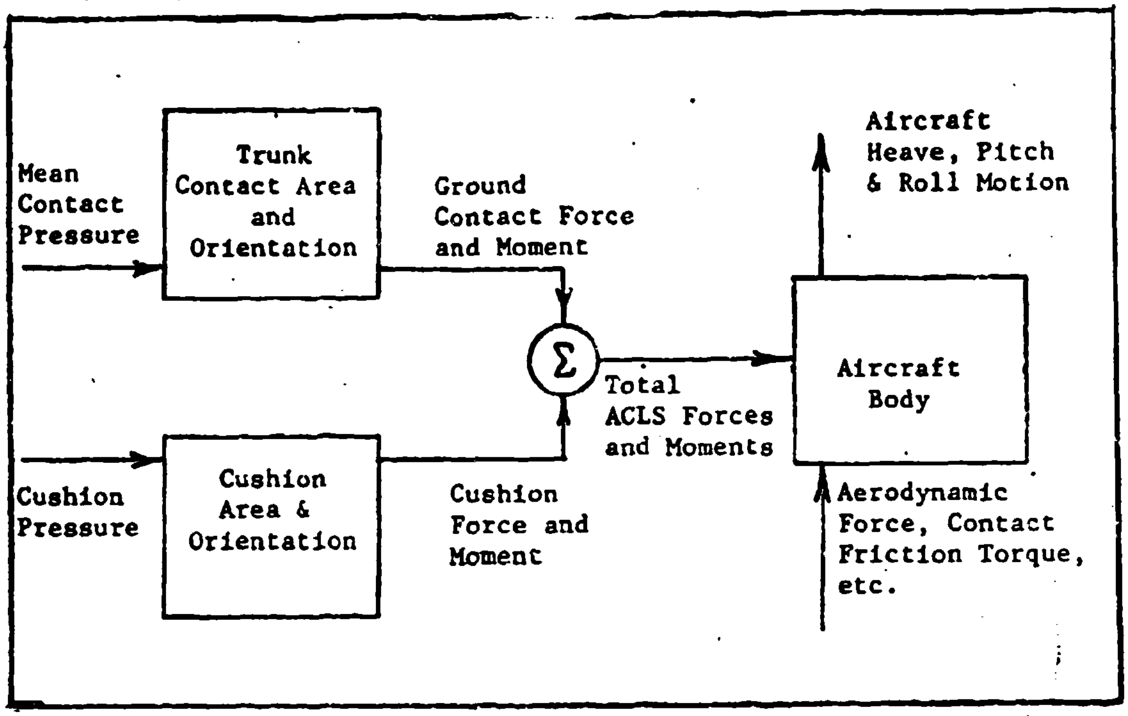
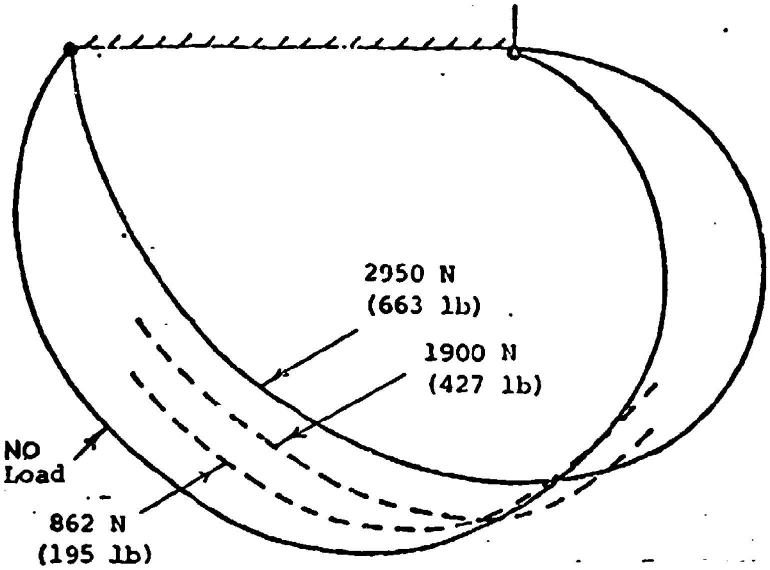
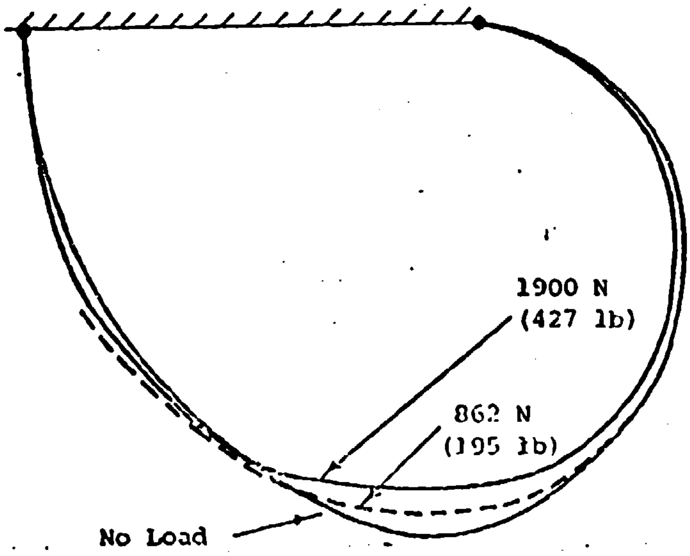


Fig. 9. ACTS Force System

in the region that actually touches the ground. This difference in behavior occurs because the front segment is much smaller than the side segment and is curved. When the cushion pressure increases due to an increase in the load, the radially outward force causes the oval trunk planform to become more circular, as shown in Fig. 11. This causes a hoop tension force, T, to act around the trunk periphery. In the side segments, this force acts substantially normal to the side excursion, δ_s, so that its component resisting the motion is negligible and the side segment can thus bow outwards relatively unrestrained. In the end segments the situation is different, since the curvature of the segment causes the hoop tension to have a much higher component opposing the motion so that outward motion of the trunk ends is much smaller.



(a) Side Segment



(b) End Segment

Fig. 10. Measured Trunk Profile

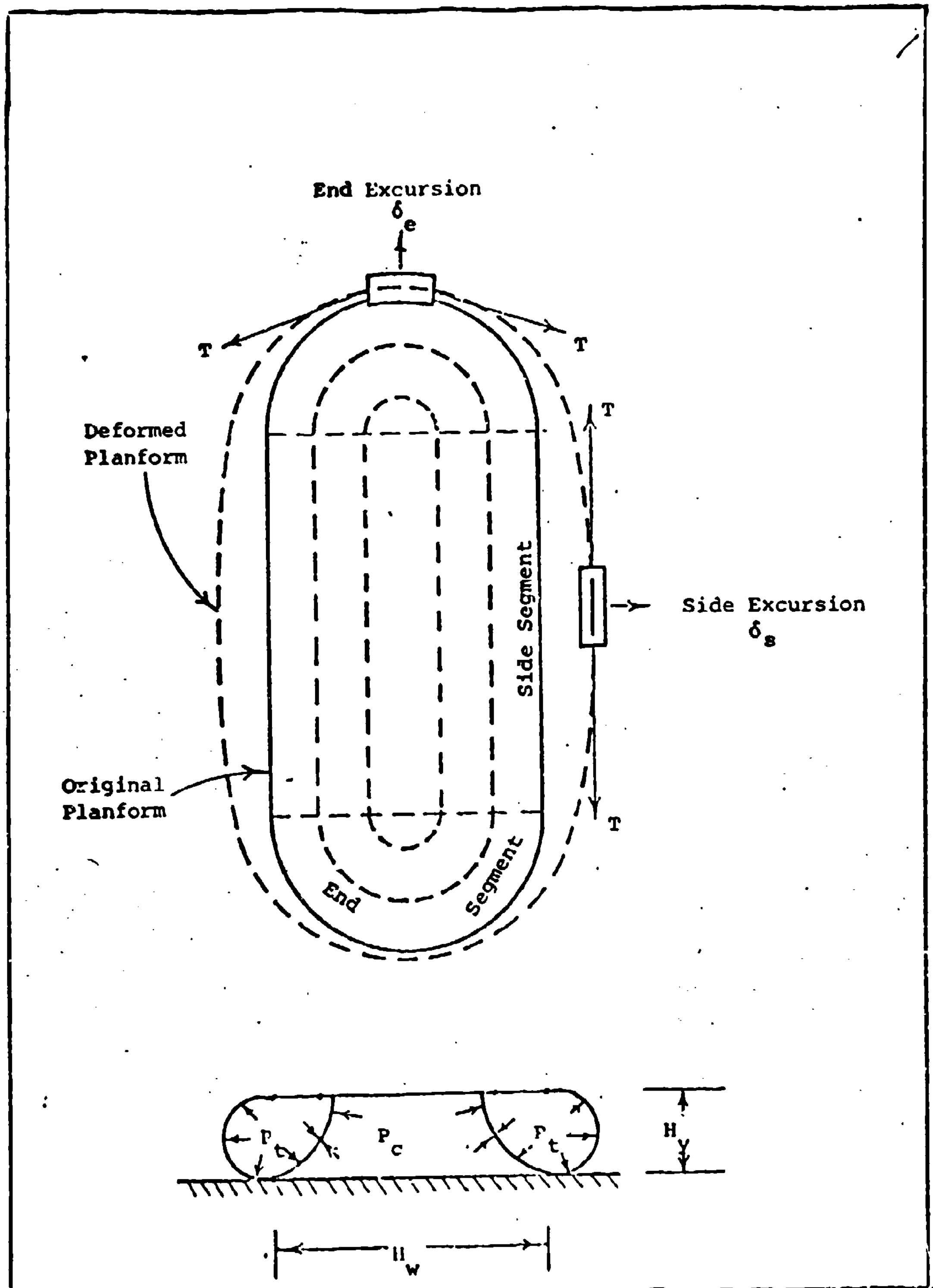


Fig. 11. Outward Excursion of Trunk Segments

Since hoop tension has very little effect on side trunk motion, the side segments can be considered as simple, two-dimensional membranes, as done in the Membrane Trunk Model. On the other hand, the fact that hoop tension restrains ("freezes") the trunk ends suggests that these segments be modeled by the Frozen Trunk Model. Thus, the logical step in trunk model improvement is to combine the two existing models and form the Hybrid Trunk Model, in which the sides are represented by the Membrane Model and the ends by the Frozen Model.

The Hybrid Trunk Model is essentially a limiting case analysis of trunk deflection. In general, best results will be obtained at the middle of the respective segments, i.e., at the center of the side segments, where the trunk behaves very much like an ideal membrane, and at the center of the end segments, where the trunk shape is truly fixed. In the transition region (at and near where the segments meet), the trunk will exhibit properties of both the membrane and frozen trunk approximations.

Contact Pressure. In addition to trunk and cushion shape, the trunk model also determines the pressure distribution in the ground contact zone. The analysis for pressure distribution is complicated by the fact that two separate effects must be considered: direct trunk-ground contact caused by the trunk pressure forcing the trunk against the ground, and airflow through the trunk holes into the interstices that remain in the contact zone.

When two bodies in contact are acted upon by a force, F, the actual contact occurs at a number of discrete regions rather than over the whole area, due to the inherent roughness of the contacting surfaces as shown in Fig. 12. Because the number of contact

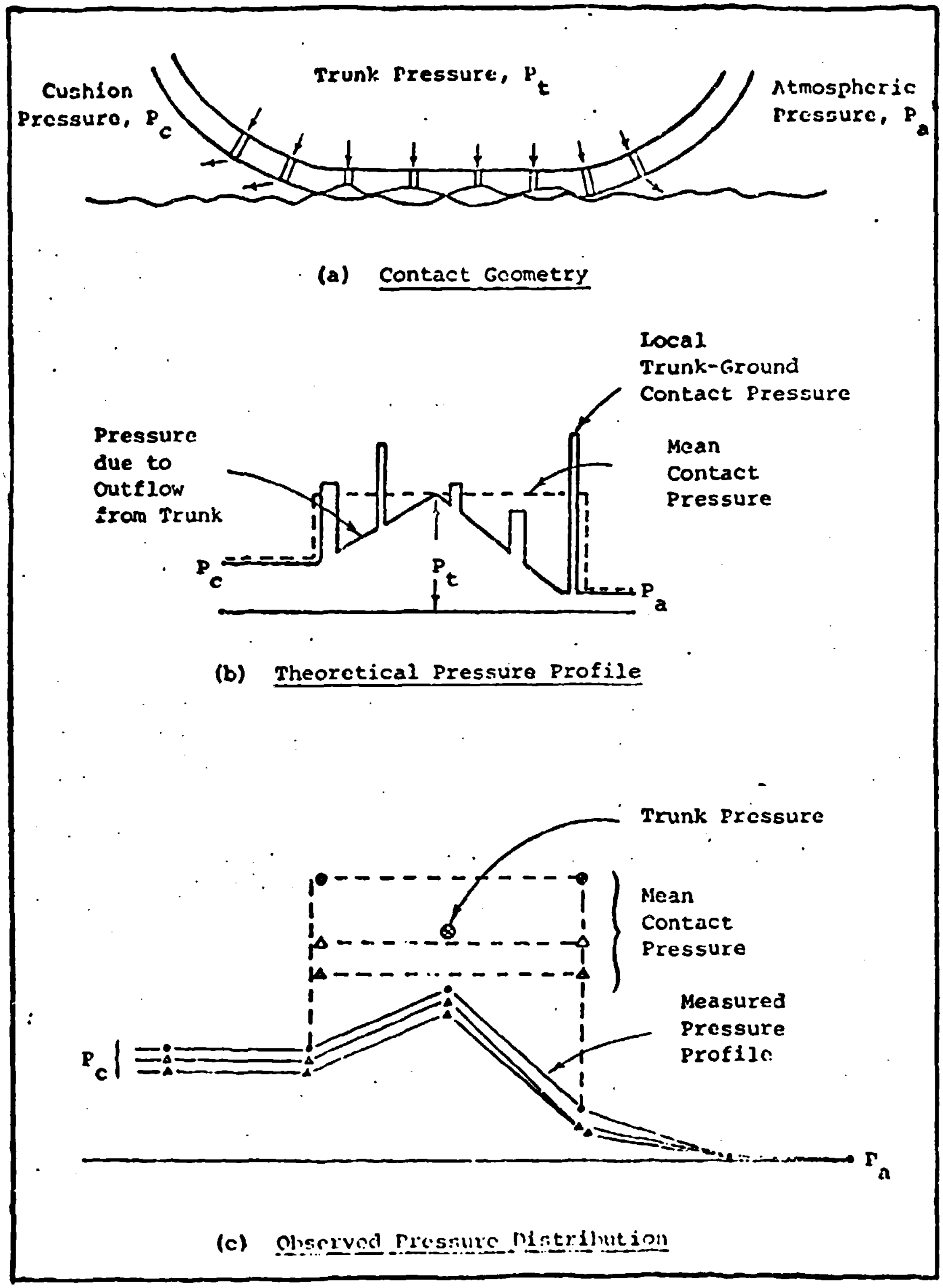


Fig. 12. Pressure Distribution in Trunk Contact Zone

regions is large for a "smooth" surface, it was convenient to define an average contact pressure, $P_{av} = F/A$, acting as though the bodies were touching uniformly over the entire area, A. In fact, P_{av} equals the trunk pressure, P_t. For purposes of trunk outflow calculation, the pressure profile in the non-contacting regions is approximated by a linearly decreasing relationship as shown in Fig. 12. The driving pressure for flow through any trunk hole is thus given by the difference between the trunk pressure and the gap pressure at that location. This pressure distribution model has been compared to experimental data and has been quite accurate (within 10%) (Ref. 4).

Trunk Damping. In dynamic operation, the trunk is deformed cyclically both in tension and flexure, and energy dissipation in the trunk material gives rise to a damping force which opposes the strain rate. Because the present trunk analysis does not solve for strain (and hence strain rate), a damping model that links trunk material properties directly to trunk damping forces cannot be developed. An alternate approach in which the damping characteristics are modeled by dimensional analysis (similarity) based on test data thus appears more appropriate. In keeping with the method of approach outlined earlier, the trunk is divided into segments (Fig. 13) and a series of dashpots, one for each segment, is included in the model such that the segment damping force is proportional to the vertical velocity of the trunk segment.

Each dashpot models the energy dissipation characteristic of the trunk segment. Although all parts of the trunk dissipate energy, the major contributions will come from those parts that undergo high stress reversals, since the strain rate is highest in these sections. Observations of a trunk in dynamic operation suggest that the high

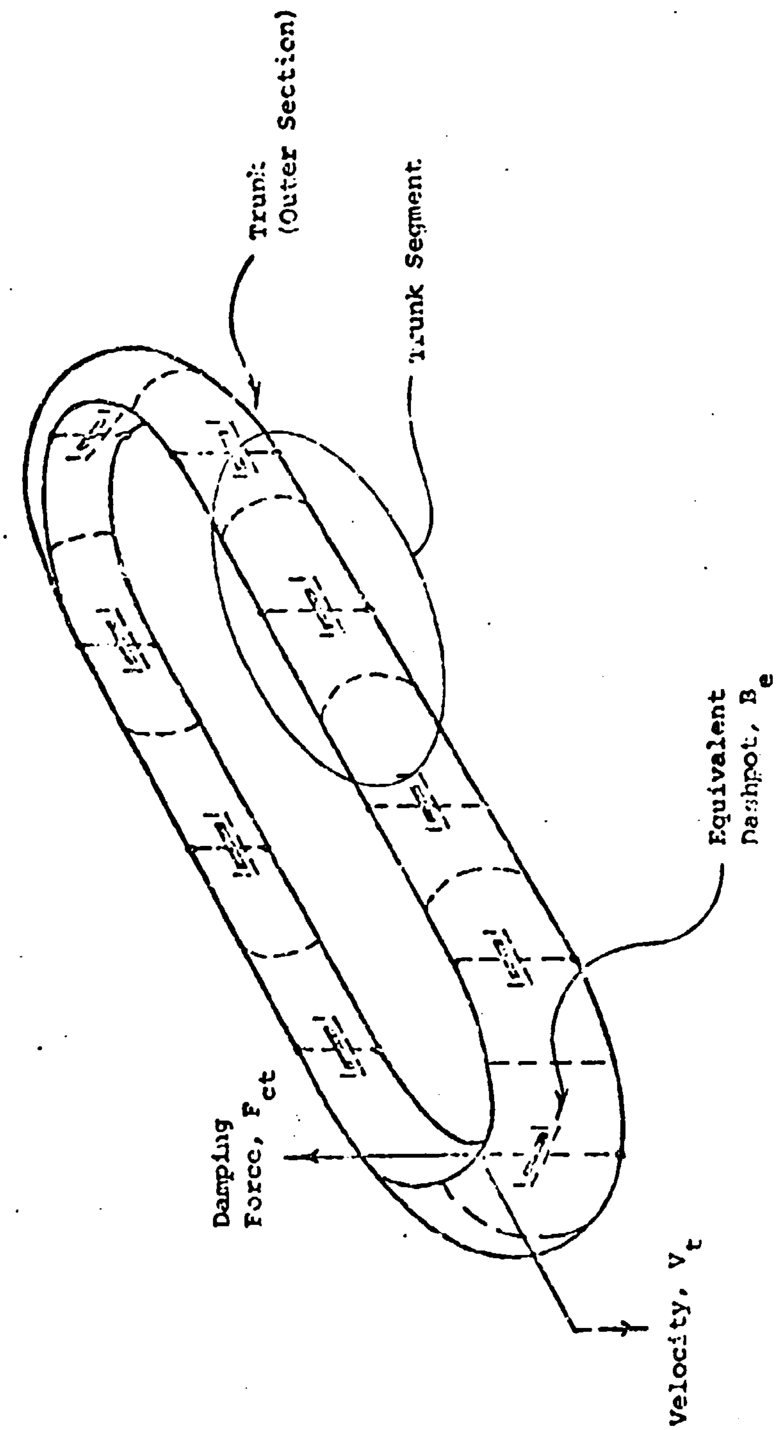


FIG. 13. The Trunk Damping Model

stress reversal regions lie along the periphery of the trunk-ground contact zone, because it is here that the rate of change of trunk slope (and hence stress) is high and constantly changing with the time as the contact area changes. As a first order approximation, the damping model derived here assumes that all the energy dissipation in the trunk is concentrated along the trunk-ground contact periphery so that the damping coefficient of each dashpot depends on the perimeter of the ground contact zone. This means that when a segment is not contacting the ground it has zero damping and when it is contacting the ground it has a damping coefficient proportional to the contact perimeter.

Model Synopsis

The Flow System

(a) The fan is characterized by a static pressure rise element for forward and back flow in series with an inertance (duct) and a capacitance (volume).

(b) The trunk and cushion volume are found from the Hybrid Trunk Model, which characterizes the side trunk segment as an ideal two-dimensional membrane and the end segment as a "frozen" trunk.

(c) The orifice areas between the trunk and cushion, trunk and atmosphere and cushion and atmosphere are found from the trunk shape as predicted by the Hybrid Trunk Model, along with the cushion orientation and ground profile.

(d) The pressure within the cushion, trunk and plenum is considered to be uniform.

(e) The pressure in the trunk/ground contact zone is found from the triangular profile given by the Hybrid Trunk Model.

(f) The flow through the plenum, trunk and cushion is governed by the unsteady state flow continuity equation in which the air is assumed to behave like a perfect gas and follow a polytropic expansion relationship.

(g) The flow through all orifices is found from the incompressible flow square-law orifice equation.

The Force System

(a) The mean contact pressure in the trunk/ground contact zone is equal to the trunk pressure.

(b) The trunk contact area and location relative to the aircraft CG is found from the trunk shape predicted by the Hybrid Trunk Model.

(c) The cushion area and location relative to the aircraft CG is found from the Hybrid Trunk Model. In width, the cushion extends between the lowest (ground tangent) points of the side trunk segments. In length, it extends between the ground tangent points of the end trunk segments, or, if in ground contact, between the inner edges of the contact zone.

(d) The total forces and moments acting on the aircraft occur due to the mean trunk contact pressure acting over the contact area, the cushion pressure acting over the cushion area, aerodynamic drag and trunk damping losses caused by aircraft heave motion, and trunk-ground friction.

(e) The forces and moments are found by dividing the cushion (and trunk) into segments, approximating the actual ground profile underneath the cushion by a similar set of segments parallel to the cushion, computing the cushion and contact pressure forces and moments for each segment, and then summing them to determine the total force and moment about the aircraft CG.

(f) The heave motion of the aircraft is found by applying Newton's law in the vertical direction to the aircraft CG.

(g) Angular accelerations in pitch and roll are obtained by applying the theorem of moment of momentum about the aircraft pitch and roll axes.

(h) A coordinate transformation is carried out to express vehicle frame velocities and accelerations in terms of Euler angles and their derivatives.

(i) The moment of momentum equations, expressed in terms of Euler angles are integrated to give the angular position of the aircraft as a function of time.

Chapter IV

Controller Design

During the low speed portion of the takeoff roll (to approximately 50 knots), the controls available are a yaw thruster on the rear fuselage and vertical roll thrusters on each wing tip. The roll thrusters can be directed up or down. During the takeoff sequence, the most unstable mode of the aircraft is the roll mode. Therefore, it was decided to control this mode and observe the control that was applied to the pitch mode through the inertial cross-coupling. Also, during takeoff the yaw angle will be controlled by the yaw thruster on the rear of the fuselage. With roll and yaw controlled, Eqn (4), which is rewritten here, can be simplified.

$$I_{xx}\dot{P} - I_{xz}(\dot{R} + PQ) + (I_{zz} - I_{yy})RQ = \mathcal{L} \tag{106}$$

Controlling roll and yaw means that $\dot{R}$ will be small and PQ and RQ (products of small numbers) will be small. Q can be considered small because the takeoff starts with zero initial conditions on P, Q, and R. With the above simplification and assuming that any roll inputs from the ground profile are impulsive, then the roll moment generated by the roll thrusters can be written as

$$\begin{aligned} I_{xx}\ddot{\phi} &= \mathcal{L}_{THRUSTERS} \\ &= l_w F_T \end{aligned} \tag{107}$$

so

$$\ddot{\phi}(t) = C F_T(t) \tag{108}$$

where

$$C = \frac{l_w}{I_{xx}} = 8.28 \times 10^{-3}$$

now let $$x_1 = \phi \tag{109}$$

$$x_2 = \dot{\phi} \tag{110}$$

and $$u(t) = F_T(t) \tag{111}$$

so Eqn (108) can be rewritten as

$$\dot{x}_2(t) = c\,u(t) \tag{112}$$

and $$\dot{x}_1(t) = x_2(t) \qquad \text{from Eqn (110)} \tag{113}$$

so $$\dot{\underline{x}}(t) = \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}}_{\underline{A}} \underline{x}(t) + \underbrace{\begin{bmatrix} 0 \\ c \end{bmatrix}}_{\underline{B}} u(t) \tag{114}$$

For a minimum time Performance Index and for $|u(t)| \leq U_{max}$ an optimal control for this system can be shown to be a bang-bang control (Ref. 14, pgs 245-248). In other words, the control is a maximum (either positive or negative) whenever it is applied. Since the eigenvalues for $\underline{A}$ are both zero, Theorems 5·4-1 and 5·4-3 of Kirk (Ref. 14) show that an optimal control exists, is unique and has at most one switching.

Therefore, the control for a specified initial state must be

$$u^*(t) = \begin{cases} +U_{max}, \ t_0 \leq t < t^*, \text{ or} \\ -U_{max}, \ t_0 \leq t < t^*, \text{ or} \\ +U_{max} \text{ for } t_0 \leq t < t_1 \text{ AND } -U_{max} \text{ for } t_1 \leq t < t^*, \text{ or} \\ -U_{max} \text{ for } t_0 \leq t < t_1 \text{ AND } +U_{max} \text{ for } t_1 \leq t < t^*, \text{ or} \\ 0 \quad \text{for } \underline{x}(t) = 0 \end{cases} \tag{115}$$

Integrating Eqns (112) and (113) with $U = \pm U_{max}$ gives

$$x_2(t) = C \int u(t)\,dt$$
$$= \pm C U_{max} t + C C_2 \quad (116)$$

where C_2 is the value of x_2 at $t = t_0$

and
$$x_1(t) = \int x_2(t)\,dt$$
$$= \int (\pm C U_{max} t + C C_2)\,dt$$

$$= \frac{\pm C U_{max}}{2} t^2 + C C_2 t + C_1 \quad (117)$$

where C_1 is the value of x_1 at $t = t_0$

solving for t in (116)

$$t = \frac{x_2(t) - C C_2}{\pm C U_{max}} \quad (118)$$

substituting t into (117)

$$x_1(t) = \frac{\pm C U_{max}}{2 C^2 U_{max}^2} \left(x_2^2(t) - 2 C C_2 x_2(t) + C^2 C_2^2 \right) + \frac{C C_2 (x_2(t) - C C_2)}{\pm C U_{max}} + C_1 \quad (119)$$

for $U = + U_{max}$

$$x_1(t) = \frac{1}{2 C U_{max}} x_2^2(t) - \frac{C_2 x_2(t)}{U_{max}} + \frac{C C_2^2}{2 U_{max}} + \frac{C_2 x_2(t)}{U_{max}} - \frac{C C_2^2}{U_{max}} + C_1$$
$$= \frac{1}{2 C U_{max}} x_2^2(t) + C_3 \quad (120)$$

where
$$C_3 = C_1 - \frac{C C_2^2}{2 U_{max}} \quad (121)$$

for $u = -U_{max}$

$$x_1(t) = \frac{-1}{2CU_{max}} x_2^2(t) + \frac{C_2 x_2(t)}{U_{max}} - \frac{CC_2^2}{2U_{max}} - \frac{C_2 x_2(t)}{U_{max}} + \frac{CC_2^2}{U_{max}} + C_1$$

$$= \frac{-1}{2CU_{max}} x_2^2(t) + C_4 \tag{122}$$

where $$C_4 = C_1 + \frac{CC_2^2}{2U_{max}} \tag{123}$$

the switching curve is

$$x_1(t) = \frac{-1}{2CU_{max}} x_2(t)\,|x_2(t)| \tag{124}$$

Let $$SX = x_1(t) + \frac{1}{2CU_{max}} x_2(t)\,|x_2(t)| \tag{125}$$

so

$$u^*(t) = -\begin{cases} -U_{max}, & SX > 0 \\ U_{max}, & SX < 0 \\ -U_{max}, & SX = 0,\ x_2(t) > 0 \\ U_{max}, & SX = 0,\ x_2(t) < 0 \\ 0, & \underline{x}(t) = 0 \end{cases} \tag{126}$$

It can be noted that this controller design is almost completely independent of the aircraft type. In the low speed range where aerodynamic controls are not effective, this design will help stabilize the roll mode of any aircraft. The only relationship between the aircraft and controller is that the thruster force is a function of roll inertia and wing span. Thus, this design becomes very versatile and applicable to stabilize the roll mode of any ACLS aircraft.

A somewhat similar analysis will be made for the controller of the yaw thruster. The criterion for directional control is to keep the aircraft on the runway centreline during takeoff; this can be accomplished by minimizing the lateral deviation, y, and rate of deviation from the centreline, $\dot{y}$. This deviation and rate will be minimized by yawing the aircraft in a direction to oppose the disturbance with the use of the yaw thruster.

Prior to the installation of the air cushion, the directional stability of the Jindivik was controlled by a batsman at the end of the runway. His job was to steer the dolly (Fig. 1) to keep the aircraft on the centreline. The same batsman will visually sense lateral deviation and deviation rate and control the yaw acceleration to indirectly control the lateral deviation and rate.

Assuming that the pitch and roll angles are kept small, then the lateral acceleration (to correct a lateral displacement) in the inertial frame of reference of the runway is a function of the thrust and yaw angle.

$$\ddot{y} \simeq \frac{THRUST}{MASS} \sin \Psi \tag{127}$$

and since Ψ will be small ($\leq 30°$) to keep the aircraft on the runway centreline, then

$$\ddot{y} \simeq \frac{THRUST}{MASS} \Psi \tag{128}$$

Equation 128 can be implemented as shown in Fig. 14. With these control loops the desired yaw angle to zero lateral displacement can be determined. Referring to Fig. 14

$$\Psi_d = -K_1 \dot{y} - K_2 y \tag{129}$$

the inner loop open loop transfer function is

$$(GH)_1 = \frac{KK_1}{S} \tag{130}$$

and the equivalent closed loop transfer function is

$$G_1 = \frac{GH}{1+GH} = \frac{KK_1}{S+KK_1} \tag{131}$$

the outer loop open loop transfer function is therefore

$$(GH)_2 = \frac{G_1 K_2}{S} = \frac{KK_1K_2}{S(S+KK_1)} \tag{132}$$

this gives a root locus as shown in Fig. 15. For a damping ratio of 0.7 the closed loop roots are located at $\left(-\frac{KK_1}{2}, -\frac{KK_1}{2}\right)$, which gives a closed loop transfer function of

$$G_2 = \frac{KK_1K_2}{\left(S+\frac{KK_1}{2} \pm j\frac{KK_1}{2}\right)} = \frac{KK_1K_2}{S^2+KK_1S+\frac{K^2K_1^2}{2}} \tag{133}$$

but

$$G_2 = \frac{(GH)_2}{1+(GH)_2} = \frac{KK_1K_2}{S^2+KK_1S+KK_1K_2} \tag{134}$$

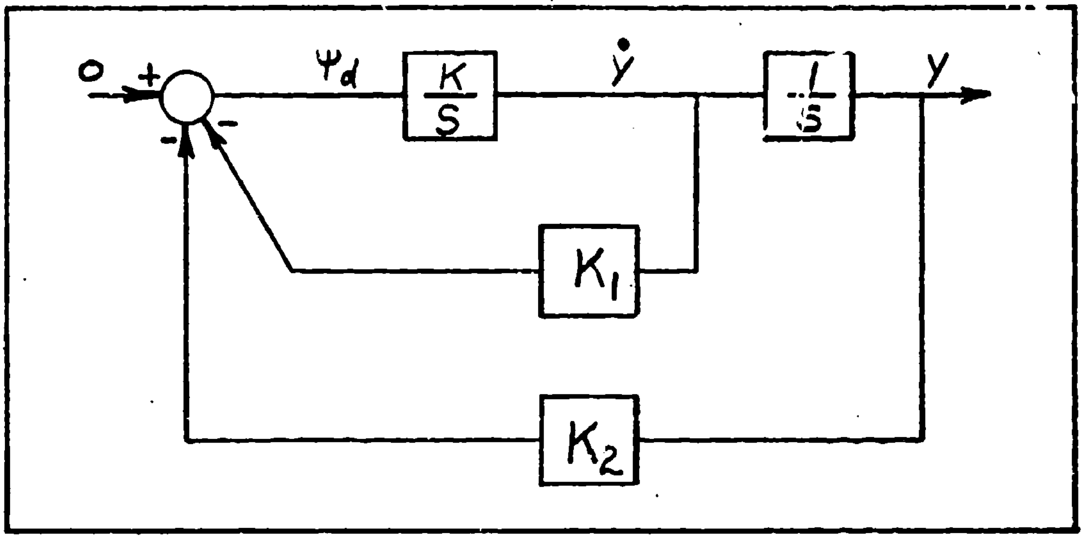


Fig. 14. Feedback Control Loops for Lateral Acceleration

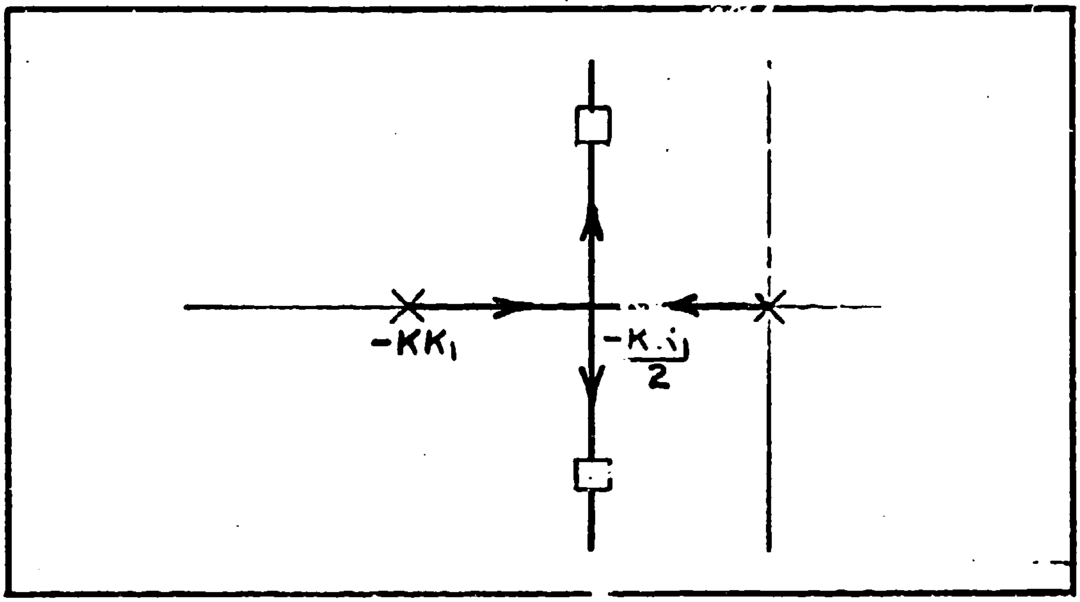


Fig. 15 Root Locus for Fig. 14

therefore equating denominator coefficients gives

$$K K_1 K_2 = \frac{K^2 K_1^2}{2}$$

or

$$K_2 = \frac{K K_1}{2}$$

$$= 9.1 K_1 \tag{135}$$

Since K_1 is arbitrary, a value of 0.3 was selected, so

$$K_1 = 0.3 \tag{136}$$

and

$$K_2 = 2.73 \tag{137}$$

so the desired yaw angle is

$$\psi_d = -0.3\dot{y} - 2.73y \tag{138}$$

The yaw angle is associated with the yaw thruster force by

$$I_{zz}\ddot{\psi} = \eta = F_{YT}\,\ell_w$$

$$\ddot{\psi} = C_5 F_{YT} \tag{139}$$

where

$$C_5 = 2.73 \times 10^{-3} \tag{140}$$

in matrix form

$$\begin{bmatrix} \dot{\psi} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \psi \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} 0 \\ C \end{bmatrix} F_{YT} \tag{141}$$

This equation is the same as Eqn. 114 for the roll thruster, and similarly a bang-bang control exists for which the switching function is

$$SXYT = \psi + \frac{\dot{\psi}\,|\dot{\psi}|}{2C\,F_{YTMAX}} \tag{142}$$

and the optimal control law is

$$F_{YT}^{*} = \begin{cases} -F_{YTMAX}, & SXYT > 0 \\ F_{YTMAX}, & SXYT < 0 \\ -F_{YTMAX}, & SXYT = 0,\ \dot{\psi} > 0 \\ F_{YTMAX}, & SXYT = 0,\ \dot{\psi} < 0 \\ 0, & \psi = 0,\ \dot{\psi} = 0 \end{cases} \tag{143}$$

The implementation of this control law would drive the yaw angle and yaw rate to zero in the minimum time, but the directional control problem requires that the yaw angle be equal to the desired angle, ψ_d. This can be accomplished by shifting the switching curve by the amount ψ_d.

$$SXYT = \psi - \psi_d + \frac{\dot{\psi}\,|\dot{\psi}|}{2C\,F_{YTMAX}} \tag{144}$$

This change in the switching curve means that the yaw rate and the quantity ($\psi - \psi_d$) will be driven to zero in the minimum time, or ψ will equal ψ_d.

Chapter V

The Computer Program and Simulation Results

General

The EASY Dynamic Analysis Program to Aircraft Modelling (Ref. 5) formed the major portion of the computer analysis and simulation and the air cushion system was modelled by the Foster-Miller model, as described in Chapter III. The computer program is listed in Appendix B. In brief, the EASY program provided the means for an analysis of the six degree of freedom rigid body dynamics of the Jindivik drone and the Foster-Miller model estimated the ground forces and moments transferred by the air cushion to the airframe. Additional FORTRAN was used to reflect the Jindivik autopilot and the designed roll, pitch and yaw controllers in the EASY program. Simulations were performed to obtain time history comparisons of the uncontrolled and controlled aircraft models with a crosswind driving function and an initial pitch angle to simulate flying off of a 2 inch step.

The Computer Program

Figures 16, 17, and 18 show the computer block diagrams of the aircraft dynamics, the longitudinal autopilot, and the lateral autopilot, respectively. Understanding the symbology used in these figures would require considerable referral to Reference 5, but a general description of the schematics will be given here.

In Figure 16, SD performs the six degree of freedom rigid body dynamics, AV calculates the aerodynamic variables, LO calculates the longitudinal force and moment sum, and LD calculates the lateral force and moment sum. The terms FX3S2, FZ3S2, TY3S2, TY3S2, TX3S2, and TZ3S2, shown feeding into LO and LD, are the sums of the engine and external

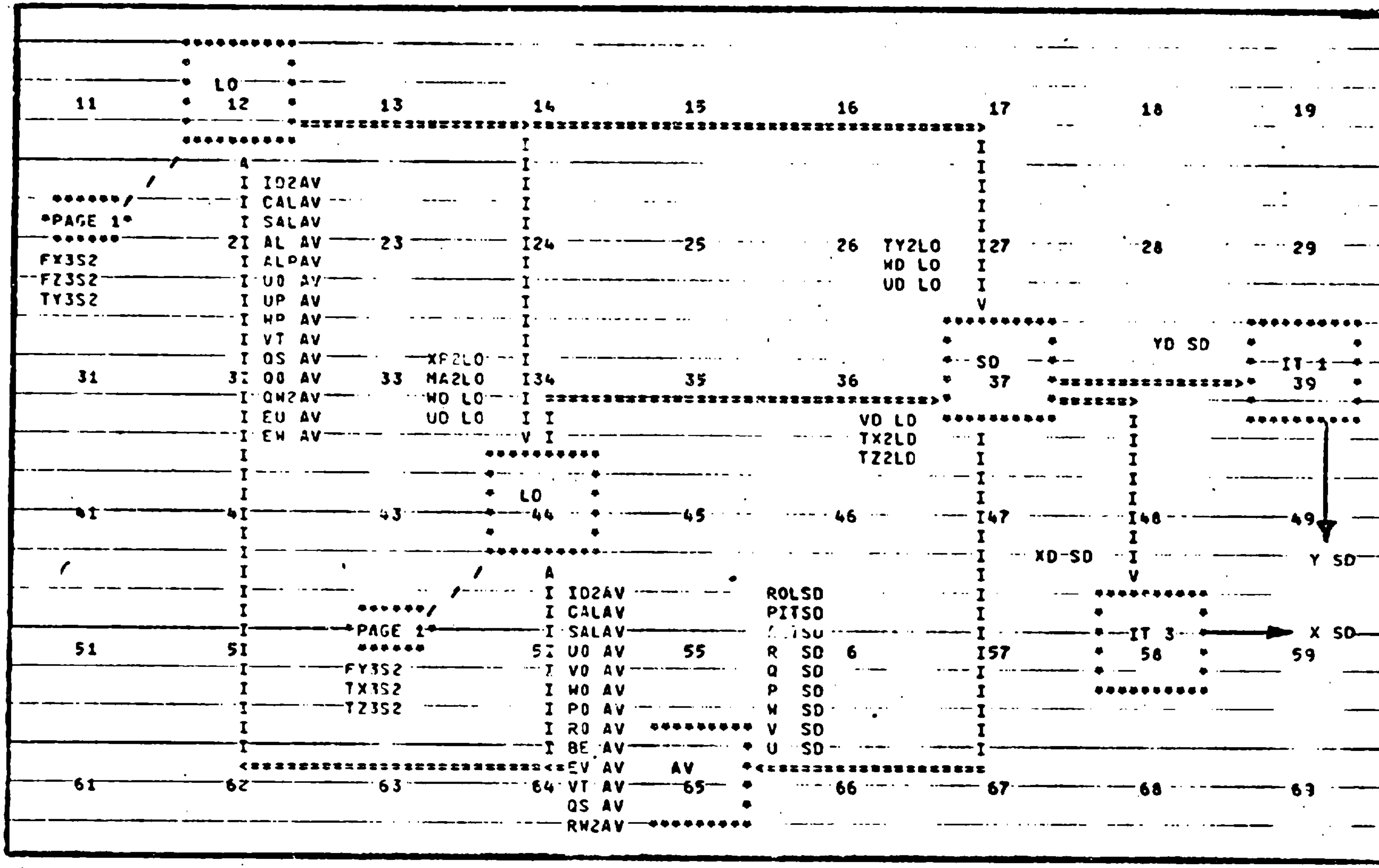


Fig. 16. Block Diagram of Aircraft Dynamics

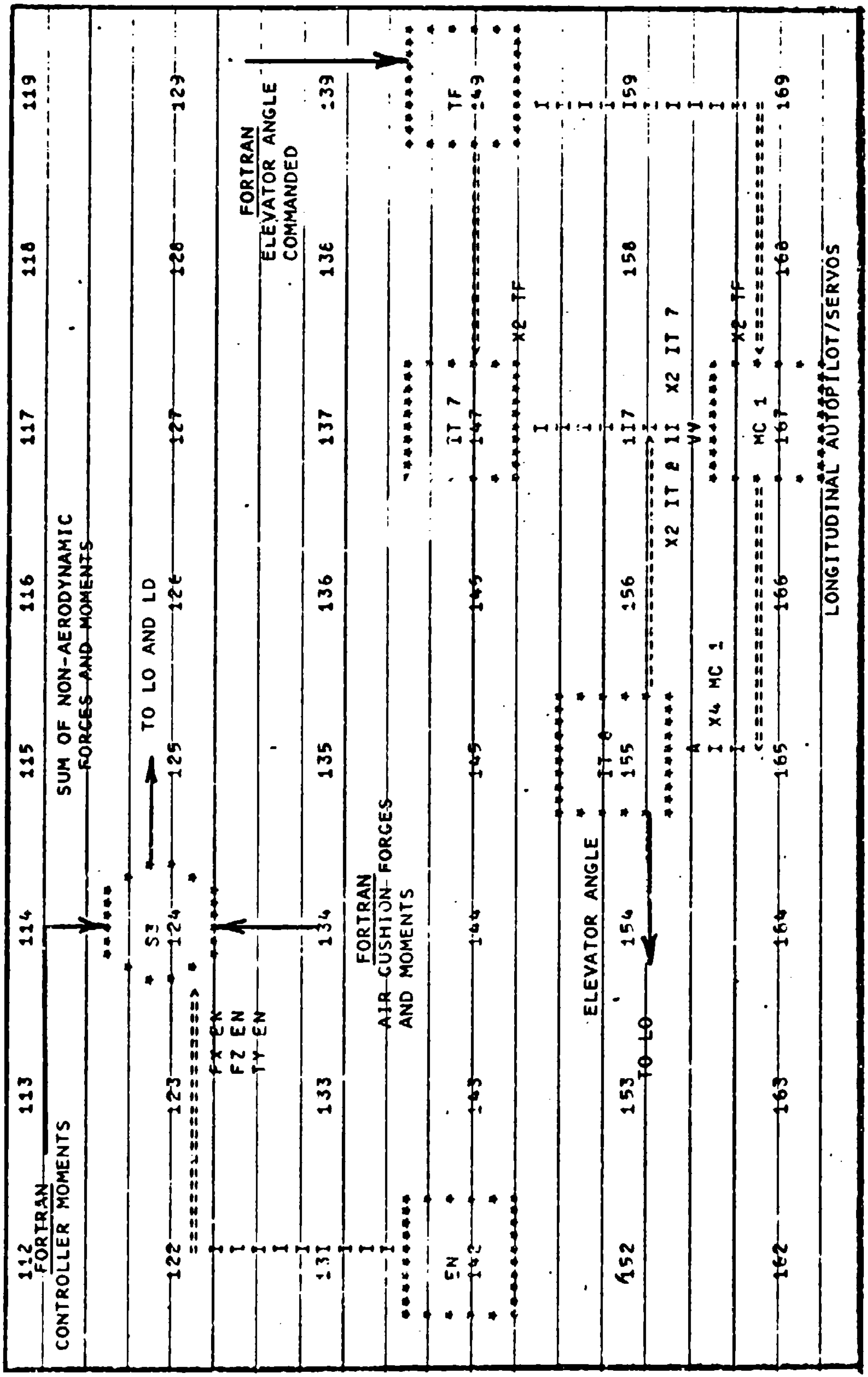


Fig. 17. Block Diagram of Longitudinal Autopilot

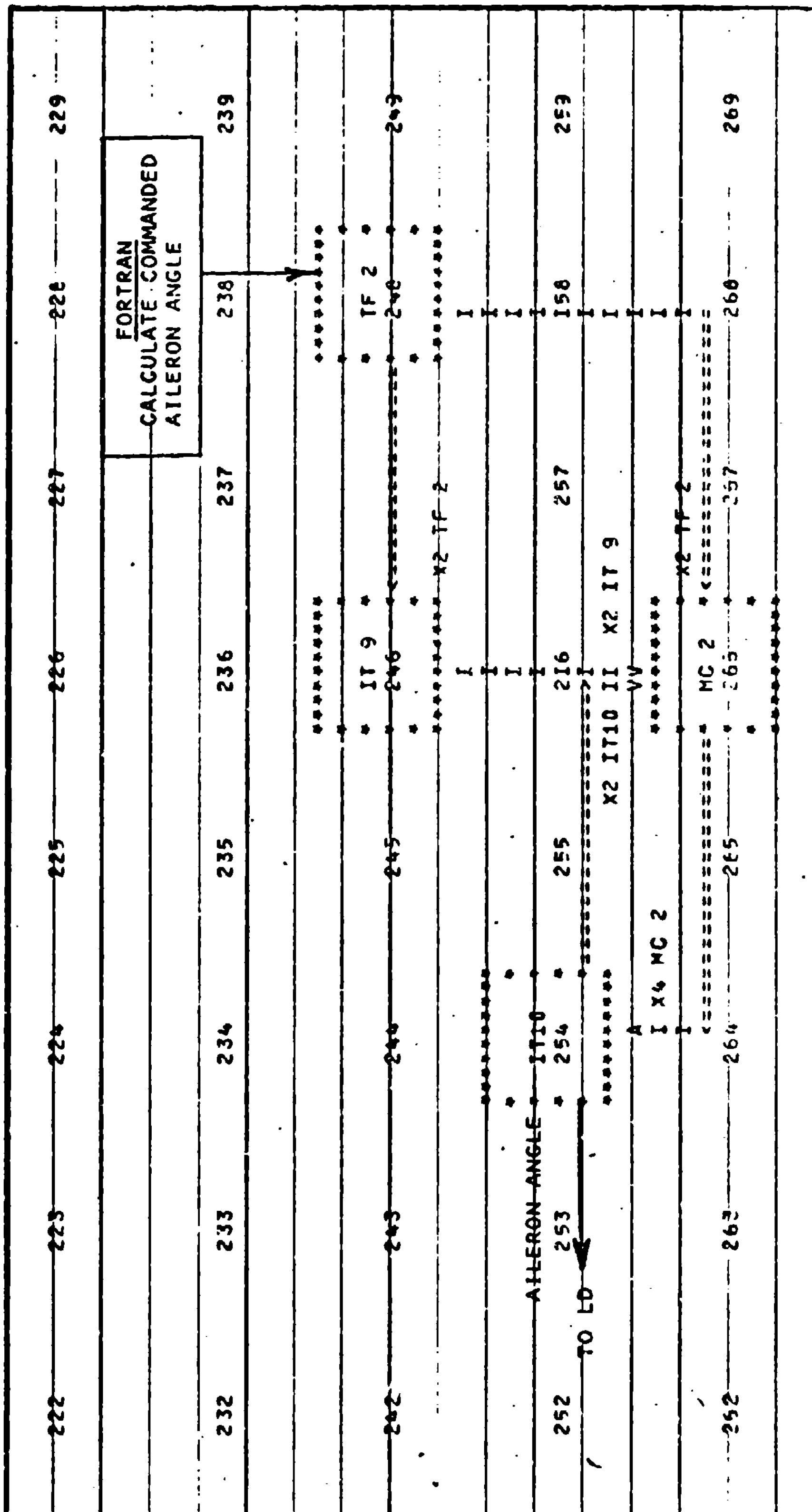


Fig. 18. Block Diagram of Lateral Autopilot

(i.e. air cushion system and controller) forces and moments.

The longitudinal and lateral autopilot functions shown in Figures 17 and 18 were developed from the elevator and aileron transfer functions given in Reference 3.. The maximum deflection of the control surfaces, the gearing ratios between control surfaces and servos, and the maximum slewing rates for the servos were also programmed into the model by FORTRAN.

Air Cushion Program

The air cushion system was programmed with ten subroutines to the EASY program and the flow chart of the subroutines is shown in Figure 19. The functions of the ten subroutines are as follows: FM is the main subroutine which calls and interacts with the remaining subroutines; it also determines the appropriate fan curve and contains the integration routine. HC, TK, SE, CO, PR, CL, S1, and SP form a set of subroutines which need the aircraft position, cushion and trunk pressures and ground profile as input parameters and they calculate various areas and volumes associated with those parameters. HC calculates the value of side trunk height for a given cushion to trunk pressure ratio. Subroutine TK takes that height and calculates trunk cross section parameters. From these parameters SE updates the trunk division parameters. Subroutine CO transforms position vectors for each trunk centre, from the vehicle frame to the ground frame, and then calculates the distance between each of the trunk segments and the ground, and it also calculates the ground coordinates above which each of the segments lie. Subroutine PR determines ground elevations (input by user) corresponding to the

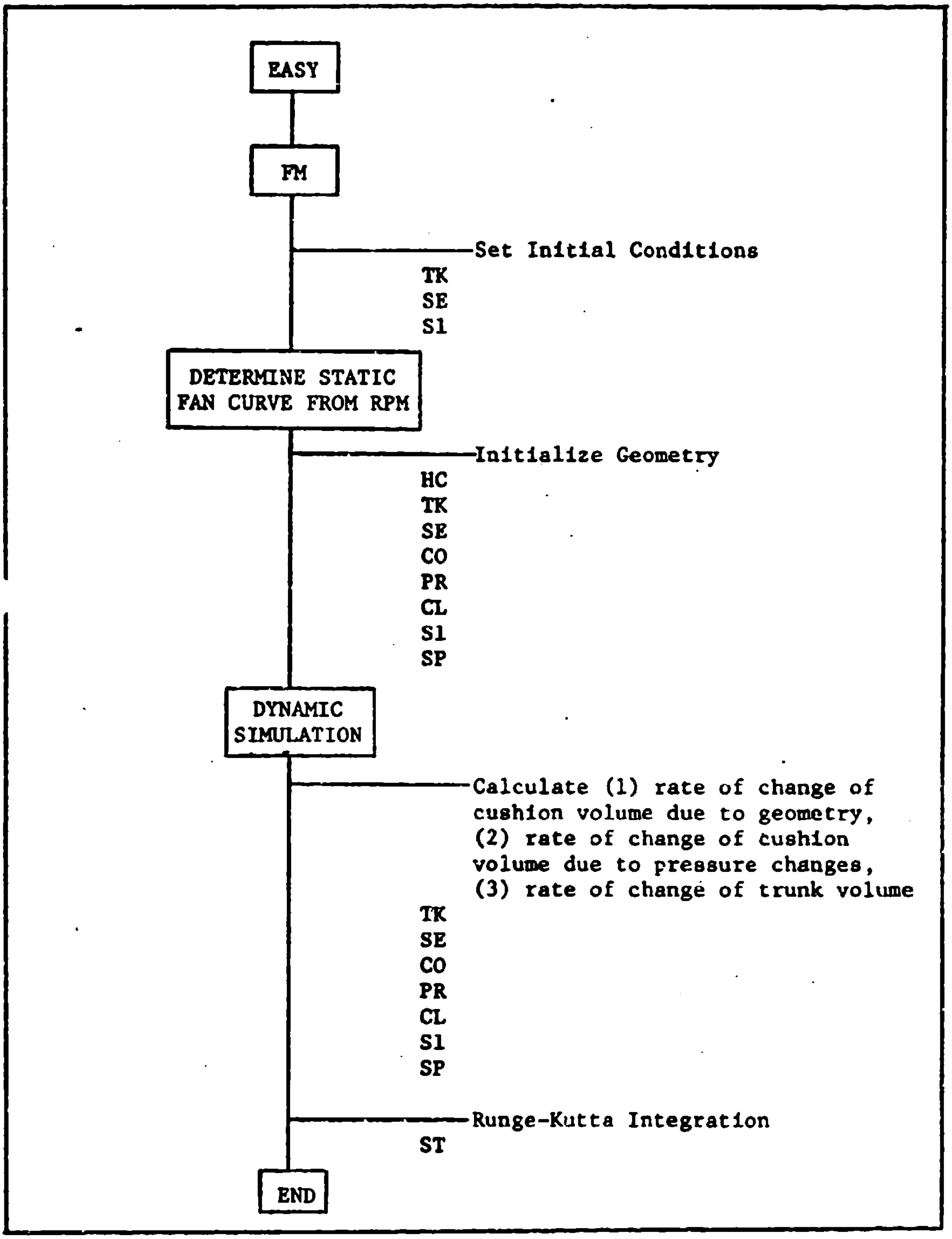


Fig. 19. Air Cushion Subroutine Flow Chart

ground coordinates generated by CO, and subroutine CL determines the hard surface clearance for each segment using (a) the ground elevation value and (b) the distance of the trunk segment from the ground. Finally, subroutine SP calculates values of different areas and volumes for each segment and adds them together to give total areas and volumes.

In addition to these seven subroutines, FM also calls subroutine ST. ST determines the value of fan flow for a given value of fan pressure rise and also calculates the forces and torques for a given ACTS orientation. Subroutine ST also incorporates all the system differential equations so that the value of the state differentials can be updated each time it is called by the Runge-Kutta integration routine. The forces and torques calculated by ST are passed to the EASY program via FM.

Simulation Results

The results of the simulation to test the controller designs of Chapter IV are shown in Tables I and II. For initial conditions of $-1°$ in pitch, $0°$ in roll, and a constant 40 ft/sec crosswind, the uncontrolled model induces a roll angle and pitch angle that are lightly damped in comparison to the controlled model. Also, the restoring yaw angle is less than the controlled yaw angle and the lateral deviation, y, is subsequently greater for the uncontrolled model. The 1.5 second simulation for the uncontrolled model required 15,000 seconds of computation time and consequently the simulation was not continued to the extent

Table I - Simulation Results Uncontrolled Model

TIME	PITCH	ROLL	YAW	ALTITUDE	LATERAL DEVIATION
0.0	-1.000	0.000	0.00	2.60	0.00
0.5	.305	.113	-.63	1.52	3.84
1.0	.451	.474	-1.94	1.02	8.78
1.5	.377	.142	-2.51	1.25	14.87

Table II - Simulation Results Controlled Model

TIME	PITCH	ROLL	YAW	ALTITUDE	LATERAL DEVIATION
0.0	-1.000	0.000	0.0	2.60	0.00
0.5	.273	.150	-.74	1.71	3.83
1.0	.282	.072	-2.05	1.25	8.77
1.5	.063	.023	-3.82	2.32	14.68
2.0	.038	.005	-5.99	2.31	21.23
2.5	.045	↓	-8.52	2.27	27.87
3.0	.052	↓	-11.37	2.23	34.12
3.5	.048	↓	-14.46	2.32	39.56
4.0	↓	↓	-17.70	2.32	43.74
4.5	↓	↓	-21.10	2.32	46.22
5.0	↓	↓	-24.50	↓	46.52
5.5	↓	↓	-27.80	↓	44.15
6.0	↓	↓	-30.90	↓	38.56
6.5	↓	↓	-33.40	↓	29.21
7.0	↓	↓	-34.80	↓	15.69

Chapter VI

Conclusions and Recommendations

The National Aeronautics and Space Administration (NASA) has accepted the Foster-Miller program as a valid air cushion model; however, in the course of the analysis of this thesis, several problems arose which prevent this program from becoming an effective design tool. The primary problem is the excessive computation time required for dynamic simulation when it is incorporated with the EASY Dynamic Analysis Program. The Fourth Order Runge-Kutta integration routine used in the air cushion model requires a time increment of 0.001 seconds for numerical stability. Since this integration routine is the prime reason for the excessive computation time, it is recommended that the routine be changed or augmented to reduce computation time.

The air cushion model assumes that the trunk is an elliptical shape rather than the actual shape, in which the aft end is 10% wider than the fore end. This discrepancy impinges on trunk and cushion volumes and areas, pitching and rolling moments, clearance and gap areas, etc. In other words, it requires considerable evaluation and extensive modification and verification of the program to change trunk shapes. It is recommended that Foster-Miller Associates be asked to modify their model to accommodate different trunk shapes as future designs may require.

The air cushion model has no provision to orient the trunk orifices other than perpendicular to the trunk surface. In fact, the Jindivik trunk orifices are drilled inward at a 45° angle to produce more cushion pressure in the region of trunk contact. Some adjustment should be made to the model to allow this orifice orientation as a design parameter. Also, the model uses a single curve to describe the fan characteristics of outflow vs. drive pressure, but the actual characteristics are dependent on more variables; hence, a fan "map" is required to replace the single curve and adequately describe the fan during all phases of its operation.

A weak part of the computer simulation is the evaluation of the Jindivik Stability Derivatives due to the fact that the static wind tunnel data was extrapolated from the Recovery Trunk Data and was suspect from the beginning of the analysis. Consequently, it is recommended that wind tunnel tests be conducted in a moving belt tunnel with the takeoff trunk and with measurement of the rate variables p, q, and r. Barring this option, the development of the derivatives should be reviewed and ammended with the use of more sophisticated data reduction techniques. Once the computer program, shown in Appendix B, is changed to encompass the previous recommendations, it can be used to define the following parameters:

a) Operational limits and directions of crosswinds.

b) A "ground roughness" criteria above which the aircraft becomes unstable.

c) A flap deflection schedule to provide minimum takeoff distance within pitch stability. The present two flap settings would provide a step input to pitch and hence should be changed.

d) All of the above for different vertical thruster sizes and locations.

This thesis has integrated the EASY Dynamic Analysis Program and a truncated version of the Foster-Miller air cushion model to simulate an air cushion vehicle during takeoff. During the process of that integration and simulation, some major deficiencies in the Foster-Miller model have been highlighted. This thesis has also developed and demonstrated a technique to control bang-bang thrusters on the wing tips and a bang-bang thrust deflector on the tail section. Complete verification of the controller design was not possible due to the large computer resources that would have been required, but the results do show the control trends that are expected. The application of wing tip and yaw thrusters to other air cushion aircraft should provide comparable results. Also, these thrusters could be used on Vertical or Short Takeoff and Landing (V/STOL) aircraft because these aircraft also have marginal stability and require control enhancement in the low speed range.

Bibliography

1. Air Cushion Launch Skirt-Jindivik. Blueprint 03481-5X2024E, B. F. Goodrich Aerospace and Defense Products, Akron, Ohio, June 1974.

2. Blakelock, J. H. Automatic Control of Aircraft and Missles. Wiley, 1965.

3. Boeing Aerospace Co. Contract Report F33615-77-C-3054, June 1977.

4. Bogani, A. B., et al. Heave, Pitch, and Roll Analysis and Testing of Air Cushion Landing Systems. NASI-12403, Langley Research Center, Hampton, Virginia, March 1977.

5. Burrough, J. D. and Warren, A. W. Applications of the EASY Dynamic Analysis Program to Aircraft Modelling. User's Manual BCS40120, Boeing Computer Services, Inc., Seattle, Washington, 1976.

6. Captain, K. M., et al. Dynamic Heave-Pitch Analysis of Air Cushion Landing Systems. NASA CR-2530, May 1975.

7. Diedrick, F. W. A Plan-Form Parameter for Correlating Certain Aerodynamic Characteristics of Swept Wings. NACA TN 2335, April 1951.

8. Digges, K. H. Theory of an Air Cushion Landing System for Aircraft. TR-71-50, Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio, June 1971.

9. Etkin, B. Dynamics of Flight: Stability and Control. Wiley, New York, 1959.

10. Fan Transition Assembly-Jindivik. USAF Blueprint 13690214-012, Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio.

11. Fitzgerald, Robert E. Jindivik Yaw Control Thrusters. Chandler-Evans Inc., West Hartford, Connecticut, August 1975.

12. General Arrangement of ACRS and ACTS-Jindivik Ground Tests. USAF Blueprint 13690214-001, Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio.

13. Government Aircraft Factories (Australia), GAF Project Report Jindivik No. B3B1/77 (Wind Tunnel Data), September 1976.

14. Kirk, Donald E. Optimal Control Theory-An Introduction. Prentice-Hall Inc., New Jersey, 1970.

15. McRuer, Askenas, and Graham, Aircraft Dynamics and Automatic Control. Prentice-Hall Inc., New Jersey, 1973.

16. Preliminary Remotely Piloted Vehicle Air Cushion Landing System Static and Ground Tests on the Jindivik Drone. TR-77-52 (Unpublished), Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio.

17. Roskam, J. Flight Dynamics of Rigid and Elastic Airplanes. Roskam Aviation and Engineering Corporation, Lawrence, Kansas, 1975.

18. USAF Stability and Control Datcom. Flight Control Division, Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio, January 1975.

Appendix A

Graphs of Aerodynamic Coefficients

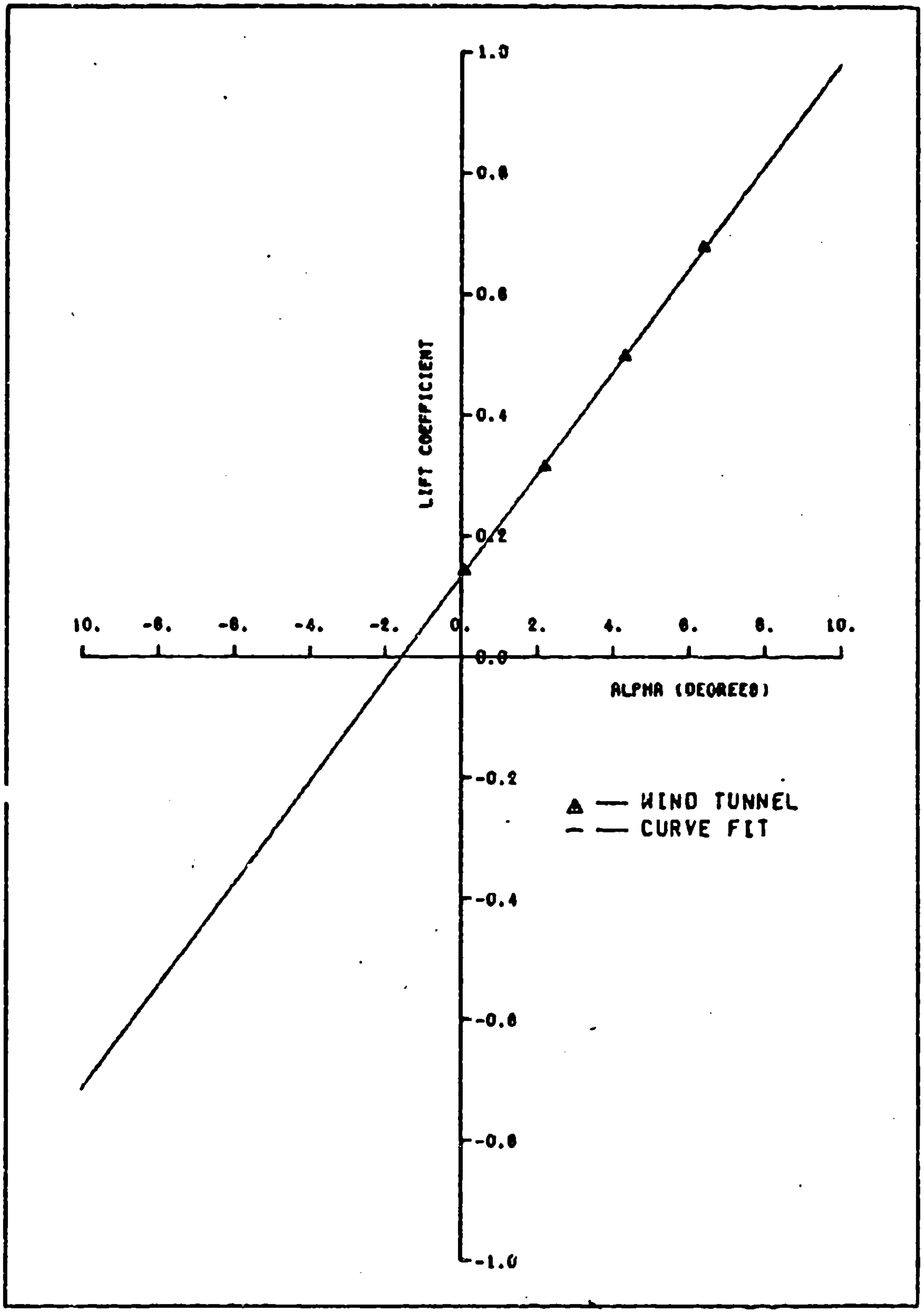


FIG. A-1 LIFT COEFFICIENT VERSUS ANGLE OF ATTACK

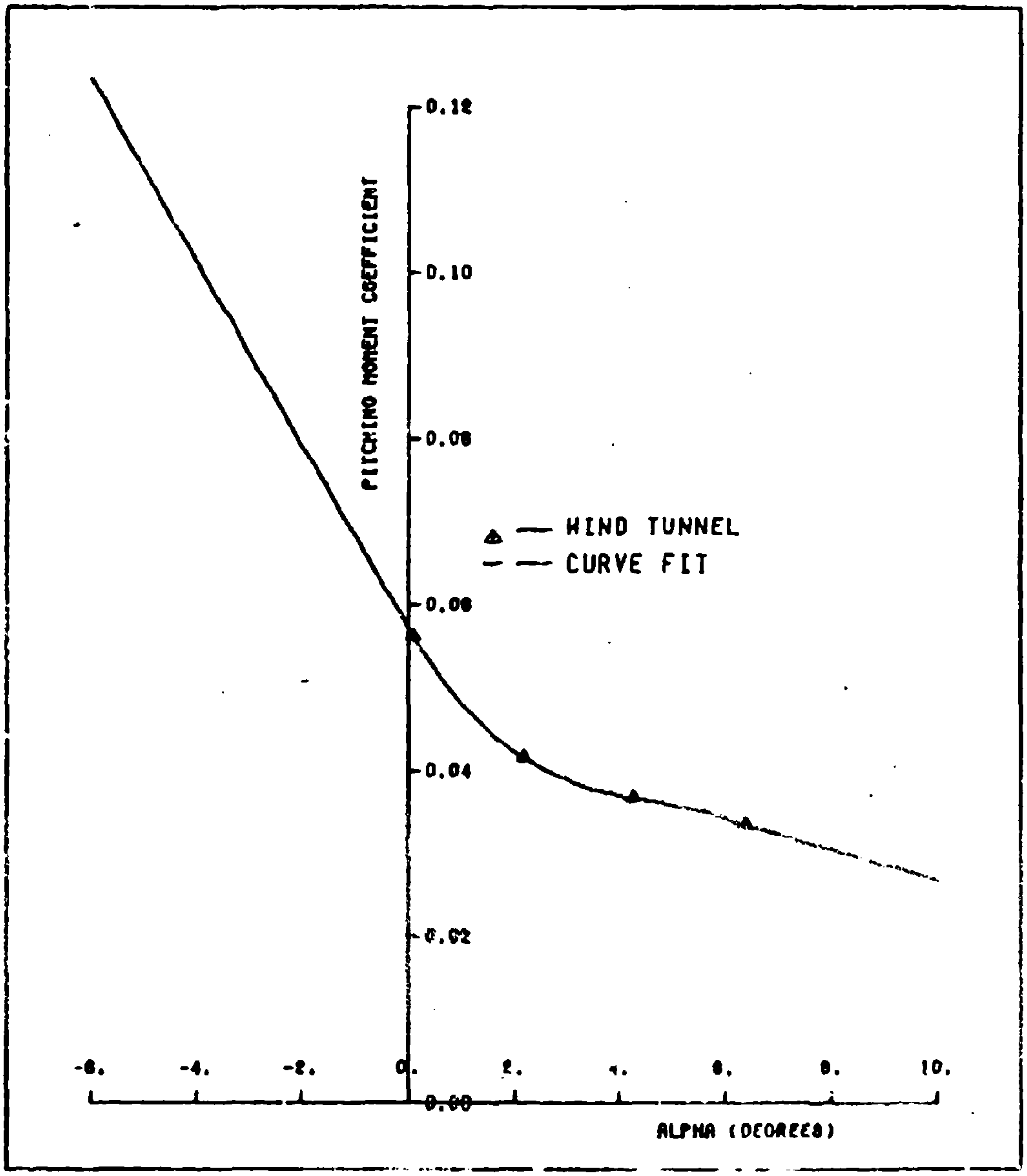


FIG. A-2 PITCHI[illegible] [illegible]OMENT COE[illegible]EN[illegible]
VERSU[illegible] [illegible]GLE OF ATTAC[illegible]

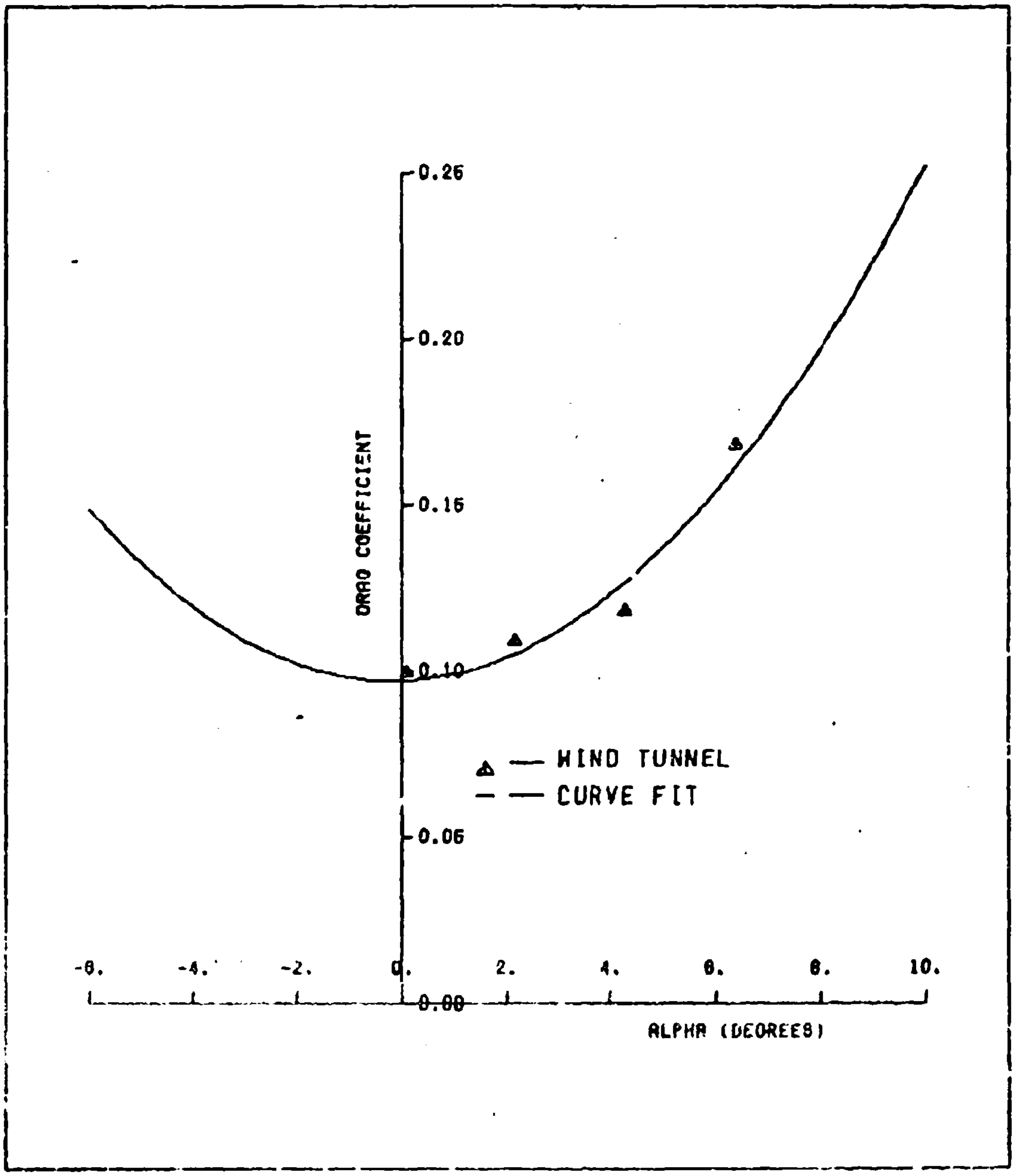


FIG. A-3 DRAG COEFFICIENT VERSUS ANGLE OF ATTACK

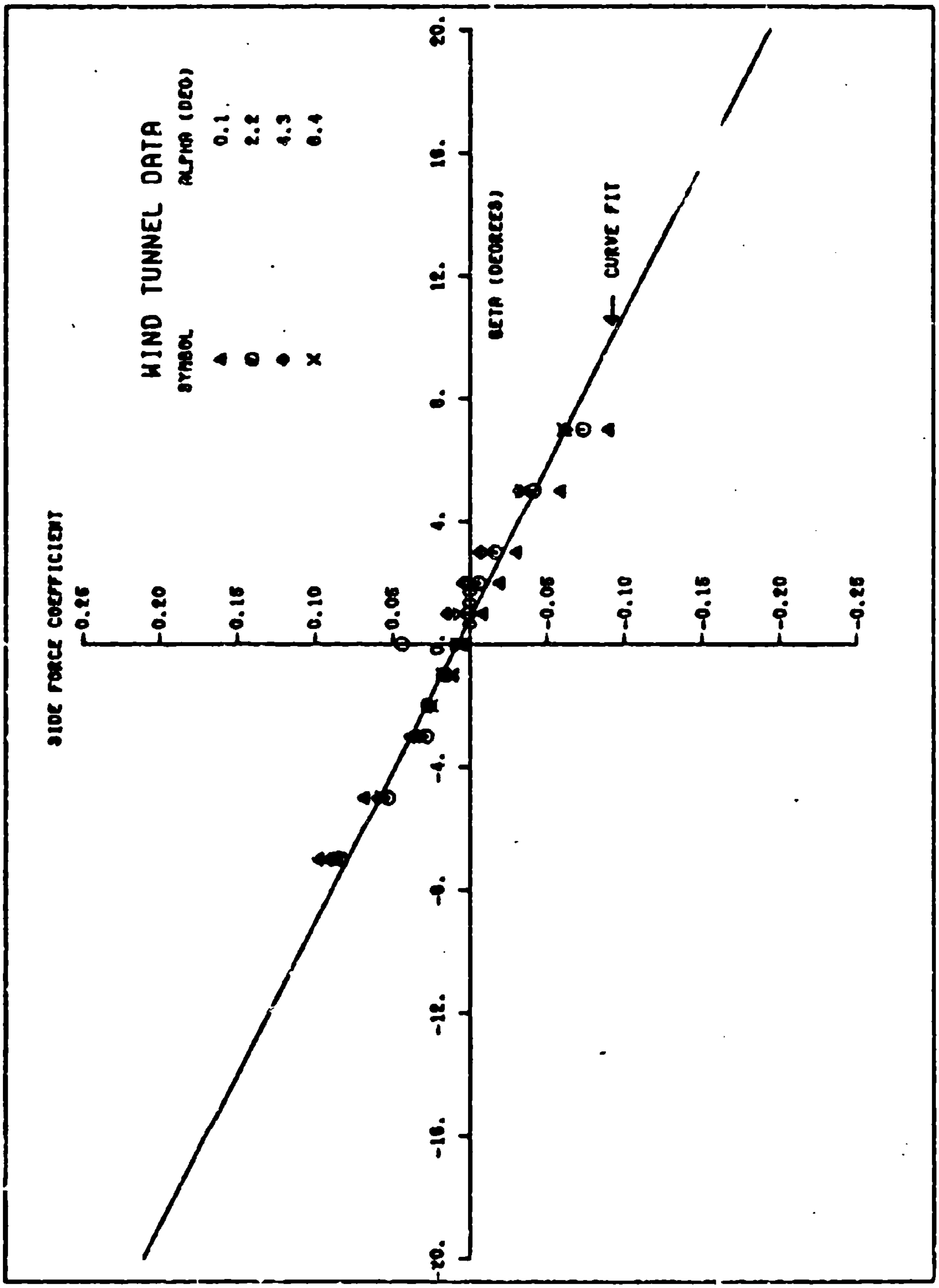


FIG. A-4 SIDE FORCE COEFFICIENT VERSUS SIDESLIP ANGLE

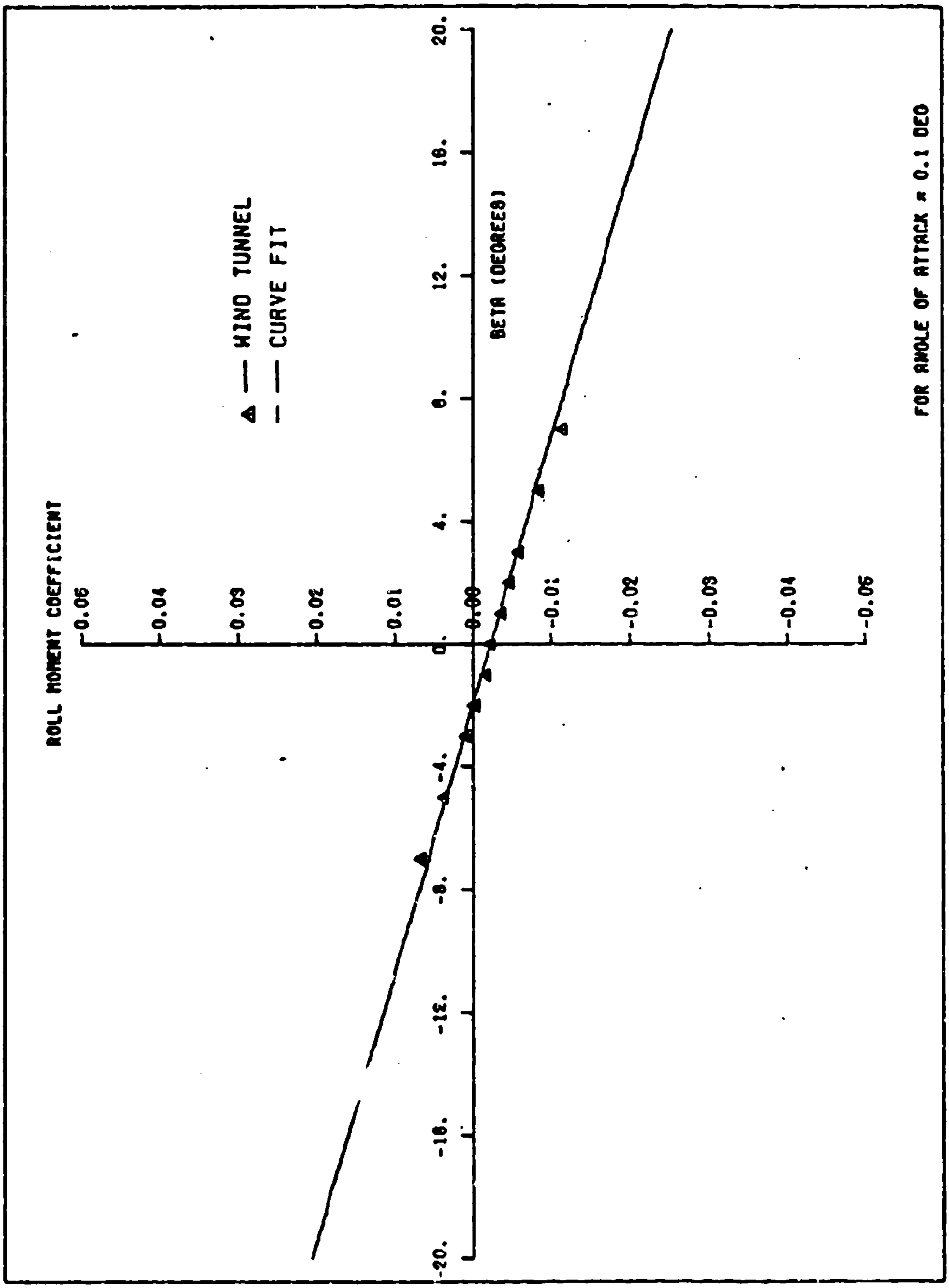


FIG. A-5 ROLL MOMENT COEFFICIENT VERSUS SIDESLIP ANGLE

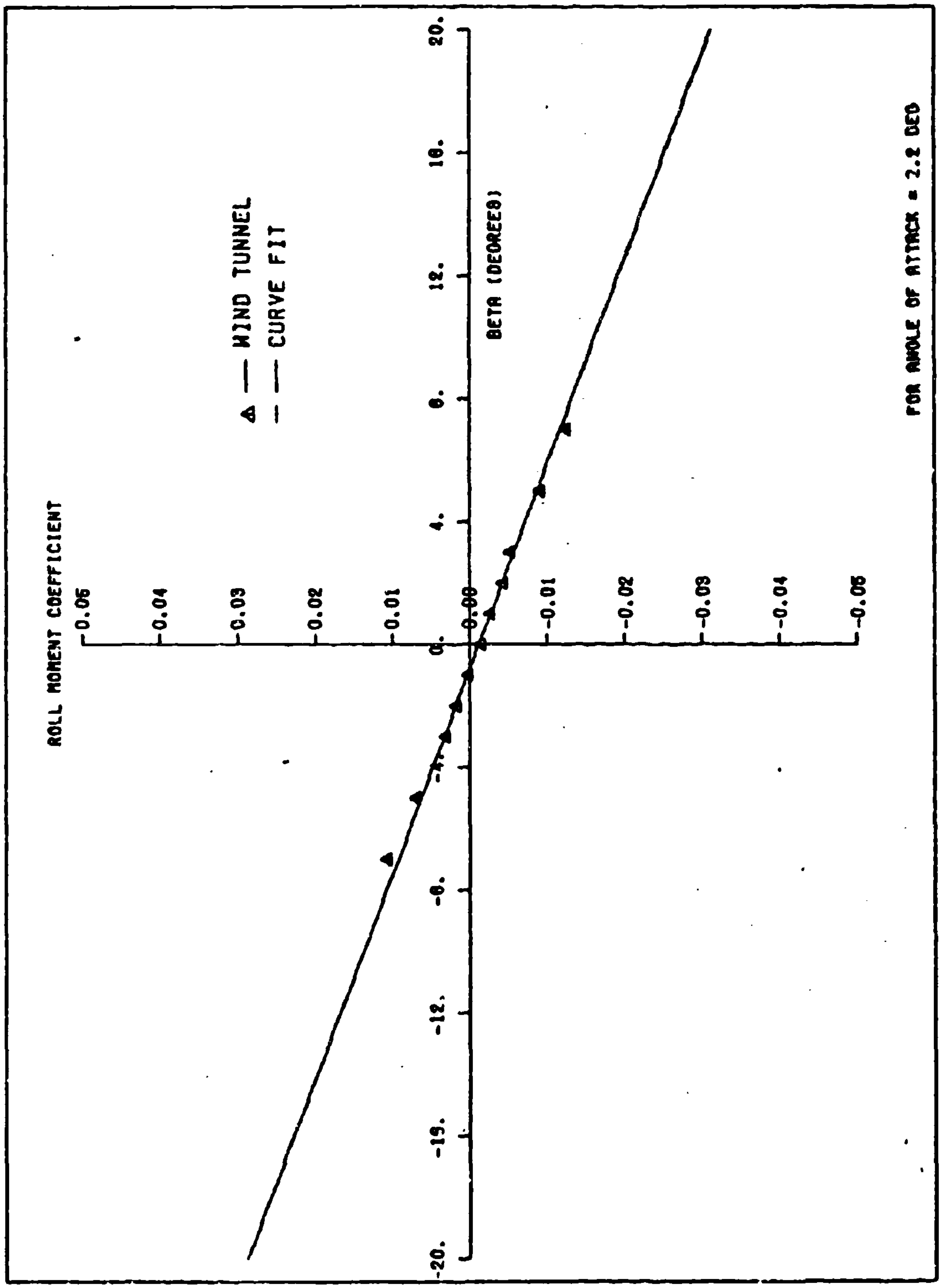


FIG. A-6 ROLL MOMENT COEFFICIENT VERSUS SIDESLIP ANGLE

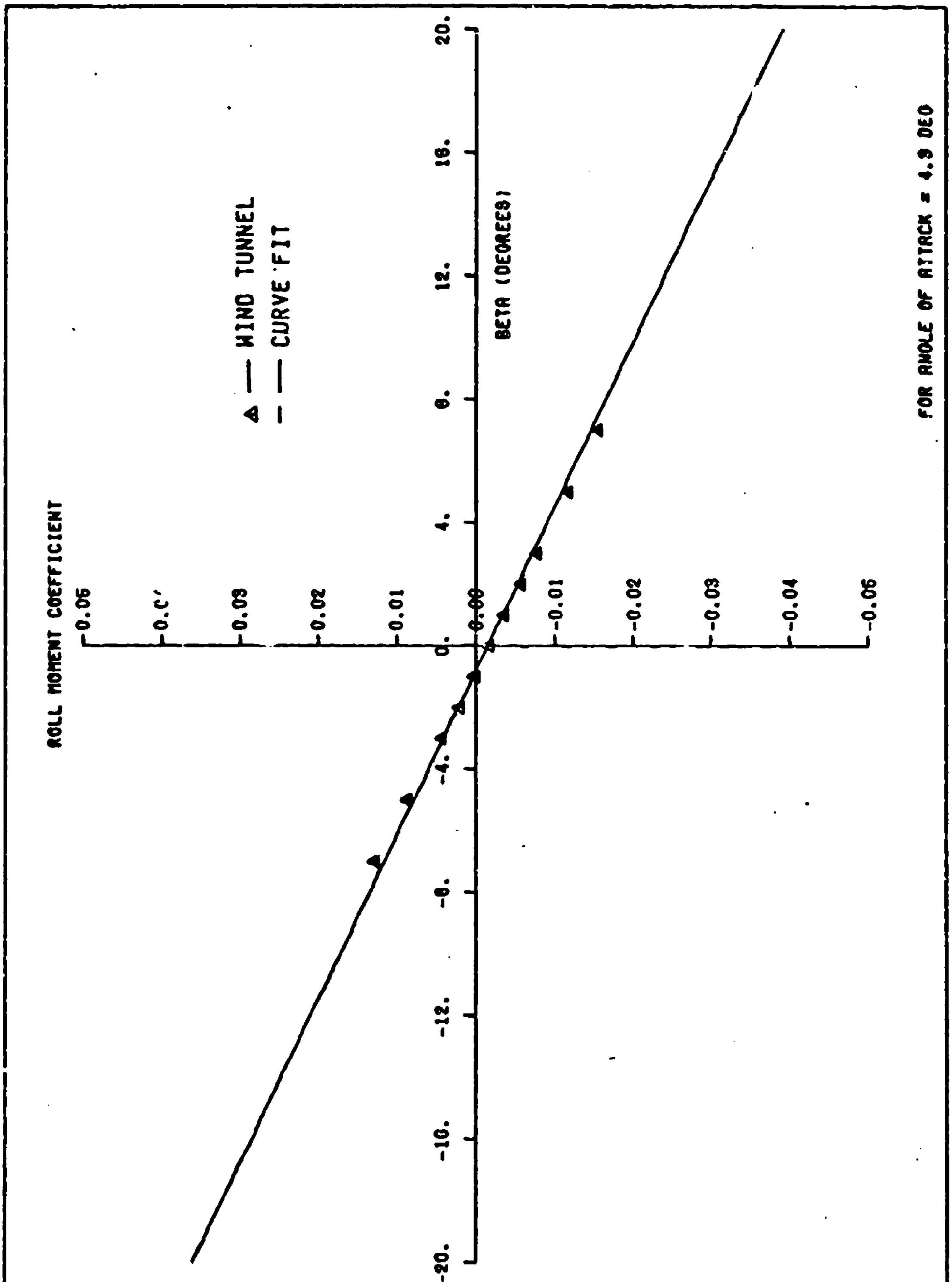


FIG. A-7 ROLL MOMENT COEFFICIENT VERSUS SIDESLIP ANGLE

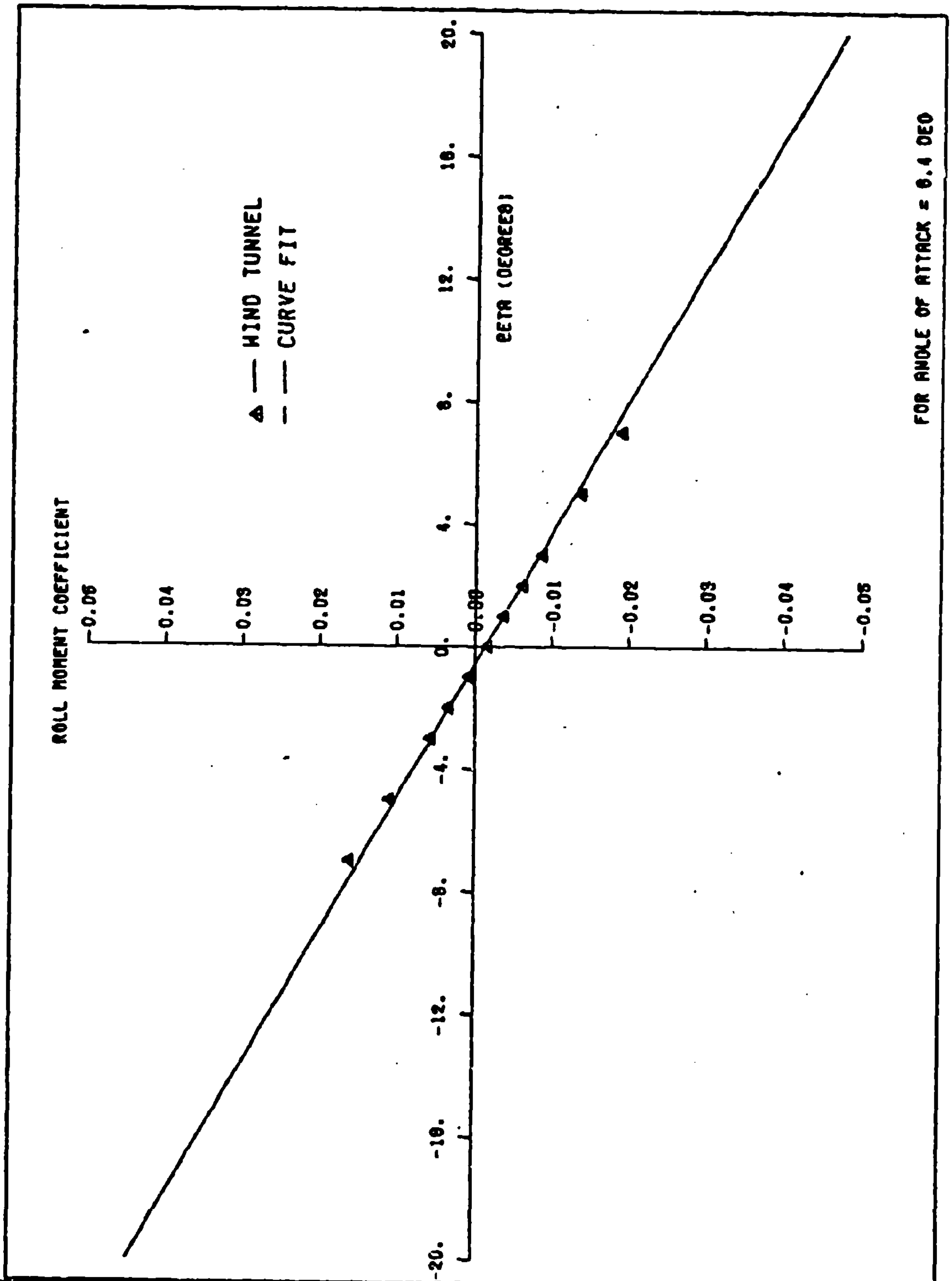


FIG. A-8 ROLL MOMENT COEFFICIENT VERSUS SIDESLIP ANGLE

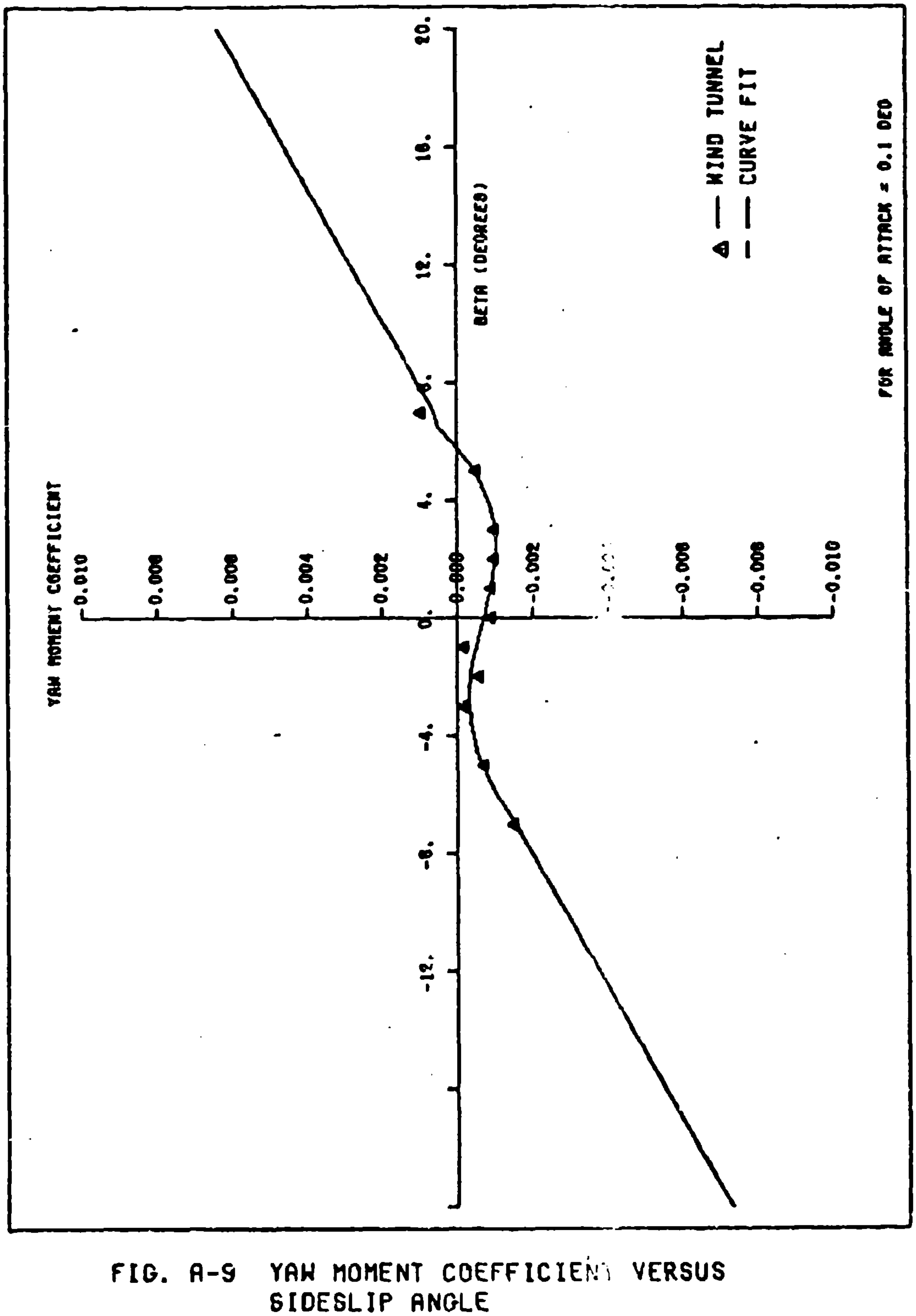


FIG. A-9 YAW MOMENT COEFFICIENT VERSUS SIDESLIP ANGLE

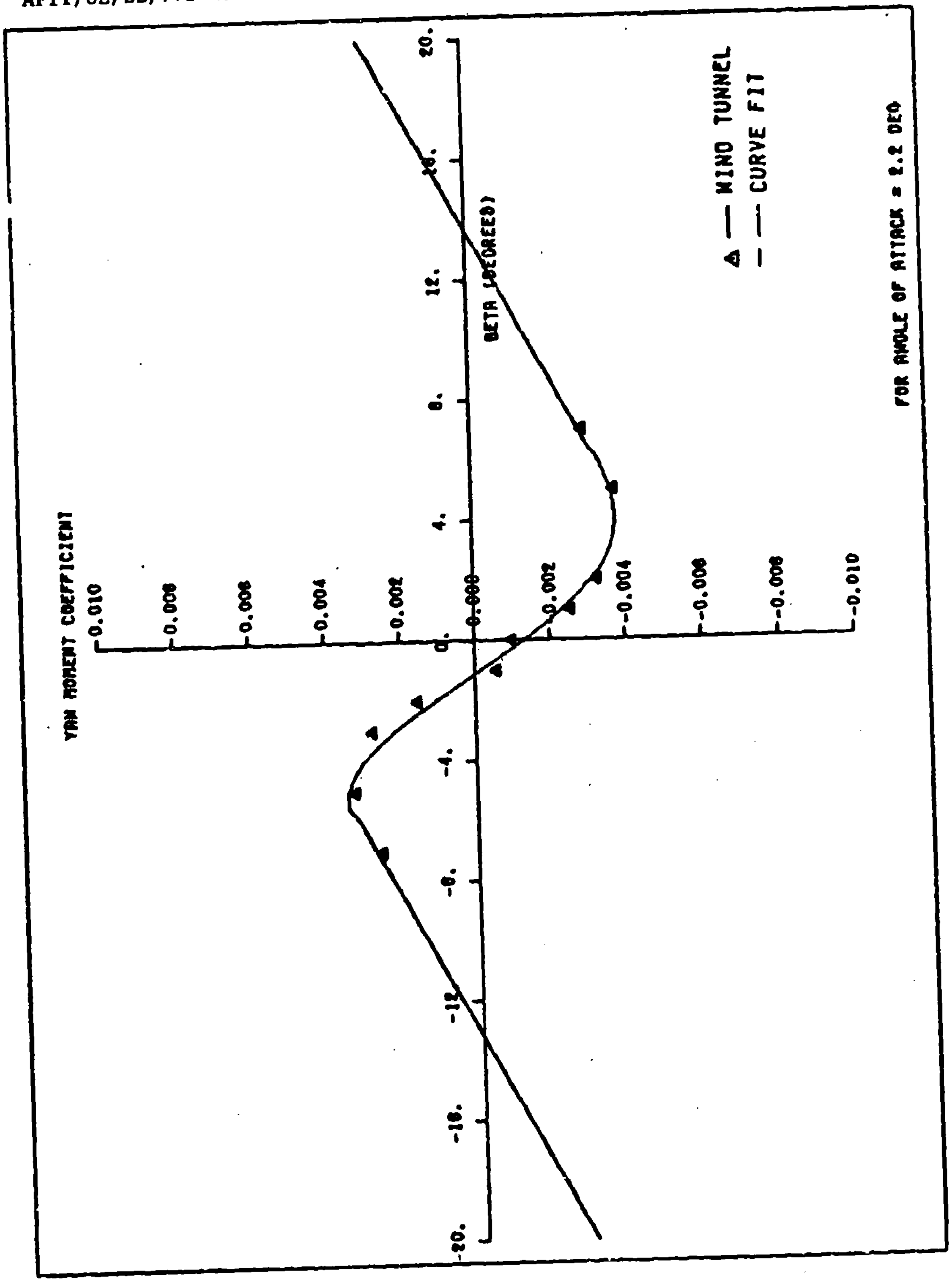


FIG. A-10 YAW MOMENT COEFFICIENT VERSUS SIDESLIP ANGLE

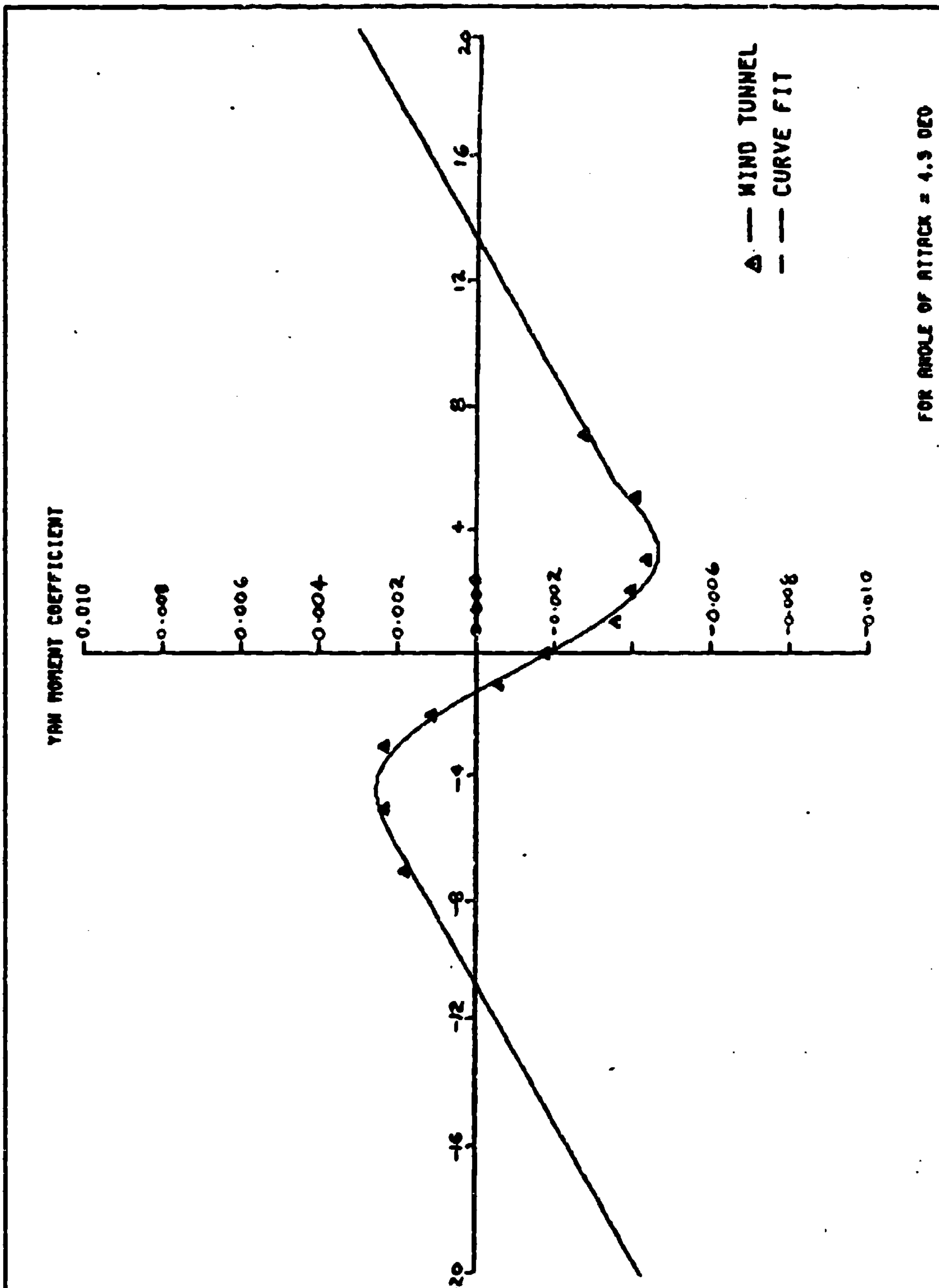


FIG. A-11 YAW MOMENT COEFFICIENT VERSUS
SIDESLIP ANGLE

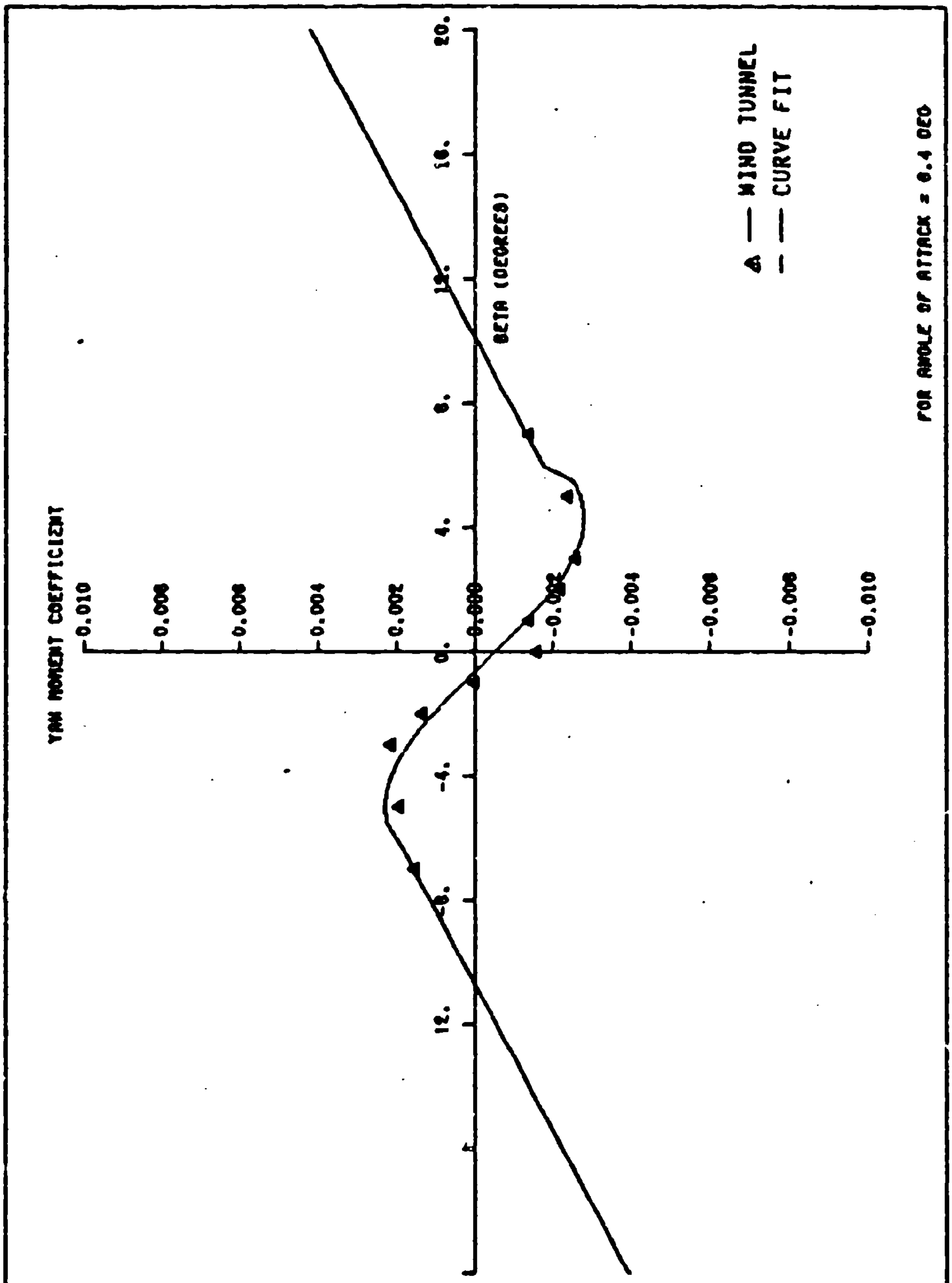


FIG. A-12 YAW MOMENT COEFFICIENT VERSUS SIDESLIP ANGLE

Appendix B

Computer Program Listing

```
MODEL DESCRIPTION JINDIVIK ACTS (CONTROLLED)
ADD STATES=PPLFM,PTKFM,PCHFM
ADD PARAMETERS=A,E,L,LS,LX,MSS,   D,CC,FF,GG,     VCD,AH,       HYI,SH,
                   CKK,PAT,GEC.                       TEM,
                JPV,                              CAF,CFX,CGP,CPA,CTC,PHA,
        CFZ,CPC,CPT,CTA,DPC,HDC               AAT,   AIF,                APC,APA,
      APT
ADD PARAMETERS=UMAX,UYTMAX
ADD VARIABLES=LCELS,RPM,R1 TK,R2 TK,  SKTFM,                    MDELS,BE SE,
   DXSE,LB,LBS,LDA,LP,LPS,LP,LRS,MAD, D2IS1,A1 S1,A2 S1,SI S1,DI S1,
  DXIS1,BEIS1,L1IS1,L2IS1,P1IS1,P2IS1,SNIS1,PHEDOT,THEDOT,    VTKS2,
  VCHS2,ACHS2,FY ST,TZ ST,TX ST,MWDS,                  HY FM,MO,MO,PFNFM,
  QFNFM,QFXFM,MDE,MDS,NPS,CSSFM,PRTFM,VCSFM,RHOFM,DVKFM,VPLFM,DVPFM,
  NP,NR,DVCFM,                      ATAS2,ATCS2,ACNS2,AACS2,ACCS2,     NPS,
      NDELS,NB,NRS,DPTFM,QTCFM,QTAFM,ATKCH,ATKAT,ATKCHC,ATKATC,OCHAT
      ,L1 TK,L2 TK,PH1TK,PH2TK
LOCATION=65,AV,INPUTS=SO
FORTRAN STATEMENTS
C     ALL ANGLES IN DEGREES
      VS AV=100.  $FLAP=1. $OS=OS AV $ELE=ELELO $ALT=ALTSO $BE=BE AV
      RO=RO AV $PO=PO AV $VT=VT AV $QO=QO AV $AL=AL AV
      RADIAN=.0174532
      VH AV=40.*COS(YAWSO*RADIAN)
      BR=BE*RADIAN $AR=AL*RADIAN
C     CALCULATION OF COEFFICIENTS
      CLAFR=1.41 $ZFIN=3. $PUD=1. $C=4. $S=76. $RHO=23.77E-4 $B=19.
      STAIL=13. $XTAIL=10. $CLATR=3.05 $SFIN=7.2 $XFIN=10. $CLADR=1.5
      CLQR=(2.*CLATR*STAIL*XTAIL)/(C*S)
      CMQR=-2.*(XTAIL**2)*CLATR*STAIL/((C**2)*S)
      CYPR=2.*CLAFR*SFIN*XFIN/(S*B)
      CYPP=-2.*SFIN*CLAFR*ZFIN/(S*B)
C     INITIALIZE CONSTANTS
      CL0=.13 $SLOPE=.352 $CLALD=.0857 $TL=.156 $CLALR=4.91
      GOEFF=.1-.005*ALT $CLAIL=-.193*(1.-4.84E-9*VT**2)/57.3
      IF(ALT.GT.20.)GOEFF=0.
   12 CL=(CL0+CLALD*AL+GOEFF)*COS(BR)
      CD1=.075+TL*CL**2+(1.6E-4*(ELE+10.)**2)
      CDALR=2.*TL*CLALR*CL
      CMADR=-2.*CLATR*(XTAIL**2)*STAIL*SLOPE/((C**2)*S)
      CNRR=-2.*CLAFR*SFIN*(XFIN**2)/(S*(B**2))-CD*RO*B/(8.*VT)
      CNPR=-2.*CLAFR*ZFIN*SFIN*XFIN/(S*B**2)-CLALR*PO*C/(8.*VT)+CDALR/6.
      CLPR=-CLALR/12.
      CLRR=2.*CLAFR*SFIN*XFIN*ZFIN/(S*B**2)+RO*CL*B/(8.*VT)
      CLDAR=34.73*CLALR*C/(S*B)
      CLELR=(5.43E-6*ELE**2-2.E-5*ELE+4.73E-3)*57.3
      CM=(-1.47E-4*AL**3+2.09E-3*AL**2-1.1E-2*AL+.057)*COS(BR)
      CMELR=(-19.2E-6*ELE**2 5.22E-5*ELE-1.2E-2)*57.3
      IF(ABS(BE).LT.60.) GO TO 210
      CYAW=SIN(BR)*5.6E-2
  210 IF(ABS(BE).LE.30.) GO TO 200
      CROLL=-SIN(BR)*.2858
      CY=-SIN(BR)*3.1270
  200 IF(AL.LT.1.)GO TO 170
      IF((AL.GE.1.).AND.(AL.LT.3.))GO TO 180
      IF((AL.GE.3.).AND.(AL.LT.5.))GO TO 190
      IF(AL.GE.5.)GO TO 195
C   ANGLE OF ATTACK IS LESS THAN 1.0
  170 CD=CD1+(2.75E-6*ABS(BE)**3+2.25E-4*ABS(BE)**2-6.07E-4*ABS(BE))*COS
     X(BR)
      IF(ABS(BE).GE.60.) GO TO 2000
      IF(BE.LT.-7.) GO TO 30
      IF(BE.GE.7.) GO TO 40
      CYAW=-5.6E-9*BE**5+1.26E-7*BE**4+1.06E-5*BE**3+1.98E-6*BE**2-2.1E-
     X4*BE-7.03E-4
```

```
      CY=-4.39E-3*BE**3+1.68E-5*BE**2-1.33E-2*BE+2.56E-3
      CROLL=-1.148E-3*BE-2.355E-3
      GO TO 2000
 30   CYAW=-.42E-4*BE+1.54E-3
      IF(ABS(BE).GT.30.) GO TO 2000
      CY=-4.39E-3*BE**3+1.68E-5*BE**2-1.33E-2*BE+2.56E-3
      CROLL=-1.148E-3*BE+2.355E-3
      GO TO 2000
 40   CYAW=4.42E-4*BE-.0025
      IF(ABS(BE).GT.30.) GO TO 2000
      CY=-4.39E-3*BE**3+1.68E-5*BE**2-1.33E-2*BE+2.56E-3
      CROLL=-1.148E-3*BE-2.355E-3
      GO TO 2000
C   ANGLE OF ATTACK IS LESS THAN 3.0
 180  CD=CD1+(-5.14E-6*ABS(BE)**3+3.03E-4*ABS(BE)**2-2.37E-4*ABS(BE)) *C
     XOS(BR)
      IF(ABS(BE).GE.60.) GO TO 2000
      IF(BE.LT.-7.) GO TO 50
      IF(BE.GE.7.) GO TO 60
      CYAW=-7.0E-9*BE**5-1.07E-6*BE**4+1.85E-5*BE**3+7.42E-5*BE**2-1.13E
     X-3*BE-1.34E-3
      CY=3.03E-7*BE**5+2.14E-6*BE**4-9 .83E-5*BE**3-1.64E-4*BE**2-7.21E-
     X3*BE+8.16E-3
      CROLL=-1.591E-3*BE-1.26E-3
      GO TO 2000
 50   CYAW=-.3E-4*BE+5.67E-3
      IF(ABS(BE).GT.30.) GO TO 2000
      CY=3.03E-7*BE**5+2.14E-6*BE**4-9 .83E-5*BE**3-1.64E-4*BE**2-7.21E-
     X3*BE+8.16E-3
      CROLL=-1.591E-3*BE-1.26E-3
      GO TO 2000
 60   CYAW=4.3E-4*BE-5.89E-3
      IF(ABS(BE).GT.30.) GO TO 2000
      CY=3.03E-7*BE**5+2.14E-6*BE**4-9 .83E-5*BE**3-1.64E-4*BE**2-7.21E-
     X3*BE+8.16E-3
      CROLL=-1.591E-3*BE-1.26E-3
      GO TO 2000
C   ANGLE OF ATTACK IS LESS THAN 5.0
 190  CD=CD1+(-4.12E-5*ABS(BE)**3+2.5E-4*ABS(BE)**2-2.34E-4*ABS(BE))*COS
     X(BR)
      IF(ABS(BE).GE.60.) GO TO 2000
      IF(BE.LT.-7.) GO TO 70
      IF(BE.GE.7.) GO TO 80
      CYAW=-4.14E-7*BE**5-7.87E-7*BE**4+4.36E-5*BE**3+6.54E-5*BE**2-1.47
     XE-3*BE-1.83E-3
      CY=2.93E-6*BE**3+3.72E-7*BE**4-2.83E-4*BE**3-1.9E-5*BE**2 -4.06E-3
     X*BE+1.36E-2
      CROLL=-1.977E-3*BE-1.555E-3
      GO TO 2000
 70   CYAW=4.3E-4*BE+4.82E-3
      IF(ABS(BE).GT.30.) GO TO 2000
      CY=2.93E-6*BE**5+3.72E-7*BE**4-2.83E-4*BE**3-1.9E-5*BE**2 -4.06E-3
     X*BE+1.36E-2
      CROLL=-1.977E-3*BE-1.555E-3
      GO TO 2000
 80   CYAW=4.5E-4*BE-5.05E-3
      IF(ABS(BE).GT.30.) GO TO 2000
      CY=2.93E-6*BE**5+3.72E-7*BE**4-2.83E-4*BE**3-1.9E-5*BE**2 -4.06E-3
     X*BE+1.36E-2
      CROLL=-1.977E-3*BE-1.555E-3
      GO TO 2000
C   ANGLE OF ATTACK IS GREATER THAN 5.
 195  CD=CD1+(2.15E-5*ABS(BE)**3+1.61E-4*ABS(BE)**2-1.24E-3*ABS(BE))*COS
     X(BR)
      IF(ABS(BE).GE.60.) GO TO 2000
      IF(BE.LT.-7.) GO TO 90
```

```
      IF(BE.GE.7.) GO TO 100
      CYAW=1.28E-5*BE**3+1.23E-5*BE**2-8.21E-4*BE-5.33E-4
      CY=1.76E-6*BE**5-1 .91E-7*BE**4-1.8E-4*BE**3+2.72E-5*BE**2-5.82E-3
     X*BE+1.04E-2
      CROLL=-2.302E-3*BE+1.445E-3
      GO TO 2000
90    CYAW=-.24E-4*BE+.0045
      IF(ABS(BE).GT.30.) GO TO 2000
      CY=1.76E-6*BE**5-1 .91E-7*BE**4-1.8E-4*BE**3+2.72E-5*BE**2-5.82E-3
     X*BE+1.04E-2
      CROLL=-2.302E-3*BE-1.445E-3
      GO TO 2000
100   CYAW=4.24E-4*BE-4.3E-3
      IF(ABS(BE).GT.30.) GO TO 2000
      CY=1.76E-6*BE**5-1 .91E-7*BE**4-1.8E-4*BE**3+2.72E-5*BE**2-5.82E-3
     X*BE+1.04E-2
      CROLL=-2.302E-3*BE-1.445E-3
2000  CONTINUE
C     DERIVATION OF DIMENTIONAL DERIVATIVES (STABILIT Y AXIS)
      ZWDS=-RHO*S*C/4.*CLADR
      ZQS=-QS/VT*C/2.*CLQR
      ZDELS=-QS*CLELR
      MWDS=RHO*S*C**2*CMADR/4.
      MQS=QS/VT*C**2*CMQR/2.
      MDELS=QS*CMELR*10.
      YDR=QS* CY
      NDR=QS*B*CYAW
      LDR=QS*B*CROLL
      YRS=RHO*S*B/4.*CYRR
      YPS=RHO*S*B/4.*CYPR
      NRS=QS/VT*B**2/4.*CNRR
      NPS=QS/VT*B**2/4.*CNPR
      LRS=QS/VT*B**2*CLRR/2.
      LPS=QS/VT*B**2*CLPR/2.
      LDELS=QS*B*CLAIL
      XQS=0. $XWDS=0. $NDELS=0.
C     DERIVATION OF DIMENTIONAL DERIVATIVES (BODY AXIS FROM STABILITY AXIS)
      X0=CL*QS*SIN(AR)-CD*QS*COS(AR)
      Z0=-CL*QS*COS(AR)-CD*QS*SIN(AR)
      M0=CM*QS*C
      XWD=(XWDS*COS(AR)**2-ZWDS*SIN(AR)*COS(AR))*COS(BR)
      ZAD=(ZWDS*COS(AR)**2+XWDS*SIN(AR)*COS(AR))*COS(BR)
      MAD=MWDS*COS(AR)*COS(BR)
      YP=YPS*COS(AR)-YRS*SIN(AR)
      YR=YRS*COS(AR)+YPS*SIN(AR)
      XDE=-ZDELS*SIN(AR)*COS(BR)
      XQ=(XQS*COS(AR)-ZQS*SIN(AR))*COS(BR)
      ZQ=(ZQS*COS(AR)+XQS*SIN(AR))*COS(BR)
      ZDE=ZDELS*COS(AR)*COS(BR)
      LP=(LPS*COS(AR)**2-(LRS+NPS)*SIN(AR)*COS(AR)+NRS*SIN(AR)**2)*COS(B
     XR)
      LDA=(LDELS*COS(AR)-NDELS*SIN(AR))*COS(BR)
      LR=(LRS*COS(AR)**2-(NRS-LPS)*SIN(AR)*COS(AR)-NPS*SIN(AR)**2)*COS(B
     XR)
      NP=(NPS*COS(AR)**2-(NRS-LPS)*SIN(AR)*COS(AR)-LRS*SIN(AR)**2)*COS(B
     XR)
      NR=(NRS*COS(AR)**2+(LRS+NPS)*SIN(AR)*COS(AR)+LPS*SIN(AR)**2)*COS(B
     XR)
              MQ=MQS $MDE=MDELS
20    CONTINUE
      XQ LO=XQ $XDELO=XDE $ZQ LO=ZQ $ZADLO=ZAD $Z0 LO=Z0 $ZDELO=ZDE
      MQ LO=MQ $MADLO=MAD $M0 LO=M0 $MDELO=MDE $YR LO=YR $YB LO=YB
      YP LO=YP $LB LO=LB $LP LO=LP $LR LO=LR $LDALO=LDA $NB LO=NB
      NP LO=NP $NR LO=NR $QS AV=QS $ELELO=ELE $ALTSO=ALT $AL AV=AL
      BE AV=BE $RO AV=RO $PO AV=PO $VT AV=VT $QQ AV=QQ $NSP=32
      LDRLO=LDR $YDRLO=YDR $NDRLO=NDR
```

```
LOCATION=142,EN
FORTRAN STATEMENTS
      RPM=330.+.81+18.47*TH EN-1.84E-2*TH EN**2+9.23E-6*TH EN**3-1.67E-9*
     XTH EN**4
 10   CALL FM(HY FM,PCHFM,PTKFM,PPLFM,VTKS2,PFNFM,OFNFM,QFXFM,        IPPF
     1M,CSSFM,RPM,  VCSFM,RHOFM,DVKFM,VPLFM,DVPFM,          DVCFM,SKIFM,
     2 ALTSD,ROLSD,PITSD,PHEDOT,THEDOT,VCHS2,ACHS2,U  SD,W  SD,AL AV,P
     X SD,C  SD,R  SD,AM,MSS,PAI,      TEM,HYI,         OP1,R1 TK,R2 TK,L1
     XTK,L2 TK,PH1TK,             TX ST,FY ST,TZ ST,YAWSD,X2 IT 3,OPTFM,OTC
     XFM,OTAFM,ATKCH,ATKAT,ATKCHC,ATKATC,QCHAT,V  SD,PH2TK,L1IS1,L2IS1,I
     XSHAPE)
C******************** ROLL CONTROLLER ********************
      SX=ROLSD+(P0 AV)*ABS(P0 AV)/(2.*UMAX*8.16E-3)
      IF(SX.GT.0.)TXCON=-UMAX*20.
      IF(SX.LT.0.)TXCON= UMAX*20.
      IF((SX.EQ.0.).AND.(P0 AV.GT.0.))TXCON=-UMAX*20.
      IF((SX.EQ.0.).AND.(P0 AV.LT.0.))TXCON=UMAX*20.
      IF((ABS(ROLSD).EQ.0.0).AND.(ABS(P0 AV).EQ.0.0)) TXCON=0.
C***********************************************************************
C****************** YAW CONTROLLER ***********************************
      UYTMAX=TH EN/23.
      SIEC=-0.30 *YD SD-X2 IT 1
      SYTX=YAWSD-SIEC+R  SD*ABS(R  SD)*183.15/UYTMAX
      IF(SYTX.GT.0.)TZCON=-UYTMAX*10.
      IF(SYTX.LT.0.)TZCON= UYTMAX*10.
      IF((SYTX.EQ.0.).AND.(R  SD.GT.0 .))TZCON=-UYTMAX*10.
      IF((SYTX.EQ.0.).AND.(R  SD.LT.0.  ))TZCON=UYTMAX*10.
      IF((YAWSD.EQ. SIEC).AND.(R  SD.EQ.0.))TZCON=0.
C********** PITCH CONTROLLER *****************************************
      SXPITCH=PITSD+(Q0 AV)*ABS(Q0 AV)/(2.*UMAX*6.52E-3)
      IF(SXPITCH.GT.0.)TYCON=-UMAX*12.
      IF(SXPITCH.LT.0.)TYCON= UMAX*12.
      IF((SXPITCH.EQ.0.).AND.(Q0 AV.GT.0.))TYCON=-UMAX*12.
      IF((SXPITCH.EQ.0.).AND.(Q0 AV.LT.0.))TYCON= UMAX*12.
      IF((ABS(PITSD).EQ.0.0).AND.(ABS(Q0 AV).EQ.0.0)) TYCON=0.
C*********************************************************************
      TX3S3=TXCON $TY3S3=TYCON $TZ3S3=TZCON $FZ2S3=-FY ST $TX2S3=TX ST
      TY2S3=TZ ST
LOCATION=124,S3,INPUTS=EN
LOCATION=12,LO,INPUTS=AV,S3,MC 1(X=ELE)
LOCATION=44,LO,INPUTS=AV,LO,S3,MC 2(X=AIL)
LOCATION=37,SO,INPUTS=LO,LO
FORTRAN STATEMENTS
      [illegible]
      ELEC=.29*(Q  SD-0.)+.533*(PITSD-0.)
      X1 TF=ELEC
LOCATION=149,TF
LOCATION=147,IT 7,INPUTS=TF
LOCATION=157,MC 1,INPUTS=TF,IT 7,IT 8
FORTRAN STATEMENTS
      IF(X4 MC 1.GE.10.) X4 MC 1=10.
      IF(X4 MC 1.LE.-15.) X4 MC 1=-15.
LOCATION=155,IT 3,INPUTS=MC 1
LOCATION=51,IT 3,INPUTS=SD(X0=X)
LOCATION=39 ,IT 1,INPUTS=SD(YD=X)
FORTRAN STATEMENTS
      AILC= .196*(P  SD-0.)+.42*(ROLSD-0.)+.2*R  SD
      X1 TF 2=AILC
LOCATION=248,TF 2
LOCATION=245,IT 9,INPUTS=TF 2
LOCATION=266,MC 2,INPUTS=TF 2,IT 9,IT10
FORTRAN STATEMENTS
      IF(X4 MC 2.GE.8. ) X4 MC 2=8.
      IF(X4 MC 2.LE.-8. ) X4 MC 2=-8.
LOCATION=254,IT1),INPUTS=MC 2
END OF MODEL
```

FORTRAN STATEMENTS

```
      SUBROUTINE FM(HY,PCH,PTK,PPL,VTK,PFN,QFN,QFX,     IPP,CSS,RPM,VCS,R
     1HO,DVK,VPL,DVP,DVC,SKT,YCG,THE,PHE,PHEDOT,THEDOT,VCHS2,ACHS2    ,U
     X,W,AL,F,Q,R,AH,MSS,PAT,     TEM,HYI,          DP1,R1 TK,R2 TK,L1 TK,L
     X2 TK,PH1TK,      TX ST,FY ST,TZ ST,SIE,XCG,QPTFM,QTCFM,QTAFM,ATCS2,
     XATAS2,ACCS2,AACS2,QCHAI,V,PH2TK,L1IS1,L2IS1,ISHAPE)
      REAL L,LS,    MASS,     LX,L1IS1,L1TK,L2IS1,L2TK,MSS
      DIMENSION Y(13),SY(13),DELTA(32 ),SL4CO(32 ),ZCXSE(32 ),YCXSE(32 ),
     X,DELSE(32 ),ITYSE(32 ),XCISE(32 ),ZCISE(32 ),ISGSE(32 ),VTIS1(32 )
     X,ACIS1(32 ),YGHCL(32 ),ACFS2(32 ),ATIS2(32 ),PERS2(32 ),ATRS2(32 )
     X,XCHS2(32 ),XTKS2(32 ),ZCHS2(32 ),ZTKS2(32 ),DY ST(13),YG PR(32 ),
     XY0(13),Y1(13),Y2(13)
      DATA PI,RADIAN/3.14159265 3.0.0174532/
      DTIME=.0003     STINC=1.
C     AIR CUSHION LANDING SYSTEM PROGRAM
      MSS=77.5 $M=4 $N=4 ZCKK=1.4 SCAF=.5 $NSP=32 $VCD=0.       $CC=-1.17
      GG= 1.5  $FF=0. $PAT=2116.3         $LS=4.125 $D=.417 $A=.695    $E
     E=.524$L=5.17 $HYI=1.0  $NX=9 $NH=195 $AH=3.4E-4 $SH=.0933 $LX=0.80
                  CGP=.5 $GCC=.1         SCFX= 1.17 $TEM=70.
      THEDOT=P $PHEDOT=Q $SKT=-W
      MASS=MSS              $TEMPAT=TEM
                  THETAS=THE $PHIS=PHE $DPHI=PHEDOT $DTHETA=THEDOT
                  NSTOP=NSP              $PPLM=PPL
C DATA ACQUISITION
      VPLM=.313 $VFAN=.468
      VPL=VPLM+VFAN
      DO 100 I=1,NSTOP
      DELTA(I)=0.0
  100 CONTINUE
      RHO=1.241/(450.0+TEMPAT)
      HY=HYI
C  MINIMUM TRUNK HEIGHT IS PARKING SLATTER HEIGHT
      HY=AMAX1(HY,.78)
      CALL TK(ISHTK,PH2TK,R2 TK,R1 TK,FH1TK,L1 TK,L2 TK,HY,A,E,L,LS)
      ISHAPE=ISHTK
      IF(ISHAPE.EQ.0)  GO TO 199
      ICLFM=0
      CALL SE(ITYSE,BE SE,DY SE,ZCXSE,XCXSE,DELSE,XCISE,ZCISE,ICLFM,R2 T
     1K,PH2TK,D2IS1,LS,M,N,D,ISGSE)
      CALL S1(D2IS1,A1 S1,A2 S1,SI S1,CI S1,DXIS1,BEIS1,L1IS1,L2IS1,R1IS
     11,R2IS1,SNIS1,PH2TK,R2 TK,DX SE,BE SE,HY,R1 TK,PH1TK,ICLFM,L1 TK,L
     X2 TK,ITYSE,D,A,E,N,M,VTIS1,ACIS1,LS)
      FPLM=57.025-3.17E-2*RPM+5.53E-6*RPM**2-2.086E-10*RPM**3
      PFAN=PFLM
C INPUT PRESSURE AND OUTPUT FLOW
      IF((RPM.GE.11937.).AND.(RPM.LT.12075.)) GO TO 87
      IF((RPM.GE.12075.).AND.(RPM.LT.12213.)) GO TO 88
      IF((RPM.GE.12213.).AND.(RPM.LT.12351.)) GO TO 89
      IF((RPM.GE.12351.).AND.(RPM.LT.12489.)) GO TO 90
      IF((RPM.GE.12489.).AND.(RPM.LT.12627.)) GO TO 91
      IF((RPM.GE.12627.).AND.(RPM.LT.12765.)) GO TO 92
      IF((RPM.GE.12765.).AND.(RPM.LT.12903.)) GO TO 93
      IF((RPM.GE.12903.).AND.(RPM.LT.13041.)) GO TO 94
      IF((RPM.GE.13041.).AND.(RPM.LT.13179.)) GO TO 95
      IF((RPM.GE.13179.).AND.(RPM.LT.13317.)) GO TQ 96
      IF((RPM.GE.13317.).AND.(RPM.LT.13455.)) GO TO 97
      IF((RPM.GE.13455.).AND.(RPM.LT.13593.)) GO TO 98
      IF((RPM.GE.13593.).AND.(RPM.LT.13731.)) GO TO 99
      IF((RPM.GE.13731.).AND.(RPM.LE.14000.)) GO TO 101
      IF( RPM.GT.14000.) GO TO 4
      IF( RPM.LT.11173.) GO TO 2
      IF((RPM.GE.11173.).AND.(RPM.LT.11385.)) GO TO 82
      IF((RPM.GE.11385.).AND.(RPM.LT.11523.)) GO TO 83
      IF((RPM.GE.11523.).AND.(RPM.LT.11661.)) GO TO 84
      IF((RPM.GE.11661.).AND.(RPM.LT.11799.)) GO TO 85
      IF((RPM.GE.11799.).AND.(RPM.LT.11937.)) GO TO 86
```

```
2     PRINT 3
3     FORMAT(*OFF LOWER END OF FAN MAP*)
      GO TO 199
4     PRINT 5
5     FORMAT(* OFF UPPER END OF FAN MAP*)
      GO TO 199
C FAN CURVE FOR STATIC ITERATIONS
82    QFAN=15.06     -.353     *PFAN+4.34E-3    *PFAN**2-1.69E-5    *PFAN
     X**3      $GO TO 110
83    QFAN=13.08     -.339     *PFAN+3.84E-3    *PFAN**2-1.39E-5    *PFAN
     X**3      $GO TO 110
84    QFAN=11.33     -.164     *PFAN+1.74E-3    *PFAN**2-6.3E-6     *PFAN
     X**3      $GO TO 110
85    QFAN=10.82     -.129     *PFAN+1.29E-3    *PFAN**2-4.56E-6    *PFAN
     X**3      $GO TO 110
86    QFAN=9.73      -.077     *PFAN+7.19E-4    *PFAN**2-2.6E-6     *PFAN
     X**3      $GO TO 110
87    QFAN=9.97      -.077     *PFAN+6.95E-4    *PFAN**2-2.4E-6     *PFAN
     X**3      $GO TO 110
88    QFAN=12.39     -.15      *PFAN+1.32E-3    *PFAN**2-3.96E-6    *PFAN
     X**3      $GO TO 110
89    QFAN=18.38     -.333     *PFAN+2.87E-3    *PFAN**2-7.79E-6    *PFAN
     X**3      $GO TO 110
90    QFAN=15.03     -.211     *PFAN+1.74E-3    *PFAN**2-4.67E-6    *PFAN
     X**3      $GO TO 110
91    QFAN=12.73     -.13      *PFAN+1.03E-3    *PFAN**2-2.78E-6    *PFAN
     X**3      $GO TO 110
92    QFAN=13.78     -.153     *PFAN+1.17E-3    *PFAN**2-3.01E-6    *PFAN
     X**3      $GO TO 110
93    QFAN=13.74     -.142     *PFAN+1.05E-3    *PFAN**2-2.6E-6     *PFAN
     X**3      $GO TO 110
94    QFAN=12.05     -.086     *PFAN+5.92E-4    *PFAN**2-1.5E-6     *PFAN
     X**3      $GO TO 110
95    QFAN=12.97     -.104     *PFAN+6.93E-4    *PFAN**2-1.63E-6    *PFAN
     X**3      $GO TO 110
96    QFAN=11.59     -.062     *PFAN+3.78E-4    *PFAN**2-9.23E-7    *PFAN
     X**3      $GO TO 110
97    QFAN=12.6      -.079     *PFAN+4.79E-4    *PFAN**2-1.09E-6    *PFAN
     X**3      $GO TO 110
98    QFAN=18.29     -.206     *PFAN+1.29E-3    *PFAN**2-2.61E-6    *PFAN
     X**3      $GO TO 110
99    QFAN=19.03     -.213     *PFAN+1.29E-3    *PFAN**2-2.53E-6    *PFAN
     X**3      $GO TO 110
101   QFAN=32.07     -.492     *PFAN+2.98E-3    *PFAN**2-5.63E-6    *PFAN
     X**3
C          CHANGED Q FROM MASS FLOW TO VOLUME FLOW
110   QFAN=QFAN/(RHO*32.2)
      QFANX=QFAN
      QFX=QFANX
      QFN=QFAN
      PFN=PFAN
      PPL=PPLM
C     CONVERT DEGREES TO RADIANS
      THETAE=THETAE*RADIAN
      PHIE=PHIE*RADIAN
      OTHETA=OTHETA*RADIAN
      OPHI=OPHI*RADIAN $SIE=SIE*RADIAN
      ICLN=0
      ICN=ICLN
C     OBTAIN INITIAL VALUE OF OCCHP AND INITIALIZE GEOMETRY
      PCHSS=(PCH+(PTK-PCH)*0.1)/PTK
      CSS=PCHSS
      CALL HC(Z  HC,CSS)
      HY=HYI*7  HC
C  MINIMUM TRUNK HEIGHT IS PARKING BLATTER HEIGHT
      HY=AMAX1(HY,.78)
```

```
      CALL TK(ISHTK,PH2TK,R2 TK,R1 TK,PH1TK,L1 TK,L2 TK,HY,A,E,L,LS)
      ISHAPE=ISHTK
      IF(ISHAPE.EQ.0)GO TO 199
      ICLFM=1
      CALL SE(ITYSE,BE SE,DX SE,ZCXSE,XCXSE,DELSE,XCISE,ZCISE,ICLFM,R2 T
     1K,PH2TK,D2IS1,LS,M,N,D,ISGSE)
      CALL CO(SL4CO,XCXSE,ZCXSE,YCG,XCG,SIF,PHE,THE,CC,FF,GG)
      CALL PR(YG PR)
      CALL CL(YGHCL,ICN,PHE,THE,SL4CO,YG PR)
      RHOFM=RHO
      CALL S1(D2IS1,A1 S1,A2 S1,SI S1,DI S1,DXIS1,BEIS1,L1IS1,L2IS1,R1IS
     11,R2IS1,SNIS1,PH2TK,R2 TK,DX SE,BE SE,HY,P1 TK,PH1TK,ICLFM,L1 TK,L
     X2 TK,ITYSE,D,A,E,N,M,VTIS1,ACIS1,LS)
      CALL SP(VTKS2,VCHS2,ACHS2,PH2TK,R2 TK,D2IS1,BE SE,A1 S1,A2 S1,DX S
     1E,SI S1,      ITYSE,YGHCL,HY    ,DI S1,DXIS1,BEIS1,XCISE,ZCISE,L2 T
     2K,L1IS1,L2IS1,R1IS1,R2IS1,DELSE,SNIS1,XCXSE,D,LS,AH,NX,NH,HYI,LX,S
     3M,VCC,P1 TK,AGPS2,ATAS2,ATCS2,ACNS2,AACS2,ACCS2,ISGSE,VTIS1,ACIS1,
     XACRS2,ATIS2,ATRS2,PERS2,XCHS2,XTKS2,ZCHS2,ZTKS2,L,L1 TK)
      VCHSS=VCHS2
      VCS=VCHSS
      CALL HC(Z  HC,(PCH/PTK))
      HY=HYI*Z  HC
C  MINIMUM TRUNK HEIGHT IS PARKING BLATTER HEIGHT
      HY=AMAX1(HY,.78)
      CALL TK(ISHTK,PH2TK,R2 TK,R1 TK,PH1TK,L1 TK,L2 TK,HY,A,E,L,LS)
      ISHAPE=ISHTK
      IF(ISHAPE.EQ.0)GO TO 199
      CALL SE(ITYSE,BE SE,DX SE,ZCXSE,XCXSE,DELSE,XCISE,ZCISE,ICLFM,R2 T
     1K,PH2TK,D2IS1,LS,M,N,D,ISGSE)
      CALL CO(SL4CO,XCXSE,ZCXSE,YCG,XCG,SIF,PHE,THE,CC,FF,GG)
      CALL PR(YG PR)
      CALL CL(YGHCL,ICN,PHE,THE,SL4CO,YG PR)
      CALL S1(D2IS1,A1 S1,A2 S1,SI S1,DI S1,DXIS1,BEIS1,L1IS1,L2IS1,R1IS
     11,R2IS1,SNIS1,PH2TK,R2 TK,DX SE,BE SE,HY,P1 TK,PH1TK,ICLFM,L1 TK,L
     X2 TK,ITYSE,D,A,E,N,M,VTIS1,ACIS1,LS)
      CALL SP(VTKS2,VCHS2,ACHS2,PH2TK,R2 TK,D2IS1,BE SE,A1 S1,A2 S1,DX S
     1E,SI S1,      ITYSE,YGHCL,HY    ,DI S1,DXIS1,BEIS1,XCISE,ZCISE,L2 T
     2K,L1IS1,L2IS1,R1IS1,R2IS1,DELSE,SNIS1,XCXSE,D,LS,AH,NX,NH,HYI,LX,S
     3M,VCC,R1 TK,AGPS2,ATAS2,ATCS2,ACNS2,AACS2,ACCS2,ISGSE,VTIS1,ACIS1,
     XACRS2,ATIS2,ATRS2,PERS2,XCHS2,XTKS2,ZCHS2,ZTKS2,L,L1 TK)
      DVCHP=(VCHS2-VCHSS)/((PCH/PTK)-PCHSS)
      DVP=DVCHP
C*****DYNAMIC SIMULATION***********************************
      INUM=0
       DVTK=0.0
      DVK=DVTK
      DVCH=SKT*ACHS2
      DVC=DVCH
      CALL ST(FY ST,TZ ST,TX ST,DY ST,IPP,FPL,VPL,VTKS2,CFX,DVK,PTK,DVC,
     1YCG,DVP,PCH,VCHS2,SKT,PHE,PHEDOT,THE,THEDOT,CKK,PAT,GG,HYI,SEC,MSS
     2, RPM,             U,         SIF,RHO,AGPS2,ACHS2,ATAS2,ATCS2,ACNS
     32,AACS2,ACCS2,CAF,CFX,CGP,      ACRS2,ATIS2,ATRS2,PERS2,XCHS2,ZCHS2,
     3XTKS2,ZTKS2,ACIS1,YGHCL,OPTFM,OTCFM,CTAFM,OCHAT)
C*********CALCULATE DVCH,DVCHP,DVTK AS DVC,DVP,DVK    ********************
      IF(INUM)200,1,200
 1    VCHS=VCHS2 $VTKS=VTKS2
 200  CALL TK(ISHTK,PH2TK,R2 TK,R1 TK,PH1TK,L1 TK,L2 TK,HY,A,E,L,LS)
      ISHAPE=ISHTK
      ICLFM=1
      CALL SE(ITYSE,BE SE,DX SE,ZCXSE,XCXSE,DELSE,XCISE,ZCISE,ICLFM,R2 T
     1K,PH2TK,D2IS1,LS,M,N,D,ISGSE)
      CALL CO(SL4CO,XCXSE,ZCXSE,YCG,XCG,SIF,PHE,THE,CC,FF,GG)
      CALL PR(YG PR)
      CALL CL(YGHCL,ICN,PHE,THE,SL4CO,YG PR)
      RHOFM=RHO
      CALL S1(D2IS1,A1 S1,A2 S1,SI S1,DI S1,DXIS1,BEIS1,L1IS1,L2IS1,R1IS
```

```
     11,R2IS1,SNIS1,PH2TK,P2 TK,DX SE,BE SE,HY,R1 TK,PH1TK,ICLFM,L1 TK,L
     X2 TK,ITYSE,D,A,E,N,M,VTIS1,ACIS1,LS)
      CALL SP(VTKS2,VCHS2,ACHS2,PH2TK,P2 TK,D2IS1,BE SE,A1 S1,A2 S1,DX S
     1E,SI S1,      ITYSE,YGHCL,HY    ,DI S1,DXIS1,BEIS1,XCISE,ZCISE,L2 T
     2K,L1IS1,L2IS1,R1IS1,R2IS1,DELSE,SNIS1,XCXSE,D,LS,AH,NX,NH,HYI,LX,S
     3H,VCD,R1 TK,AGPS2,ATAS2,ATCS2,ACHS2,AACS2,ACCS2,ISGSE,VTIS1,ACIS1,
     XACRS2,ATIS2,ATRS2,PERS2,XCHS2,XTKS2,ZCHS2,ZTKS2,L,L1 TK)
      DVC=(VCHS2-VCHS)/.01
C******* DVCHP CALCULATION *************************
      VCHSS=VCHS2 $PRAT=(PCH+(PTK-PCH)*.1)/PTK
      CALL HC(HX,PRAT)
      HY=HX*HYI
C  MINIMUM TRUNK HEIGHT IS PARKING BLATTER HEIGHT
      HY=AMAX1(HY,.78)
      CALL TK(ISHTK,PH2TK,R2 TK,R1 TK,PH1TK,L1 TK,L2 TK,HY,A,E,L,LS)
      ISHAPE=ISHTK
      IF(ISHAPE.EQ.0) GO TO 199
      CALL SE(ITYSE,BE SE,DX SE,ZCXSE,XCXSE,DELSE,XCISE,ZCISE,ICLFM,R2 T
     1K,PH2TK,D2IS1,LS,M,N,D,ISGSE)
      CALL CC(SL4CO,XCXSE,ZCXSE,YCG,XCG,SIE,PHE,THE,CC,FF,GG)
      CALL PR(YG PR)
      CALL CL(YGHCL,ICN,PHE,THE,SL4CO,YG PR)
      RHOFM=RHO
      CALL S1(D2IS1,A1 S1,A2 S1,SI S1,DI S1,DXIS1,BEIS1,L1IS1,L2IS1,R1IS
     11,R2IS1,SNIS1,PH2TK,R2 TK,DX SE,BE SE,HY,R1 TK,PH1TK,ICLFM,L1 TK,L
     X2 TK,ITYSE,D,A,E,N,M,VTIS1,ACIS1,LS)
      CALL SP(VTKS2,VCHS2,ACHS2,PH2TK,P2 TK,D2IS1,BE SE,A1 S1,A2 S1,DX S
     1E,SI S1,      ITYSE,YGHCL,HY    ,DI S1,DXIS1,BEIS1,XCISE,ZCISE,L2 T
     2K,L1IS1,L2IS1,R1IS1,R2IS1,DELSE,SNIS1,XCXSE,D,LS,AH,NX,NH,HYI,LX,S
     3H,VCD,R1 TK,AGPS2,ATAS2,ATCS2,ACHS2,AACS2,ACCS2,ISGSE,VTIS1,ACIS1,
     XACRS2,ATIS2,ATRS2,PERS2,XCHS2,XTKS2,ZCHS2,ZTKS2,L,L1 TK)
      DVP=(VCHSS-VCHS2)/((PCH/PTK)-PRAT)
      PRAT=AMAX1(0.0,AMIN1(1.0,(PCH/PTK)))
C******* CALCULATE DVTK ***************************
      CALL HC(HX,PRAT)
      HY=HX*HYI
C  MINIMUM TRUNK HEIGHT IS PARKING BLATTER HEIGHT
      HY=AMAX1(HY,.78)
      CALL TK(ISHTK,PH2TK,R2 TK,R1 TK,PH1TK,L1 TK,L2 TK,HY,A,E,L,LS)
      ISHAPE=ISHTK
      IF(ISHAPE.EQ.0) GO TO 199
      CALL SE(ITYSE,BE SE,DX SE,ZCXSE,XCXSE,DELSE,XCISE,ZCISE,ICLFM,R2 T
     1K,PH2TK,D2IS1,LS,M,N,D,ISGSE)
      CALL CC(SL4CO,XCXSE,ZCXSE,YCG,XCG,SIE,PHE,THE,CC,FF,GG)
      CALL PR(YG PR)
      CALL CL(YGHCL,ICN,PHE,THE,SL4CO,YG PR)
      RHOFM=RHO
      CALL S1(D2IS1,A1 S1,A2 S1,SI S1,DI S1,DXIS1,BEIS1,L1IS1,L2IS1,R1IS
     11,R2IS1,SNIS1,PH2TK,R2 TK,DX SE,BE SE,HY,R1 TK,PH1TK,ICLFM,L1 TK,L
     X2 TK,ITYSE,D,A,E,N,M,VTIS1,ACIS1,LS)
      CALL SP(VTKS2,VCHS2,ACHS2,PH2TK,P2 TK,D2IS1,BE SE,A1 S1,A2 S1,DX S
     1E,SI S1,      ITYSE,YGHCL,HY    ,DI S1,DXIS1,BEIS1,XCISE,ZCISE,L2 T
     2K,L1IS1,L2IS1,R1IS1,R2IS1,DELSE,SNIS1,XCXSE,D,LS,AH,NX,NH,HYI,LX,S
     3H,VCD,R1 TK,AGPS2,ATAS2,ATCS2,ACHS2,AACS2,ACCS2,ISGSE,VTIS1,ACIS1,
     XACRS2,ATIS2,ATRS2,PERS2,XCHS2,XTKS2,ZCHS2,ZTKS2,L,L1 TK)
      DVK=(VTKS2-VTKS)/.01
C*********** FOURTH ORDER RUNGE-KUTTA INTEGRATION ROUTINE ***************
      Y(1)=PPL  $Y(2)=PCH $Y(3)=PTK
      TIME=0.
 50   H=DTIME/2.
      DO 10 I=1,4
      SY(I)=Y(I)
      Y0(I)=DY ST(I)
 10   Y(I)=Y(I)+H*DY ST(I)
      TIME=TIME+H
      Y(1)=PPL  $Y(2)=PCH $Y(3)=PTK
```

```
      CALL ST(FY ST,TZ ST,TX ST,DY ST,IPP,PPL,VPL,VTKS2,OFX,DVK,PTK,DVC,
     1YCG,DVP,PCH,VCHS2,SKT,PHE,PHEDOT,THE,THEDOT,CKK,PAT,GG,HYI,GEC,MSS
     2, RPM,                U,        SIE,RHO,AGPS2,ACHS2,ATAS2,ATCS2,ACNS
     32,AACS2,ACCS2,CAF,CFX,CGP,      ACFS2,ATIS2,ATRS2,PERS2,XCHS2,ZCHS2,
     3XTKS2,ZTKS2,ACIS1,YGHCL,OPTFM,OTCFM,OTAFM,OCHAT)
      DO 20 I=1,4
      Y1(I)=DY ST(I)
20    Y(I)=SY(I)+H*DY ST(I)
      Y(1)=PPL  $Y(2)=PCH $Y(3)=PTK
      CALL ST(FY ST,TZ ST,TX ST,DY ST,IPP,PPL,VPL,VTKS2,OFX,DVK,PTK,DVC,
     1YCG,DVP,PCH,VCHS2,SKT,PHE,PHEDOT,THE,THEDOT,CKK,PAT,GG,HYI,GEC,MSS
     2, RPM,                U,        SIE,RHO,AGPS2,ACHS2,ATAS2,ATCS2,ACNS
     32,AACS2,ACCS2,CAF,CFX,CGP,      ACFS2,ATIS2,ATRS2,PERS2,XCHS2,ZCHS2,
     3XTKS2,ZTKS2,ACIS1,YGHCL,OPTFM,OTCFM,OTAFM,OCHAT)
      DO 30 I=1,4
      Y2(I)=DY ST(I)
30    Y(I)=SY(I)+DTIME*DY ST(I)
      Y(1)=PPL  $Y(2)=PCH $Y(3)=PTK
      CALL ST(FY ST,TZ ST,TX ST,DY ST,IPP,PPL,VPL,VTKS2,OFX,DVK,PTK,DVC,
     1YCG,DVP,PCH,VCHS2,SKT,PHE,PHEDOT,THE,THEDOT,CKK,PAT,GG,HYI,GEC,MSS
     2, RPM,                U,        SIE,RHO,AGPS2,ACHS2,ATAS2,ATCS2,ACNS
     32,AACS2,ACCS2,CAF,CFX,CGP,      ACFS2,ATIS2,ATRS2,PERS2,XCHS2,ZCHS2,
     3XTKS2,ZTKS2,ACIS1,YGHCL,OPTFM,OTCFM,OTAFM,OCHAT)
      TIME=TIME+H
      H=H/3.
      DO 40 I=1,4
      PPT1=2.0*(Y1(I)+Y2(I))
      PRT2=Y0(I)+DY ST(I)
      Y(I)=SY(I)+H*PRT1+H*PRT2
40    CONTINUE
      PPL=Y(1)  $PCH=Y(2) $PTK=Y(3)
      CALL ST(FY ST,TZ ST,TX ST,DY ST,IPP,PPL,VPL,VTKS2,OFX,DVK,PTK,DVC,
     1YCG,DVP,PCH,VCHS2,SKT,PHE,PHEDOT,THE,THEDOT,CKK,PAT,GG,HYI,GEC,MSS
     2, RPM,                U,        SIE,RHO,AGPS2,ACHS2,ATAS2,ATCS2,ACNS
     32,AACS2,ACCS2,CAF,CFX,CGP,      ACFS2,ATIS2,ATRS2,PERS2,XCHS2,ZCHS2,
     3XTKS2,ZTKS2,ACIS1,YGHCL,OPTFM,OTCFM,OTAFM,OCHAT)
      IF((PPL-SY(1)).LT.1.) GO TO 60
      IF(TIME.LT.TINC) GO TO 50
60    CONTINUE
      INUM=1
C*********CALCULATE DVCH,DVCHP,DVTK AS DVC,DVP,DVK   *********************
      IF(INUM)201,11,201
11    VCHS=VCHS2 $VTKS=VTKS2
201   CALL TK(ISHTK,PH2TK,R2 TK,R1 TK,PH1TK,L1 TK,L2 TK,HY,A,E,L,LS)
      ISHAPE=ISHTK
      ICLFM=1
      CALL SE(ITYSE,BE SE,DX SE,ZCXSE,XCXSE,DELSE,XCISE,ZCISE,ICLFM,R2 T
     1K,PH2TK,D2IS1,LS,M,N,D,ISGSE)
      CALL CC(SL,CO,XCXSE,ZCXSE,YCG,XCG,SIE,PHE,THE,CC,FF,GG)
      CALL PR(YG PR)
      CALL CL(YGHCL,ISN,PHE,THE,SLACO,YG PR)
      RHOFM=RHO
      CALL S1(D2IS1,A1 S1,A2 S1,SI S1,DI S1,DXIS1,BEIS1,L1IS1,L2IS1,R1IS
     11,R2IS1,SNIS1,PH2TK,R2 TK,DX SE,BE SE,HY,R1 TK,PH1TK,ICLFM,L1 TK,L
     X2 TK,ITYSE,D,A,E,N,M,VTIS1,ACIS1,LS)
      CALL SF(VTKS2,VCHS2,ACHS2,PH2TK,R2 TK,D2IS1,BE SE,A1 S1,A2 S1,DX S
     1E,SI S1,      ITYSE,YGHCL,HY    ,LI S1,DXIS1,BEIS1,XCISE,ZCISE,L2 T
     2K,L1IS1,L2IS1,R1IS1,R2IS1,DELSE,SNIS1,XCXSE,D,LS,AH,NX,NH,HYI,LX,S
     3H,VCO,R1 TK,AGPS2,ATAS2,ATCS2,ACNS2,AACS2,ACCS2,ISGSE,VTIS1,ACIS1,
     XACPS2,ATIS2,ATRS2,PERS2,XCHS2,XTKS2,ZCHS2,ZTKS2,L,L1 TK)
      DVC=(VCHS2-VCHS)/.01
C******** DVCHP CALCULATION **************************
      VCHSS=VCHS2 $PRAT=(PCH+(PTK-PCH)*.1)/PTK
      CALL HC(HX,PRAT)
      HY=HX*HYI
      CALL TK(ISHTK,PH2TK,R2 TK,R1 TK,PH1TK,L1 TK,L2 TK,HY,A,E,L,LS)
```

```
      ISHAPE=ISHTK
      IF(ISHAPE.EQ.0) GO TO 199
      CALL SE(ITYSE,BE SE,DX SE,ZCXSE,XCXSE,DELSE,XCISE,ZCISE,ICLFM,R2 T
     1K,PH2TK,D2IS1,LS,M,N,D,ISGSE)
      CALL CO(SL,CO,XCXSE,ZCXSE,YCG,XCG,SIE,PHE,THE,CC,FF,GG)
      CALL PR(YG PR)
      CALL CL(YGHCL,ICN,PHE,THE,SL4CO,YG PR)
      RHOFM=RHO
      CALL S1(D2IS1,A1 S1,A2 S1,SI S1,CI S1,DXIS1,BEIS1,L1IS1,L2IS1,R1IS
     11,R2IS1,SNIS1,PH2TK,R2 TK,DX SE,BE SE,HY,R1 TK,PH1TK,ICLFM,L1 TK,L
     X2 TK,ITYSE,D,A,E,N,M,VTIS1,ACIS1,LS)
      CALL SP(VTKS2,VCHS2,ACHS2,PH2TK,R2 TK,D2IS1,BE SE,A1 S1,A2 S1,DX S
     1E,SI S1,      ITYSE,YGHCL,HY    ,CI S1,DXIS1,BEIS1,XCISE,ZCISE,L2 T
     2K,L1IS1,L2IS1,R1IS1,R2IS1,DELSE,SNIS1,XCXSE,D,LS,AH,NX,NH,HYI,LX,S
     3H,VCO,R1 TK,AGPS2,ATAS2,ATCS2,ACHS2,AACS2,ACCS2,ISGSE,VTIS1,ACIS1,
     XACRS2,ATIS2,ATRS2,PERS2,XCHS2,XTKS2,ZCHS2,ZTKS2,L,L1 TK)
      DVP=(VCHSS-VCHS2)/((PCH/PTK)-PRAT)
      PRAT=AMAX1(0.0,AMIN1(1.0,(PCH/PTK)))
C******* CALCULATE DVTK ****************************
      CALL HC(HX,PRAT)
      HY=HX*HYI
      CALL TK(ISHTK,PH2TK,R2 TK,R1 TK,PH1TK,L1 TK,L2 TK,HY,A,E,L,LS)
      ISHAPE=ISHTK
      IF(ISHAPE.EQ.0) GO TO 199
      CALL SE(ITYSE,BE SE,DX SE,ZCXSE,XCXSE,DELSE,XCISE,ZCISE,ICLFM,R2 T
     1K,PH2TK,D2IS1,LS,M,N,D,ISGSE)
      CALL CO(SL,CO,XCXSE,ZCXSE,YCG,XCG,SIE,PHE,THE,CC,FF,GG)
      CALL PR(YG PR)
      CALL CL(YGHCL,ICN,PHE,THE,SL4CO,YG PR)
      RHOFM=RHO
      CALL S1(D2IS1,A1 S1,A2 S1,SI S1,CI S1,DXIS1,BEIS1,L1IS1,L2IS1,R1IS
     11,R2IS1,SNIS1,PH2TK,R2 TK,DX SE,BE SE,HY,R1 TK,PH1TK,ICLFM,L1 TK,L
     X2 TK,ITYSE,D,A,E,N,M,VTIS1,ACIS1,LS)
      CALL SP(VTKS2,VCHS2,ACHS2,PH2TK,R2 TK,D2IS1,BE SE,A1 S1,A2 S1,DX S
     1E,SI S1,      ITYSE,YGHCL,HY    ,CI S1,DXIS1,BEIS1,XCISE,ZCISE,L2 T
     2K,L1IS1,L2IS1,R1IS1,R2IS1,DELSE,SNIS1,XCXSE,D,LS,AH,NX,NH,HYI,LX,S
     3H,VCO,R1 TK,AGPS2,ATAS2,ATCS2,ACHS2,AACS2,ACCS2,ISGSE,VTIS1,ACIS1,
     XACRS2,ATIS2,ATRS2,PERS2,XCHS2,XTKS2,ZCHS2,ZTKS2,L,L1 TK)
      DVK=(VTKS2-VTKS)/.01
      VTK=VTKS2
 199  CONTINUE
C           CONVERT RADIANS TO DEGREES
      THETAE=THETAE/RADIAN
      PHIE=PHIE/RADIAN
      DTHETA=DTHETA/RADIAN
      DPHIE=DPHIE/RADIAN
      SIE=SIE/RADIAN
      RETURN $END
      SUBROUTINE SE(ITYP,BETA,DELX,ZCX,XCX,DELTA,XCHI,ZCHI,ICALL,R2,PHI2
     X,D2I,LS,M,N,D,ISEG)
C DIVISION OF THE TRUNK INTO SEGMENTS
      REAL LS
      DIMENSION ZCX(32),XCX(32),ITYP(32),DELTA(32),ISEG(32),XCHI(32),ZCH
     XI(32)
      DATA PI/3.14159265/
                 NSTOP=32
C IF FIRST CALL,COMPUTE PARTIAL TERMS AND NUMBER SEGMENTS
      IF(ICALL) 20,30,20
 30   RLSH=0.5*LS
C BETA IS CURVED SEGMENT ARC ANGLE
      BETA=PI/2./FLOAT(N)
C DELX IS STRAIGHT SEGMENT LENGTH
      DELX=LS/FLOAT(2*M)
      BETA2=1.33333*SIN(BETA/2.)/BETA
C NUMBERING OF SEGMENTS ACCORDING TO THEIR POSITION IN THE TRUNK
      DO 11 I=1,NSTOP
```

```
      IF(I.LE.N)ISEG(I)=1
      IF(I.GT.N.AND.I.LE.N+M)ISEG(I)=2
      IF(I.GT.N+M.AND.I.LT.N+2*M)ISEG(I)=3
      IF(I.GT.N+2*M.AND.I.LE.2*(N+M))ISEG(I)=4
      IF(I.GT.2.*(N+M).AND.I.LE.3.*N+2*M) ISEG(I)=5
      IF(I.GT.3*N+2*M.AND.I.LE.3*(N+M))ISEG(I)=6
      IF(I.GT.3*(N+M).AND.I.LE.3*N+4*M) ISEG(I)=7
      IF(I.GT.3*N+4*M.AND.I.LE.4*(N+M)) ISEG(I)=8
 11   CONTINUE
C EVALUATING PROPERTIES OF SEGMENTS
C ITYP=1 FOR CURVED SEGMENT,=0 FOR STRAIGHT SEGMENT
C XCX AND ZCX ARE X AND Z COORDINATES RESP. OF THE SEGMENT CENTER
C XCHI AND ZCHI ARE X AND Z COORDINATES RESP. OF THE CUSHION
C PRESSURE CENTER FOR A SEGMENT.WHEN IT IS OUT OF GROUND CONTACT
C DELTA IS SEGMENT CENTER ANGLE RELATIVE TO CG
 20   CONTINUE
      D2=0.5*D+R2*SIN(PHI2)
      DO 10 I=1,NSTOP
      KGO=ISEG(I)
      GO TO (1,2,3,4,5,6,7,8), KGO
C CURVED SEGMENT
C IF NOT INITIAL CALL SKIP CALCULATIONS
 1    IF(ICALL) 9,100,9
 100  ITYP(I)=1
      DELTA(I)=(FLOAT(I-1)+0.5)*BETA
      COSDEL=COS(DELTA(I))
      XCX(I)=-(RLSH+D2I*COSDEL)
      ZCX(I)=D2I*SIN(DELTA(I))
      XCHI(I)=-(RLSH+D2I*BETA2*COSDEL)
      ZCHI(I)=ZCX(I)*BETA2
      GO TO 9
C STRAIGHT SEGMENT
 2    ITYP(I)=0
      XCX(I)=-RLSH+(FLOAT(I-1-N)+0.5)*DELX
      ZCX(I)=D2
      XCHI(I)=XCX(I)
      ZCHI(I)=ZCX(I)*0.5
      GO TO 9
C STRAIGHT SEGMENT
 3    ITYP(I)=0
      XCX(I)=(FLOAT(I-N-M-1)+0.5)*DELX
      ZCX(I)=D2
      XCHI(I)=XCX(I)
      ZCHI(I)=ZCX(I)*0.5
      GO TO 9
C CURVED SEGMENT
C IF NOT INITIAL CALL SKIP CALCULATIONS
 4    IF(ICALL) 9,400,9
 400  DELTA(I)=(FLOAT(I-N-2*M-1)+0.5)*BETA
      SINDEL=SIN(DELTA(I))
      XCX(I)=RLSH+D2I*SINDEL
      ZCX(I)=D2I*COS(DELTA(I))
      XCHI(I)=RLSH+D2I*BETA2*SINDEL
      ZCHI(I)=ZCX(I)*BETA2
      GO TO 9
C CURVED SEGMENT
C IF NOT INITIAL CALL SKIP CALCULATIONS
 5    IF(ICALL)  9,500,9
 500  ITYP(I)=1
      DELTA(I)=(FLOAT(I-2*N-2*M-1)+0.5)*BETA
      COSDEL=COS(DELTA(I))
      XCX(I)=RLSH+D2I*COSDEL
      ZCX(I)=-D2I*SIN(DELTA(I))
      XCHI(I)=RLSH+D2I*COSDEL*BETA2
      ZCHI(I)=ZCX(I)*BETA2
      GO TO 9
```

```
C STRAIGHT SEGMENT
 6      ITYP(I)=0
        XCX(I)=RLSH-(FLOAT(I-3*N-2*M-1)+0.5)*DELX
        ZCX(I)=-D2
        XCHI(I)=XCX(I)
        ZCHI(I)=ZCX(I)*0.5
        GO TO 9
C STRAIGHT SEGMENT
 7      ITYP(I)=0
        XCX(I)=-(FLOAT(I-3*N-3*M-1)+0.5)*DELX
        ZCX(I)=-D2
        XCHI(I)=XCX(I)
        ZCHI(I)=ZCX(I)*0.5
        GO TO 9
C CURVED SEGMENT
C IF NOT INITIAL CALL SKIP CALCULATIONS
 8      IF(ICALL)  9,800,9
 800    ITYP(I)=1
        DELTA(I)=(FLOAT(I-3*N-4*M-1)+0.5)*BETA
        SINDEL=SIN(DELTA(I))
        XCX(I)=-(RLSH+D2I*SINDEL)
        ZCX(I)=-D2I*COS(DELTA(I))
        XCHI(I)=-(RLSH+D2I*SINDEL*BETA2)
        ZCHI(I)=ZCX(I)*BETA2
 9      CONTINUE
 10     CONTINUE
        RETURN
        END
        SUBROUTINE TK(ISHAPE,PHI2,R2,R1,PHI1,L1,L2,HY,A,B,L,LS)
C TRUNK GEOMETRY CALCULATIONS
        REAL L,L1,L2,LS
        RTOL=.1
        IF(HY.LE.0.0) GO TO 111
C***********************************************************************
C ITERATION FOR R2
C COMPUTE INNER RADIUS OF CURVATURE
        R2=SQRT(A*A*0.25+HY*HY)
C
C ITERATION LOOP FOR L2,SU,P1,R2
        DO 102 I=1,50
        PHI2=ABS(ACOS(AMAX1(-1.0,AMIN1(1.0,((R2-HY)/R2)))))
         SINPH2=SIN(PHI2)
C COMPUTE OUTER RADIUS OF CURVATURE
        R1=((A-R2*SINPH2)**2+(B+HY)**2)/(2.*(B+HY))
        PHI1=ABS(ACOS(AMAX1(-1.0,AMIN1(1.0,((R1-HY-B)/R1)))))
         XS=A-R2*SINPH2
         IF (XS.LE.0.0) PHI1=6.2831852-PHI1
         L2=L-PHI1*R1
C R2S IS RESULTANT RADIUS FOR COMPUTED L2&IN ITERATION
        IF(ABS(PHI2).LT.1.0E-2) PHI2=1.0E-2
         R2S=L2/PHI2
C TEST IF TOLERANCE .GT. ERROR
         IF(ABS(R2-R2S).LE.RTOL) GO TO 50
        R2=(R2+R2S)*0.5
 102    CONTINUE
C***********************************************************************
C ITERATED 50 TIMES WITHOUT SUCCESS,ERROR RETURN
 111    CONTINUE
        WRITE(6,9011)
 9011     FORMAT(10X,' INFEASABLE TRUNK GEOMETRY '//)
        ISHAPE=0
        RETURN
C TRUNK OK,RETURN
 50      L1=L-L2
        ISHAPE=1
        RETURN
```

```
      END
      SUBROUTINE S1(D2I,A1,A2,SI,D2D2HBI,DXAMABI,BETAD2I,L1I,L2I,R1I,R2I
     X,SINPHRI,PHI2,R2,DELX,BETA,HY,R1,PHI1,ICALL,L1,L2,ITYP,D,A,E,N,M,V
     XTKI,ACHI,LS)
C INITIAL ASSESSMENT OF AREAS,VOLUMES ASSUMING
C NO GROUND CONTACT
      REAL L1,L2,L1I,L2I,LS
      DIMENSION ITYP(32),ATKI(32),VTKI(32),ACHI(32)
      NSTOP=32       B=E
C COMPUTE GEOMETRY TERMS
      SINPH2=SIN(PHI2)
      SINPHR=SINPH2*R2
      D2=D/2.+SINPHR
      DD2=DELX*D2
      BDD2=BETA*D2*D2*0.5
       X=B*(A-SINPHR)/(B+HY-R1)
C COMPUTE AREAS OF TRUNK SECTORS
      A1=PHI2/2.0*R2**2
      A2=(R2-HY)/2.0*SINPHR
      A3=PHI1/2.0*R1**2
      A4=X*B/2.0
      A5=(A-SINPHR-X)/2.0*(HY-R1)
      X1=SINPHR-4.0*(SIN(PHI2/2.0))**2*R2/(3.0*PHI2)
      X2=0.66667*SINPHR
       X3=SINPHR+4.0*(SIN(PHI1/2.0))**2*R1/(3.0*PHI1)
      X4=A-0.333333*X
       X5=SINPHR+0.333333*(A-SINPHR-X)
      S=2.0*LS+6.28318*D2
      AA=A1+A3+A5-A2-A4
      AX=A1*X1-A2*X2+A3*X3-A4*X4+A5*X5
      IF(ICALL.GT.0) GO TO 20
C SAVE TRUNK GEOMETRY TERMS FOR END TRUNK CALCULATIONS
      R1I=R1
      R2I=R2
      PHI1I=PHI1
      PHI2I=PHI2
      L1I=L1
      L2I=L2
      A1I=A1
       A2I=A2
      SINPH2I=SINPH2
      SINPHRI=SINPHR
      X1I=X1
      X2I=X2
      A1MA2I=A1-A2
      D2I=D2
      SI=S
      BETAD2I=BETA*D2
      X12=(X1*A1-X2*A2)/A1MA2I
      DXAMABI=(D*0.5+X12)*A1MA2I*BETA
      D2D2HBI=D2*D2*0.5*BETA
      DD2I=DELX*D2
      BDD2I=BETA*D2*D2*0.5
   20 CONTINUE
C COMPUTE TRUNK SEGMENT AREA,VOLUME,CUSHION AREA
      DO 103 I=1,NSTOP
      IF(ITYP(I).EQ.1)GO TO 112
C STRAIGHT PART OF TRUNK
  131 ATKI(I)=AA
      VTKI(I)=DELX*ATKI(I)
      ACHI(I)=DD2
      GO TO 102
C CURVED PART OF TRUNK
  112 IF(ICALL.GT.0) GO TO 102
      ATKI(I)=AA
      XE=AX/ATKI(I)
```

```
      VTKI(I)=BETA*(D/2.+XE)*ATKI(I)
      ACHI(I)=BOD2
  102 CONTINUE
  103 CONTINUE
      RETURN $END
      SUBROUTINE HC(Z,X)
C SUBROUTINE TO CALCULATE POSITIONS OF
C SIDE TRUNK LOBES
C THE POSITION  (EXPRESSED BY HY) DEPENDS ON PRESSURES
C I.E. HY/HYI=F(PCH/PTK)
C FORCE INPUT PRESSURE RATIO BETWEEN 0.0 AND 1.0
      AH0=1.003 $AH1=-.865 $AH2=.328 $AH3=-.463
      X=AMIN1(1.0,AMAX1(0.0,X))
      Z=AH0+AH1*X+AH2*X*X+AH3*X*X*X
      IF(Z.LT.0.1)Z=0.1
      IF (Z.GT.1.0) Z=1.0
      RETURN
      END
      SUBROUTINE CL(YGH,ICLN,PHIE,THETAE,SL4,YG)
      DIMENSION SL4(32 ),YG(32 ),YGH(32 )
      NSTOP=32
C CALCULATION OF TRUNK GROUND CLEARANCE FOR EACH SEGMENT
      ICLNS=ICLN
      COSCOS=COS(PHIE)*COS(THETAE)
C CALCULATE SEGMENT GAP
      DO 161 I=1,NSTOP
      YGH(I)=SL4(I)-YG(I)*COSCOS
C IF NEGATIVE SET GAP TO ZERO
      YGH(I)=AMAX1(YGH(I),0.0)
      IF(YGH(I).LE.0.0) ICLN=ICLN+1
  161 CONTINUE
      ICLN=ICLNS
        RETURN
       END
      SUBROUTINE CO(SL4,XCX,ZCX,YCG,XCG,SIE,PHIE,THETAE,CC,FF,GG)
C THIS SUBROUTINE CALCULATES X AND Z COORDINATES OF THE GROUND
C POINT CORRESPONDING TO EACH SEGMENT, FOR A PARTICULAR ACLS
C ORIENTATION
      DIMENSION SL4(32 ),XCX(32 ),ZCX(32 ),XG(32 ),ZG(32 )
      NSTOP=32
C CALL BMATRIX FOR SPACIAL TRANSFORMATION
C BMATRIX TRANSFORMS A VECTOR FROM VEHICLE FREAME TO INERTIAL FRAME
C
C
C CALCULATE TRANSCENDENTALS
      CSIE=COS(SIE)
      CPHIE=COS(PHIE)
      CTHETAE=COS(THETAE)
      SSIE=SIN(SIE)
      SPHIE=SIN(PHIE)
      STHETAE=SIN(THETAE)
C
C COMPUTE TRANSLATION MATRIX ELEMENTS
      B11=CSIE*CPHIE+STHETAE*SPHIE*SSIE
      B12=SPHIE*CTHETAE
      B13=-SSIE*CPHIE+STHETAE*SPHIE*CSIE
      B21=-SPHIE*CSIE+SSIE*CPHIE*STHETAE
      B22=CPHIE*CTHETAE
      B23=SPHIE*SSIE+CSIE*CPHIE*STHETAE
      B31=SSIE*CTHETAE
      B32=-STHETAE
      B33=CSIE*CTHETAE
C
C DO LOOP OF ALL SEGMENTS TO GROUND POSITION
      DO 104 I=1,NSTOP
      XCXCC=(XCX(I)-CC)
```

```
      ZCXFF=(ZCX(I)-FF)
C
C CALCULATE VECTOR DA FOR SEGMENT
      SL4(I)=(YCG+XCXCC*B12+ZCXFF*B32)/B22-GG
      SL4GG=(SL4(I)+GG)
C
C CALCULATE X-GROUND COORDINATE
      XG(I)=XCXCC*B11-SL4GG*B21+ZCXFF*B31+XCG
C
C CALCULATE Z-GROUND COORDINATE
      ZG(I)=XCXCC*B13-SL4GG*B23+ZCXFF*B33
  104 CONTINUE
      RETURN
      END
      SUBROUTINE PR(YG)
C USER SPECIFIED GROUND PROFILE.
C ELEVATION YG(I) IS EXPRESSED AS A FUNCTION OF X AND Z COORDINATRES
C OF GROUND POINT I,I.E. XG(I) AND ZG(I)
      DIMENSION YG(32 )
      NSTOP=32
      DO 105 I=1,NSTOP
C SET FOR FLAT TERRAIN
      YG(I)=0.
  105 CONTINUE
      RETURN
      END
      SUBROUTINE SP(VTK,VCH,ACH,PHI2,R2,D2I,BETA,A1,A2,DELX,SI,ITYP,YGH,
     XHY,D2D2HBI,DXAHA9I,BETAD2I,XCHI,ZCHI,L2,L1I,L2I,F1I,R2I,DELTA,SINP
     XHRI,XCK,D,LS,AH,NX,NH,HYI,LX,SH,VCHD,R1,AGAP,ATKAT,ATKCH,ATKCN,ATK
     XATC,ATKCHC,ISEG,VTKI,ACHI,ACHR,ATKCNI,ATKCNR,PERI,XCH,XTK,ZCH,ZTK,
     XL,L1)
C CALCULATION OF AREAS AND VOLUMES ASSOCIATED WITH ACLS,KNOWING ITS
C ORIENTATION
      REAL L2,LS,LX,L1I,L2I,L,L1
      DIMENSION ATKR9(32 ),VTKRA(32 ),VTKRD(32 ),PERI(32 ),XCH(32 ),ZCH(
     X32 ),ATKCNI(32 ),ATKCNR(32 ),ATKATI(32 ),ATKATR(32 ),VCHI(32 ),AGA
     XPI(32 ),AGAPR(32 ),XTK(32 ),ZTK(32 ),ATKR(32 ),ACHI(32),XCHI(32 ),
     XZCHI(32 ),YGH(32 ),ISEG( 32),VTKI( 32),XCY(32),ITYP(32),DELTA(32),
     XACHR(32),VCHR(32),VTKF(32),ATKCHR(32),ATKCHI(32)
      NSTOP=32                                        $PI=3.141592653
      ATOTAL=.6
C COMPUTE PARTIAL TERMS
      SINPH2=SIN(PHI2)
      SINPHR=SINPH2*R2
      D2=D*0.5+SINPHR
      D2ISQ=D2I*D2I
      A1MA2=A1-A2
      BETA2=1.33333*SIN(BETA/2.)/BETA
      RLSH=LS*0.50
      ADS=AH*DELX/SI
      ABDSI=AH*BETA*D2I/SI
      RNN=FLOAT(NX*NH)
C*****************************************************************
C PART 1 I VALUE OF VCH AND AGAP
C*****************************************************************
      DO 17 I=1,NSTOP
C TEST FOR TRUNK SEGMENT, WHETHER CURVED OR STRAIGHT
      IF(ITYP(I).EQ.1)GO TO 11
C
C STRAIGHT PART OF TRUNK
C
C CALCULATE CUSHION SEGMENT INITIAL VOLUME
 13   VCHI(I)=(YGH(I)*D2-A1MA2)*DELX
C CALCULATE SEGMENT GAP AREA
      AGAPI(I)=(YGH(I)-HY)*DELX
      GO TO 10
```

```
C
C CURVED PART OF TRUNK
C
 11    VCHI(I)=YGH(I)*DZDZHBI-DXAMABI
       ACAPI(I)=(YGH(I)-HYI)*BETADZI
C*******************************************************************
C PART 2 R VALUE CALCULATIONS
C*******************************************************************
C
C TEST FOR GROUND CONTACT AT EACH SEGMENT
 10    CONTINUE
C FORCE VOLUME AREAS .GE.0
       VCHI(I)=AMAX1(0.0,VCHI(I))
       ACAPI(I)=AMAX1(0.0,ACAPI(I))
C TEST SEGMENT FOR CONTACT
       IF(ITYP(I).EQ.1.AND.YGH(I).LE.HYI) GO TO 14
       IF(ITYP(I).EQ.0.AND.YGH(I).LE.HY) GO TO 23
C
C NO GROUND CONTACT
C SET CONTACT AND REMOVE TERMS TO ZERO
       ATKR(I)=0.0
       ACHR(I)=0.0
       VTKR(I)=0.0
       VCHR(I)=0.0
       ACAPH(I)=0.0
       ATKCNI(I)=0.0
       ATKCNR(I)=0.0
       ATKCHR(I)=0.0
       ATKATR(I)=0.0
       PERI(I)=0.0
C SET DISTANCES X,Z TO FREE TRUNK VALUES
       XCH(I)=XCHI(I)
       ZCH(I)=ZCHI(I)
       ZTK(I)=ZCH(I)
       XTK(I)=XCH(I)
C COMPUTE TRUNK-CUSHION-ATMOSPHERE BLEED AREAS
       IF(ITYP(I)) 16,16,18
 16    CONTINUE
C NO CONTACT STRAIGHT SECTIONS
       ATKCHI(I)=FLOAT(IFIX((L2-LX)/SH+1.0)*NH)*ADS
       ATKATI(I)=RNN*ADS-ATKCHI(I)
       GO TO 17
 18    CONTINUE
C NO CONTACT CURVED SECTIONS
       ATKCHI(I)=FLOAT(IFIX((L2I-LX)/SH+1.0)*NH)*ABOSI
       ATKATI(I)=RNN*ABOSI-ATKCHI(I)
       GO TO 17
C*******************************************************************
C
C TRUNK GROUND CONTACT
C
C CURVED PART OF TRUNK
C CALCULATE DEFORMATION ANGLES FOR SEGMENT
 14    PHI3=ACOS((R2I-(HYI-YGH(I)))/R2I)
       PHI4=ACOS((R1I-(HYI-YGH(I)))/R1I)
       SINPH3=SIN(PHI3)
       SINPH4=SIN(PHI4)
C COMPUTE PARTIAL TERMS
       DRSP=(D2I-R2I*SINPH3)
       DRSP2=DRSP*DRSP
       COSDEL=COS(DELTA(I))
       SINDEL=SIN(DELTA(I))
       DFDRSN=DETA2*DRSP*SINDEL
       DFDRCS=DETA2*DRSP*COSDEL
C COMPUTE REMOVAL SECTORS
       A6=R2I*R2I*PHI3*0.5
```

```
      A7=(R2I-HYI*YGH(I))*0.5*R2I*SINPH3
      A6MA7=A6-A7
      A8=R1I*R1I*PHI4*0.5
      A9=(R1I-HYI*YGH(I))*0.5*R1I*SINPH4
      A10=A9
      A11=A8
C COMPUTE SECTOR CENTROIDS
      X6=SINPHRI-1.333333*(SIN(PHI3*0.5)**2)*R2I/PHI3
      X7=SINPHRI-0.333333*R2I*SINPH3
      X8=SINPHRI+1.3333333*(SIN(PHI4*0.5)**2)*R1I/PHI4
      X9=SINPHRI+0.3333333*R1I*SINPH4
      X10=X9
      X11=X8
      PII2=PI*0.5
      IF(PHI4.LT.PII2) GO TO 50
C IF PHI4 GREATER THAN 90 DEGREES, SET TO 90 DEGREES
      PHI4=PII2
      SINPH4=SIN(PHI4)
      A10=(R1I-HYI*YGH(I))*R1I
      X10=SINPHRI+0.5*R1I
      A11=R1I*R1I*PHI4*0.5
      X11=SINPHR+1.333333*(SIN(PHI4*0.5)**2)*R1I/PHI4
 50   CONTINUE
C COMPUTE TRUNK AREA CHANGE
      ATKR(I)=A6MA7+A8-A9
      ATKRB(I)=A6MA7+A11-A10
      XER=(A6*X6-A7*X7+A8*X8-A9*X9)/ATKR(I)
      XERB=(A6*X6-A7*X7+A11*X11-A10*X10)/ATKRB(I)
C COMPUTE TRUNK VOLUME CHANGE
      VTKRA(I)=BETA*(D*0.5+XER)*ATKR(I)
      VTKRB(I)=ATKRB(I)*BETA*(D*0.5+XERB)
      VTKR(I)=2.*VTKRA(I)-VTKRB(I)
      XCR=(A6*X6-A7*X7)/A6MA7
C COMPUTE TRUNK EXIT AREAS
      ATKCHI(I)=FLOAT(IFIX((L2I-LX-R2I*PHI3)/SH+1.0)*NH)*ABOSI
      ATKATI(I)=FLOAT(IFIX((L1I-L+LX+FLOAT(NX+1)*SH-R1I*PHI4)/SH+1.0)*NH
     1)*ABOSI
      ATKCHR(I)=FLOAT(IFIX((L2I-LX)/SH+1.0)*NH)*ABOSI-ATKCHI(I)
      ATKATR(I)=NHH*ABOSI-ATKCHI(I)-ATKATI(I)-ATKCHR(I)
      PERI(I)=BETA*(DRSP+D2I+R1I*SINPH4)
C COMPUTE CONTACT PERIMETER
      ATKCNI(I)=BETA*0.5*(D2ISQ-DRSP2)
C COMPUTE TRUNK CONTACT AREA
      ATKCNR(I)=BETA*0.5*((D2I+R1I*SINPH4)**2-D2ISQ)
      ACHR(I)=ATKCNI(I)
C COMPUTE CUSHION VOLUME CHANGE
      VCHR(I)=-BETA*A6MA7*(D*0.5+XCR)
C COMPUTE GAP AREA CHANGE
      AGAPR(I)=AGAPI(I)
C DISTANCE OF SEGMENT PRESSURE CENTERS FROM CUSHION CENTER
 29   RR=D2I+R1I*SINPH4
      RR1=C2I-R2I*SINPH3
      XX2=1.333333*SIN(BETA*0.5)/BETA*(RR**3-RR1**3)/(RR*RR-RR1*RR1)
      KGO=ISEG(I)
      GO TO (61,23,23,64,65,23,23,68),KGO
 61   XCH(I)=-RLSH-DEDRCS
      ZCH(I)=DEDRSN
      XTK(I)=-RLSH-XX2*COSDEL
      ZTK(I)=XX2*SINDEL
      GO TO 17
 64   XCH(I)=RLSH+DEDRSN
      ZCH(I)=DEDRCS
      XTK(I)=RLSH+XX2*SINDEL
      ZTK(I)=XX2*COSDEL
      GO TO 17
 65   XCH(I)=RLSH+DEDRCS
```

```
      ZCH(I)=-BEDRSN
      XTK(I)=RLSH+XX2*COSDEL
      ZTK(I)=-XX2*SINDEL
      GO TO 17
 60   XCH(I)=-RLSH-BEDRSN
      ZCH(I)=-BEDRCS
      XTK(I)=-RLSH-XX2*SINDEL
      ZTK(I)=-XX2*COSDEL
      GO TO 17
C*********************************************************************
C
C TRUNK GROUND CONTACT
C STRAIGHT PART OF TRUNK
 23   CONTINUE
C COMPUTE DEFORMATION ANGLES
      RHY=((R2-(HY-YGH(I)))/R2)
      PHI3=ACOS(AMAX1(-1.0,AMIN1(1.0,RHY)))
      RHY=((R1-(HY-YGH(I)))/R1)
      PHI4=ACOS(AMAX1(-1.0,AMIN1(1.0,RHY)))
C DO TRANSCENDENTALS ONLY ONCE
      SINPH3=SIN(PHI3)
      SINPH4=SIN(PHI4)
C COMPUTE PARTIAL TERMS
      DRSP=(D2-R2*SINPH3)
      DRSP2=DRSP*DRSP
      COSDEL=COS(DELTA(I))
      SINDEL=SIN(DELTA(I))
      BEDRSN=BETA2*DRSP*SINDEL
      BEDRCS=BETA2*DRSP*COSDEL
C COMPUTE REMOVAL SECTORS
      A6=R2*R2*PHI3*0.5
      A7=(R2-HY +YGH(I))*0.5*R2*SINPH3
      A6MA7=A6-A7
      A8=R1*R1*PHI4*0.5
      A9=(R1-HY +YGH(I))*0.5*R1*SINPH4
      A10=A9
      A11=A8
C COMPUTE SECTOR CENTROIDS
      X6=SINPHR-1.333333*(SIN(PHI3*0.5)**2)*R2/PHI3
      X7=SINPHR-0.333333*R2*SINPH3
      X8=SINPHR+1.3333333*(SIN(PHI4*0.5)**2)*R1/PHI4
      X9=SINPHR+0.3333333*R1*SINPH4
      X10=X9
      X11=X8
      PII2=PI*0.5
      IF(PHI4.LT.PII2) GO TO 70
C IF PHI4 IS GREATER THAN 90 DEGREES, SET TO 90 DEGREES
      PHI4=PII2
      SINPH4=SIN(PHI4)
      A10=(R1-HY+YGH(I))*R1
      X10=SINPHR+0.5*R1
      A11=R1*R1*PHI4*0.5
      X11=SINPHR+1.3333333*(SIN(PHI4*0.5)**2)*R1/PHI4
 70   CONTINUE
C COMPUTE TRUNK AREA CHANGE
      ATKR(I)=A6MA7+A8-A9
      ATKR3(I)=A6MA7+A11-A10
C COMPUTE TRUNK VOLUME CHANGE
      VTKRA(I)=ATKR(I)*DELX
      VTKR3(I)=ATKR3(I)*DELX
      VTKR(I)=2.*VTKRA(I)-VTKR3(I)
C COMPUTE CUSHION VOLUME CHANGE
      VCHR(I)=-DELX*A6MA7
C COMPUTE TRUNK EXIT AREAS
      ATKATI(I)=FLOAT(IFIX((L1-L+LX+FLOAT(NX-1)*SH-R1*PHI4)/SH+1.0)*NH)
     1*ADS
```

```
      ATKCHI(I)=FLOAT(IFIX((L2-LX-R2*PHI3)/SH+1.)*NH)*ADS
      ATKCHR(I)=FLOAT(IFIX((L2-LX)/SH+1.0)*NH)*ADS-ATKCHI(I)
      ATKATR(I)=RNN*ADS-ATKCHI(I)-ATKATI(I)-ATKCHR(I)
      PERI(I)=2.*DELX
C COMPUTE TRUNK CONTACT PERIMETER
C COMPUTE TRUNK CONTACT AREA
      ATKCNI(I)=R2*SINPH3*DELX
      ATKCNR(I)=R1*SINPH4*DELX
C COMPUTE GAP AREA CHANGE
      ACHR(I)=ATKCNI(I)
      AGAPR(I)=AGAPI(I)
      KGO=ISEG(I)
C COMPUTE SEGMENT CONTACT CENTER OF PRESSURE FOR CUSHION AND TRUNK
      GO TO (17,62,62,17,17,66,66,17),KGO
 62   XCH(I)=XCX(I)
      ZCH(I)=0.5*(D2-R2*SINPH3)
      XTK(I)=XCX(I)
      ZTK(I)=D2+0.5*(R1*SINPH4-R2*SINPH3)
      GO TO 17
 66   XCH(I)=XCX(I)
      ZCH(I)=-0.5*(D2-R2*SINPH3)
      XTK(I)=XCX(I)
      ZTK(I)=-(D2+0.5*(R1*SINPH4-R2*SINPH3))
 17   CONTINUE
C*****************************************************************$
C PART 3 SUMMATION OF SEGMENT AREAS VOLUMES
C*****************************************************************$
C
C SET TOTAL AREA AND VOLUMES TO ZERO
      ATKCN=0.0
      VTK=0.0
      ACH=0.0
      ATKCH=0.0
      ATKAT=0.0
      VCH=0.0
      AGAP=0.0
      ATKATC=0.
      ATKCHC=0.
C LOOP ON SEGMENTS TO FIND TOTALS OF AREAS AND VOLUMES.
      DO 30 I=1,NSTOP
      VTK=VTK+ (VTKI(I)-VTKR(I))
      ACH=ACH+ (ACHI(I)-ACHR(I))
      ATKCH=ATKCH+ATKCHI(I)
      ATKAT=ATKAT+ATKATI(I)
      VCH=VCH+ (VCHI(I)-VCHR(I))
      ATKCN=ATKCN+ATKCNI(I)+ATKCNR(I)
      AGAP=AGAP+ (AGAPI(I)-AGAPR(I))
      ATKATC=ATKATC+ATKATR(I)
      ATKCHC=ATKCHC+ATKCHR(I)
 30   CONTINUE
      AGAP=AMAX1(AGAP,0.0)
      VTK=AMAX1(0.0,VTK)
      VCH=AMAX1(0.0,(VCH+VCHO))
      VCH=AMAX1(0.0,VCH)
      ATKCH=AMAX1(0.000,ATKCH)
      ATKAT=AMAX1(0.000,ATKAT)
      ACH=AMAX1(0.0,ACH)
      ATKATC=AMAX1(0.0,ATKATC)
      ATKCHC=AMAX1(0.0,ATKCHC)
C   FORCE SUM OF NOZZLE AREAS TO BE EQUAL TO TOTAL NOZZLE AREA
      SUM=ATKAT+ATKCH+ATKATC+ATKCHC
      ATKCH=ATKCH/SUM*ATOTAL $ATKAT=ATKAT/SUM*ATOTAL
      ATKCHC=ATKCHC/SUM*ATOTAL $ATKATC=ATKATC/SUM*ATOTAL
      RETURN
      END
      SUBROUTINE STIF(Y,TZ,TX,DERY,IPP,PPLM,VPLM,VTK,QFANX,DVTK,PTK,DVCH,
```

```
     XYCG,CVCMP,PCH,VCH,SINKRT,PHIE,DPHI ,THETAE,DTHETA,CKK,PAT,GG,HYI,G
     XEC,MASS,RPM,                        VELX,          SIE,RHO,AGAP,ACH,ATK
     XAT,ATKCH,ATKCN,ATKATC,ATKCHC,CAF,CENFX,CGAP,       ACHR,ATKCNI,ATKCN
     CR,PERI,XCH,ZCH,XTK,ZTK,ACHI,YGH,QPLTK,QTKCH,QTKAT,QCHAT)
C DYNAMIC FAN VERSION FOR FMA.
C STATE EQUATIONS FOR THE DYNAMIC SYSTEM
      REAL MASS
C FOLLOWING SUBROUTINES ARE CALLED TO UPDATE VALUES OF
C FORCES,TORQUES AND FLOWS, GIVEN THE NEW VALUES OF THE
C STATE VARIABLES
      DIMENSION DERY(13),ACHI(32),ACHR(32),ATKCNI(32),ATKCNR(32),PERI(32
     X),XCH(32),ZCH(32),XTK(32),ZTK(32),YGH(32)
      CC=-1.175AATFN=.796         $CPT=.6 $CTA=.4          $QVENT=0.
            CTC=.4 $FF=0.
                 HDC=1.          $PHA=155.          $U=0.03
                          APLTK=.698  $DAMPC=3.2          $AIFAN=.072
                         CENFZ=0.     $NSTOP=32
C SUBROUTINE TO FIND FLOW AND PRESSURE VALUES DURING DYNAMIC SIMULATION
      TIRHO=2.0/RHO
C PLENUM TO TRUNK FLOW
      SIGN=1.0
      IF((PPLM-PTK).LT.0.0) SIGN=-1.0
      QPLTK=SIGN*CPT*APLTK*SQRT(ABS(TIRHO*(PPLM-PTK)))
C TRUNK TO CUSHION FLOW
      SIGN=1.
      IF((PTK-PCH).LT.0.) SIGN=-1.
      QTKCH=SIGN*CTC*SQRT(ABS(TIRHO*(PTK-PCH)))*(ATKCH+0.66667*ATKCHC)
C TRUNK TO ATMOSPHERE FLOW
      SIGN=1.0
      IF(PTK.LT.0.0)SIGN=-1.
      QTKAT=SIGN*CTA*SQRT(TIRHO*ABS(PTK))*(ATKAT+0.66667*ATKATC)
C CUSHION TO ATMOSPHERE FLOW
      SIGN=1.0
      IF(PCH.LT.0.0) SIGN=-1.0
      QCHAT=AGAP*CGAP*SQRT(TIRHO*ABS(PCH))*SIGN
C FORCES AND TORQUES ASSOCIATED WITH A PARTICULAR ACLS ORIENTATION
C ARE CALCULATED
C CALCULATE TRANSCENDENTALS ONLY ONCE
      CSSCS=COS(PHIE)*SIN(THETAE)*SIN(SIE)-COS(SIE)*SIN(PHIE)
      CPCT=COS(PHIE)*COS(THETAE)
C CLEAR TOTAL FORCES AND TORQUES TO ZERO
      FORCT=0.0
      TTPX=0.0
      TTPZ=0.0
      TCPX=0.0
      TCPZ=0.0
      TORFZ=0.0
      TORQTX=0.0
      TORQTZ=0.0 $FORCEY=0. $TORQUEX=0. $TORQUEZ=0.
C  FORCES AND TORQUES INDEPENDENT OF SEGMENTS INDIVIDUALLY
C HEAVE FORCES CUSHION AND TRUNK
      FCP=PCH*ACH
      FTP=PTK*ATKCN
C COMPUTE VELOCITY FOR DRAG FORCE
      V=VELX*CSSCS+SINKRT*CPCT
      SIGN=1.
      IF(V.GT.0.0)SIGN=-1.
C HEAVE DRAG FORCE
      FDF=0.5*HDC*PHA*RHO*V*V*SIGN
C DRAG TORQUE
      TDFZ=FDF*CENFX
      TDFX=-FDF*CENFZ
C FORCES AND TORQUES DEPENDENT ON SEGMENTS INDIVIDUALLY
C SUM INDIVIDUAL SEGMENTS TO FIND TOTALS
      DO 101 I=1,NSTOP
C CUSHION PRESSURE TORQUES
```

```
      TCPZ=TCPZ+(XCH(I)-CC)*PCH*(ACHI(I)-ACHR(I))
      TCPX=TCPX-(ZCH(I)-FF)*PCH*(ACHI(I)-ACHR(I))
      TTPZ=TTPZ+(XTK(I)-CC)*(PTK*(ATKCNI(I)+ATKCNR(I)))
      TTPX=TTPX-(ZTK(I)-FF)*(PTK*(ATKCNI(I)+ATKCNR(I)))
      IF((ATKCNI(I).GT.0.).OR.(ATKCNR(I).GT.0.)) GO TO 111
      GO TO 101
111   VELT=              SINKRT*CPCT+DPHI*(XTK(I)-CC)+DTHETA*(ZTK(I)-FF)
      FORD=-VELT*DAMPC*PERI(I)
      FORCT=FORCT+FORD
      TOROTZ=TOROTZ+(XTK(I)-CC)*FORD
      TOROTX=TOROTX-(ZTK(I)-FF)*FORD
      IF(VELX.EQ.0.0) GO TO 101
      TORFZ=TORFZ-(YCG*CPCT) *PTK*(ATKCNI(I)+ATKCNR(I))*U
 101  CONTINUE
C SUMMATION OF FORCE AND TORQUE COMPONENTS
C TOTAL HEAVE FORCE
      FORCEY=(FCP+FTP+FORCT+FDF)*CPCT
      FY=AMAX1(FORCEY,0.)
C TOTAL TORQUE X AXIS
      TORQUEX=TCPX+TTPX+TOROTX+TOFX
      TX=TORQUEX
C TOTAL TORQUE Z AXIS
      TORQUEZ=TCPZ+TTPZ+TOROTZ+TOFZ+TORFZ
      TZ=TORQUEZ
C STATE EQUATIONS
C***THE STATE VARIABLES***
C 1)PPLM..PLENUM PRESSURE (GAGE)
C 2)PCH..CUSHION PRESSURE (GAGE)
C 3)PTK..TRUNK PRESSURE (GAGE)
C 4)SINKRT..VERTICAL SINK RATE, POSITIVE UPWARDS
C 5)YCG..CG ELEVATION
C 6)DPHI..PITCH RATE,VEHICLE FRAME
C 7)DTHETA..ROLL RATE, VEHICLE FRAME
C 8)THETAE..EULERIAN ROLL ANGLE
C 9)PHIE..EULERIAN PITCH ANGLE
C 10)SIE..EULERIAN YAW ANGLE (APPROX. ZERO)
C 11)XV..DISPLN OF PRESSURE RELIEF VALVE
C 12)VV..VELOCITY OF PRESSURE RELIEF VALVE
C 13)QFANX..FAN AIR INERTANCE FLOW
      DERY(1)=(CKK*(PPLM+PAT)/VPLM)*(QFANX-QPLTK)
      DERY(3)=(CKK*(PTK+PAT)/VTK)*(QPLTK-QTKCH-QTKAT-QVTK)
C CUSHION FLOW ABOVE GROUND EFFECT TRANSITION ZONE
      QCHFT=QPLCH+QTKCH-QVCH
C CALCULATE GROUND EFFECT TRANSITION ZONE
      TBOUND=GG+HYI*(1.+GEC)
      BBOUND=GG+HYI
C DETERMINE IF ACLS IN TRANSITION ZONE
      IF(YCG.GT.TBOUND)GO TO 13
       IF(YCG.GT.BBOUND) GO TO 14
      GO TO 16
C
C ABOVE TRANSITION ZONE
 13   QCHAT=QCHFT
      IFLAG=0
      IO=0
      NO=0
      GO TO 15
C
C IN TRANSITION ZONE
 14   IFLAG=1
      NO=100
      IO=IFIX(ABS((TBOUND-YCG)/(TBOUND-BBOUND)*FLOAT(NO)))
      IF(IO.GT.NO)IO=NO
      GO TO 17
C
C IN GROUND EFFECT ZONE
```

```
 16     NO=1
        IO=1
        IFLAG=2
C COMPUTE CUSHION TO ATMOSPHERE FLOW
 17     QCHAT=FLOAT(NO-IO)/FLOAT(NO)*QCHFT+FLOAT(IO)/FLOAT(NO)*QCHAT
C CUSHION PRESSURE DERIVATIVE
 15     DERY(2)=(QPLCH+QTKCH-QCHAT-DVCH+DVCHF*PCH*DERY(3)/(PTK*PTK))/
       1(VCH/(CKK*(PCH+PAT))+DVCHP/PTK)
C CUSHION PRESSURE IS ZERO ABOVE TRANSITION ZONE
        IF(IFLAG.EQ.0.0) PCH=0.0
 66     RETURN
        END
PARAMETER VALUES=IO1AV=0
        UMAX=20.
       MSS=77.5                                                        CC=-1.17
       GG= 1.5    ,FF=0. ,PHA=155. ,HOC=1. ,LS=4.125 ,D=.417 ,A=.695
       E=.524,L=5.17 ,HYI=1.00 ,                       AH=3.4E-4 ,SH=.0833 ,LX=0.80
        AIF=.072
                   APC=0. ,APT=.698 ,APA=0. ,AAT=.796             ,VCD=0.
        DPV=8,                                FAT=2116.8 ,TEM=70.
                          ,CKK=1.4 ,CPA=0. ,CAF=.5 ,CPC=0. ,CPT=.5 ,CTC=.4
        CTA=.4 ,CGP=.5 ,GEC=.1 ,      CFX= 1.17 ,CFZ=0. ,DPC=3.2
XA LO=0.   ,XU LO=0.     ,ZA LO=0.     ,MALLO=0.     ,MU LO=0.     ,XP1LO=-.2
                         Y9DLO=0.      ,YB LO=0.     ,YDALO=0,     ,LBDLO=0.
LB LO=0.   ,NBDLO=0.     ,NB LO=0.     ,NDALD=0.     ,RUDLO=1.     ,B  LO=19.
TCOEN=6.4 ,THREN=2353. ,X0 FN=-6.1,GAXEN=1.   ,GAZEN=0.   ,P0 TF=158.73,
P1 TF=17.46,Z0 TF=0.   ,Z1 TF=158.73
                                                       ,Z0 TF 2=158.73
Z1 TF 2=0.,P0 TF 2=153.73,P1 TF 2=17.46,
GKIIT 3=1.                               ,GKLIT 3=1.
C1 MC 1=1.2,C2 MC 1=3.75,C3 MC 1=2.5,C4 MC 1=0.,GKIIT 7=1.,GKLIT 7=1.
GKIIT 8=-1.,GKLIT 8=1.,C1 MC 2=1.94,C2 MC 2=5.75,C3 MC 2=2.5,C4 MC 2=0.,
GKIIT 9=1.,GKLIT 9=1. ,GKIIT10=-1.,GKLIT10=1. ,ALSAV=0.    ,S  AV=76.
UW AV=0. ,VW AV=+0.   ,WW AV=0.   ,PW AV=0.    ,QW1AV=0.     ,RW1AV=0.
MA1LO=77.5,C  LO=4.    ,GKLIT 1=1. ,GKIIT 1=1. ,IXXSO=2414.,IYYSO=1840. ,
IZZSO=+082.,IXZSO=-203. ,Z0 EN=0. ,ZU LO=0.
        FY1S3=0. ,FY2S3=0. ,TZ2S3=0. ,FX2S3=0. ,TX1S3=0. ,TZ1S3=0. ,
        FX3S3=0. ,FY3S3=0. ,FZ3S3=0.
                 X0 LO=-39.692,ZADLO=- .12366,YP LO=-.035692,ZQ LO=-8.0257,
MADLO=-.75699,YR LO=.12081,LP LO=-113.74,XDELO=-.065052,MQ LO=-175.86,
YDRLO=350.17,LR LO=-19.316,NP LO=-11.851,Z0 LO=-10.967,ZDELO=-15.562,
LDRLO=607.,NR LO=100.93,MO LO=13.725,MDELO=-865.13,LOALC=-3.6745,
NDRLO=-119.
INITIAL CONDITIONS=
PPLFM=137.93,PTKFM=126.06,PCHFM=0.    ,TH EN=1515.6,U  SD=17.033,
V  SD=6.5235,W  SD=0.      ,P  SD=0.     ,Q  SD=0 .0      R  SD=0.
ROLSD=0.      ,PITSD=-1.0    ,YAWSD=0.       ,ALTSD=2.6  0,XI TF=0.
X2 TF=0.     ,X2 IT 7=0.        ,X2 IT 8=0.        ,X2 IT 3=0.
X2 IT 1=0.,XI TF 2=0.,X2 TF 2=0.,X2 IT 9=0.,X2 IT10=0.
  PRINT CONTROL=4,TINC=0.5 ,TMAX=10.,OUTRATE=1   ,PRATE= 1,SIMULATE
```

Vita

Edward Arthur Kenney was born 18 July 1948 in London, Ontario, Canada to James A. and Evelyn V. Kenney. After graduating from G. A. Wheable Secondary School in 1967, he entered the Royal Military College (RMC) in Kingston, Ontario. He graduated from RMC in 1971 with the degree of Bachelor of Science in Electrical Engineering, and a commission in the Canadian Armed Forces. After graduating from the Primary Flying School and attending the Flying Training School at CFB Moosejaw, Saskatchewan, he reclassified to be an Aerospace Engineer. Subsequent tours of duty included CFB Shearwater, Nova Scotia, as the Deputy Maintenance Records Officer and Heavy Maintenance Repair Officer; H.M.C.S. Preserver as the Fleet Air Maintenance Officer; and National Defence Headquarters, Ottawa, in the Directorate of Avionics and Armament Subsystems Engineering. Capt Kenney was assigned to the United States Air Force Institute of Technology in June 1976 in the Graduate Guidance and Control Program.

Mailing Address: National Defence Headquarters
Ottawa, Ontario Canada
K1A OK2
Attn: DPCO/AERE

AFIT/GE/EE/77D-43 UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFIT/GE/EE/77D-43	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): The Design and Simulation of a Takeoff Stabilization System for an Aircraft with an Air Cushion Landing System	5. TYPE OF REPORT & PERIOD COVERED: Master's thesis
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Edward A. Kenney	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Air Force Institute of Technology (AFIT/EN) Wright-Patterson AFB, Ohio 45433	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: Vehicle Equipment Division, Mechanical Branch, (FEMB), Air Force Flight Dynamics Laboratory, Wright-Patterson AFB, Ohio 45433	12. REPORT DATE: Dec[redacted]77
	13. NUMBER OF PAGES: 117
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Approved for public release; IAW AFR190-17

JERRAL F. GUESS, Captain, USAF
Director of Information

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

ACLS
SAS
Coanda Effect
Thrust Vector Device
Aircraft
EASY
ACTS

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

The inherent instability in pitch and roll associated with an Air Cushion Landing System (ACLS) aircraft at low airspeeds was investigated, and a means to aid control in pitch and roll was developed. The control system required the use of vertical wing tip thrusters which provided thrust up or down depending on the control signal (similar to space vehicle thrusters). These thrusters could be activated alternately to control roll angle and roll rate with the use of a bang-bang optimal controller.

As well, the thrusters would be set forward of the aircraft center of gravity and could be activated in tandum to aid in pitch control.

The Jindivik Remotely Piloted Vehicle, an Australian target drone, was fitted with an ACLS and taxi tests showed the instability and need for a stabilization system. Subsequent use of Jindivik wind tunnel and taxi test data served as the basis for the development of the roll/pitch control system presented in this paper. Due to computational problems with the air cushion model of the computer program, the controller designs could not be completely verified; but expected trends in pitch, roll, and yaw control were shown.