(NASA-CR-152303-Vol-6-Pt-1) GASP- GENERAL AVIATION SYNTHESIS PROGRAM. VOLUME 6: PERFORMANCE. PART 1: THEORETICAL DEVELOPMENT (Aerophysics Research Corp., Bellevue, Wash.) 121 p HC A06/MF A01 N81-19093

Unclas G3/05 16994

GASP - GENERAL AVIATION SYNTHESIS PROGRAM

NASA-CR-152303

VOLUME VI - PERFORMANCE

PART 1 - THEORETICAL DEVELOPMENT

JANUARY 1978

Prepared for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Ames Research Center
Moffett Field, California

Under

CONTRACT NAS 2-9352

AEROPHYSICS RESEARCH CORPORATION

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF FIGURES

VI.1 PERFORMANCE

Aircraft performance modeling requires consideration of propulsion, aerodynamics and weight characteristics, and the interaction between the subroutines may be quite complex. This is particularly true when a large number of realistic physical constraints are included in the performance specification. Table VI.1.1 shows the interaction between the subroutines as discussed in this volume of the study.

Table VI.1.1

	PERFORMANCE SUBROUTINES
PERFRM (70 cards)	DEFINES MISSION
ACCEL (70)	RECTILINEAR SPEED-UP
CLIMB (250)	CONSTRAINED CLIMBING FLIGHT
DLAND (170)	LANDING MANEUVER
TAKOFF (350)	TAKEOFF MANEUVER TO FLAPS-UP
DERIV (90)	DETERMINES DERVIATIVES AND CHECKS CONSTRAINTS FOR TAKOFF ROUTINE
TAXI (25)	FUEL CONSUMPTION DURING GROUND IDLE
TURN (40)	CONSTANT ALTITUDE TURN MANEUVER
XRANGE (340)	CRUISE RANGE COMPUTATION
ASPEED (90)	DETERMINING SPEED AT SPECIFIED POWER FOR EQUILIBRIUM HORIZONTAL FLIGHT
RGBAL (120)	CONTROL ROUTINE WHEN A RANGE OR ENDURANCE IS SPECIFIED IN INPUT

VI.1.1 Subroutine PERFRM

This subroutine is called only by MAIN, and it acts principally to organize and unify the computations carried out by other performance subroutines. The call has the form CALL PERFRM (I_{SEG}, I_{COND}, I_{FLY}). The mission segment number is I_{SEG}, which is associated with a specific subroutine; i.e., as shown in Figure VI.1.1, the mission segment varies as follows:

$$I_{SEG} = \begin{cases} 1, \text{ Call TAXI} \\ 2, \text{ Call TAKOFF} \\ 3, \text{ Call ACCEL} \\ 4, \text{ Call CLIMB} \\ 5, \text{ Write Performance Data and Advance } I_{SEG} \\ 6, \text{ Call XRANGE} \\ 7, \text{ Call DLAND} \\ 8, \text{ Call TURN} \\ 9, \text{ Return} \end{cases}$$

The other call parameters are

$$I_{COND} = \begin{cases} 0, \text{ Write data heading and set } I_{COND} = 1 \\ 1, \text{ Initialize weight, altitude, etc., and continue} \end{cases}$$

$$I_{FLY} = \begin{cases} 1, \text{ Call all performance subroutines, unless } I_{DC} = 99 \\ 2, \text{ Call TAXI and TAKOFF only} \\ 3, \text{ Call TAXI, TAKOFF and CLIMB only} \end{cases}$$

The other significant input parameters to the subroutine will be described as they are needed in the discussion to follow.

The program logic begins before statement 5, where I_{SEG} is augmented for later use;

$$I_{SEG} = I_{SEG} + 1 \qquad \text{(VI.1.1)}$$

and the gross weight W and altitude H are specified in terms of the input parameters W_G, WT_{MISN}, and H_{OO}. The various mission segments are then analyzed in the order implied by I_{SEG}. The basic order is taxi, takeoff, climb, cruise, reserve, and landing with acceleration between takeoff and climb and climb and cruise.

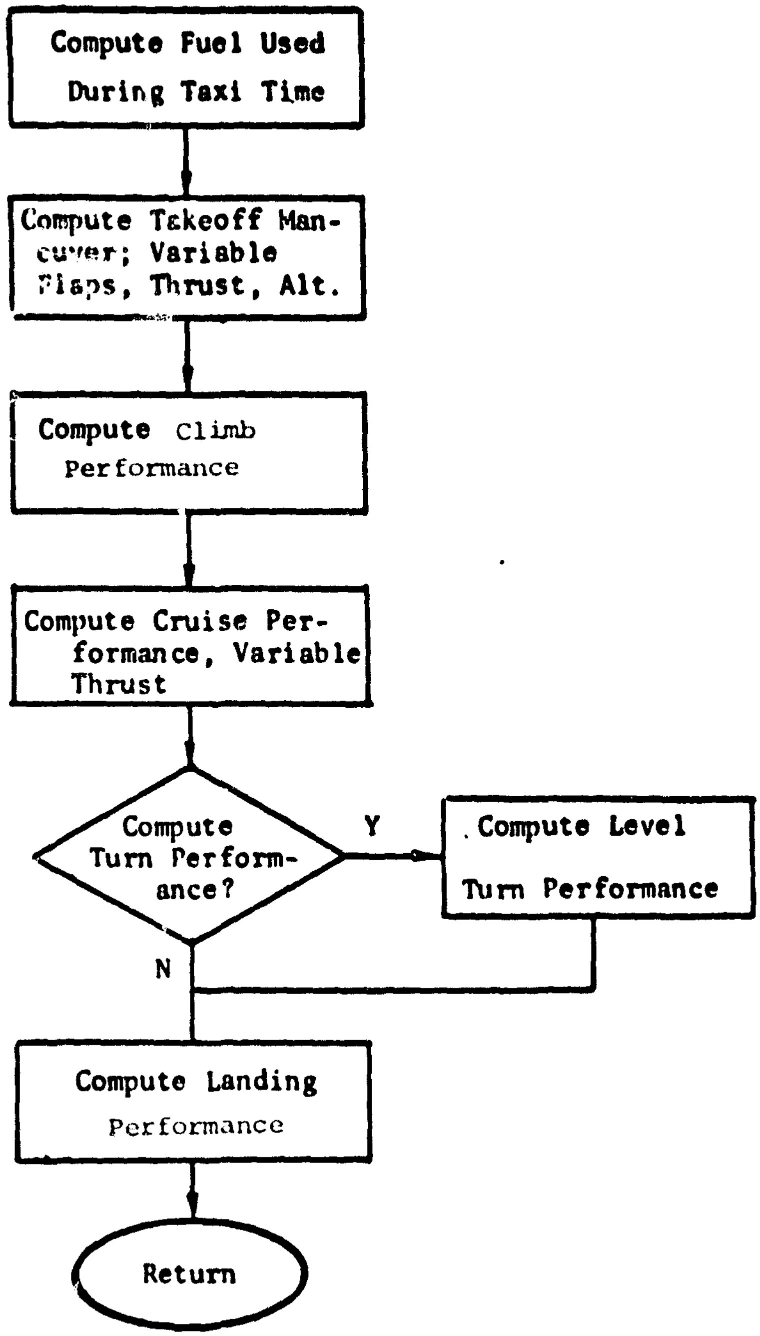


Figure VI.1.1 Subroutine PERFRM Flow Chart

For $I_{SEG} = 1$, the subroutine TAXI requires as input the runway altitude H, the initial time S_T, and the taxi time, DELTT. It returns the parameters

W = gross weight after taxi, lb

WF = fuel burned in taxi, lb

ST = time at end of taxi, sec

and then if $I_{DC} \neq 99$, I_{SEG} is augmented.

For $I_{SEG} = 2$, the subroutine TAKOFF is called, which computes the time history of the take-off maneuver, from ground roll to a specified height. The standard maneuver is divided into sub-segments, or "events," as follows:

(1) Begin ground roll
(2) Rotate aircraft
(3) Lift off
(4) Begin gear retraction
(5) Distance to an altitude of 35 ft
(6) Distance to an altitude of 50 ft
(7) Begin flap retraction
(8) End take-off

In addition, for multi-engine aircraft, the optional engine out takeoff performance can be computed. This consists of the continued takeoff performance to an altitude of 50 ft with an engine failed and the accelerate-stop distance divided into the events,

(1) Begin ground roll
(2) Engine failure
(3) Remove power
(4) Apply brakes
(5) Stop aircraft

The subroutine ACCEL (SPEED) is called, between mission segments to accelerate the aircraft to the speed of the next segment. This subroutine models the rectilinear speed-up maneuver at constant altitude, during which weight and speed change monotonically, until the velocity reaches the value SPEED. SPEED may be specified in knots or Mach number.

When I_{SEG} is 4, CLIMB is called, with the altitude argument

$$H_{ENDCL} = \begin{cases} CR_{ALT}, & \text{if } EM_{TURN} = 0 \\ H_{TURN}, & \text{if } EM_{TURN} \neq 0 \end{cases} \qquad \text{(VI.1.2)}$$

EM_{TURN} is the Mach number during the turn. If turning performance is desired it will be computed after the climb otherwise the climb is to the cruise altitude CR_{ALT}. If no value is input for CR_{ALT} it will be defaulted to EM_{CRU}.

The cruise portion of the mission (I_{SEG} = 6) is analyzed by a call to subroutine XRANGE (W_{FCRU}, W_{FRES}), where

W_{FCRU} = fuel used during cruise, lb

W_{FRES} = reserve fuel, lb

are calculated by XRANGE in terms of three different payloads and three different Mach numbers.

Subsequently, I_{SEG} is advanced to 7, for which subroutine DLAND is called, using the flag I_{LER} = 99. This subroutine computes the landing field performance.

If the optional turning performance is desired, subroutine TURN is called, after the climb segment. This brief subroutine calculates aircraft turning performance at constant altitude, Mach number, and load factor, and it calculates various characteristics of this steady state maneuver.

VI.1.2 Subroutine ACCEL

This subroutine models the thrust and drag characteristics of an aircraft while its forward velocity is changed from an initial value to a required final value.

The subroutine begins by calling TPALT, which returns various atmospheric properties existing at the initial altitude. The static pressure P_o is converted to lb per sq ft, and the speed of sound is given in ft/sec as

$$S_A = 49.1 \text{ SQRT } (T_o) \quad \text{(VI.1.3)}$$

The air density is a function of pressure, temperature and gravity,

$$RHO = P_o / [53.32 \ T_o \ G] \quad \text{(VI.1.4)}$$

and these units are in slugs per cu ft. The single calling argument of the subroutine is a command Mach number or velocity, SPEED, which must be greater than the current Mach number, EM, in order for the positive acceleration modeling to be effective. If the calling argument, SPEED, is greater than 5.0 the argument is considered to be the command velocity in knots. The initial velocity in ft per sec is given by

$$V = EM * S_A \quad \text{(VI.1.5)}$$

the time-step interval is 0.5 sec, and the velocity difference to be made up is

$$DEL_V = X_{MN} \ S_A - V \quad \text{(VI.1.6)}$$

The initial value of time is then defined before the loop which begins at statement 1. After augmenting the time, which is measured in hours by the variable TIME, this value is compared to 0.2 hr. That is, twelve minutes is considered time enough to accelerate to any practical value of Mach number. The lift coefficient is

$$C_L = W/[.7 \ S_W \ P_o \ EM^2] \quad \text{(VI.1.7)}$$

where other input parameters are

S_W = reference wing area, sq ft
W = current weight, lbs

This value of lift coefficient and the Mach number are input to subroutine DRAG, and the appropriate atmospheric properties are input to subroutine ENGINE. Together, these permit computation of thrust in lb,

$$T = F_N \, EN_P \, , \tag{VI.1.8}$$

fuel consumption in lb per hr,

$$W_{FUEL} = W_{FUEL} \, EN_P \, , \tag{VI.1.9}$$

true airspeed in knots,

$$TAS = EM \; S_A/1.69 \, , \tag{VI.1.10}$$

and equivalent airspeed,

$$EAS = TAS/SQRT \; (RHO/.0023769) \tag{VI.1.11}$$

where

EN_P = number of engines

F_N = thrust per engine, lb

EM = Mach number

RHO = air density at altitude, slugs per cu ft

The acceleration is then expressed by Newton's Law as

$$DVDT = (32.2/W)*(T - C_D*S_W*.7 \; P_o \; EM^2) \tag{VI.1.12}$$

where the drag coefficient C_D is known from the earlier call to subroutine DRAG. It is expected that the thrust exceeds the drag, so that this forward acceleration is positive. If not, the subroutine merely writes a number of output quantities before returning. Otherwise, the velocity is augmented,

$$V = V + DVDT * DT \tag{VI.1.13}$$

and the Mach number is

$$EM = V/S_A \tag{VI.1.14}$$

The aircraft weight and range are found, in lb and nm, as

$$W = W - W_{FUEL} * DT/3600. \tag{VI.1.15}$$

$$R = R + V * DT/6076. \tag{VI.1.16}$$

and the fuel consumed

$$W_F = W_F + W_{FUEL}\ DT/3600. \qquad \text{(VI.1.17)}$$

If the current Mach number is less than the command value, the time is augmented at statement 1, and the loop is repeated. Otherwise, the final values of Mach number and true and equivalent airspeed are calculated corresponding to the input value, X_{MN}, at statement 2, and a final line of output is printed.

VI.1.3 Subroutine CLIMB

This is a subroutine of 250 cards which is called in order to combine aerodynamic characteristics of the aircraft with its thrust characteristics, to simulate the planar climb maneuver to the altitude H_{ENDCL}. The external subroutines CLIFT, DRAG, ENGINE and TPALT are called, and three internal functions are defined in this calculation. Additional input quantities are constraints on pitch angle, altitude step size, maximum airspeed and atmospheric properties.

The subroutine first initializes a set of parameters, ending with an estimated value of maximum lift coefficient C_{LL}. The input parameter ICLM specifies the type of climb maneuver desired,

$$ICLM = \begin{cases} 1, \text{ Climb at maximum rate of climb} \\ 2, \text{ Climb at maximum allowable operating speeds} \\ 3, \text{ Climb at specified equivalent airspeed} \end{cases} \quad (VI.1.18)$$

Following a write statement, the loop which begins at Statement 1 is the time-varying outer loop governing the climb maneuver. It begins with the call to TPALT, which returns static pressure P_o, gravity G and temperature T_o corresponding to the current (initial) altitude, H (ft). The static pressure is expressed in lb per sq ft and the speed of sound is given in kts,

$$S_A = 49.1\ (T_o)^{1/2}/1.689 \quad (VI.1.19)$$

and the local air density in slugs per cu ft follows according to the equation of state,

$$RHO = P_o/(53.32\ T_o\ G) \quad (VI.1.20)$$

the minimum climb speed is expressed in kts and based on the estimated maximum lift coefficient, CLL,

$$V_L = 1.1\ SQRT[W/(1.426\ S_W\ RHO\ C_{LL})] \quad (VI.1.21)$$

where W is the current weight and S_W the wing area, and the factor 1.1 provides a margin above the stall speed. If this velocity is less than 50 kts, it is raised to 50.

Following the setting to zero of several integer variables, the maximum allowable velocity at the current altitude is calculated as limited by the allowable equivalent airspeed, V_{MO}, the maximum operating Mach number, EM_{MO}, the speed limit below 10,000 feet altitude, or the cruise Mach number, EM_{CRU}.

The next lines of the subroutine related these constraints on the maximum velocity:

$$V_{MAX} = V_{MO}/SQRT(RHO/.0023769) \tag{VI.1.22}$$

(a) If $I_{CLM} = 3$ and $V_{CLMB} < V_{MO}$,

$$V_{MAX} = V_{CLMB}/SQRT(RHO/.0023769) \tag{VI.1.23}$$

(b) If $H < 10000.$ ft and $V_{MO} > 250$ kts,

$$V_{MAX} = 250./SQRT(RHO/.0023769) \tag{VI.1.24}$$

(c) If $V_{MAX} > S_A EM_{MO}$ (speed corresponding to maximum Mach number),

$$V_{MAX} = S_A\ EM_{MO} \ . \tag{VI.1.25}$$

The velocity V_2 is next defined as

$$V_2 = \begin{cases} \min(V_{MAX}, V_{CRU}, V_{LIMIT}), & I_{CLM} = 1 \\ \min(V_{MAX}, V_{CRU}), & I_{CLM} = 2 \text{ or } 3 \end{cases} \tag{VI.1.26}$$

where

$$V_{CRU} = SA\ EM_{CRU} \tag{VI.1.27}$$

$$V_{LIMIT} = V_{LIMX}/SQRT(RHO/.0023769) \tag{VI.1.28}$$

and

EM_{CRU} = cruise Mach number, from COMMON/UNIVX/

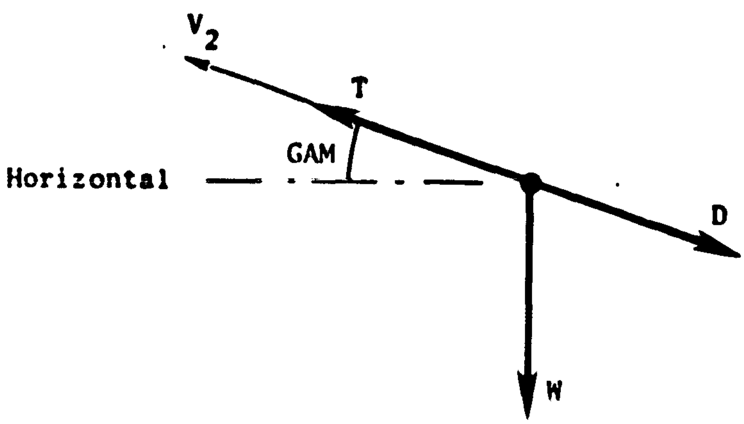
V_{LIMX} = sea level limit speed, from COMMON /TOCLM/

All of these velocities correspond to steady-state horizontal flight, and are measured in kts.

The loop begins with the dynamic pressure and lift coefficient,

$$Q = 1.426 \text{ RHO } V_2^2 \qquad \text{(VI.1.29)}$$

$$C_L = W \text{ COS}(RGAM_1)/(Q * S_W) \qquad \text{(VI.1.30)}$$



$$T = D + W \sin(GAM)$$

$$\text{Rate of Climb} = V_2 \sin(GAM) = V_2(T - D)/W$$

Figure VI.1.2 Force Equilibrium in Steady Climb

where the numerical factor accounts for the units of V_2 (knots) and of Q (lb per sq ft).

The subroutine CLIFT returns the angle of attack, ALPHA, required for this lift coefficient, and a rate of climb multiplier is defined as

$$PHI = \begin{cases} 1. & (DELTA_H = 0) \\ 1 + V_2(V_2-V_1)/(11.278\ DELTA_H) & \end{cases} \qquad (VI.1.31)$$

The parameter $DELTA_H$ is initially at zero, but it later measures the altitude increment remaining to be climbed. The subroutine DRAG returns a value for C_D, and ENGINE returns a value for thrust per engine, T_A, which is multiplied by the number of engines, EN_P. The rate of climb is expressed in ft per min as (see Fig. VI.1.2).

$$RC_{POW} = 101.34\ V_2(T_A - C_D S_W Q)/W \qquad (VI.1.32)$$

If the velocity V_2 is too large, the rate of climb is negative so that the end of this iterative loop is indicated by,

$$V_2 = V_2 - 10. \qquad (VI.1.33)$$

$$RC_1 = RC_{POW} \qquad (VI.1.34)$$

If V_2 is now less than V_L, as given by Eq. VI.1.21, insufficient power is available for climbing, and the subroutine returns. Otherwise, another iteration of the rate of climb loop is performed at this lower speed, V_2. However, when RC_{POW} is positive, control passes to statement 6, for computation of the flight path angle,

$$R_{GAM2} = ARSIN[RC_{POW}/(101.34\ V_2)] \qquad (VI.1.35)$$

When this is within 0.1 deg of its value on the previous iteration, convergence has occurred and control passes to statement 7. Similarly, the next seven lines are concerned with finding the flight path angle, R_{GAM4}, at which the pitch attitude angle is a maximum at THE_{MAX}, and the corresponding rate of climb is RC_{TETA}.

Assuming that $I_{CLM} = 1$, such that the aircraft climbs at maximum rate, the equations between statements 9 and 18 are devoted to an analysis of the rate of climb with the pitch angle constraint, RC_{TETA}, relative to the rate of climb with the power constraint, RC_{POW}. This portion of the program involves the use of rather complex iterative procedures, including the three functions:

$$PART(X,X_1,X_2,X_3,Y) = \frac{(X-X_1)(X-X_2)}{(X_3-X_1)(X_3-X_2)} \qquad \text{(VI.1.36)}$$

$$BO2A(V_1,V_2,V_3,U_1,U_2,U_3) = \frac{V_1^2(U_2-U_3)+V_2^2(U_3-U_1)+V_3^2(U_1-U_2)}{2[V_1(U_3-U_2)+V_2(U_1-U_3)+V_3(U_2-U_1)]} \qquad \text{(VI.1.37)}$$

$$ADEN(W_1,W_2,W_3,Z_1,Z_2,Z_3) = Z_1(W_3-W_2)+Z_2(W_1-W_3)+Z_3(W_2-W_1) \qquad \text{(VI.1.38)}$$

When both of these constraints have been satisfied, control passes to statement 20, where the angle of attack is computed as

$$ALPHA = THE_{MAX} + EYE_W - RGAM_4 * ROTD \qquad \text{(VI.1.39)}$$

where

EYE_W = input wing incidence angle, deg.

The lift coefficient is then found by a call to CLIFT, and the drag coefficient follows from a call to DRAG. The total thrust required in the equilibrium climb condition is found as

$$T_A = W * PHI * RC_{TETA}/(101.34 * V_2) + C_D\ S_W\ Q \qquad \text{(VI.1.40)}$$

and after division by the number of engines EN_P, subroutine ENGINE returns the fuel flow rate per engine and other quantities related to the propulsion system. Following computation of the equivalent airspeed EAS, the altitude is compared with H_{ENDCL}, and until it is greater, the integers

$$N_{QUO} = H/DEL_H + .05 \qquad \text{(VI.1.41)}$$

and

$$YALE_{22} = N_{QUO} + 1 \quad . \tag{VI.1.42}$$

are found, so that the altitude

$$H_9 = YALE_{22} * DEL_H \tag{VI.1.43}$$

will be the next value of H, whether or not H_9 is less than H_{ENDCL}. The apparent purpose of this logic is to develop an integral number of additional computational steps for the remaining altitude.

The closing computations involves the routine computations for the rate of climb,

$$RC = RC_{POW}/PHI, \tag{VI.1.44}$$

and the flight path angle,

$$RGAM_2 = ARSIN[RC/(101.34 * V_2)] \quad , \tag{VI.1.45}$$

which in degrees is,

$$GAMMA = RGAM_2 * RTOD \quad , \tag{VI.1.46}$$

and the lift coefficient,

$$C_L = W\ COS(RGAM_2)/(Q * S_W) \tag{VI.1.47}$$

The angle of attack ALPHA is then returned in degrees by a call to CLIFT, and the pitch angle is

$$THETA_F = GAMMA - EYE_W + ALPHA \quad . \tag{VI.1.48}$$

Finally, the velocity is $V = V_2$.

VI.1.4 Subroutine DLAND

This subroutine models the landing maneuver in four segments; a glide from 50 ft altitude, a flare to touchdown, a delay before braking and a braked roll. Figure VI.1.3 shows the flare and touchdown schematics.

The program begins with the specification of landing weight, which is determined by the input integer I_{WLD}, which is defined as

$$I_{WLD} = \begin{cases} 0, & \text{landing weight equals gross weight} \\ 1, & \text{landing weight equals weight at end of previous mission segment} \\ 2, & \text{landing weight equals input percent of gross weight} \end{cases}$$

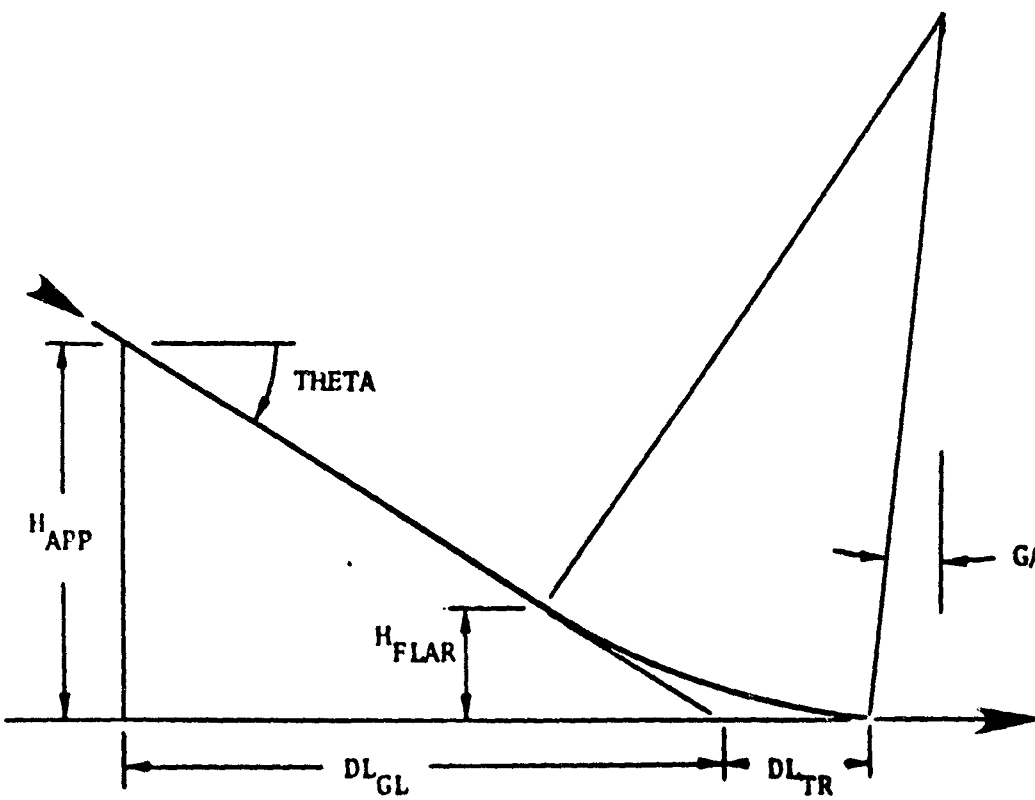


Figure VI.1.3 Flare and Touchdown Maneuver

The next five cards specify the landing weight W_L (lb) and the wing loading WOS (lb per sq ft), and the initial altitude is

$$H_{IN} = H_{APP} + ALT_{LND} \tag{VI.1.49}$$

where

H_{APP} = approach altitude above runway, ft

ALT_{LND} = altitude above sea level of runway, ft

The subroutine TPALT again provides static temperature T_o and pressure P_o at the initial altitude, which may then be modified for an increment $TDEL_{LD}$ to the temperature, and by converting the pressure to lb per sq ft. The air density in slugs per cu ft is

$$RHO = P_o/(53.32\ G\ T_{AMB}) \tag{VI.1.50}$$

where the ambient temperature, in degrees Rankine, is

$$T_{AMB} = T_o + TDEL_{LD} \tag{VI.1.51}$$

and

G = acceleration of gravity, ft per sec^2

The square root of the density ratio is useful in computing the equivalent airspeed at sea level. This factor is denoted by

$$SR_{DRAT} = SQRT(RHO/.002378) \tag{VI.1.52}$$

Next, the flight path angle $RGAM_1$, the angle of attack ALPHA, the approach altitude in wing spans HOB, and the stall Mach number EM_S are defined, prior to a call of subroutine CLIFT, which returns the appropriate maximum lift coefficient C_{LMX}. The associated stall speed in ft per sec is given by

$$V_{ST} = SQRT[2.\ WOS/(RHO * C_{LMX})] \tag{VI.1.53}$$

and this is converted to the equivalent airspeed

$$V_{STEAS} = SR_{DRAT}\ V_{ST} \tag{VI.1.54}$$

If the aircraft is single engine, and this velocity is less than the maximum stall speed of 61 kts or 103.03 fps, per the Federal Air Regulations Part 23, the maximum wing loading is calculated and printed as

$$WGS_{MAX} = 12.6212\ C_{LMX} \quad , \tag{VI.1.55}$$

where the numerical factor is the dynamic pressure at sea level, in lb per sq ft, corresponding to this stall speed.

The glide velocity in ft/sec is next defined as the product

$$V_{GL} = V_{RAT}\ V_{ST} \tag{VI.1.56}$$

where

V_{RAT} = input approach speed margin (default value = 1.3)

and the height above the ground in wing spans at the beginning of the glide is expressed as

$$H_{OB} = H_{APP}/B \tag{VI.1.57}$$

where

H_{APP} = approach altitude, ft

B = wing span, ft

The Mach number and Reynolds number per unit length are given by

$$EM_{G} = V_{GL}/SQRT(1.4 * G * 53.32 * T_{AMB}) \tag{VI.1.58}$$

$$RE_{LI} = V_{GL}/X_{KV} \tag{VI.1.59}$$

where

X_{KV} = kinematic viscosity, as returned by TPALT.

Next, the subroutine AERO is called, which returns a number of coefficients appropriate to the drag coefficient of the aircraft of interest. The dynamic pressure is calculated by the usual equation, and if $I_{ENGSZ} = 1$, subroutine ENGINE returns a value for idling thrust, T_{IDLE}.

The flight path angle iteration loop begins at statement 7, where the lift coefficient is computed in terms of the flight path angle,

$$C_L = WOS * COS(R_{GAM1})/Q_{GL} \tag{VI.1.60}$$

Here, R_{GAM1} is the variable being calculated in this loop, and it is positive when the sink rate is positive. The wing loading is WOS lb per sq ft and the dynamic pressure Q_{GL} corresponds to the velocity, V_{GL}. The drag coefficient appropriate to the approach configuration is returned by subroutine DRAG, and the positive rate of sink is expressed for the idle thrust as

$$RS_{IDLE} = 60.0\ V_{GL}\ (C_D\ S_W\ Q_{GL} - T_{IDLE} * EN_P)/W_L \tag{VI.1.61}$$

where the numerical factor is to get the units of ft/min.

This equation is followed by another for the rate of sink RS, in which the actual thrust is expressed as T_A. The equilibrium flight path angle associated with RS is given by

$$RGAM_2 = ARSIN[RS/60.0 * V_{GL}] \tag{VI.1.62}$$

and if this is within .1 deg of the previous value, convergence has occurred. Otherwise, $RGAM_1 = RGAM_2$ and control returns to statement 7, where the lift coefficient is recomputed according to Eq. VI.1.60.

After this loop has converged, several constraints must be checked, which may require the loop to be re-entered. The sink speed is checked to see if it is limited by the idle thrust value found in Eq. VI.1.61 or exceeds the maximum allowable value RS_{MX}. If the maximum allowable value is exceeded, the necessary thrust increase is computed to provide the desired sink rate,

$$T_A = C_D\ S_W\ Q_{GL} - W_L\ RS_{MX}/60.0\ V_{GL}) \tag{VI.1.63}$$

before reconverging to the flight path angle $RGAM_2$.

At statement 15, the flare maneuver is initiated with computation of the initial pitch angle and the final flight path angle at touchdown,

$$THETA = ARSIN[RS/60.0\ V_{GL})] \tag{VI.1.64}$$

$$GAM_{TD} = SINK_{TD}/V_{GL}\ , \tag{VI.1.65}$$

where

$SINK_{TD}$ = vertical velocity at touchdown (3 ft per sec or input)

The input variable XLF_{MX} indicates the maximum allowable load factor in g's if less than 4., and the radius during the flare is computed as

$$R_Z = V_{GL}^2/[G(XLF_{MX} - 1.)] \tag{VI.1.66}$$

and the altitude at which the flare is initiated is approximated as

$$H_{FLAR} = V_{GL}^2(THETA^2 - GAM_{TD}^2)/[2.\ G(XLF_{MAX} - 1.)] \tag{VI.1.67}$$

Otherwise, the flare radius is

$$R_Z = V_{GL}^2/[G(XLF - 1.)] \tag{VI.1.68}$$

where

$$XLF = V_{GL}^2(THETA^2 - GAM_{TD}^2)/(2 * G * H_{FLAR}) + 1. \tag{VI.1.69}$$

However, if the input variable XLF_{MX} is greater or equal to 4 it is taken as the flare height in ft,

$$H_{FLAR} = XLF_{MX} \tag{VI.1.70}$$

The computation continues with the air distance from approach altitude H_{APP} to intersection with the ground, if H_{FLAR} is less than H_{APP},

$$DL_{GL} = H_{APP}/TAN(THETA) \tag{VI.1.71}$$

and from this point to touchdown (see Fig. VI.1.3).

$$DL_{TR} = R_Z \; THETA/2.(1 - GAM_{TD}/THETA)^2 \tag{VI.1.72}$$

The latter expression is a small-angle approximation to

$$DL_{TR} = R_Z\left[\sin(THETA)-\sin(GAM_{TD}) + \frac{\cos(THETA)-\cos(GAM_{TD})}{\tan(THETA)}\right]$$

If H_{FLAR} exceeds H_{APP}, $DL_{GL} = 0$ and the air track distance is, to first order,

$$DL_{TR} = R_Z[SQRT(2\; H_{APP}/R_Z + GAM_{TD}^2) - GAM_{TD}] \tag{VI.1.73}$$

The touchdown velocity is expressed as the average of the stall speed and the approach speed

$$V_{TD} = (V_{RAT} + 1.)V_{ST}/2. \tag{VI.1.74}$$

where $V_{RAT} > 1$ is the ratio of approach speed to stall speed. The delay distance, over which the velocity changes, is

$$D_{DELAY} = V_{TD}\; T_{DELAY} \tag{VI.1.75}$$

where T_{DELAY} is input or has the default value 1 sec.

Other parameters which are then computed are the height in wing spans above the ground and the Mach number at touchdown. These are given respectively by

$$HOB = H_{TG}/B \tag{VI.1.76}$$

and

$$EM_L = V_{TD}/SQRT(1.4 * G * 53.32 * T_{AMB}) \tag{VI.1.77}$$

where

H_{TG} = altitude at touchdown

These are required inputs to the subroutines CLIFT and DRAG, which are next called. They return the gear-down lift and drag coefficients, C_{LRL} and C_{DRL}. The net drag is the sum of aerodynamic drag and friction drag, which is proportional to the difference of weight and lift. A one-dimensional equation of motion takes the form,

$$\frac{dV}{dt} = V\frac{dV}{dX} = \frac{g}{W}\left[T - D - \mu(W - L)\right]$$

where the symbols have standard definitions and where μ is the coefficient of friction. In the present case, D and L are both proportional to V^2, and the equation becomes separable, as follows:

$$\frac{VdV}{F_{RAT} + A_{RAT}\ V^2} = -\ 2.\ A_{RAT}\ dX$$

where

$$A_{RAT} = DL_{RL}/[C_{LMX}\ V_{ST}^2] \tag{VI.1.78}$$

$$F_{RAT} = MU_B - TOW_L \tag{VI.1.79}$$

$$DL_{RL} = C_{DRL} - MU_B\ C_{LRL} \tag{VI.1.80}$$

and

MU_B = friction coefficient

TOW_L = thrust to weight ratio at landing

The equation can then be integrated, to give the ground roll distance X as

$$DL_G = -13.0287[W_{OS}/(D_{RAT}\ DL_{RL})]\ \log\left[\frac{F_{RAT}}{F_{RAT} + A_{RAT}}\right] \tag{VI.1.81}$$

where the numerical factor is the inverse of the product ... sea level air density and gravity. The decceleration during the ground roll maneuver in g's is the quantity

$$D_{OG} = V_{TO}^2/(2 * G * D_{LG}) \tag{VI.1.82}$$

Finally, the total landing distance is the sum,

$$DL_T = DL_G + DL_{TR} + D_{DELAY} + DL_{GL} \tag{VI.1.83}$$

The remaining steps in the subroutine relate to satisfying a runway length constraint by iterating on the airplane's wing loading, W_{GS}. This logic is dependent on the input value of runway length requirement, X_{LDGRQ}, and it requires two iterations on the wing loading. An interpolation or extrapolation of the associated runway lengths then develops a required maximum wing loading.

The procedure is described as follows. Initially $I_{LER} = 1$ and $I_{WGS} = 1$, when DLAND is first called by MAIN. If DL_T is found to be less than the required value of runway length, the only remaining computations deal with conversion of velocities to knots and with augmenting the runway length to meet FAR Part 25/121 regulations; i.e., the required runway length in ft is

$$D_{FAR} = DL_T/.6 \ . \tag{VI.1.84}$$

On the other hand, if DL_T exceeds the limiting value, X_{LDGRQ}, the initial values of wing loading and runway length are relabelled W_{GSP} and X_{PAS}, and the wing loading is reduced by 30%;

$$W_{GS} = .7 \ W_{GS} \tag{VI.1.85}$$

After augmenting I_{WGS}, and setting the subroutine argument to

$$I_{LER} = 2 \tag{VI.1.86}$$

control is returned to the MAIN calling program. On the following iteration, $I_{WGS} = 2$ and the shorter runway length DL_T is found for the reduced wing loading, W_{GS}. The sensitivity is found as

$$SLOPE = (W_{GSP} - W_{GS})/(X_{PAX} - DL_T) \tag{VI.1.87}$$

and the counter I_{WGS} is augmented to 3, while $I_{LER} = 2$. This slope is positive unless an error has occurred, and the required wing loading is found at statement 75 as

$$W_{GS} = W_{GSP} + SLOPE * (X_{LDGRQ} - X_{PAS}) \tag{VI.1.88}$$

Finally, if the runway length requirement is met, no further iterations are needed, and at statement 100, the counters I_{LER} and I_{WGS} are set to 1, and the concluding computations ending with Eq. VI.1.83 are performed.

VI.1.5 Subroutine TAKOFF

The simulation of the take-off maneuver requires consideration of several smaller segments or "events." Each event is terminated when a particular performance variable reaches a given value. These events are summarized as follows, with key variables in parentheses:

All engines at take-off power:

(1) Begin ground roll (velocity)
(2) Rotation (velocity)
(3) Lift-off (lift = weight)
(4) Begin gear retraction (altitude)
(5) Altitude = 35 ft
(6) Altitude = 50 ft
(7) Begin flap retraction (altitude)
(8) End of take-off (altitude)

One engine out: accelerate-stop distance

(1) Begin ground roll (velocity)
(2) Engine failure (velocity)
(3) Remove power (time)
(4) Apply brakes (time)
(5) Stop (velocity)

The take-off computation is represented by simultaneous integration of the differential equations for velocity, flight path angle, distance and altitude, using a time interval of 0.2 sec.

The subroutine DERIV, to be discussed in the next subsection, performs the computation of the time derivatives, and at each time step, updated values for weight, thrust, air density, etc., are found before the integration subroutine INTS is called.

Subroutine TAKOFF begins with the definition of a linear interpolation function of the form,

$$YYY(X) = Y_1 + \frac{X-X_1}{X_2-X_1} (Y_2-Y_1) \qquad (VI.1.89)$$

which defines a value YYY between Y_1 and Y_2, when X falls between X_1 and X_2. This is followed by initialization of some 30 numerical factors, ending with the vector values T(1)-T(11), used by the integration routine.

The subroutine TPALT then returns atmospheric properties at the input altitude of the airport and runway, H_{APT}, and the local speed of sound and air density are given as

$$S_A = 49.1\ SQRT(T_{AMB}) \tag{VI.1.90}$$

$$RHO = P_o/[53.32\ T_{AMB}\ G] \tag{VI.1.91}$$

where

T_{AMB} = ambient airport temperature, deg Rankine

G = local gravity, ft per sec^2

P_o = local static pressure, lb per sq ft

The sea level stall speed is given in kts by

$$V_{STLKT} = SQRT\left[\frac{2W}{.0023769\ S_W\ C_{LMAX}}\right]\left(\frac{1}{1.689}\right) \tag{VI.1.92}$$

where the symbols have their traditional definitions.

$$C_{LTO} = C_{LMAX}/V^2_{RAT} \tag{VI.1.93}$$

is determined where V_{RAT} is the input takeoff speed margin or has a default value 1.10.

Speed increments are then used to define the engine failure speed, V1, and the rotation speed, VR,

$$V_1 = V_{STLKT} + DV_1 \tag{VI.1.94}$$

$$V_R = V_1 + DV_R \tag{VI.1.95}$$

where

DV_1 = increment of "decision" speed above stall, kts

DV_R = increment of rotation speed above V_1, kts

The loop between statements 180 and 186 finds the speed VEL for the best rate of climb, beginning with the stall velocity. This loop uses lift coefficient C_L and the Mach number EM, as inputs to subroutine DRAG, which returns the drag coefficient, C_D. Subroutine ENGINE returns the thrust available, T_A, and together these permit computation of the power required,

$$PWR_{RQR} = .5\ RHO\ S_W\ VEL^2\ C_D * VEL \tag{VI.1.96}$$

and of the power available,

$$PWR_{AVL} = T_A\ EN_P\ VEL \tag{VI.1.97}$$

The velocity is augmented by DEL_Z = 10. feet per second, and the loop is repeated, until the difference $PWR_{AVL} - PWR_{RQR}$ is less than it was on the previous iteration. The velocity increment is then reduced to 1 foot per second, and the velocity is eventually found at which the available power exceeds the required power by the maximum amount. This permits computation of the best rate of climb velocity in kts,

$$V_{END} = .592087\ VEL_N * SQRT(RHO/.0023769) \tag{VI.1.98}$$

which must satisfy the constraints,

$$V_{END} \leq 250.\ \text{kts} \tag{VI.1.99}$$

and

$$V_{END} \leq V_{CLMB} \qquad (I_{CLM} = 3) \tag{VI.1.100}$$

The next important portion of the subroutine deals with the loop between statements 1 and 300, which treats the ground roll, rotation, lift-off and climb phases of the take-off maneuver. The sequence begins with the ground roll equation of motion, for which control passes to statement 230. The Mach number and altitude are input to ENGINE, which returns the thrust and fuel flow. The horizontal acceleration is then expressed as

$$T(8) = (32.16/W)[T_A \cos(ALPHA_R) - D$$
$$- MU * (W - L - T_A \sin(ALPHA_R))] \tag{VI.1.101}$$

where all of the terms are time varying during this phase, except the angle of attack and the friction coefficient. Control passes back to statement 225, after setting $I_M = 1$, and the ground roll integration continues after a call to INTM. Until the velocity exceeds V_R, the integration proceeds with T(5) = flight path angle = 0.

When the rotation speed is exceeded, interpolation develops values for the time and value of the lift-off speed, TIMR and VW, using the function YYY defined in Eq. VI.1.89. This requires setting $I_R = 1$, and the lift and drag coefficients C_{LR} and C_{DR} are also found at this time by linear interpolation. The time after the rotation, when a more or less steady positive flight path angle has been acquired, is

$$TIM_{END} = TIM_R + DELT_{VR} \qquad \text{(VI.1.102)}$$

where $DELT_{VR}$ is input, or defaulted to 3.5 sec. During this brief interval, of course, the exact dynamic response of the aircraft would require consideration of the aerodynamic derivatives, the ground effect on these derivatives and the aircraft inertia in pitch. These details are not considered in this subroutine, which is intended to model the aircraft as a point mass during the take-off maneuver.

Following the computation of angle of attack, it is tested whether the vertical forces (lift and upward component of thrust) exceed the weight, with T(5) = 0. If so, the friction is set to MU = 0, and the take-off speed is calculated in kts as

$$V_{TOKTS} = T(4) * .592087 \qquad \text{(VI.1.103)}$$

The parameter values at lift-off are initial conditions for the fourth event (gear retraction) and this is modeled by a reduction in drag coefficient. This begins when the altitude equals H_{GR} = 20 ft, and the corresponding time is found by interpolation as T_G. The gear retracting operation requires an interval of DT_{GR} = 7 sec. and the gear drag increment falls linearly to zero over this time interval. The flag I_G is set to 1 at the beginning of retraction, and the lift varies linearly with time from C_{LR} to C_{LTO}, at beginning and end of gear retraction. This permits computation of angle of attack during gear retraction.

The next two events correspond to take-off altitude heights of 35 ft and 50 ft, and these occur at runway distances found by interpolation as S_{35} and S_{50} ft, when $I_{PAS} = 1$ or 2, respectively. When this flag has been set at 2, the altitude exceeds 50 ft and control passes over the 23 statements dealing with these two events.

The flaps are deflected at the angle $DFLP_{TO}$, and are to be retracted at the input altitude $H_{FLP} = 400$ ft. Retraction requires a time interval which is calculated as DT_{FLP}, which is defined as equivalent to a retraction rate of 3-1/3 deg per sec. Over this time interval lift and drag characteristics vary linealy between flaps down and flaps up conditions. The take-off maneuver terminates at the specified altitude H_{MAX}. If H_{MAX} is less than H_{FLP} it terminates when the flaps are retracted.

Engine failure is simulated by repeating the ground run portion of the subroutine, when the input flag is $N_{FAIL} = 0$. This failure is assumed to occur at the ground speed, V_1, where all parameters are calculated by interpolation. The flag I_M is set to 2, indicating engine failure has been initiated, unless the total number of engines is $EN_P = 1$. The continued takeoff computations after engine failure is handled by the same equations as the all engine takeoff except the thrust is reduced by the loss of the engine. The continued takeoff calculations are terminated at an altitude 50 feet above the airport. The accelerat-stop computations after engine failure begin at statement 400. Following initialization, the loop beginning at statement 920 integrates the equations of motion until the brakes are applied or until the velocity is less than 1 ft/sec. This is done by implementing the following logic:

(1) Engine thrust is reduced to zero 1.8 sec after failure is initiated.

(2) Brakes are applied 3.5 sec after failure.

A closed form solution of the one-dimensional motion is then used with the braking value of friction, MU_B, and this computation has been described in subroutine DLAND. The result is the incremental runway distance DL_G, which is the distance traveled after brakes are applied. The total runway distance is then given by

$$D_{STOP} = T(6) + DL_G \,, \tag{VI.1.104}$$

where T(6) is the runway distance preceding brake application.

VI.1.6 Subroutine DERIV

This 90-card subroutine is called by TAKOFF, as an argument of subroutine INTS, and it computes the time derivatives of the aircraft position and velocity. The two kinematic equations are integrated to give horizontal and vertical distance traveled, and the two acceleration equations in general yield the velocity and the flight path angle, while ignoring the rigid body pitch dynamics of the aircraft. The accelerations and velocities are limited by input values of the following quantities:

(1) Maximum load factor
(2) Maximum velocity
(3) Maximum lift coefficient (angle of attack)
(4) Maximum pitch angle
(5) Minimum (zero) load factor
(6) No de-acceleration along flight path

In simplest form, the forces acting on the center of mass of the aircraft are lift, drag, thrust and weight, and if the thrust is oriented by the angle $\alpha - i_W$, the acceleration components acting on the aircraft mass are

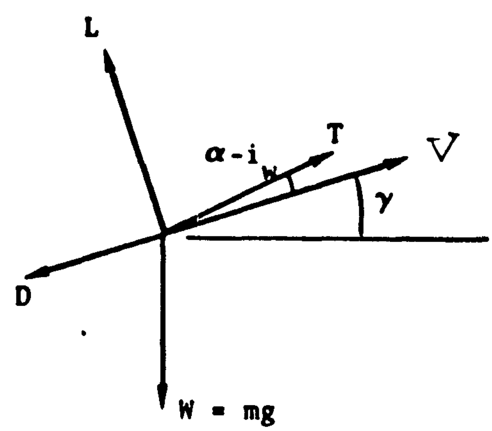


Figure VI.1.4 Applied Forces During Take-off

$$\dot{V} = (g/W)[T\cos(a - i_W) - D - W\sin\gamma - \mu F_N]$$
$$V\dot{\gamma} = (g/W)[T\sin(a - i_W) + L - W\cos\gamma] \qquad \text{(VI.1.105)}$$

where $\mu = 0$ when the aircraft is airborne, and where the normal force is otherwise:

$$F_N = W - L - T\sin(\alpha - i_W). \qquad \text{(VI.1.106)}$$

The subroutine accounts for weight loss due to fuel consumption, and the thrust varies according to the speed and altitude, as found by subroutine ENGINE. The angle of attack is equal to the wing incidence angle until the aircraft rotation is initiated during takeoff.

The subroutine begins with the modification of the angle of attack, ALPHA, during the airborne portion of the take-off maneuver. As measured by the load factor X_{LF}, the angle of attack in degrees is increased according to:

$$\text{ALPHA} = \begin{cases} \text{ALPHA} + .1 & X_{LF} > .85 \\ \text{ALPHA} + .3 & .8 < X_{LF} \leq .85 \\ \text{ALPHA} + .5 & X_{LF} \leq .8 \end{cases}$$

Whether or not the aircraft is airborne, subroutines CLIFT and DRAG are called in mode 2, returning values of C_L and C_D. The next 12 lines of code deal with the drag and lift increments due to landing gear and flaps. These events occur at the times T_G (begin landing gear retraction) and T_{FLP} (begin flap retraction), and quantities returned by DRAG are

C_{DGU} = drag coefficient, takeoff flaps down and gear up

C_{DFU} = drag coefficient, flaps up and gear down

The corrected coefficients of drag and lift are then

$$C_D = C_D - DC_{DG}\,G_{FAC}/DT_{GR} - DC_{DF}F_{FAC}/DT_{FLP}$$
$$C_L = C_L - DC_{LF}F_{FAC}/DT_{FLP} \qquad \text{(VI.1.107)}$$

where DC_{DG} and DC_{DF} are increments in drag due to gear and flaps and DC_{LF} is the lift increment due to flaps. The remaining factors represent the linear change of these increments over the time interval DT_{GR} or DT_{FLP}.

Maximum and limit values of lift coefficient are then found, to account for takeoff speed margins. If C_L exceeds the limit value C_{LLMT}, the subroutine CLIFT returns a lower angle of attack, and a second iteration from statement 1010 develops a lift coefficient less than C_{LLMT}.

The fuselage attitude angle may also limit the angle of attack, if $THETA_F$ exceeds THE_{MAX}. In this case, the angle of attack is defined in degrees as

$$ALPHA = THE_{MAX} + EYE_W - T(5) * 57.29578 \qquad (VI.1.108)$$

where T(5) is the flight path angle. Control again returns to the beginning of this loop, at statement 1000. Otherwise, the normal load factor increment due to lift and thrust is found as

$$X_{LF} = [C_L * QSW + T_A \sin(ALPHA_R)]/W \cos(T(5)) \qquad (VI.1.109)$$

and if this exceeds X_{LFMAX} when the aircraft is airborne, the angle of attack is reduced by .1 deg for the next iteration. This reduction is only .05 deg if the flaps have already begun to be retracted.

The equivalent airspeed is then calculated in kts as

$$EAS = T(4) * KTFPS * SQRT(RHO/RHO_{SL}) \qquad (VI.1.110)$$

where

KTFPS = .592087 and RHO_{SL} = .0023769.

This is required to be less than the velocity for maximum rate of climb, V_{END}. Otherwise, T(4) is reduced to this equivalent value and the velocity acceleration is set to T(8) = 0.

The tangential acceleration in the unconstrained case is then given in statement 4200 as

$$T(8) = (32.15/W)[T_A \cos(ALPHA_R) - D - MU * F - W*SIN(T(5))] \qquad (VI.1.111)$$

where drag and normal force during takeoff are

$$D = C_D * QSW$$

$$F = W - C_L * QSW - T_A * SIN(ALPHA_R) \qquad (VI.1.112)$$

If $I_{OUT} = 2$ or $T(8)$ is positive, the normal acceleration $T(9)$, and horizontal and vertical velocity components $T(10)$, $T(11)$, are computed by straightforward equations before returning. If $T(8)$ is negative, the drag exceeds the thrust, so the angle of attack is reduced and the counter I_{ALOP} is augmented. If 20 iterations do not develop a positive acceleration $T(8)$, a lower flap deflection is suggested by the print statement when $I_{ALOP} = 20$.

VI.1.7 Subroutine TAXI

This is a very brief subroutine with the purpose of computing the fuel burned while taxiing at ground idle. This is used to account for taxi time at the start and end of mission. This fuel weight is subtracted from the initial gross weight to provide the aircraft weight at take-off.

At each time step, the subroutine is called, and it calls TPALT for values of pressure and temperature, P_{AMB}, T_{AMB}, at the runway altitude, H. These are input to the subroutine ENGINE, which returns fuel flow per engine, FF, in lb per hr. This is multiplied by the number of engines, EN_P,

$$FF = EN_P * FF \tag{VI.1.113}$$

Next, the weight of the fuel burned during the time interval $DELT_T$ is

$$WF_T = FF * DELT_T \tag{VI.1.114}$$

and this reduces the aircraft weight,

$$W = W - WF_T \tag{VI.1.115}$$

The total fuel burned and the total taxi time are then augmented as

$$WF = WF + WF_T \tag{VI.1.116}$$

and

$$ST = ST + DELT_T \tag{VI.1.117}$$

If K_{WRITE} is nonzero, both input and output values of these time-varying parameters are printed before control is returned to subroutine PERFRM.

VI.1.8 Subroutine TURN

This is another very brief subroutine, which calculates certain parameters descriptive of the steady turn maneuver. The subroutine begins by determining the static pressure P_o and temperature T_o at the input altitude H_{TURN}, by calling subroutine TPALT. The speed of sound is then given in ft per sec by

$$A_{SON} = 49.1\ \mathrm{SQRT}(T_o) \qquad \text{(VI.1.118)}$$

and the dynamic pressure in lb per sq ft is

$$Q_{TURN} = .7 * P_o * EM_T^2 \qquad \text{(VI.1.119)}$$

where EM_T is the airplane's Mach number in the turn.

The load factor in the turn is the ratio of lift to weight, as shown in Figure VI.1.5. With the load factor XLF_{TRN} input as greater than one, this permits the required lift coefficient to be found as

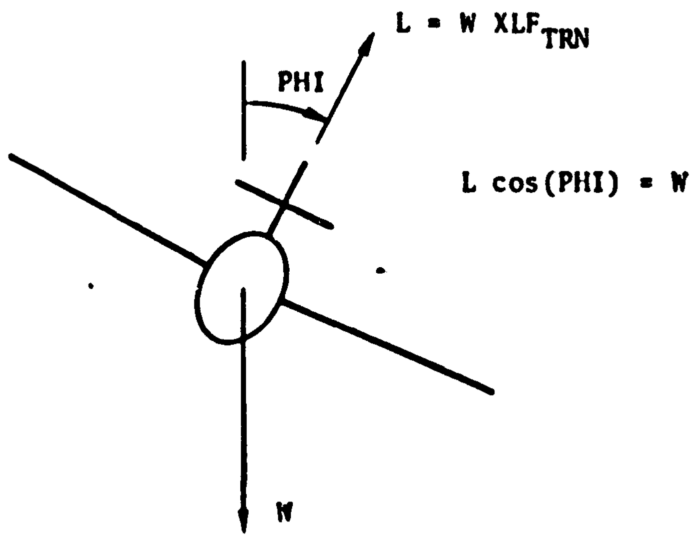


Figure VI.1.5 Aircraft in Coordinated Turn

$$C_{LRQT} = XLF_{TRN}\ WOS/Q_{TURN} \quad , \tag{VI.1.120}$$

where WOS is the wing loading in lb per sq ft, as calculated a few lines earlier. If this is less than the limit value C_{LTLMT}, the bank angle is found. Otherwise, the lift coefficient is C_{LTLMT} and the limit load factor is

$$XLF_{TRN} = C_{LRQT}\ Q_{TURN}/WOS \tag{VI.1.121}$$

In either case, the bank angle is given by

$$PHI = ARCOS(1/XLF_{TRN}) \tag{VI.1.122}$$

as suggested by Figure VI.1.5. The Mach number and lift coefficient together imply a drag coefficient C_{DTURN}, as provided by subroutine DRAG, and the minimum thrust required for maintaining velocity during the turn is equal to the drag,

$$TR_{TRN} = C_{DTURN}\ Q_{TURN}\ S_W \tag{VI.1.123}$$

The steady turn radius in ft is then found by equating horizontal forces to the centripetal acceleration, or

$$TRN_{RAD} = EM_T^2\ A_{SON}^2/[G * XLF_{TRN}\ SIN(PHI)] \tag{VI.1.124}$$

Finally, if I_{DC} is not equal to 99, subroutine ENGINE determines the thrust available at this flight condition, and if it is less than that required, an appropriate message is printed before returning to subroutine PERFRM.

If $I_{DC} = 99$ subroutine TURN is being used by subroutine ENGSZ to determine the thrust required in sizing the engines to the specified turn requirements.

VI.1.9 Subroutine XRANGE

This subroutine computes the cruise range using the Brequet range equation, for both propeller and turbojet aircraft. The equilibrium, in cruise, of lift and weight and of thrust and drag together lead to cruise angle of attack and cruise velocity, for a particular power setting. The equilibrium equations are nonlinear and must be solved iteratively, and subject to several constraints.

The subroutine begins by determining static pressure P_o and the speed of sound A_{SON} at the input cruise altitude, H, through a call to TPALT. Initial values for aircraft weight and Mach number are also specified, and the square root of the density ratio is defined as

$$SQRDR = \left[\frac{P_o/P_{REF}}{T_o/T_{REF}}\right]^{1/2} \qquad \text{(VI.1.125)}$$

where P_{REF} = 2116.22 lb per sq ft and T_{REF} = 518.67 deg R. This scale factor will be needed at several points later in the program.

The subroutine ASPEED is called, the engine operating condition for normal cruise power having been specified with the indicator K_{ENG}. The Mach number returned as EM_{NP} is the "normal power" value, and it corresponds to the velocity in kts,

$$VEL_K = 29.07\ EM_{NP}\ SQRT(T_o) \qquad \text{(VI.1.126)}$$

This also serves as an initial value for the loop between statements 20 and 30, which leads to the speed for the best "specific range," in nautical miles per lb of fuel;

$$SPEC_R = VEL_K/(FF\ EN_P) \qquad \text{(VI.1.127)}$$

where FF is the fuel flow returned by subroutine ENGINE, and EN_P is the number of engines. This iterative loop varies the velocity, first in steps of 10 kts and, once the peak is surpassed, in 2 kt steps. The resulting "best range" Mach number is denoted EM_{BRG}.

The next phase of the subroutine deals with cruise performance at a maximum of three fuel capacities and at three Mach numbers. The fuel weights are input through COMMON/WGTRGE/ as

W_{FAMAX} = maximum available fuel weight (minimum payload)

W_{FAMIN} = minimum available fuel weight (maximum payload)

W_{FADES} = design available fuel weight

The Mach numbers have been specified as

EM_X = cruise Mach number, input as CR_{MACH}, or defaulted to EM_{CRU}

EM_{NP} = normal power Mach number

EM_{BRG} = maximum specific range Mach number

The parameter SPEED used by subroutine ACCEL is equated to one of these Mach numbers, depending on the value of the input flag I_{CRUS}, and a number of other definitions are made at this point.

The subroutine ACCEL returns the time ST, the range R and the fuel weight DEL_{WGT} required to accelerate from the "climb" Mach number EM_{CLB} to the "cruise" Mach number SPEED. The integers I_{DC} and I_{SEGX} are defined as 1 and 9 for later use in the subroutine ENGINE, and weight of available fuel is defined as W_{FAVLB}, which depends on the mission weight and on W_{FAMAX}, W_{FAMIN} and W_{FADES}. The "mission weight," WT_{MISN}, if input is a take-off weight which is less than the gross weight but greater than the empty weight of the aircraft.

After setting to zero the parameters OUT(I,J) and RES_F(I), the index K is initialized at zero, before the loop 120 is entered. This loop is passed through for each of the three Mach numbers, EM_X (K=1), EM_{NP} (K=2), and EM_{BRG} (K=3). Initially, the Mach numbers are checked to determine if they are within the operating envelope of the aircraft. The velocity V_{MAX} is expressed in kts, and if the operating velocity V_X exceeds V_{MAX}, then V_X and EM are re-defined to correspond to V_{MAX}, which must satisfy the maximum operating airspeed and Mach number constraints.

For all but fixed-pitch propeller systems (i.e., for KODE $\neq$ 8), subroutines CTAER and ENGINE are called to determine the drag DRG and the required thrust of the engines, FN_{RQ}, at the flight condition of interest. These parameters are defined at statement 145 at the altitude, H, and aircraft weight, W.

A longer, iterative process is required for fixed pitch propeller systems, or KODE = 8. In this case the aircraft speed is determined by the engine power setting, to find a Mach number SMN at which thrust and drag are in balance, i.e., at which

$$ERROR = (FN_{RQ} - DRG)/DRG < .002 \quad , \qquad (VI.1.128)$$

where FN_{RQ} is the thrust as output by ENGINE and DRG is the drag as output by CTAER.

On convergence of this loop, if shaft horsepower SHP is less than available horsepower (HP_{AVLB}), control passes to statement 146. Otherwise the engine RPM is reduced by 5% and the loop is reinitiated. This can be repeated until convergence occurs. The equations between statements 146 and 145 are concerned only with the time spent in accelerating to the cruise Mach number, and with the fuel consumption during this interval. When KODE = 8 and I_1 = 0, as on the initial arrival at 146, subroutine ACCEL returns the aircraft weight and range covered during the constant altitude acceleration to the Mach number SPEED. This is necessary regardless of the iterations needed in the initial loop between statements 142 and 146.

The Breguet range constant at the start of cruise is then found in units of nautical miles, as

$$BREG_1 = V_X \; W_1 \; 3600./(FF_1 * 6076.1) \qquad (VI.1.129)$$

where aircraft velocity in ft per sec is the product of Mach number and the speed of sound,

$$V_X = S_{MN} A_{SON} \quad . \tag{VI.1.130}$$

aircraft weight is W_1 lb and fuel flow is,

$$FF_1 = FF * EN_P \tag{VI.1.131}$$

where FF is fuel flow per engine in lb per hr, and the subscript 1 indicates parametric values at the beginning of the cruise phase.

The fuel weight available for cruise is the difference

$$W_{FCRU} = W_{FAVLB} - W_{F1} - W_{FRES} \tag{VI.1.132}$$

where

W_{F1} = fuel weight used in taxi, takeoff, and climb, lb

W_{FRES} = reserve fuel weight, lb

Aircraft and fuel weights at the end of cruise are W_2 and W_{F2}, respectively, and these are found by the same iterative procedure as described above for the beginning of cruise. The subscript 2 indicates end of cruise and is used to specify fuel flow FF_2, Mach number EM_2, velocity,

$$V_{KTS2} = .5921\ EM_2\ A_{SON} \tag{VI.1.133}$$

and equivalent airspeed

$$V_{KES2} = V_{KTS2}\ SQRDR, \tag{VI.1.134}$$

where SQRDR is defined in Eq. VI.1.125.

This permits the computation of the end of cruise Breguet factor $BREG_2$ by an equation similar to that of Eq. VI.1.129, following which the "average" factor is

$$BREG_A = (BREG_1 + BREG_2)/2 \quad . \tag{VI.1.135}$$

The cruise range is then calculated as

$$R_{CRNM} = BREG_A \; ALOG[1./(1-W_{FCRU}/W_1)] \qquad (VI.1.136)$$

and other parameters at the end of cruise are given as range R_2, angle of attack $ALPH_2$, attitude angle $THET_2$, lift coefficient C_{L2}, lift to drag ratio $ZLOD_2$, and time ST_2.

The array OUT(6,16) is then used for storing computed quantities at start and end of cruise for the three Mach numbers, as indicated by the counter K. A large number of WRITE statements follow, which are self-explanatory, and these are followed by a redefinition of available fuel; i.e., the first pass through the loop beginning at line 100 is with the available fuel weight given by

$$W_{FAVLB} = W_{FAMIN} - DEL_{WGT} \qquad (VI.1.137)$$

while the subsequent passes define this weight as W_{FAMAX} and W_{FADES}, respectively.

The closing computations deal with the specification of the output parameters, at the design payload fuel weight, computed on the final pass through XRANGE, which are used for checking range or endurance requirements in subroutine RGBAL or operating cost in subroutine GACOST.

The input parameter I_{CRUS} specifies the speed condition selected for the design mission performance and the index

$$KK = 2(I_{CRUS} + 1) - 1,$$

and the following parameters are defined:

R_{CLBNM} = range after climbing, to start of cruise, nm

R_{CRNM} = cruise range, nm

T_{CRHR} = time spent in cruise, hr

T_{CLHR} = time spent in climbing to start of cruise, hr

EM_{SAVE} = Mach number at start of cruise

H_{SAVE} = altitude of cruise, ft

W_{FCRU} = fuel consumed in cruise, lb

R = total range, after cruise, nm

ST = total time from take-off, after cruise, hr

W_F = total fuel used, after cruise, lb

VI.1.10 Subroutine ASPEED

This subroutine is used to calculate the maximum Mach number for equilibrium flight at given values of weight, altitude and engine power setting.

The subroutine is called by subroutines GACOST, XRANGE and PNOYS, and it calls on subroutines CTAER, ENGINE, ITRMHW and TPALT. The iterative method begins with an estimate of Mach number, to which corresponds a thrust value based on the engine data, at a specified engine power condition, KENGX. Subroutine CTAER then determines the cruise angle of attack at which lift and weight are equal, and returns a value for drag, which is compared with the thrust before modifying the Mach number.

The call arguments of ASPEED are the output Mach number XM_N, altitude (ft) H_{IN}, weight (lb) W_{IN}, and engine power indicator K_{ENGX}. After specification of certain parameters, a call to TPALT returns the static pressure P_o and temperature T_o at the altitude H_{IN}. The pressure is then expressed in lb per sq ft and the density follows by the equation of state as

$$RHO = P_o/[53.32(32.2)\ T_o] \qquad \text{(VI.1.138)}$$

as expressed in slugs per cu ft. The Mach number to start the iteration is then computed as the maximum value, corresponding to the maximum operating airspeed V_{MO};

$$X_{MN} = V_{MO}/(29.0721\sqrt{RHO * T_o/.0023769}) \qquad \text{(VI.1.139)}$$

The denominator of this expression is the speed of sound in kts, for any values of air density and temperature. Since $I_{TOM} = 0$, the first loop is entered in order to find the Mach number for minimum power required. The drag is initially set equal to the weight, and the Mach number is reduced from the value X_{MN} in steps of 0.1, until the value of drag returned by CTAER is greater than the value on the previous iteration. After this overshoot, the Mach number is increased by .01 and on the next iteration,

$$DRG > DRG_P,$$

so control is transferred to statement 8, where X_{MIN} is the Mach number at which drag is a minimum.

The longer loop, between statements 4 and 10, begins by calling CTAER at the Mach number X_{MN}. The cruise drag returned is DRG, and set equal to the required thrust

$$F_{NRQ} = DRG \ , \qquad \text{(VI.1.140)}$$

before subroutine ENGINE is called, at the same values of Mach and altitude. The thrust per engine is returned, in terms of which the total thrust available is

$$F_{NAV} = EN_P * TH_{PENG} \qquad \text{(VI.1.141)}$$

where EN_P is the number of engines, which is input in the common statement, UNIV.

Now, if thrust available, F_{NAV} exceeds the thrust required, F_{NRQ} on the first iteration (while $J_C = 0$) it means that a velocity greater than the "maximum" value V_{MO} is possible, and control passes to the write statement 20 before returning. Otherwise, the dimensionless error is

$$ERR_M = (F_{NAV} - F_{NRQ})/F_{NRQ} \qquad \text{(VI.1.142)}$$

and this difference, together with its previous value, permits use of a linear iteration procedure in subroutine ITRMHW, leading to a Mach number for which thrust available and thrust required are more nearly equal. If the error is less than .002, the computation is complete, and the true airspeed in kts is found at statement 10,

$$V_{KTAS} = .5921\ X_{MN} * 49.1 * \sqrt{T_o} \qquad \text{(VI.1,143)}$$

following which the subroutine returns control to the calling program.

VI.1.11 Subroutine RGBAL

This subroutine is called only by MAIN, and its purpose is to compare the total range or endurance computed in XRANGE with an input required mission range or endurance, R_{CRRQ}. If $R_{CRRQ} < 24$, it assumes an endurance requirement is being specified.

$$RAG_I(1) = T_{CRHR} + T_{CLHR} \quad (\text{If } R_{CRRQ} < 24.)$$

If the error exceeds one percent, the aircraft gross weight is adjusted and the aircraft entirely resized before reflying the mission to generate an updated figure for range or endurance. This subroutine controls the program flow during this resizing iteration to determine the aircraft size required for the range or endurance requirement, R_{CRRQ}.

A flow diagram for subroutine RGBAL is shown in Figure VI.1.t, which outlines the calculations corresponding to the following logic. When RGBAL is initially called, the range or endurance corresponding to the initial gross weight is known, as is the range or endurance requirement. Assuming the difference exceeds one percent, the gross weight is changed by a scale factor, FAC_{W1}. There are two options for resizing the aircraft, one holds wing loading fixed, the other fixes wing area, and is followed by reflying the mission and calculating the new range or endurance. This permits linear estimation of the gross weight needed to meet the required range or endurance condition. Recomputation leads to a new range or endurance and to a new estimate of the gross weight. A third iteration yields either less than 1 percent error or the printing of an error "failure" message before returning to MAIN.

The subroutine begins with the initialization of the gross weight and range or endurance parameters,

$$W_{GRI}(1) = W_G \tag{VI.1.144}$$

$$RAG_I(1) = R_{CRNM} + R_{CLBNM} \tag{VI.1.145}$$

where

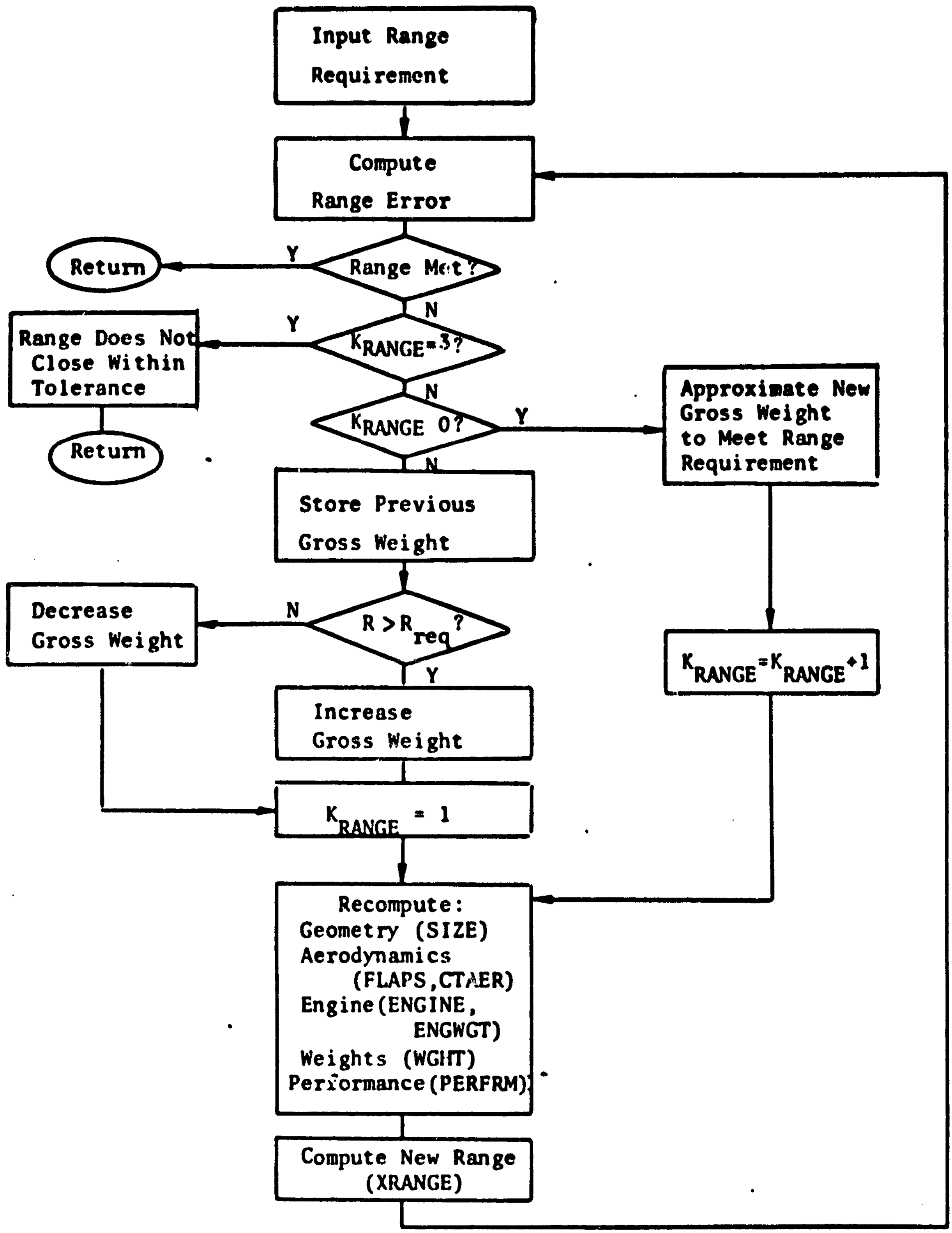


Figure VI.1.6 Subroutine RGBAL

W_G = input gross weight, lb

R_{CRNM} = cruise range, nm

R_{CLBNM} = climb range, nm

T_{CRHX} = time in cruise/hrs

T_{CLHR} = time to climb, hrs

where the climb and cruise times on the right are input through COMMON/RANGX/, as computed in subroutine XRANGE. The scale factor FAC_{W1} is input through COMMON/INRBAL/, and the range error is expressed in dimensionless form as

$$RG_{ERR} = (R_{CRNM} + R_{CLBNM} - R_{CRRQ})/R_{CRRQ} \tag{VI.1.146}$$

or as

$$RG_{ERR} = (T_{CLHR} + T_{CRHR} - R_{CRRQ})/R_{CRRQ} \tag{VI.1.147}$$

depending again upon whether or not R_{CRRQ} exceeds 24 in magnitude. If this figure is below .01 in magnitude, the subroutine returns, after printing the significant results. Otherwise, the flow depends on the iteration number K_{RANGE}, which is initially 0.

When $K_{RANGE} = 0$, the gross weight is increased or decreased by the scale factor, according to

$$W_G = \begin{cases} FAC_{W1} \cdot W_G \ , & RG_{ERR} > 0 \\ W_G/FAC_{W1} \ , & RG_{ERR} < 0 \end{cases} \tag{VI.1.148}$$

and subsequently $K_{RANGE} = 1$ to indicate the first iteration is underway. For subsequent iterations, when $K_{RANGE} = 1$ or 2, gradient information is provided by

$$SLOPE = ABS[(RG_{ERR} - RG_{ERR1})/(W_G - W_{G1})] \tag{VI.1.149}$$

and the updated gross weight is found as

$$W_G = W_G - RG_{ERR}/SLOPE \qquad \text{(VI.1.150)}$$

after previously defining $W_{G1} = W_G$.

The wing loading is redefined as

$$WGS = W_G/S_W \qquad \text{(VI.1.151)}$$

if $I_{SWING} = 1$, and after some lines of output printing, control passes to statement 21. Subroutine SIZE is called twice, with NPC = 3 (to initialize it) and NPC = 2 (to develop new values for wing and tail geometric parameters).

This is followed by three calls to subroutine FLAPS, corresponding to flap deflections in cruise, take-off and landing configurations. Lift and drag coefficients for the aircraft are returned for these flight conditions, and Mach number and altitude are defined as input to subroutine CTAER, which returns angle of attack, lift and drag in cruise.

Subsequent calls to subroutine ENGSZ return values of engine sizing data, depending on flight condition and aircraft geometry. These are followed by calls to subroutines ENGWGT, WGHT, and PERFRM, which finally develops values for total range or endurance in the climb and cruise configuration.

It is noted that at most three iterations are possible through the subroutine. If convergence occurs before the third iteration is complete, the variation of gross weight and corresponding total range or endurance is printed, together with the required cruise range or endurance, R_{CRRQ}.

Before returning, the subroutine calls CTAER, OUTPUT and AEROUT, to provide output of the final values of geometry, weight, and cruise, takeoff, and landing aerodynamics.

GASP - GENERAL AVIATION SYNTHESIS PROGRAM

VOLUME VI - PERFORMANCE

PART 2 - USER'S MANUAL

JANUARY 1978

Prepared for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Ames Research Center
Moffett Field, California

Under

CONTRACT NAS 2-9352

AEROPHYSICS RESEARCH CORPORATION

VI.2 PERFORMANCE MODEL USER'S MANUAL

As suggested by the descriptive material in Part I of Volume VI, the study of aircraft performance involves many subroutines and many input/output parameters. The performance subroutines given in Table VI.1.1 are tabulated alphabetically in this section, and it will be noted that a relatively small number of input parameters (10 to 20) are required by each.

VI.2.1 Input - Subroutine ACCEL

VARIABLE	DESCRIPTION
EM	= initial Mach number
ENP	= number of engines
H	= altitude, ft
SPEED	= desired velocity, kts, or (less than 5) desired Mach number
ST	= initial time, hr
SW	= wing area, sq ft
W	= initial gross weight, lb

VI.2.2 Output - Subroutine ACCEL

VARIABLE	DESCRIPTION
EM	= Mach number
R	= range, nm
ST	= time, hr
W	= gross weight, lb
WF	= fuel consumed, lb

VI.2.3 Input - Subroutine ASPEED

VARIABLE	DESCRIPTION
EM	= Mach number
EMMO	= maximum operating Mach number
ENP	= number of engines
H	= altitude, ft
HIN	= input altitude, ft
KENGX	= indicates engine power setting
KWRITE	= { 0, do not write output 1, write output
VMO	= maximum operating velocity, KEAS
WG	= gross take-off weight, lb
WIN	= input weight, lb

VI.2.4 Output - Subroutine ASPEED

VARIABLE	DESCRIPTION
DRG	= drag, lb
XMN	= Mach number for cruise equilibrium

VI.2.5 Input - Subroutine CLIMB

VARIABLE	DESCRIPTION
ALPHLO	= angle of attack for zero lift, deg
AR	= aspect ratio
DELH	= altitude increment in computing climb performance, ft
DLMC4	= quarter chord wing sweep, deg
EM	= Mach number
EM_{CRU}	= cruise Mach number
EM_{MO}	= maximum operating Mach number
EN_P	= number of engines
EYE_W	= wing incidence angle, deg.
H	= altitude, ft
H_{ENDCL}	= terminal altitude for climb maneuver, ft
I_{CLM}	= 1, maximum ROC; 2, maximum operating speed; 3, specified EAS
K_{WRITE}	= write option
R	= range, nm
SW	= reference wing area, sq ft
V_{CLMB}	= climb velocity for maximum rate of climb, ft per sec
V_{LIMX}	= limiting operating climb velocity
V_{MO}	= maximum operating airspeed, kts
W	= initial weight, lb

VI.2.6 Output - Subroutine CLIMB

VARIABLE	DESCRIPTION
EM	= Mach number
H	= altitude, ft
R	= range, nm
ST	= time, hr
TA	= engine thrust, lb
W	= weight, lb
WF	= fuel consumed, lb

VI.2.7 Input - Subroutine DERIV

VARIABLE	DESCRIPTION
ALPHLO	= zero angle of attack
AR	= aspect ratio
CDFU	= drag coefficient, take-off flaps, gear up
CLALPH	= lift curve slope, per deg
CLMAX	= maximum lift coefficient
DLMC4	= quarter chord sweep, deg
DTFLP	= time increment to retract flaps, sec
DTGR	= time increment to retract gear, sec
EM	= Mach number
EYEW	= wing incidence, deg
HOB	= altitude above ground in wing spans
IOUT	= 0, all engines 1, one engine out 2, accelerate-stop on runway
MU	= normal force friction coefficient
QSW	= product of wing area and dynamic pressure, lb
RHO	= air density, slug per cu ft
TA	= thrust, lb
TFLP	= time of beginning flap retraction, sec
TG	= time of beginning landing gear retraction, sec
TGU	= time gear retraction completed, sec
THEMAX	= maximum fuselage pitch angle, deg
VEND	= velocity for maximum climb rate
W	= weight, lb
XLFMAX	= maximum load factor during take-off

VI.2.8 Output - Subroutine DERIV

VARIABLE	DESCRIPTION
ALPHA	= angle of attack, deg
CD	= drag coefficient during take-off
CL	= lift coefficient during take-off
DCDF	= drag coefficient increment due to flaps
DCDG	= drag coefficient increment due to gear
DCLF	= lift coefficient increment due to flaps
HMAX	= altitude at end of take-off, ft
IACCL	= flag indicating limits on load factor or speed are reached
THETAF	= fuselage attitude angle during take-off, deg
XLF	= load factor

VI.2.9 Input - Subroutine DLAND

VARIABLE	DESCRIPTION
ALPHLO	= zero-lift angle of attack, deg
ALTLND	= altitude of runway above sea level, ft
AR	= wing aspect ratio
B	= wing span, ft
CLALPH	= lift curve slope, per rad
DLMC4	= quarter chord sweep angle, deg
ENP	= number of engines
EYEW	= wing incidence angle, deg
HAPP	= approach altitude, ft
HTG	= wing height above ground, ft
ILER	= { 1, runway length constraint satisfied 2, iterating on runway length constraint
IWGS	= { 1, first iteration in runway length computation 2, second iteration in runway length computation 3, last iteration in runway length computation
IWLD	= { 0, landing weight is gross weight 1, landing weight is weight at end of previous segment 2, landing weight is WL_{PCT} times gross weight
MUB	= breaking coefficient of friction
RSMX	= maximum rate of sink, ft per min
SINKTD	= sink rate at touchdown, ft per sec
SW	= wing area, sq ft
TDELAY	= pilot's brake reaction time, sec

(CONTINUED)

VARIABLE	DESCRIPTION
TDELLD	= incremental temperature above standard temperature, °F
TIDLE	= idle thrust, lb
TROTID	= ratio of reverse thrust to idle thrust
VRATT	= ratio of approach speed to stall speed
W	= aircraft weight, lb
WG	= gross weight, lb
WGS	= wing loading, lb per sq ft.
WLPCT	= landing weight divided by gross weight
XLDGRQ	= landing length requirement, ft
XLFMX	= maximum load factor during flare maneuver

VI.2.10 Output - Subroutine DLAND

VARIABLE	DESCRIPTION
TIDLE	= total idle thrust, lb
WGS	= wing loading, lb per sq ft
DLT	= total landing distance from obstacle height, ft
DLG	= distance from obstacle to start of flare, ft
DLTR	= flare distance, ft
DDELAY	= distance from brake application, ft
DLGL	= braked ground roll distance, ft
DFAR	= FAR 25/121 factored landing distance, ft
RS	= rate of sink during glide portion, ft per min
VAPEAS	= approach speed, KEAS
VSTEAS	= landing stall speed, KEAS
VTD	= touchdown speed, KEAS
DOG	= average decceleration during landing ground roll, g's
THETA	= approach flight path angle, deg
CLMX	= landing maximum lift coefficient

VI.2.11 Input - Subroutine PERFRM

VARIABLE	DESCRIPTION
CRALT	= cruise altitude, ft
EMTURN	= Mach number in turn
HPORT	= altitude of airport, ft
HTURN	= altitude of turn, ft
IDC	= control parameter
IFLY	= input parameter to control mission computation
ISEG	= 1 to 8, mission segment indicator
MUB	= braking coefficient of friction
TDELTO	= temperature increment above standard for mission
TDELTX	= temperature increment above standard for engine sizing
VLIMX	= maximum take-off velocity, ft per sec
WG	= aircraft gross weight, lb
WTMISN	= start of mission weight, lb

VI.2.12 Input - Subroutine RGBAL

VARIABLE	DESCRIPTION
DFLPLD	= landing flap deflection, deg
DFLPTO	= take-off flap deflection, deg
EMCRU	= cruise Mach number
FACW1	= weight modification factor
HNCRU	= cruise altitude, ft
IEGWGT	= { 0, propulsion weight and nacelle geomtry fixed input 1, propulsion weight and nacelle geomtry computed in ENGWGT
ISWING	= { 0, hold wing loading fixed 1, hold wing area fixed
NTYE	= input flag 1 to 13, describing engine type
NTYP	= input flag 1 to 16, describing propeller type
RCLBNM	= climb range, nautical miles
RCRNM	= cruise range, nautical miles
RCRRQ	= required total mission range, nautical miles or endurance, hrs
SW	= wing area, sq ft
TCLHR	= climb time, hrs
TCRHR	= cruise time, hrs
WG	= initial gross weight, lb

VI.2.13 Output - Subroutine RGBAL

VARIABLE	DESCRIPTION
WG	= final gross weight required to meet range condition, lb
WGS	= final wing loading, lb per sq ft

VI.2.14 Input - Subroutine TAKOFF

VARIABLE	DESCRIPTION
ALPHLO	= zero lift angle of attack, deg
AR	= aspect ratio
B	= wing span, ft
CLALPH	= lift curve slope, per rad
CLMAX	= maximum lift coefficient at stall
DELTVR	= time required to rotate fuselage in take-off, sec.
DLMC4	= quarter chord sweep angle, deg
DVR	= increment of rotation speed above decision speed, kts
DV1	= increment of decision speed above stall speed, kts
EYEW	= wing incidence angle, deg
GRCD	= drag coefficient increment of landing gear
HFLP	= altitude at which flaps are retracted, ft
HTMAX	= maximum take-off altitude, ft
MUB	= Braking coefficient of friction
NFAIL	= 0, one engine is out; 1, all engines operative
UM	= rolling coefficient of friction
VCLMB	= velocity for best rate of climb, kts
VRAT	= margin of take-off speed above stall speed (greater than one)
W	= current aircraft weight, lb
WF	= current fuel weight, lb
WGS	= wing loading, lb per sq ft
XLF	= load factor during takeoff
XTORQ	= required field length, ft

VI.2.15 **Output - Subroutine TAKOFF**

VARIABLE	DESCRIPTION
ALPHA	= angle of attack, deg
CD	= drag coefficient
CL	= lift coefficient
DTFLP	= time required to retract flaps, sec
DTGR	= time required to retract landing gear, sec.
H	= altitude during take-off, ft
HMAX	= cutoff maximum altitude, ft
QSW	= product of dynamic pressure and wing area, lb
RELI	= Reynolds number per ft of length
RHO	= air density, slugs per cu ft
ST	= time end of take-off
TA	= total engine thrust, lb
TFLP	= time at start of flap retraction, sec
TFLPU	= time at end of flap retraction, sec
TG	= time of start of gear retraction, sec
TGU	= time at end of grear retraction, sec
VEND	= best rate of climb speed, kts
VLIMX	= constrained rate of climb speed, kts
W	= aircraft weight, lb
WF	= weight of fuel consumed, lb
XTO	= all-engine runway distance needed to clear 35 ft altitude, ft
DSTOP	= accelerate-stop distance, ft
S35	= engine-out distance to 35 ft altitude, ft
S35ALL	= all-engine distance to 35 ft altitude, ft
SFAR25	= FAR 25 factored takeoff distance, ft
S50	= all-engine distance to 50 ft altitude, ft

VI.2.16 Input - Subroutine TAXI

VARIABLE	DESCRIPTION
DELTT	= time interval, hr
ENP	= number of engines
H	= altitude, ft
KWRITE	= { 0, do not write output 1, write output
ST	= initial time, hr
TA	= engine thrust, lb
W	= initial aircraft weight, lb
WF	= initial burned fuel weight, lb

VI.2.17 Output - Subroutine TAXI

VARIABLE	DESCRIPTION
ST	= final time, hr
W	= final aircraft weight, lb
WF	= final burned fuel weight, lb

VI.2.18 Input - Subroutine TURN

VARIABLE	DESCRIPTION
CLTLMT	= maximum lift coefficient in turn
EM	= Mach number in turn
EN_P	= number of engines
HTURN	= altitude of steady turn, ft
IDC	= performance or engine sizing control parameter
SW	= wing area, sq ft
W	= aircraft weight, lb
XLFTRN	= desired load factor in turn

VI.2.19 Output - Subroutine TURN

VARIABLE	DESCRIPTION
TRTRN	thrust required in steady turn, lb
XLFTRN	load factor in turn

VI.2.20 Input - Subroutine XRANGE

VARIABLE	DESCRIPTION
CRMACH	= cruise Mach number
DMACH	= wing compressibility Mach number
EM	= Mach number at end of climb
EMMO	= maximum operating Mach number
ENP	= number of engines
EYEW	= wing incidence, deg
FRESF	= reserve fuel fraction (1.0 equals 45 min) (greater 10. equals lbs fuel)
H	= altitude, ft
HNCRU	= design cruise altitude, ft
HPAVLB	= horsepower available
KODE	= engine operating condition indicator
SHP	= shaft horsepower
ST	= time to begin cruise, sec
VMO	= maximum operating air speed, kts
W	= weight, lb
WF	= fuel weight, lb
WFAMAX	= maximum weight of available fuel, lb
WFAMIN	= minimum weight of available fuel, lb
WG	= gross weight, lb
WTMISN	= weight at start of mission, lb
ZLQD	= lift to drag ratio

VI.2.21 Output - Subroutine XRANGE

VARIABLE	DESCRIPTION
EM	= Mach number
EMCOST	= Mach number used in cost calculations
EMSAVE	= Mach number used in RGBAL calculations
FPMN	= Mach number used in fixed pitch propeller iteration
H	= altitude, ft
OUT(6,16)	flight conditions at start and end of cruise, for three Mach numbers
PCRPM	= percent of rated rpm
PO	= static pressure, lb per sq ft
R	= total range, nm
RCLBNM	= climb range, nm
RCRNM	= cruise range, nm
ST	= total time, hr
TCLHR	= climb time, hr
TCRHR	= cruise time, hr
W	= weight, lb
WF	= fuel weight, lb
WFCRU	= cruise fuel weight, lb
WFRES	= reserve fuel weight, lb

GASP - GENERAL AVIATION SYNTHESIS PROGRAM

VOLUME VI - PERFORMANCE

PART 3 - PROGRAMMER'S MANUAL

JANUARY 1978

Prepared for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Ames Research Center
Moffett Field, California

Under

CONTRACT NAS 2-9352

AEROPHYSICS RESEARCH CORPORATION

VI.3 PERFORMANCE MODEL PROGRAMMER'S MANUAL

The flow charts in the following pages follow the order shown in Table VI.1.1, and it will be noted that the nominal mission sequence of Figure VI.1.1 makes iterative loops largely unnecessary. That is, each mission segment requires calling a different subroutine concerned with the segment, and the output of each segment acts as the input to the next. The individual segments are also straightforward, although the flight condition must satisfy various constraints, as tested by a sequence of IF-tests.

VI.3.1 Subroutine PERFRM, Mission Computation Control Routine

Subroutine PERFRM controls the mission computations in GASP. A series of mission segments covering:

1. Taxi
2. Take-off
3. Acceleration
4. Climb
5. Range
6. Landing
7. Turns

are controlled by PERFRM. The routine PERFRM is called from the main program.

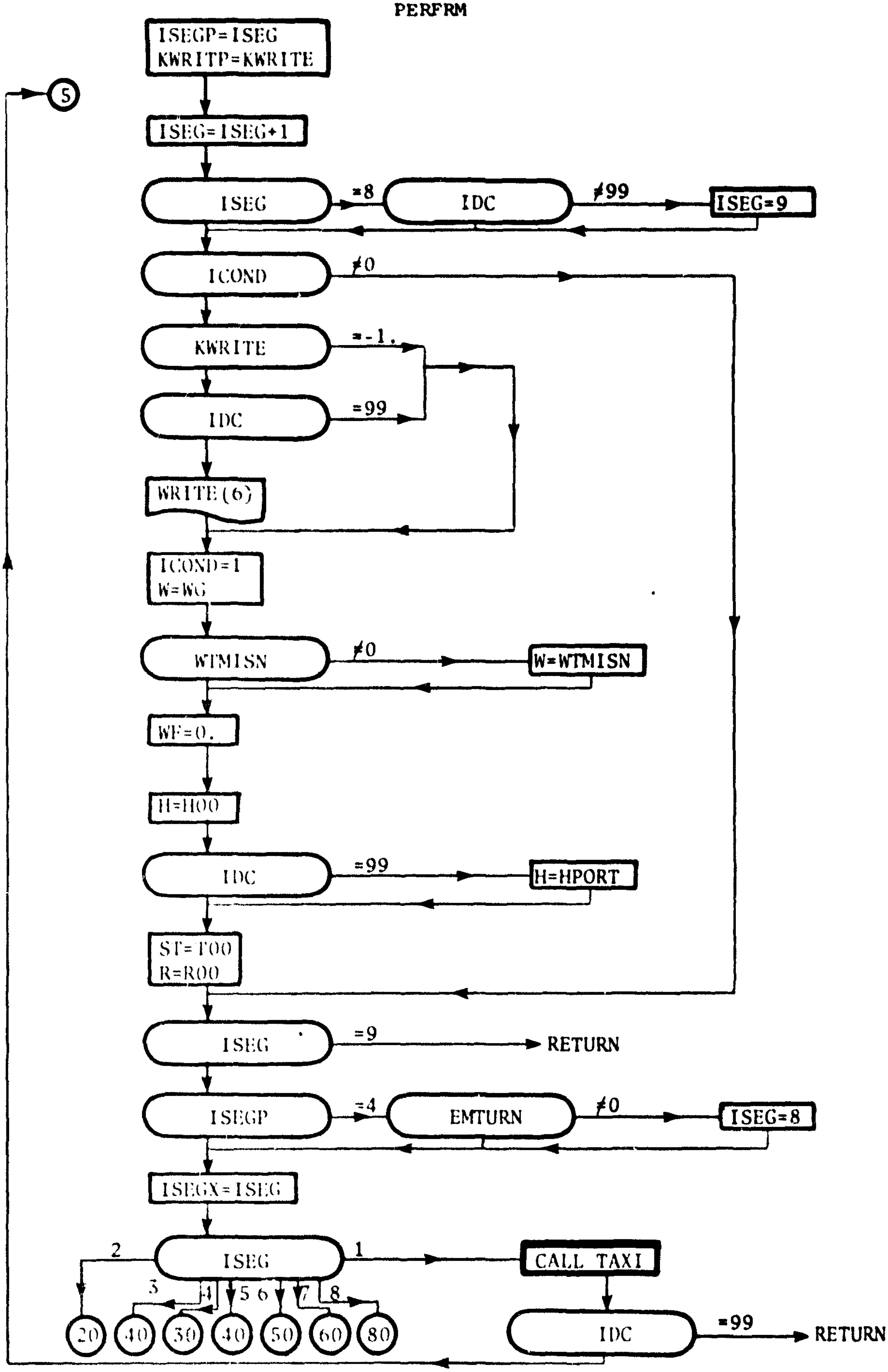
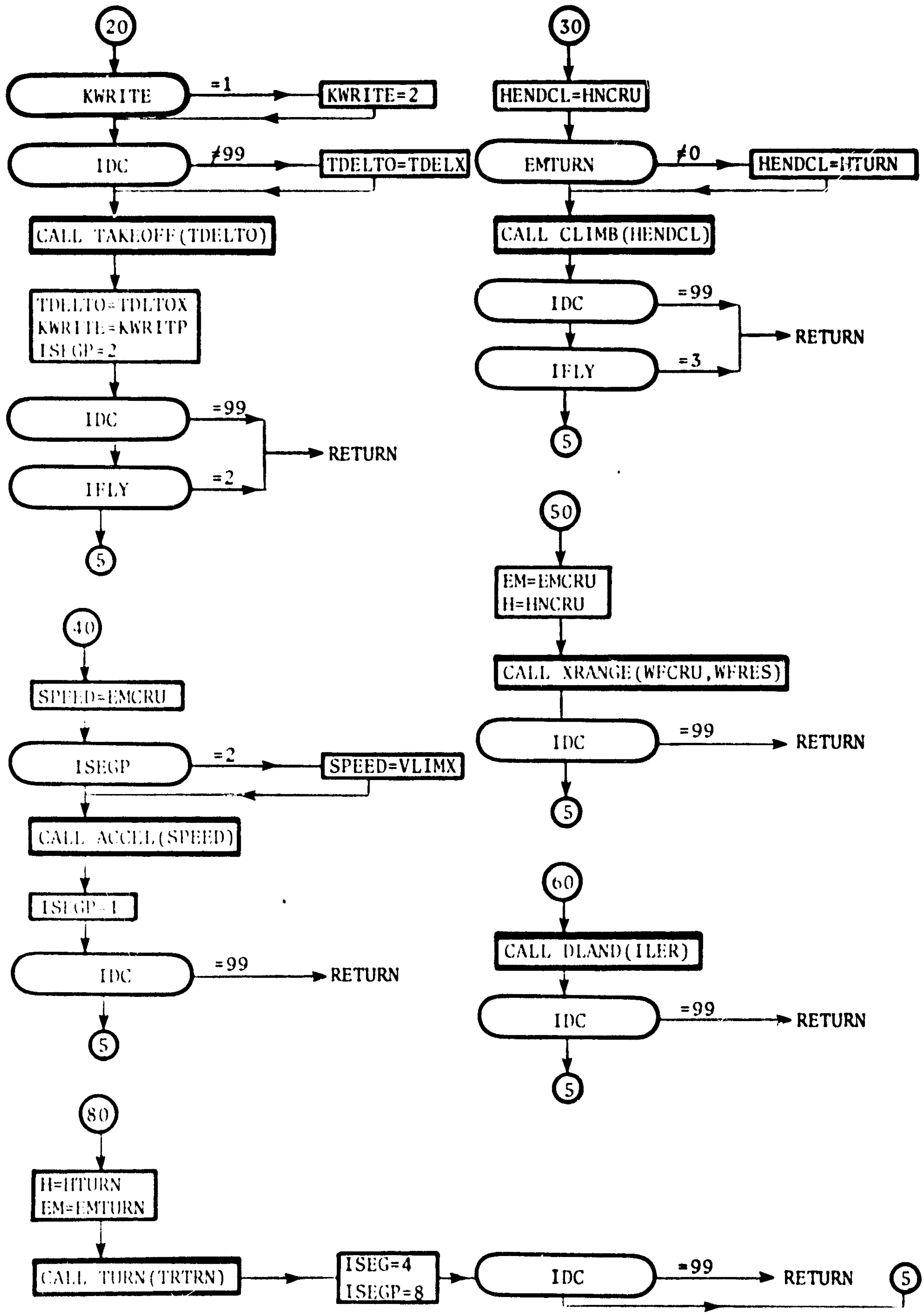


Figure VI.3.1 Subroutine PERFRM, Detailed Flow Chart


20
KWRITE
=1
KWRITE=2
IDC
≠99
TDELTO=TDELX
CALL TAKEOFF(TDELTO)
TDELTO=TDLTOX
KWRITE=KWRITP
ISEGP=2
IDC
=99
RETURN
IFLY
=2
5
30
HENDCL=HNCRU
EMTURN
≠0
HENDCL=HTURN
CALL CLIMB(HENDCL)
IDC
=99
RETURN
IFLY
=3
5
40
SPEED=EMCRU
ISEGP
=2
SPEED=VLIMX
CALL ACCEL(SPEED)
ISEGP=1
IDC
=99
RETURN
5
50
EM=EMCRU
H=HNCRU
CALL XRANGE(WFCRU,WFRES)
IDC
=99
RETURN
5
60
CALL DLAND(ILER)
IDC
=99
RETURN
5
80
H=HTURN
EM=EMTURN
CALL TURN(TRTRN)
ISEG=4
ISEGP=8
IDC
=99
RETURN
5

VI.3.2 Subroutine ACCEL, Acceleration Segment

This routine computes level flight accelerations under the control of PERFRM. Several subroutines are called by ACCEL during these calculations including:

TPALT--Atmospheric Properties

DRAG--Drag Calculations

ENGINE--Propulsion Characteristics

Major outputs computed by this routine are weight, range and time changes during the acceleration.

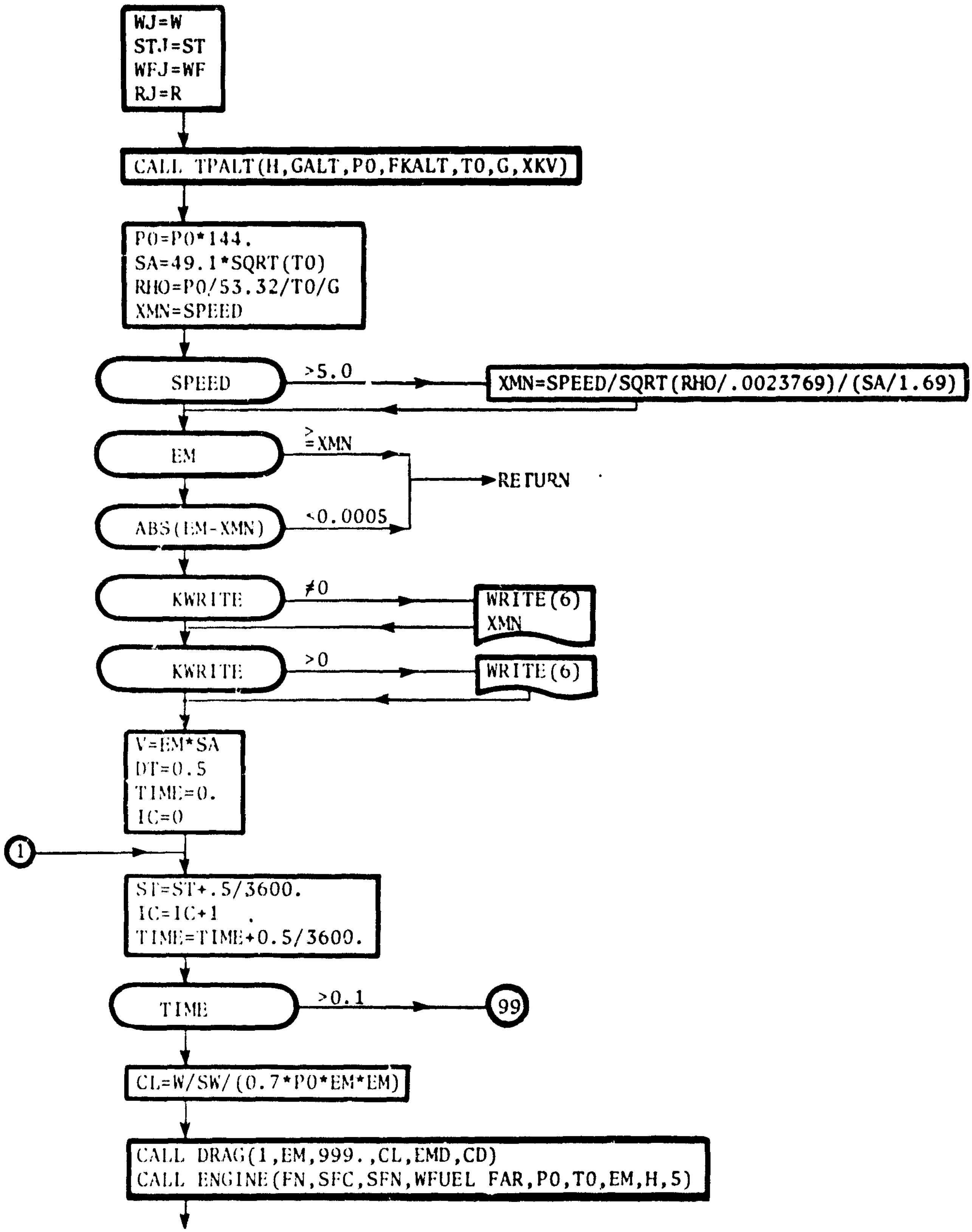
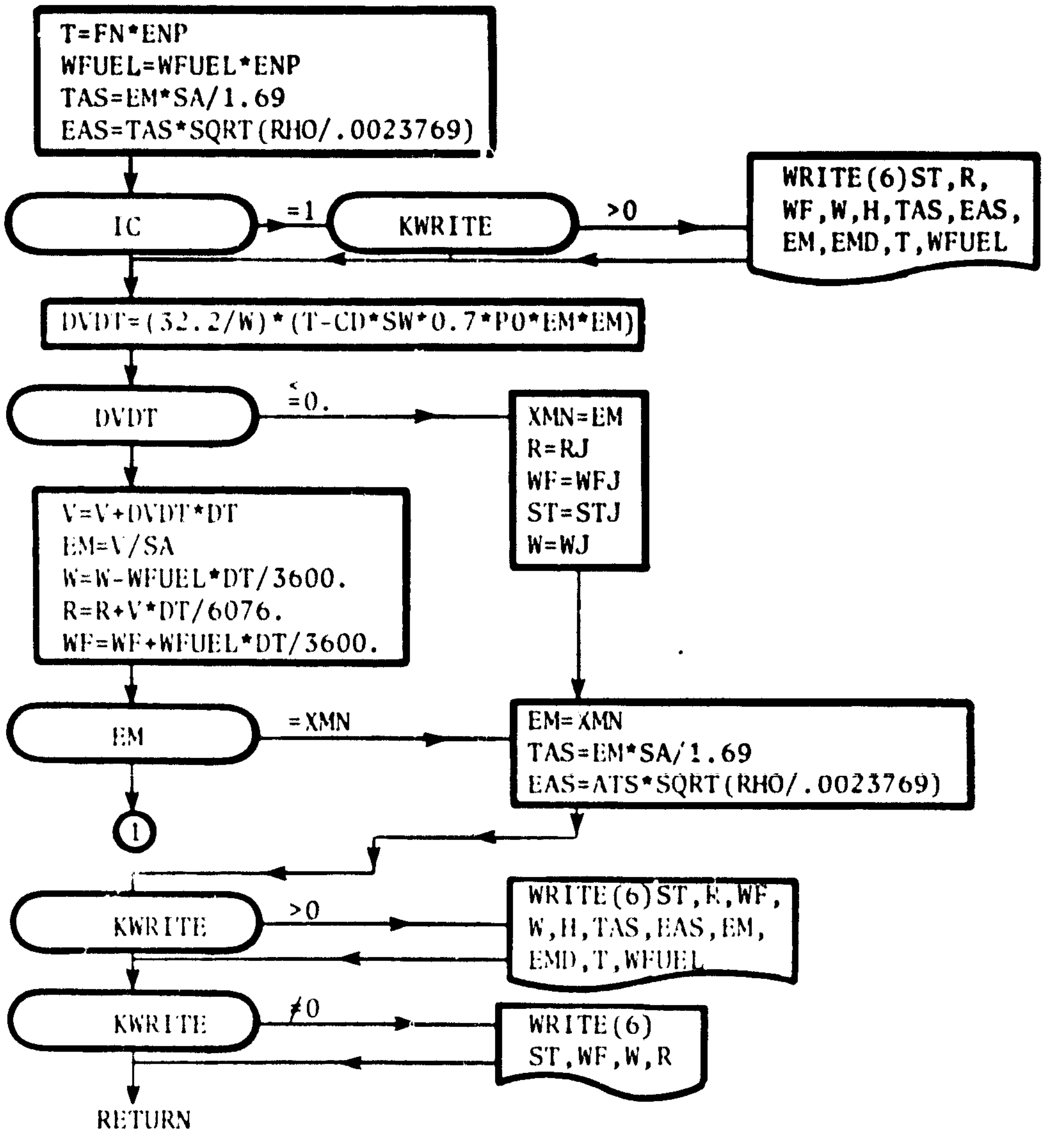


Figure VI.3.2 Subroutine ACCEL, Detailed Flow Chart


T=FN*ENP
WFUEL=WFUEL*ENP
TAS=EM*SA/1.69
EAS=TAS*SQRT(RHO/.0023769)
IC
=1
KWRITE
>0
WRITE(6)ST,R,
WF,W,H,TAS,EAS,
EM,EMD,T,WFUEL
DVDT=(32.2/W)*(T-CD*SW*0.7*PO*EM*EM)
DVDT
≤0.
XMN=EM
R=RJ
WF=WFJ
ST=STJ
W=WJ
V=V+DVDT*DT
EM=V/SA
W=W-WFUEL*DT/3600.
R=R+V*DT/6076.
WF=WF+WFUEL*DT/3600.
EM
=XMN
EM=XMN
TAS=EM*SA/1.69
EAS=ATS*SQRT(RHO/.0023769)
1
KWRITE
>0
WRITE(6)ST,R,WF,
W,H,TAS,EAS,EM,
EMD,T,WFUEL
KWRITE
≠0
WRITE(6)
ST,WF,W,R
RETURN

VI.3.3 Subroutine CLIMB, Climb Calculations

This routine computes pianar climb segment characteristics. Atmospheric properties along the climb path are determined from TPALT. Required lift coefficient is found from CLIFT and drag coefficient from DRAG. Propulsion system characteristics are obtained from ENGINE.

Climb paths may be flown at maximum rate-of-climb, at maximum allowable airspeed, at a specified equivalent air speed. Three functions PART, BOZA, and ADEN are defined and used in subroutine CLIMB.

FUNCTION DEFINITIONS

```
PART(XXX,X111,X222,X333,Y111)=(((XXX-X111)*(XXX-X222))/
    ((X333-X111)*(X333-X222)))*Y111
BO2A(V11,V22,V33,U1,U2,U3)=(V11**2*(U2-U3)-V22**2*(U1-U3)
    +V33**2*(U1-U2))/(2.*(-V11*(U2-U3)+V22*(U1-U3)-V33*(U1-U2)
ADEN=W1,W2,W3,Z1,Z2,Z3)=Z3*(W2-W1)+Z2*(W1-W3)+Z1*(W3-W2)
```

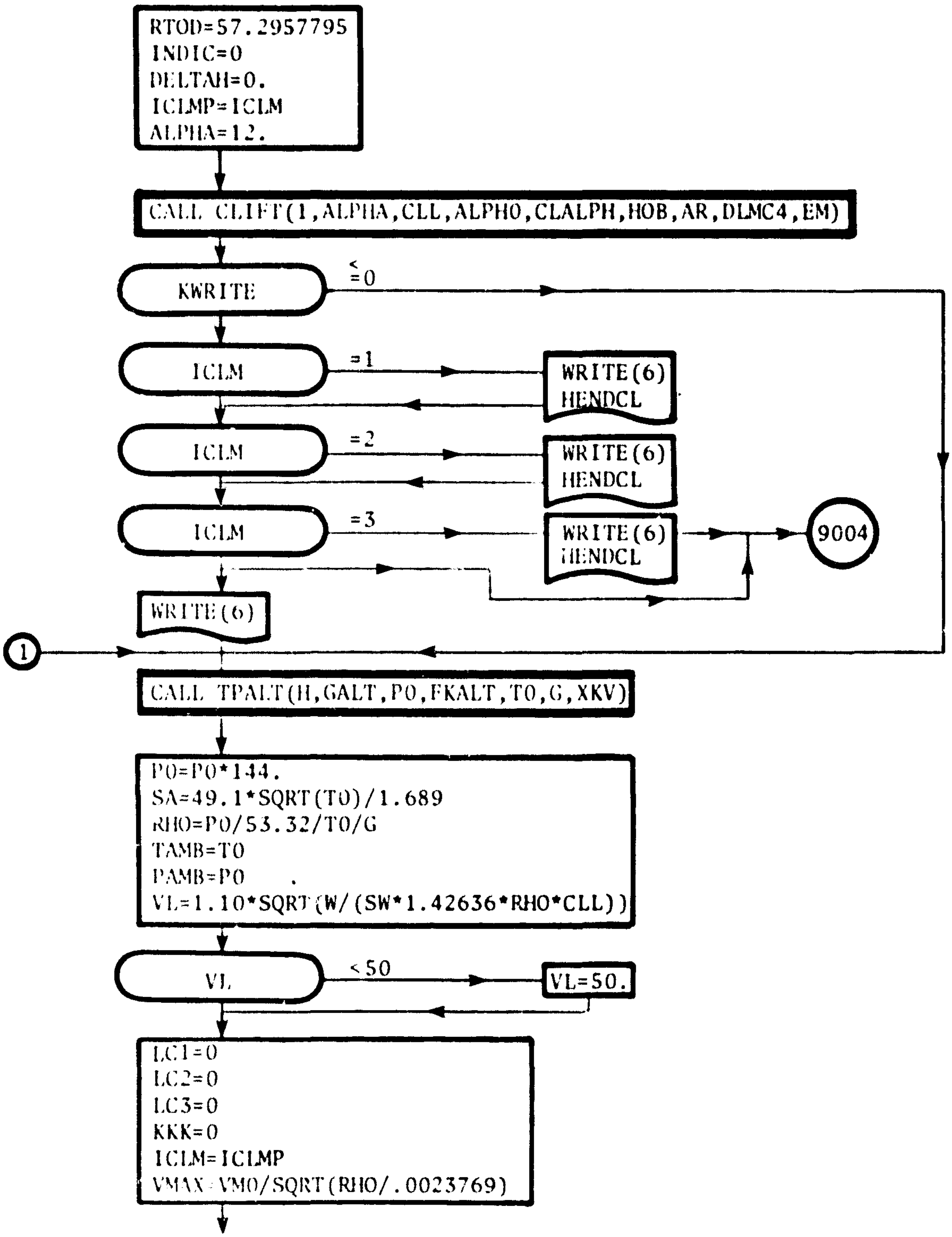
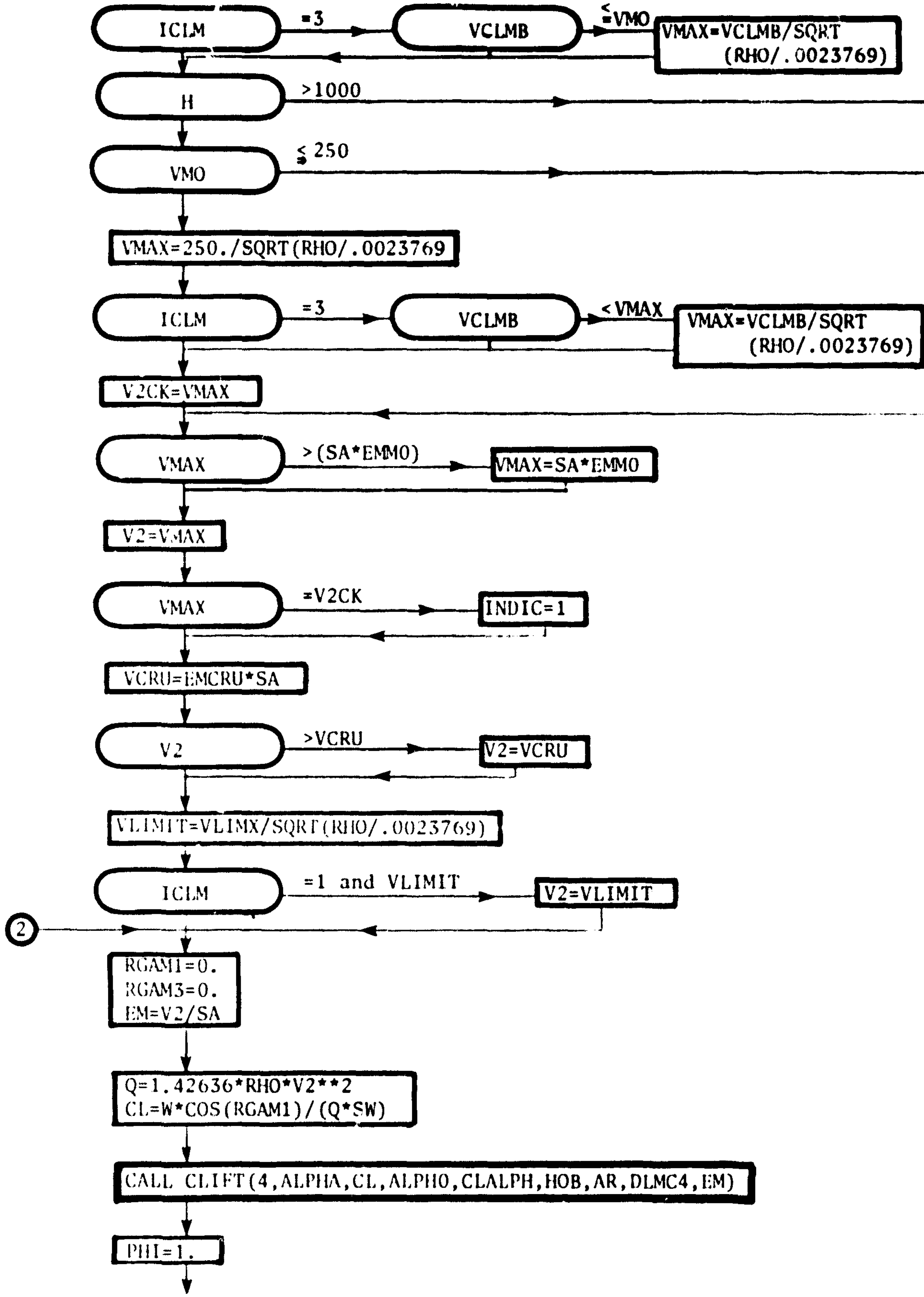
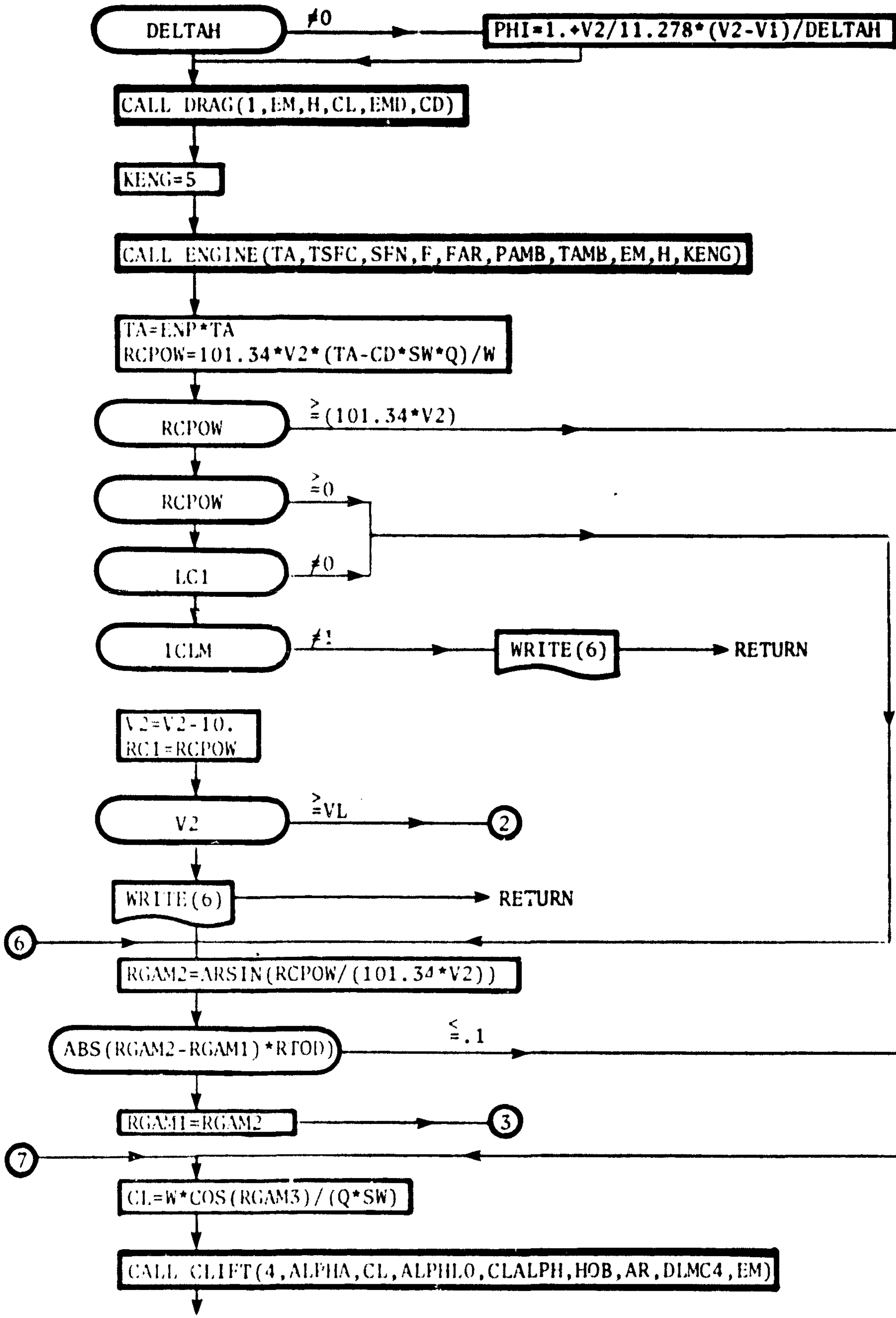
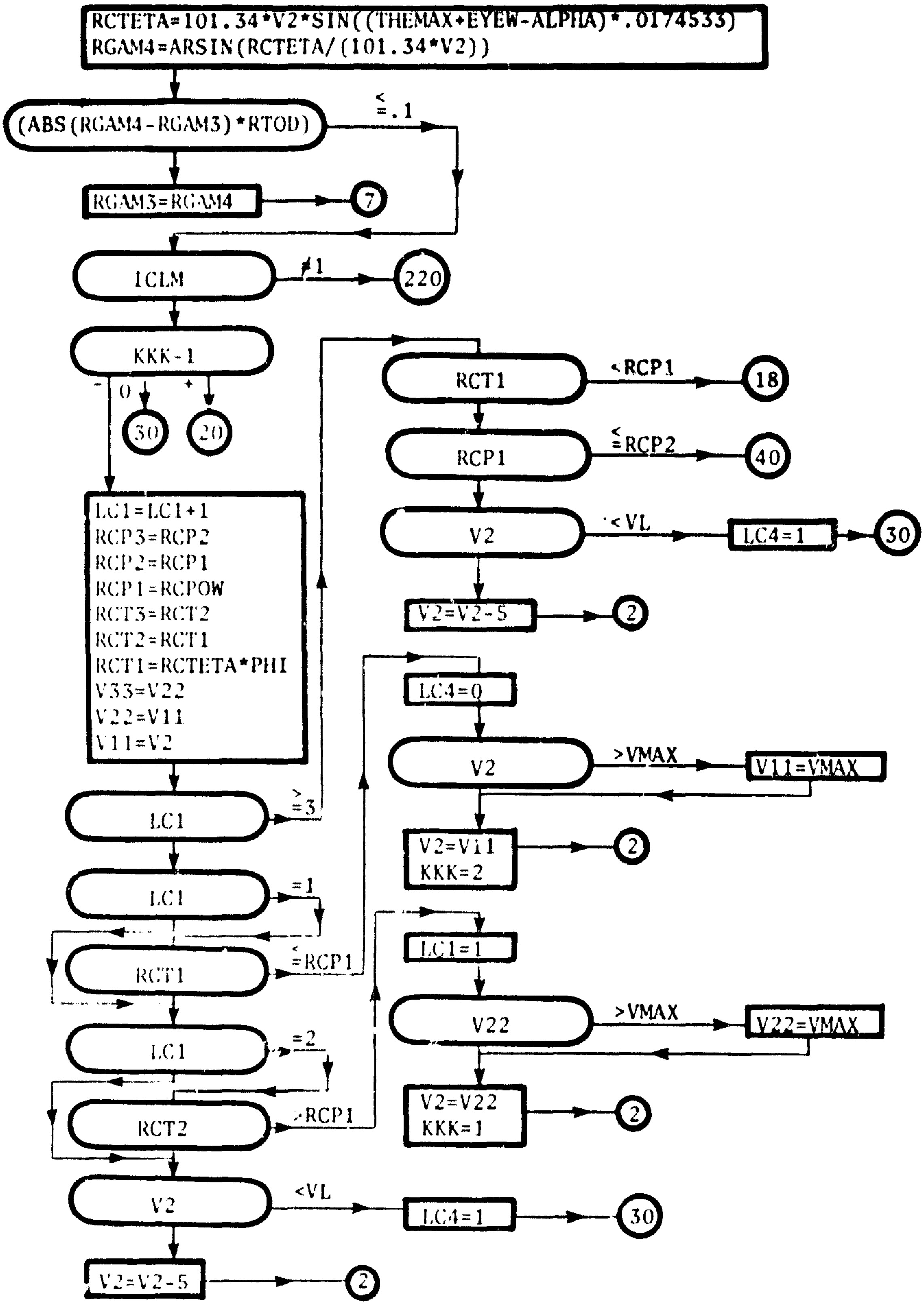


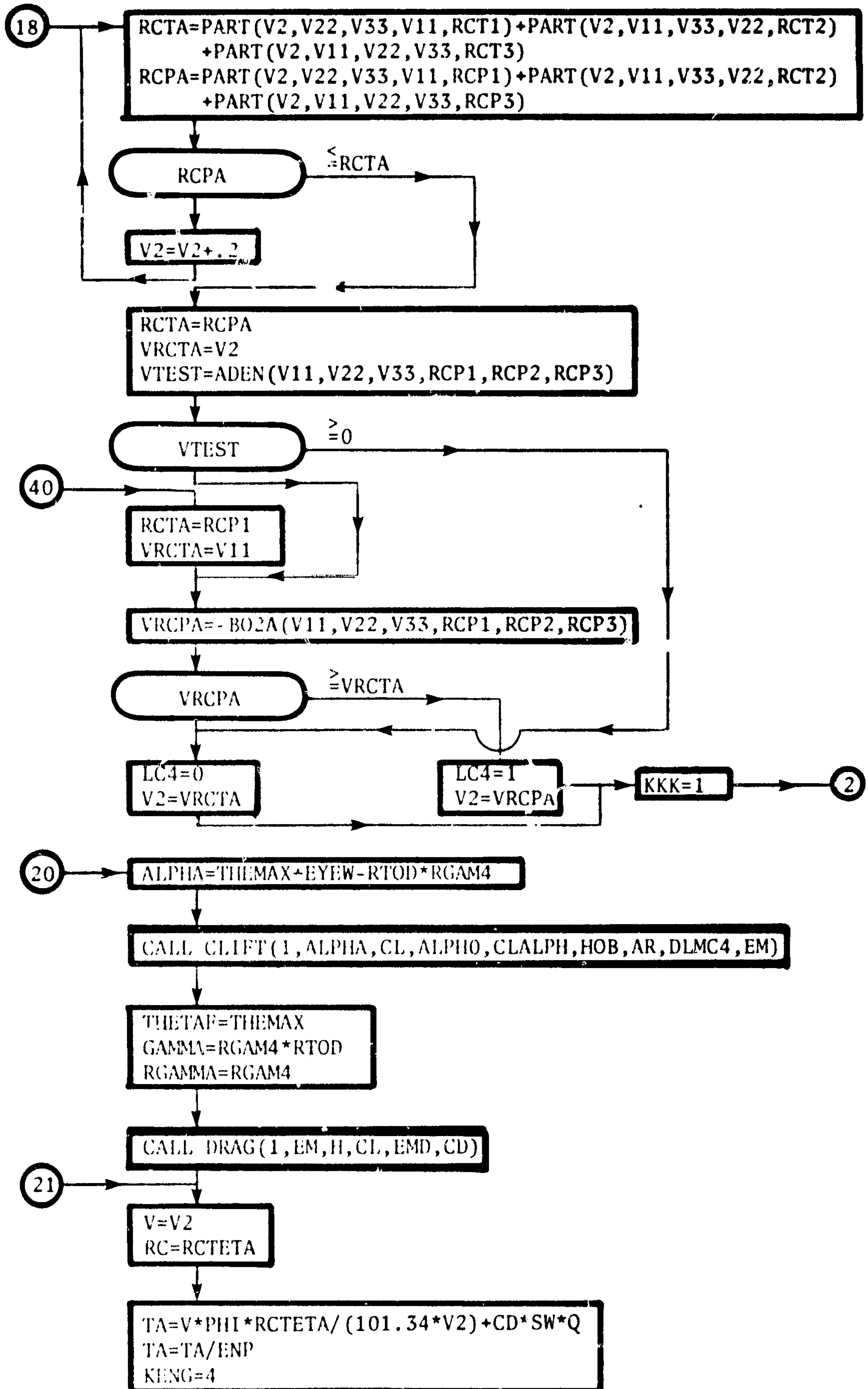
Figure VI.3.3 Subroutine CLIMB, Detailed Flow Chart

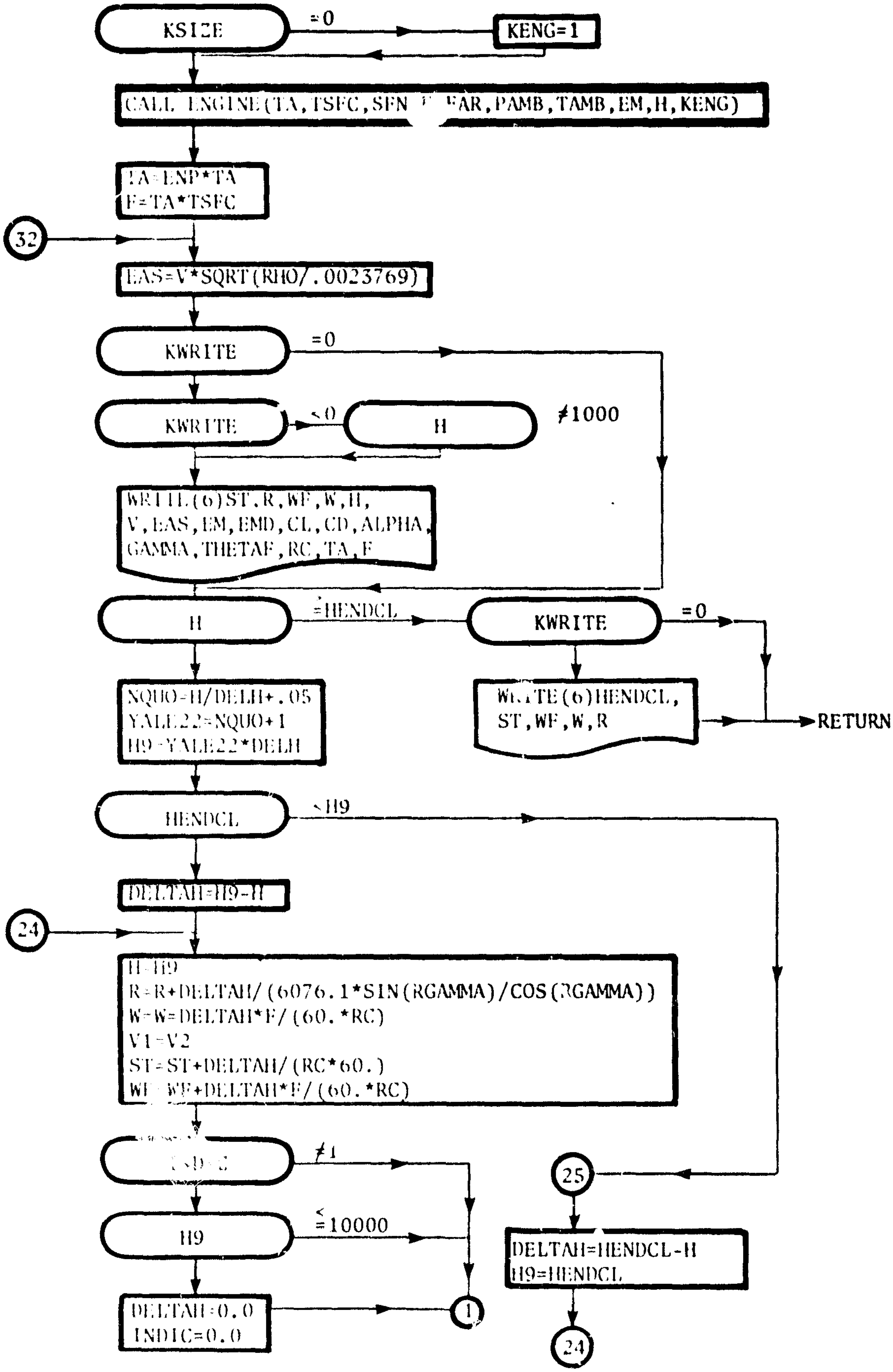

ICLM
=3
VCLMB
≤VMO
VMAX=VCLMB/SQRT (RHO/.0023769)
H
>1000
VMO
≤ 250
VMAX=250./SQRT(RHO/.0023769)
ICLM
=3
VCLMB
< VMAX
VMAX=VCLMB/SQRT (RHO/.0023769)
V2CK=VMAX
VMAX
> (SA*EMMO)
VMAX=SA*EMMO
V2=VMAX
VMAX
=V2CK
INDIC=1
VCRU=EMCRU*SA
V2
>VCRU
V2=VCRU
VLIMIT=VLIMX/SQRT(RHO/.0023769)
ICLM
=1 and VLIMIT
V2=VLIMIT
2
RGAM1=0.
RGAM3=0.
EM=V2/SA
Q=1.42636*RHO*V2**2
CL=W*COS(RGAM1)/(Q*SW)
CALL CLIFT(4,ALPHA,CL,ALPHO,CLALPH,HOB,AR,DLMC4,EM)
PHI=1.

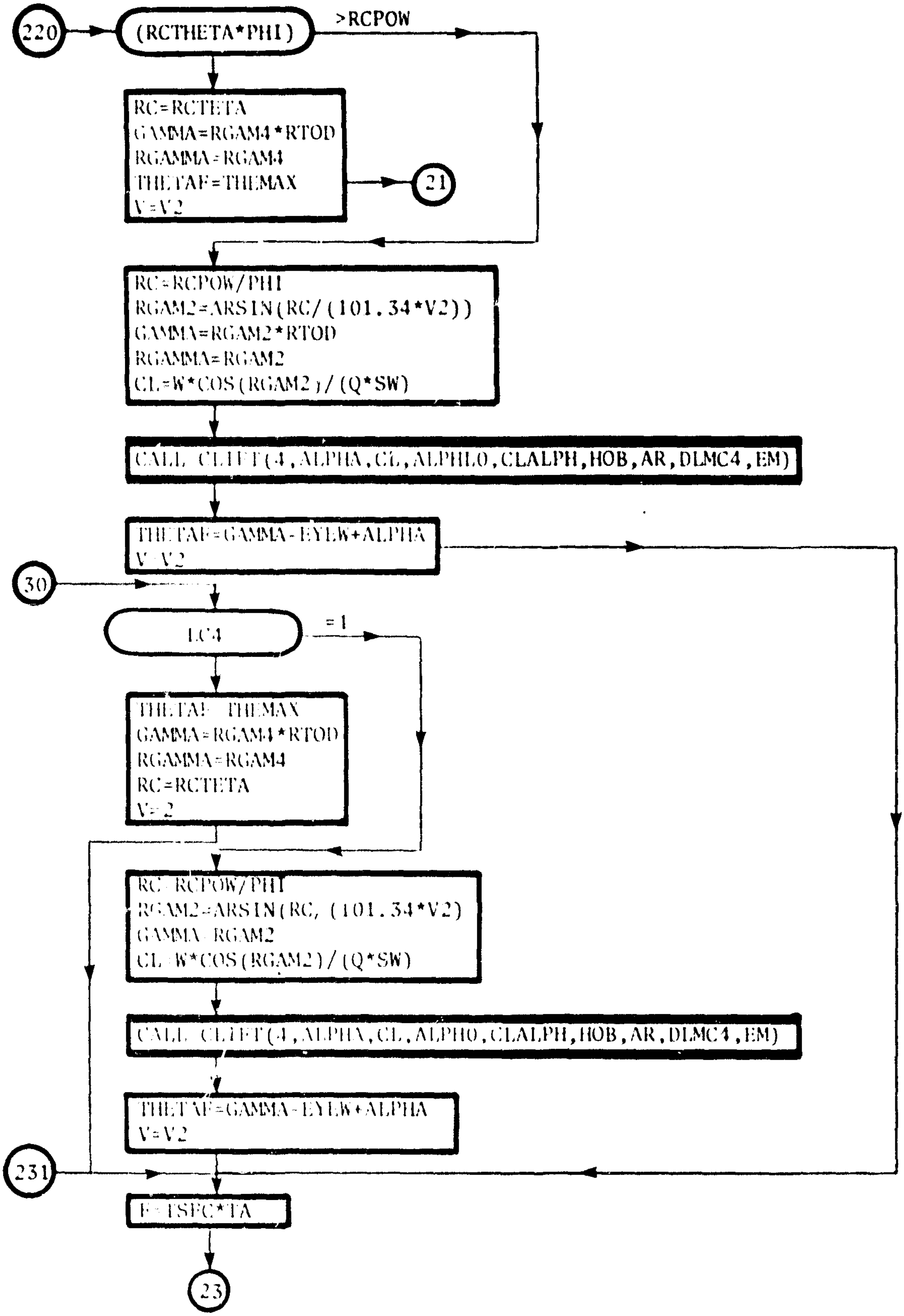

DELTAH
≠0
PHI=1.+V2/11.278*(V2-V1)/DELTAH
CALL DRAG(1,EM,H,CL,EMD,CD)
KENG=5
CALL ENGINE(TA,TSFC,SFN,F,FAR,PAMB,TAMB,EM,H,KENG)
TA=ENP*TA
RCPOW=101.34*V2*(TA-CD*SW*Q)/W
RCPOW
≥(101.34*V2)
RCPOW
≥0
LC1
≠0
ICLM
≠1
WRITE(6)
RETURN
V2=V2-10.
RC1=RCPOW
V2
≥VL
2
WRITE(6)
RETURN
6
RGAM2=ARSIN(RCPOW/(101.34*V2))
ABS(RGAM2-RGAM1)*RTOD
≤.1
RGAM1=RGAM2
3
7
CL=W*COS(RGAM3)/(Q*SW)
CALL CLIFT(4,ALPHA,CL,ALPHLO,CLALPH,HOB,AR,DLMC4,EM)



ORIGINAL PAGE IS
OF POOR QUALITY


18
RCTA=PART(V2,V22,V33,V11,RCT1)+PART(V2,V11,V33,V22,RCT2)
+PART(V2,V11,V22,V33,RCT3)
RCPA=PART(V2,V22,V33,V11,RCP1)+PART(V2,V11,V33,V22,RCT2)
+PART(V2,V11,V22,V33,RCP3)
RCPA
≤RCTA
V2=V2+.2
RCTA=RCPA
VRCTA=V2
VTEST=ADEN(V11,V22,V33,RCP1,RCP2,RCP3)
VTEST
≥0
40
RCTA=RCP1
VRCTA=V11
VRCPA=-BO2A(V11,V22,V33,RCP1,RCP2,RCP3)
VRCPA
≥VRCTA
LC4=0
V2=VRCTA
LC4=1
V2=VRCPA
KKK=1
2
20
ALPHA=THEMAX+EYEW-RTOD*RGAM4
CALL CLIFT(1,ALPHA,CL,ALPHO,CLALPH,HOB,AR,DLMC4,EM)
THETAF=THEMAX
GAMMA=RGAM4*RTOD
RGAMMA=RGAM4
CALL DRAG(1,EM,H,CL,EMD,CD)
21
V=V2
RC=RCTETA
TA=V*PHI*RCTETA/(101.34*V2)+CD*SW*Q
TA=TA/ENP
KENG=4


KSIZE
=0
KENG=1
ENP*TA
F=TA*TSFC
32
EAS=V*SQRT(RHO/.0023769)
KWRITE
=0
KWRITE
<0
H
≠1000
WRITE(6)ST,R,WF,W,H,
V,EAS,EM,EMD,CL,CD,ALPHA,
GAMMA,THETAF,RC,TA,F
H
=HENDCL
KWRITE
=0
ST,WF,W,R
RETURN
NQUO=H/DELH+.05
YALE22=NQUO+1
H9=YALE22*DELH
HENDCL
<H9
DELTAH=H9-H
24
H=H9
R=R+DELTAH/(6076.1*SIN(RGAMMA)/COS(RGAMMA))
V1=V2
ST=ST+DELTAH/(RC*60.)
≠1
H9
≤10000
INDIC=0.0
1
25
DELTAH=HENDCL-H
H9=HENDCL
24


220
(RCTHETA*PHI)
>RCPOW
RC=RCTETA
GAMMA=RGAM4*RTOD
RGAMMA=RGAM4
THETAF=THEMAX
V=V2
21
RC=RCPOW/PHI
RGAM2=ARSIN(RC/(101.34*V2))
GAMMA=RGAM2*RTOD
RGAMMA=RGAM2
CL=W*COS(RGAM2)/(Q*SW)
CALL CLIFT(4,ALPHA,CL,ALPHLO,CLALPH,HOB,AR,DLMC4,EM)
THETAF=GAMMA-EYEW+ALPHA
V=V2
30
LC4
=1
THETAF=THEMAX
GAMMA=RGAM4*RTOD
RGAMMA=RGAM4
RC=RCTETA
V=2
RC=RCPOW/PHI
RGAM2=ARSIN(RC,(101.34*V2)
GAMMA=RGAM2
CL=W*COS(RGAM2)/(Q*SW)
CALL CLIFT(4,ALPHA,CL,ALPHO,CLALPH,HOB,AR,DLMC4,EM)
THETAF=GAMMA-EYEW+ALPHA
V=V2
231
F=TSFC*TA
23

VI.3.4 Subroutine DLAND, Landing Calculations

Subroutine DLAND computes landing segment characteristics from an analysis of terminal glide, flare, touchdown and braking roll measures. Atmospheric properties for the landing field are computed from TPALT. Routine CLIFT and DRAG provide landing aerodynamic characteristics, propulsion system characteristics are obtained through the routine ENGINE.

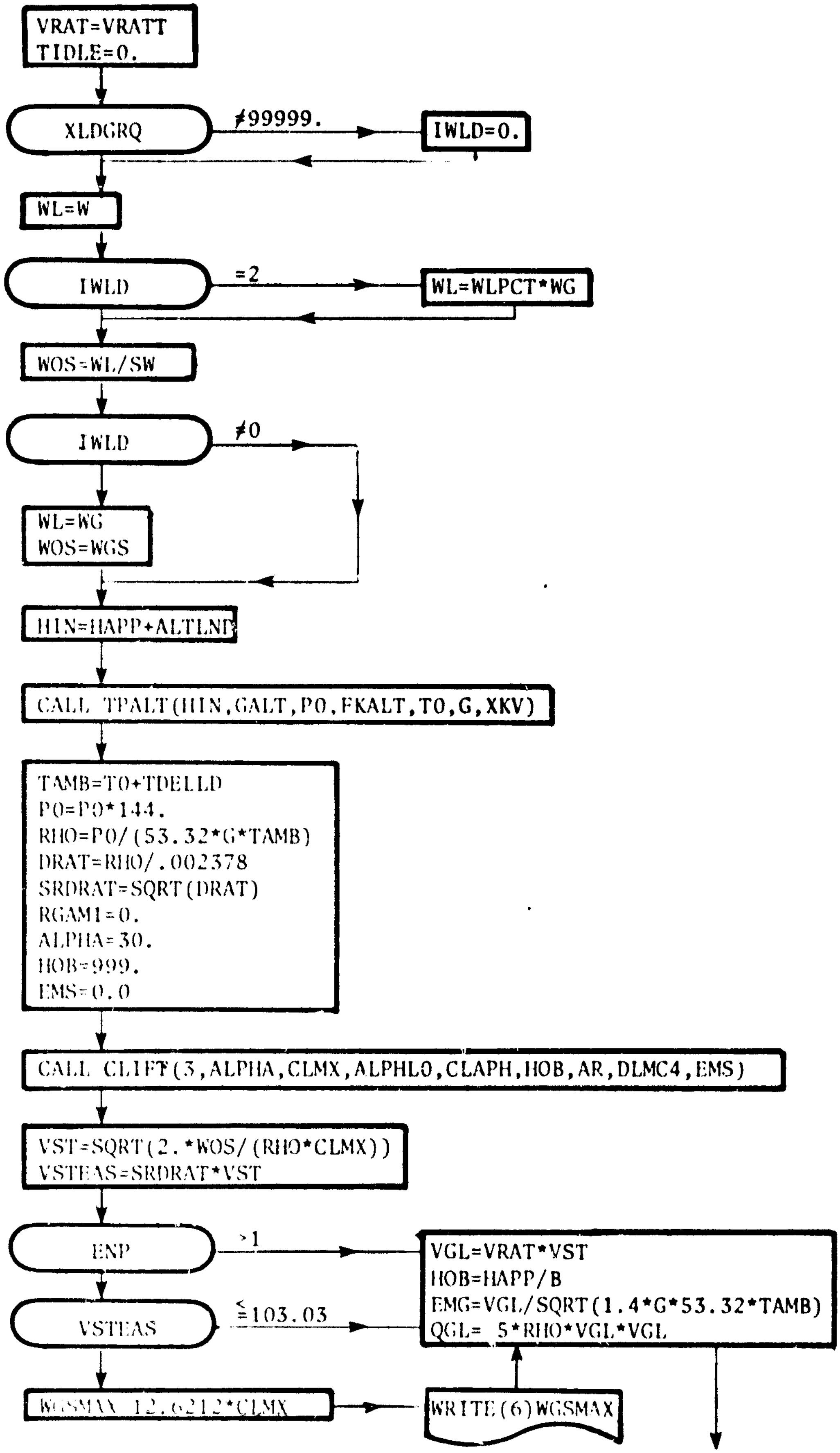
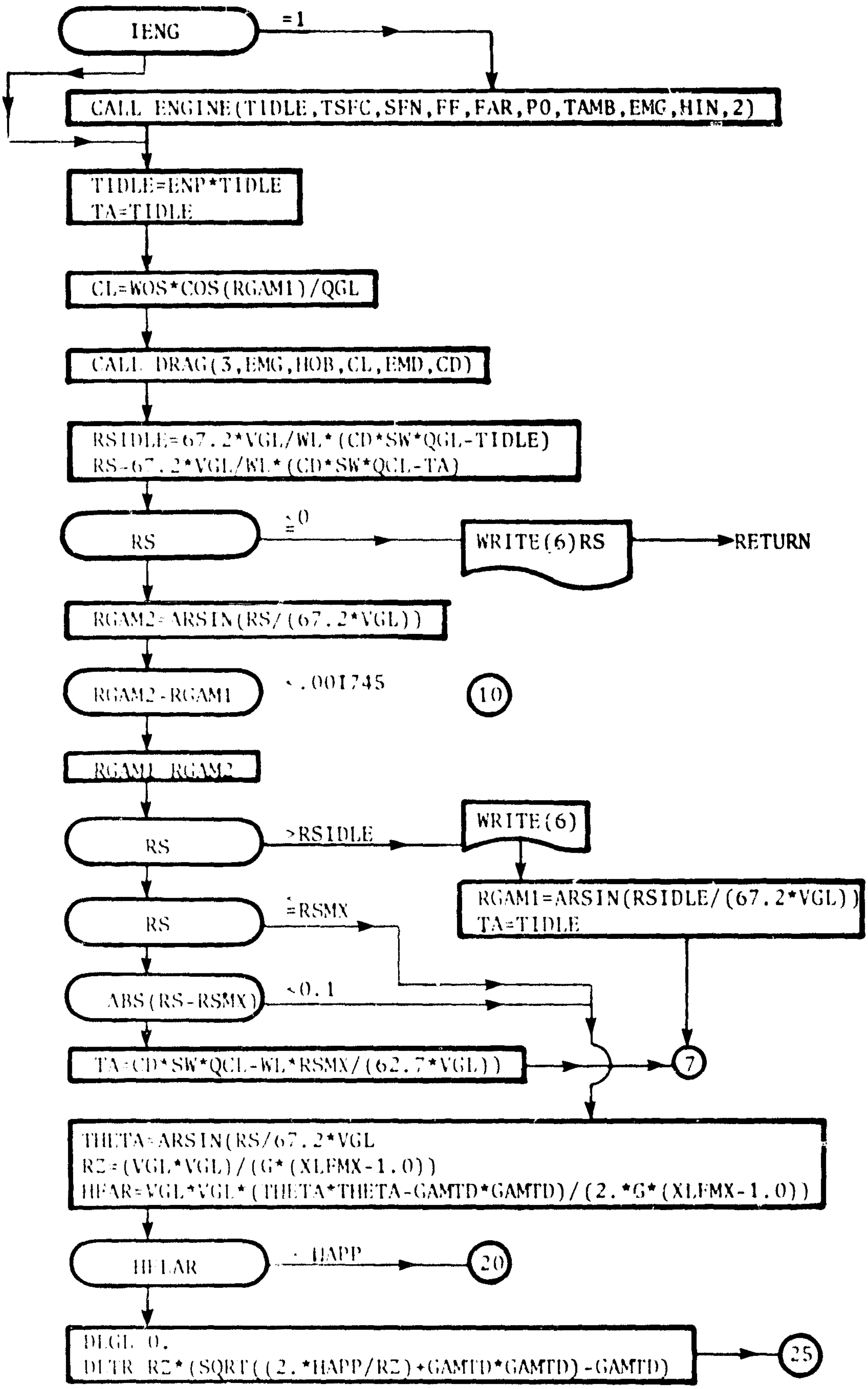
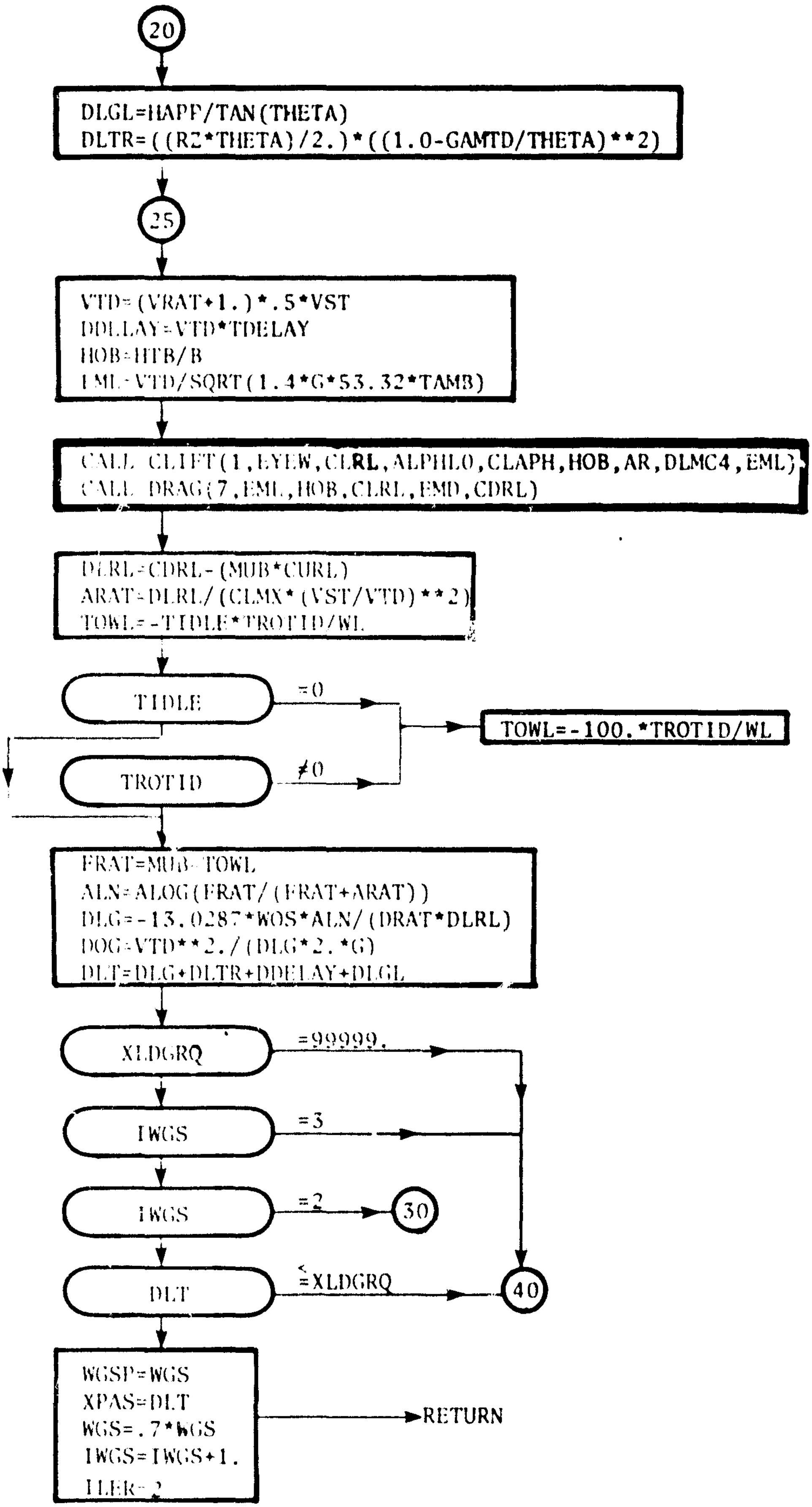
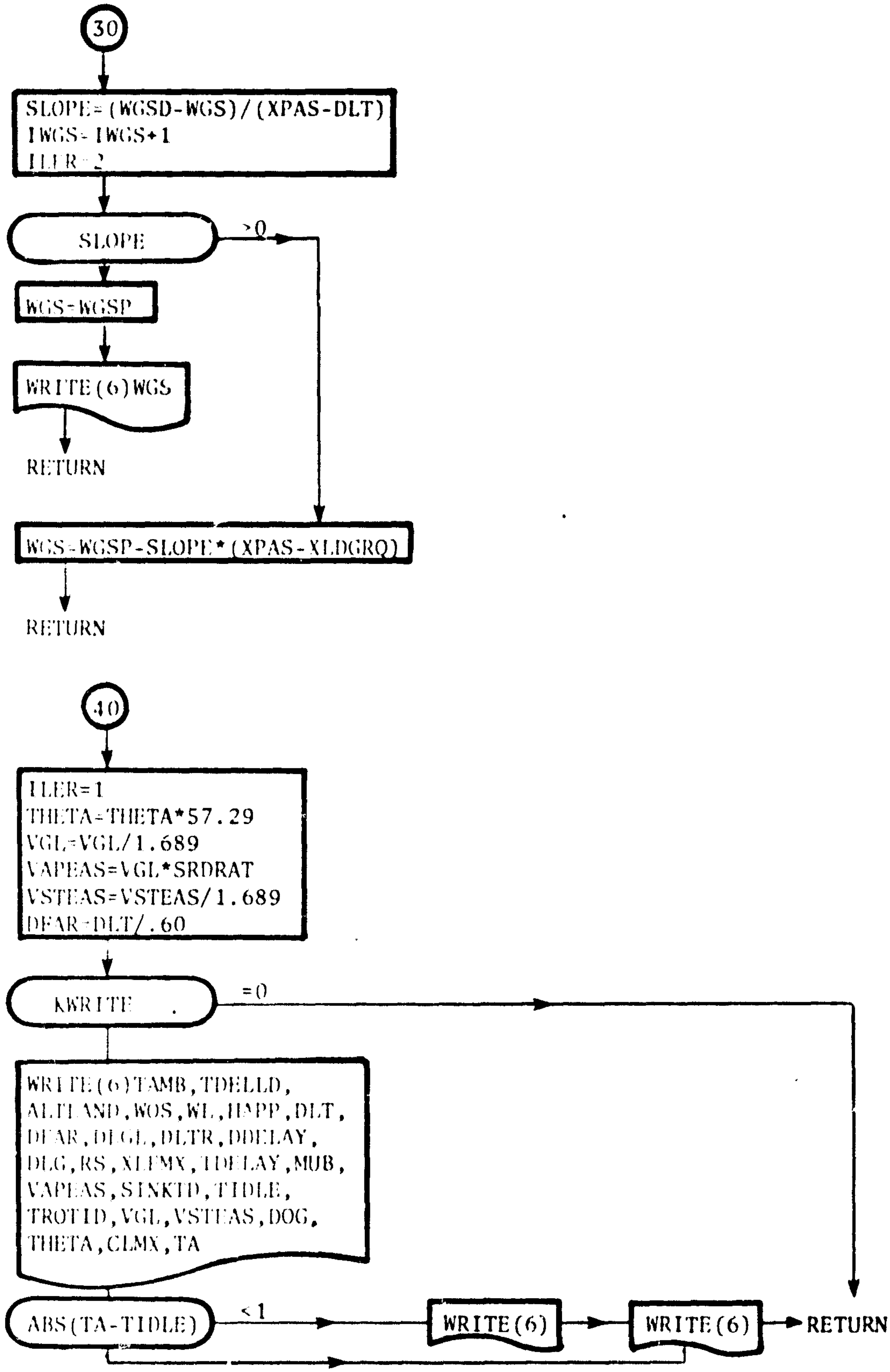


Figure VI.3.4 Subroutine DLAND, Detailed Flow Chart


IENG
=1
CALL ENGINE(TIDLE,TSFC,SFN,FF,FAR,PO,TAMB,EMG,HIN,2)
TIDLE=ENP*TIDLE
TA=TIDLE
CL=WOS*COS(RGAM1)/QGL
CALL DRAG(3,EMG,HOB,CL,EMD,CD)
RSIDLE=67.2*VGL/WL*(CD*SW*QGL-TIDLE)
RS=67.2*VGL/WL*(CD*SW*QGL-TA)
RS
≧0
WRITE(6)RS
RETURN
RGAM2=ARSIN(RS/(67.2*VGL))
RGAM2-RGAM1
<.001745
10
RGAM1=RGAM2
RS
>RSIDLE
WRITE(6)
RGAM1=ARSIN(RSIDLE/(67.2*VGL))
TA=TIDLE
RS
≧RSMX
ABS(RS-RSMX)
<0.1
TA=CD*SW*QGL-WL*RSMX/(62.7*VGL))
7
THETA=ARSIN(RS/67.2*VGL
RZ=(VGL*VGL)/(G*(XLFMX-1.0))
HFAR=VGL*VGL*(THETA*THETA-GAMTD*GAMTD)/(2.*G*(XLFMX-1.0))
HFLAR
<HAPP
20
DEGL=0.
DLTR=RZ*(SQRT((2.*HAPP/RZ)+GAMTD*GAMTD)-GAMTD)
25


20
DLGL=HAPP/TAN(THETA)
DLTR=((RZ*THETA)/2.)*((1.0-GAMTD/THETA)**2)
25
VTD=(VRAT+1.)*.5*VST
DDELAY=VTD*TDELAY
HOB=HTB/B
EML=VTD/SQRT(1.4*G*53.32*TAMB)
CALL CLIFT(1,EYEW,CLRL,ALPHLO,CLAPH,HOB,AR,DLMC4,EML)
CALL DRAG(7,EML,HOB,CLRL,EMD,CDRL)
DLRL=CDRL-(MUB*CLRL)
ARAT=DLRL/(CLMX*(VST/VTD)**2)
TOWL=-TIDLE*TROTID/WL
TIDLE
=0
TOWL=-100.*TROTID/WL
TROTID
≠0
FRAT=MUB-TOWL
ALN=ALOG(FRAT/(FRAT+ARAT))
DLG=-13.0287*WOS*ALN/(DRAT*DLRL)
DOG=VTD**2./(DLG*2.*G)
DLT=DLG+DLTR+DDELAY+DLGL
XLDGRQ
=99999.
IWGS
=3
IWGS
=2
30
DLT
≤XLDGRQ
40
WGSP=WGS
XPAS=DLT
WGS=.7*WGS
IWGS=IWGS+1.
ILER=2
RETURN


30
SLOPE=(WGSD-WGS)/(XPAS-DLT)
IWGS=IWGS+1
ILER=2
SLOPE
>0
WGS=WGSP
WRITE(6)WGS
RETURN
WGS=WGSP-SLOPE*(XPAS-XLDGRQ)
RETURN
40
ILER=1
THETA=THETA*57.29
VGL=VGL/1.689
VAPEAS=VGL*SRDRAT
VSTEAS=VSTEAS/1.689
DEAR=DLT/.60
KWRITE
=0
WRITE(6)TAMB,TDELLD,
ALTLAND,WOS,WL,HAPP,DLT,
DEAR,DEGL,DLTR,DDELAY,
DLG,RS,XLFMX,TDELAY,MUB,
VAPEAS,SINKTD,TIDLE,
TROTID,VGL,VSTEAS,DOG,
THETA,CLMX,TA
ABS(TA-TIDLE)
<1
WRITE(6)
WRITE(6)
RETURN

VI.3.5 Subroutine TAKOFF, Take-off Calculations

Subroutine TAKOFF computes take-off performance through ground roll, rotation, lift-off, gear retraction, initial climb, and flap retraction. The take-off trajectory is integrated and derivatives are computed by subroutine DERIV. Atmospheric properties are computed by TPALT. Aerodynamics are obtained from CLIFT and DRAG. Propulsion characteristics are obtained from ENGINE. Integration is performed by the utility routine INTS.

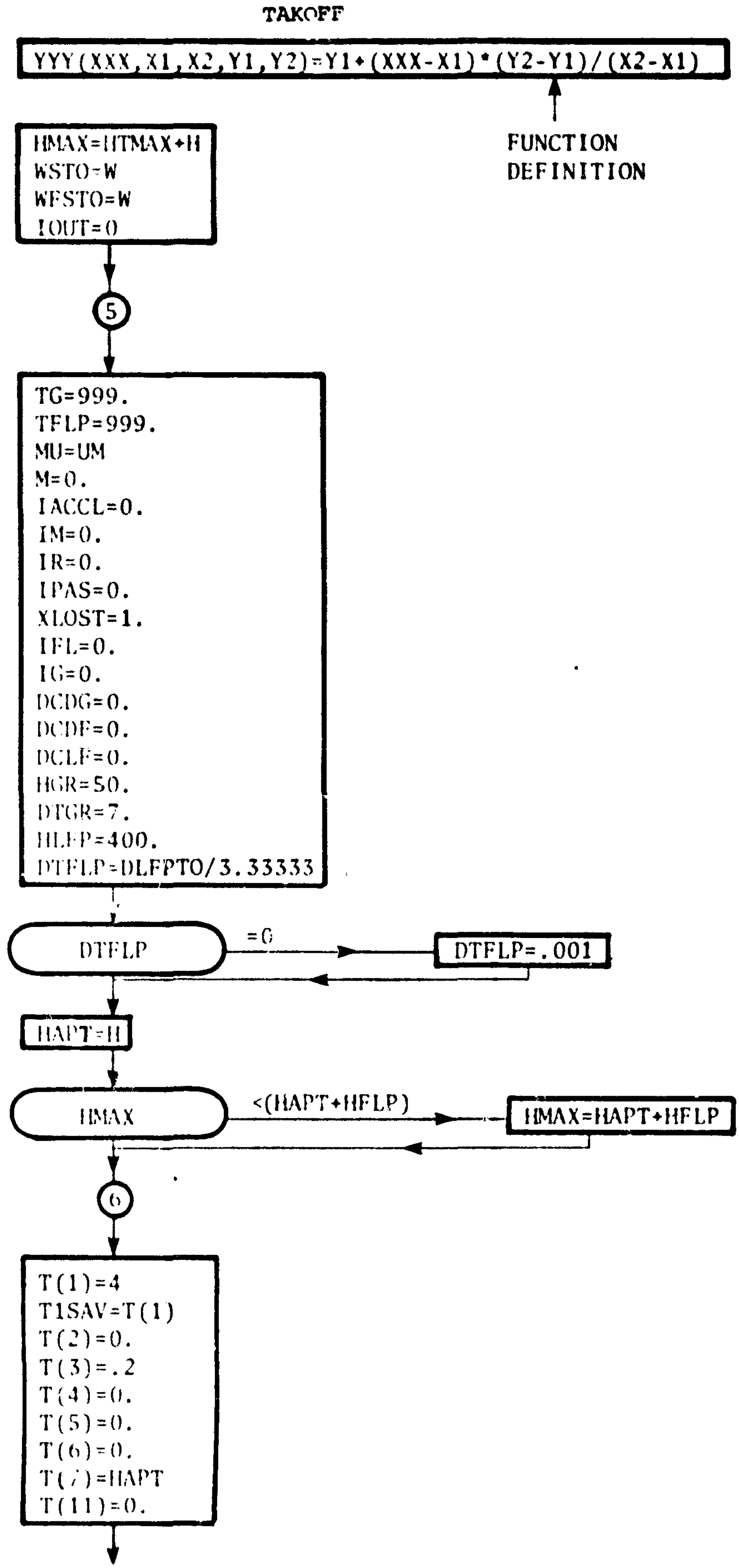
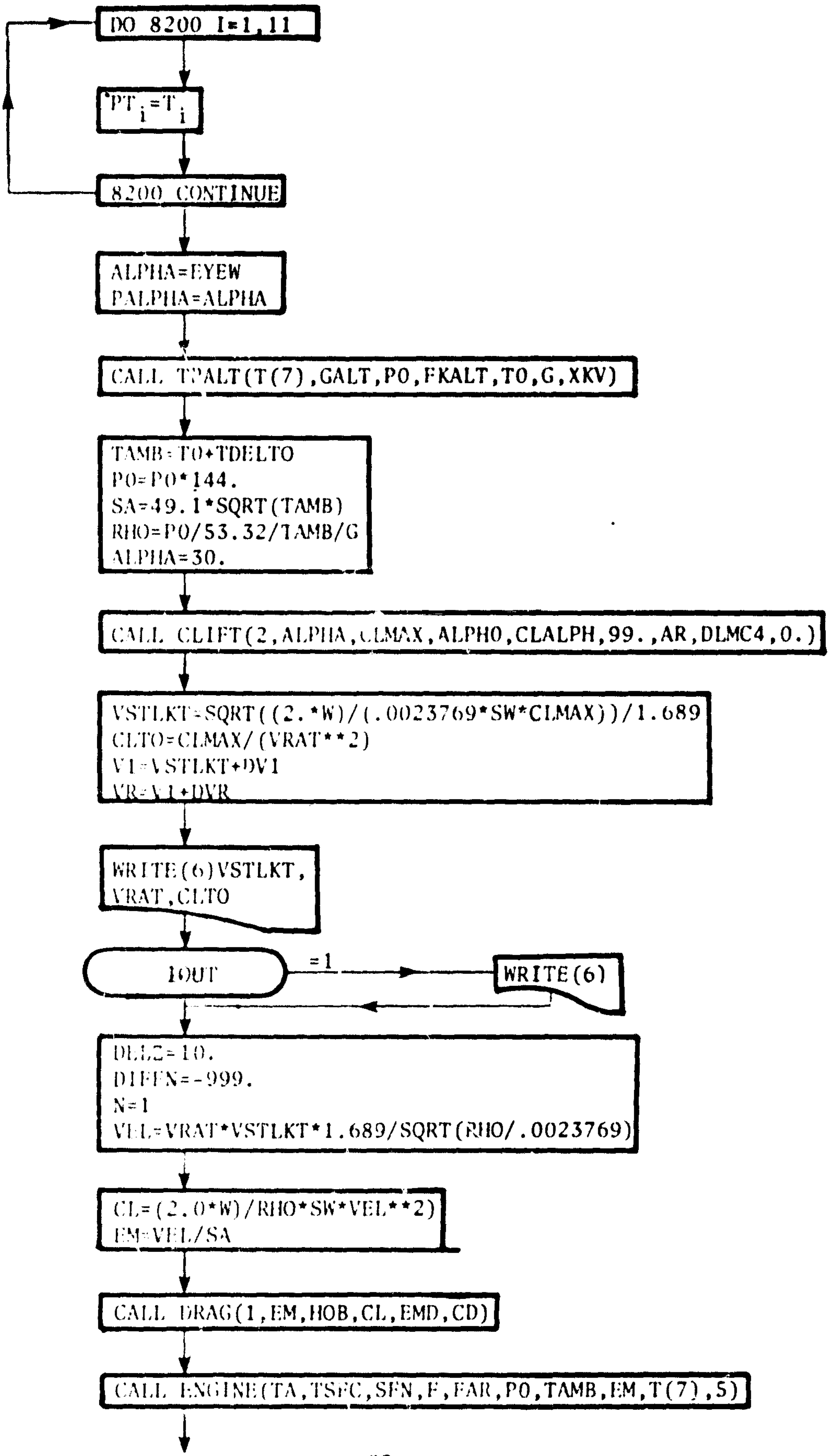
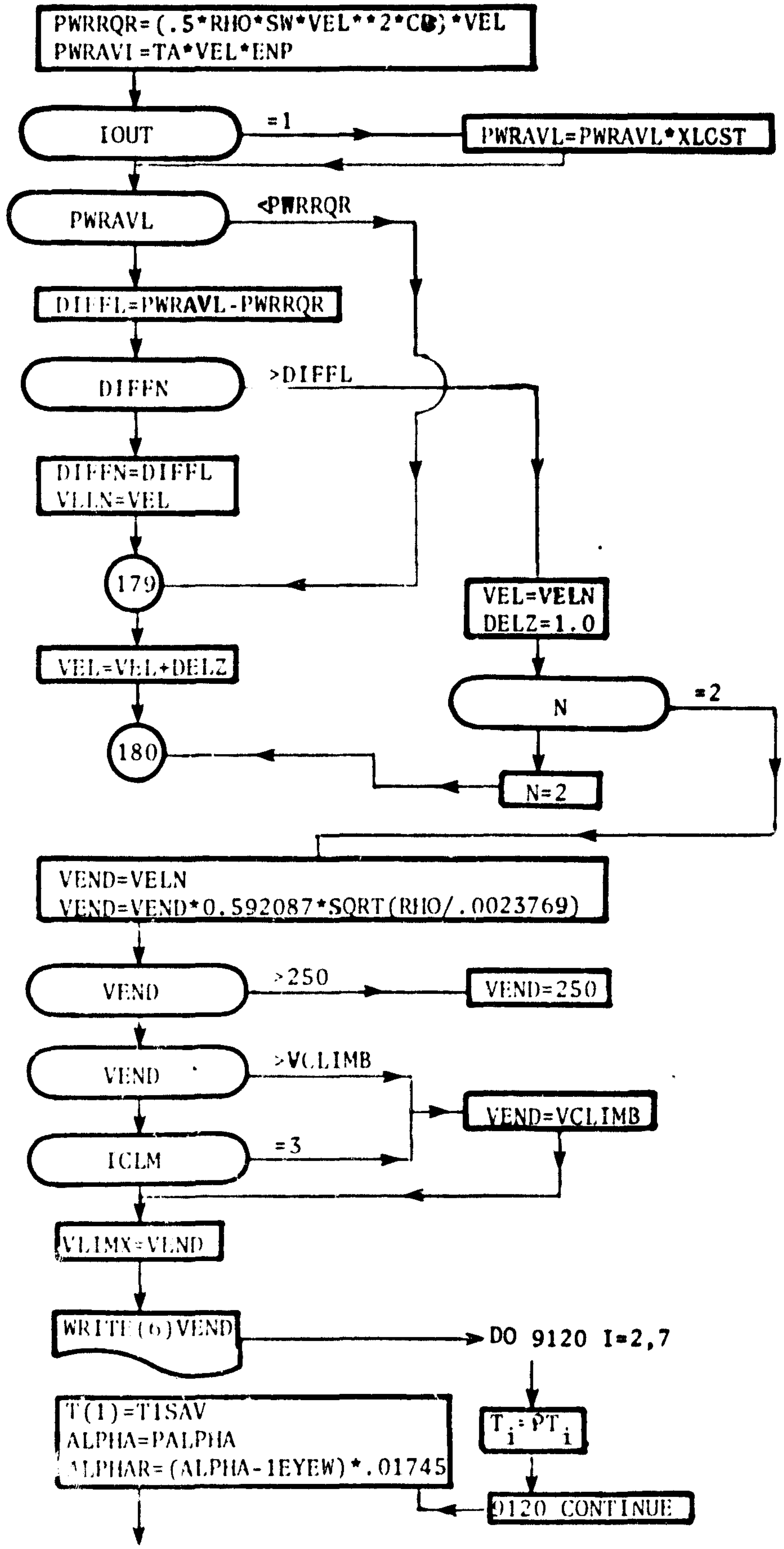
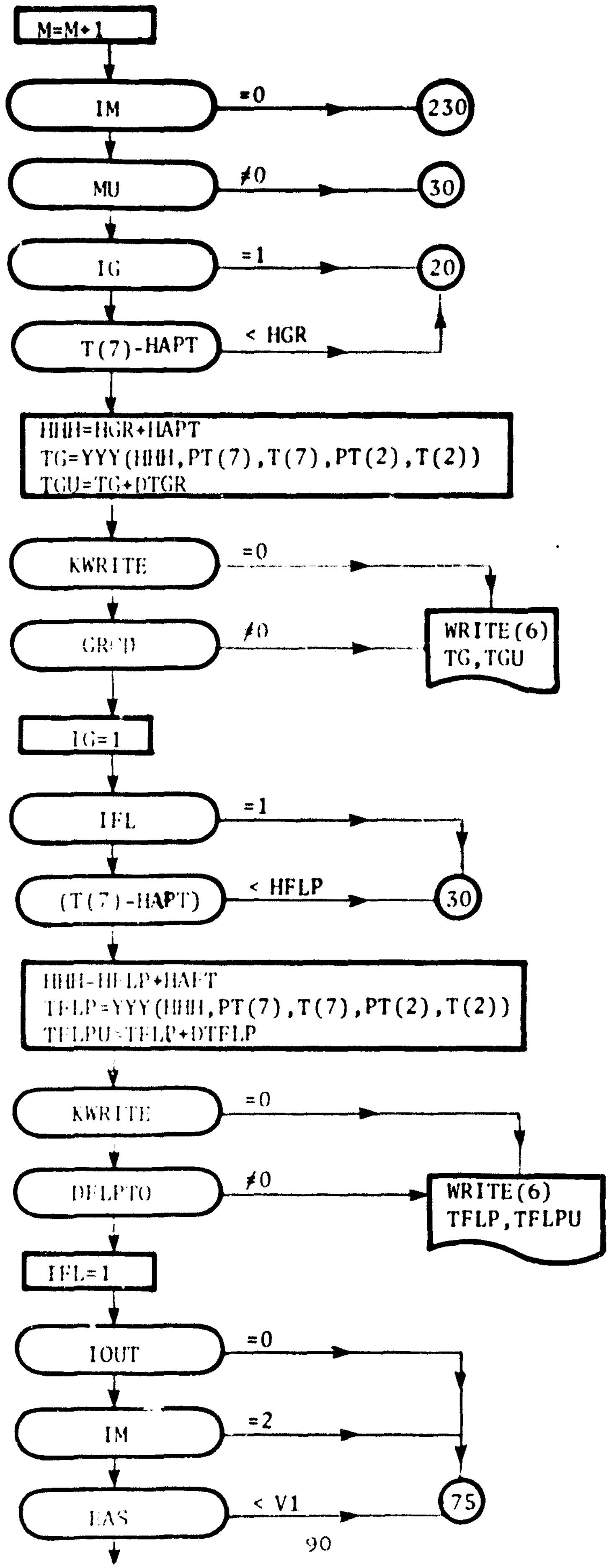
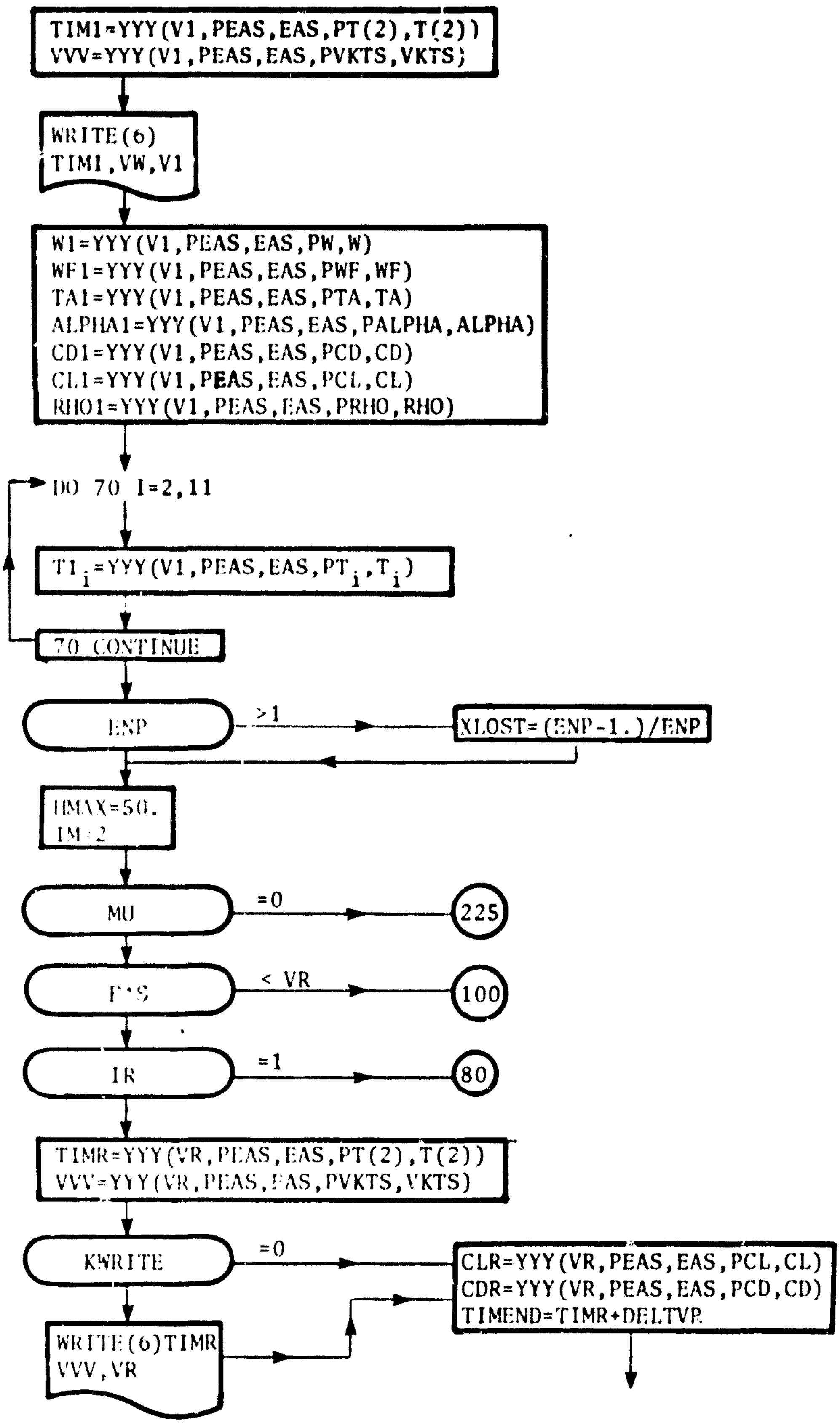


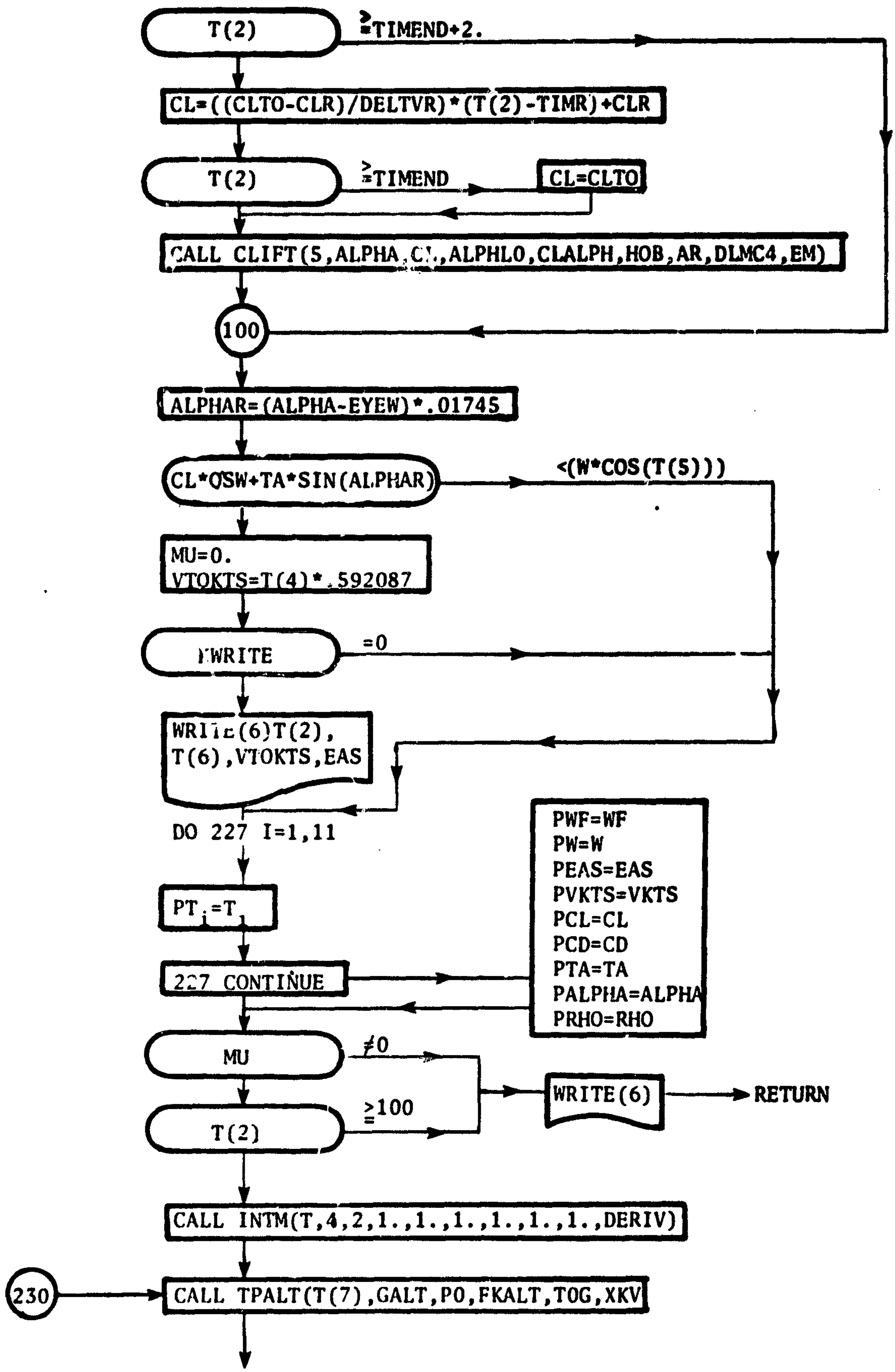
Figure VI.3.5 Subroutine TAKOFF, Detailed Flow Chart


DO 8200 I=1,11
$PT_i=T_i$
8200 CONTINUE
ALPHA=EYEW
PALPHA=ALPHA
CALL TPALT(T(7),GALT,PO,FKALT,TO,G,XKV)
TAMB=TO+TDELTO
PO=PO*144.
SA=49.1*SQRT(TAMB)
RHO=PO/53.32/TAMB/G
ALPHA=30.
CALL CLIFT(2,ALPHA,CLMAX,ALPHO,CLALPH,99.,AR,DLMC4,0.)
VSTLKT=SQRT((2.*W)/(.0023769*SW*CLMAX))/1.689
CLTO=CLMAX/(VRAT**2)
V1=VSTLKT+DV1
VR=V1+DVR
WRITE(6)VSTLKT,
VRAT,CLTO
IOUT
=1
WRITE(6)
DELZ=10.
DIFFN=-999.
N=1
VEL=VRAT*VSTLKT*1.689/SQRT(RHO/.0023769)
CL=(2.0*W)/RHO*SW*VEL**2)
EM=VEL/SA
CALL DRAG(1,EM,HOB,CL,EMD,CD)
CALL ENGINE(TA,TSFC,SFN,F,FAR,PO,TAMB,EM,T(7),5)

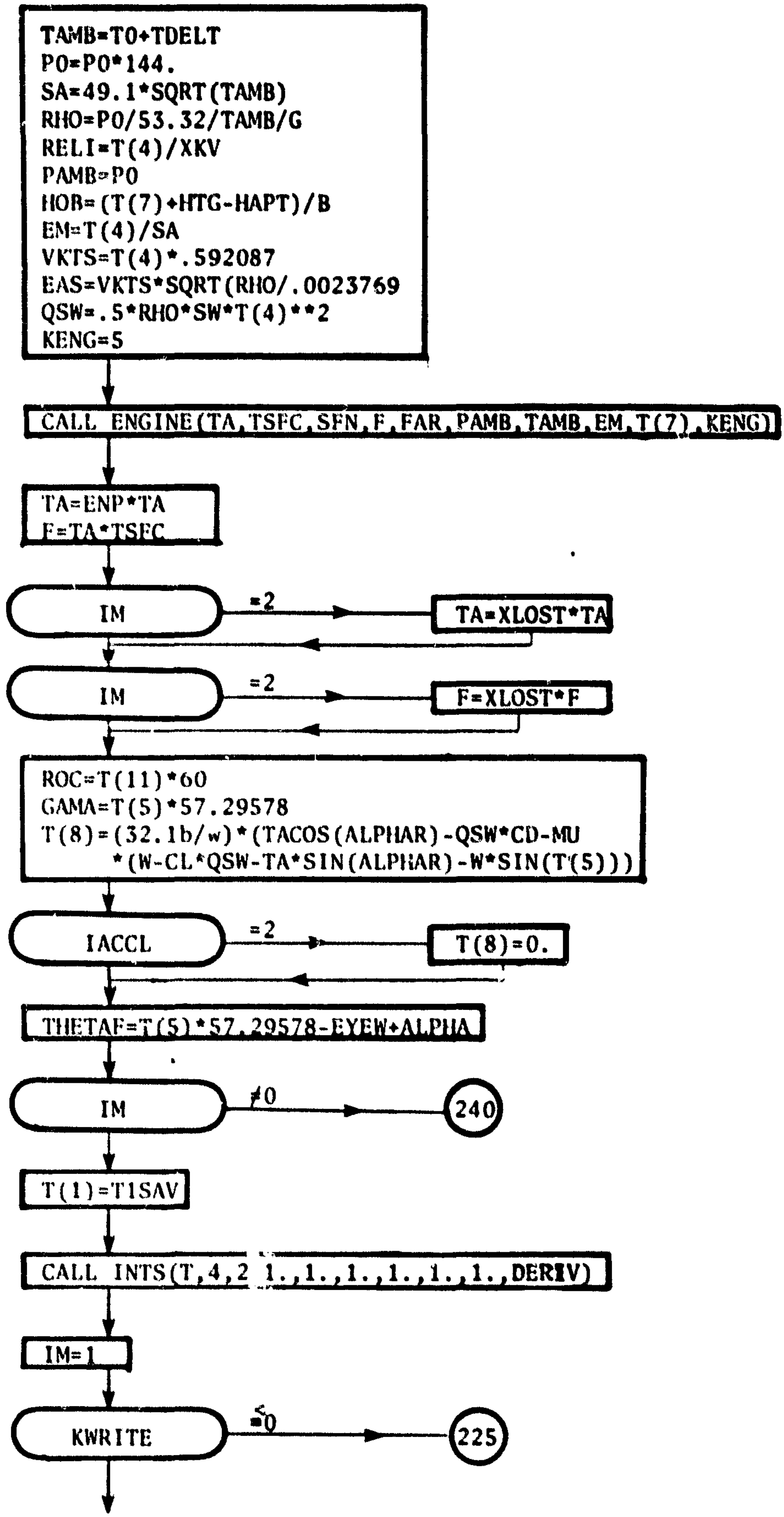

PWRRQR=(.5*RHO*SW*VEL**2*CD)*VEL
PWRAVL=TA*VEL*ENP
IOUT
=1
PWRAVL=PWRAVL*XLOST
PWRAVL
<PWRRQR
DIFFL=PWRAVL-PWRRQR
DIFFN
>DIFFL
DIFFN=DIFFL
VELN=VEL
179
VEL=VELN
DELZ=1.0
VEL=VEL+DELZ
N
=2
180
N=2
VEND=VELN
VEND=VEND*0.592087*SQRT(RHO/.0023769)
VEND
>250
VEND=250
VEND
>VCLIMB
VEND=VCLIMB
ICLM
=3
VLIMX=VEND
WRITE(6)VEND
DO 9120 I=2,7
T(1)=T1SAV
ALPHA=PALPHA
ALPHAR=(ALPHA-IEYEW)*.01745
Ti=PTi
9120 CONTINUE

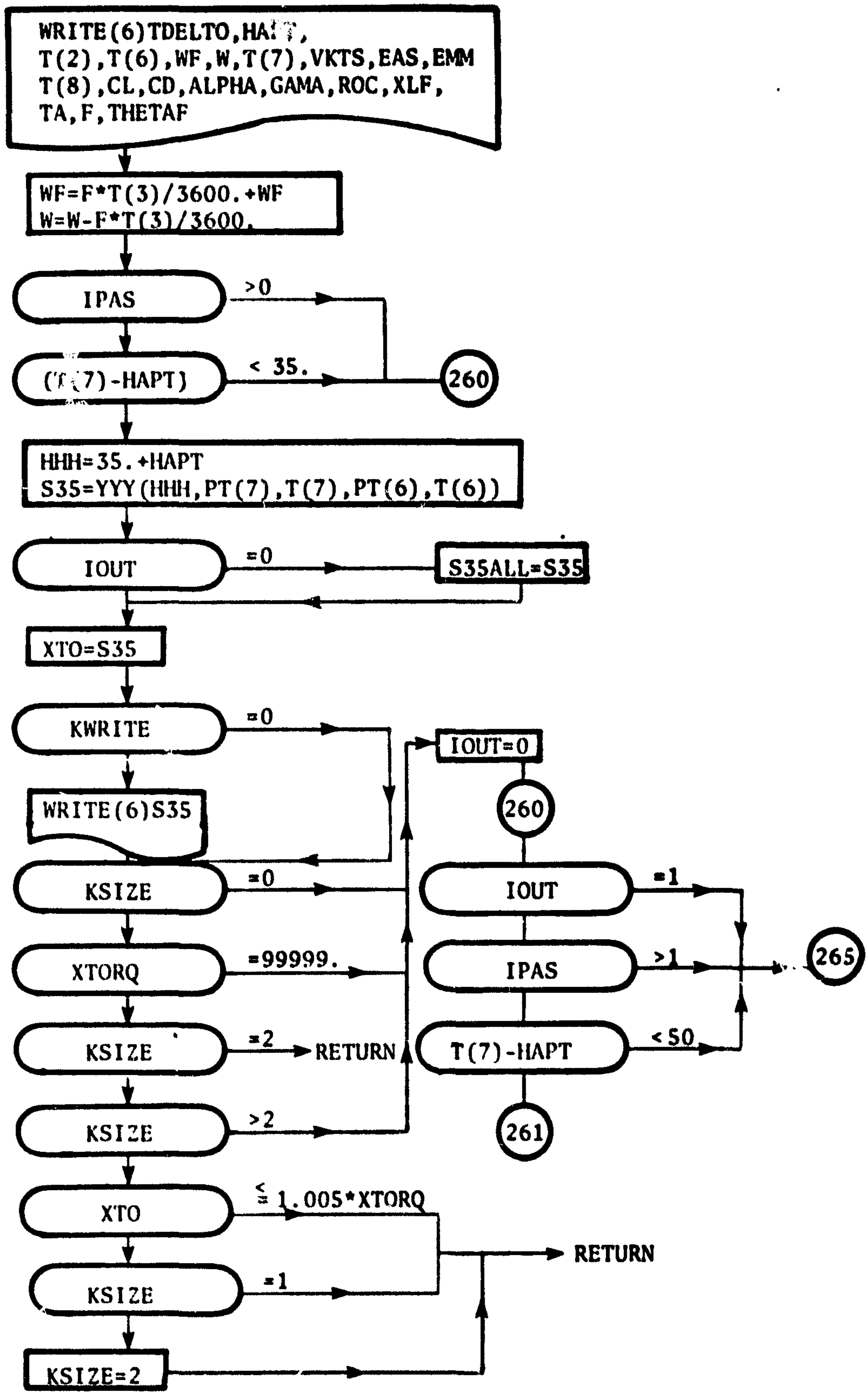

M=M+1
IM
=0
230
MU
≠0
30
IG
=1
20
T(7)-HAPT
< HGR
HHH=HGR+HAPT
TG=YYY(HHH,PT(7),T(7),PT(2),T(2))
TGU=TG+DTGR
KWRITE
=0
GRCD
≠0
WRITE(6)
TG,TGU
IG=1
IFL
=1
(T(7)-HAPT)
< HFLP
30
HHH=HFLP+HAPT
TFLP=YYY(HHH,PT(7),T(7),PT(2),T(2))
TFLPU=TFLP+DTFLP
KWRITE
=0
DFLPTO
≠0
WRITE(6)
TFLP,TFLPU
IFL=1
IOUT
=0
IM
=2
EAS
< V1
75

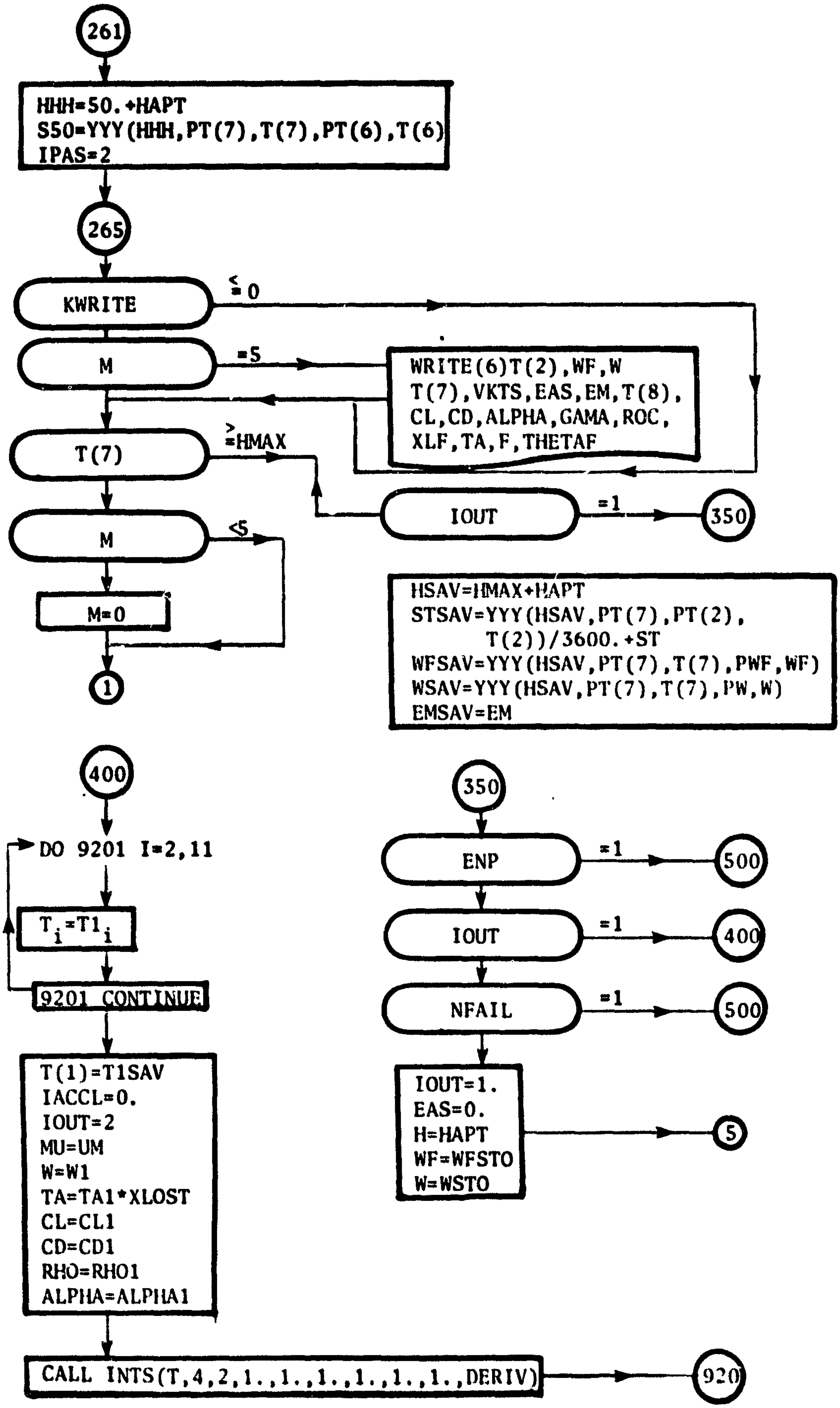

TIM1=YYY(V1,PEAS,EAS,PT(2),T(2))
VVV=YYY(V1,PEAS,EAS,PVKTS,VKTS)
WRITE(6)
TIM1,VW,V1
W1=YYY(V1,PEAS,EAS,PW,W)
WF1=YYY(V1,PEAS,EAS,PWF,WF)
TA1=YYY(V1,PEAS,EAS,PTA,TA)
ALPHA1=YYY(V1,PEAS,EAS,PALPHA,ALPHA)
CD1=YYY(V1,PEAS,EAS,PCD,CD)
CL1=YYY(V1,PEAS,EAS,PCL,CL)
RHO1=YYY(V1,PEAS,EAS,PRHO,RHO)
DO 70 I=2,11
$T1_i$=YYY(V1,PEAS,EAS,PT_i,T_i)
70 CONTINUE
ENP
>1
XLOST=(ENP-1.)/ENP
HMAX=50.
IM=2
MU
=0
225
EAS
< VR
100
IR
=1
80
TIMR=YYY(VR,PEAS,EAS,PT(2),T(2))
VVV=YYY(VR,PEAS,EAS,PVKTS,VKTS)
KWRITE
=0
CLR=YYY(VR,PEAS,EAS,PCL,CL)
CDR=YYY(VR,PEAS,EAS,PCD,CD)
TIMEND=TIMR+DELTVR
WRITE(6)TIMR
VVV,VR

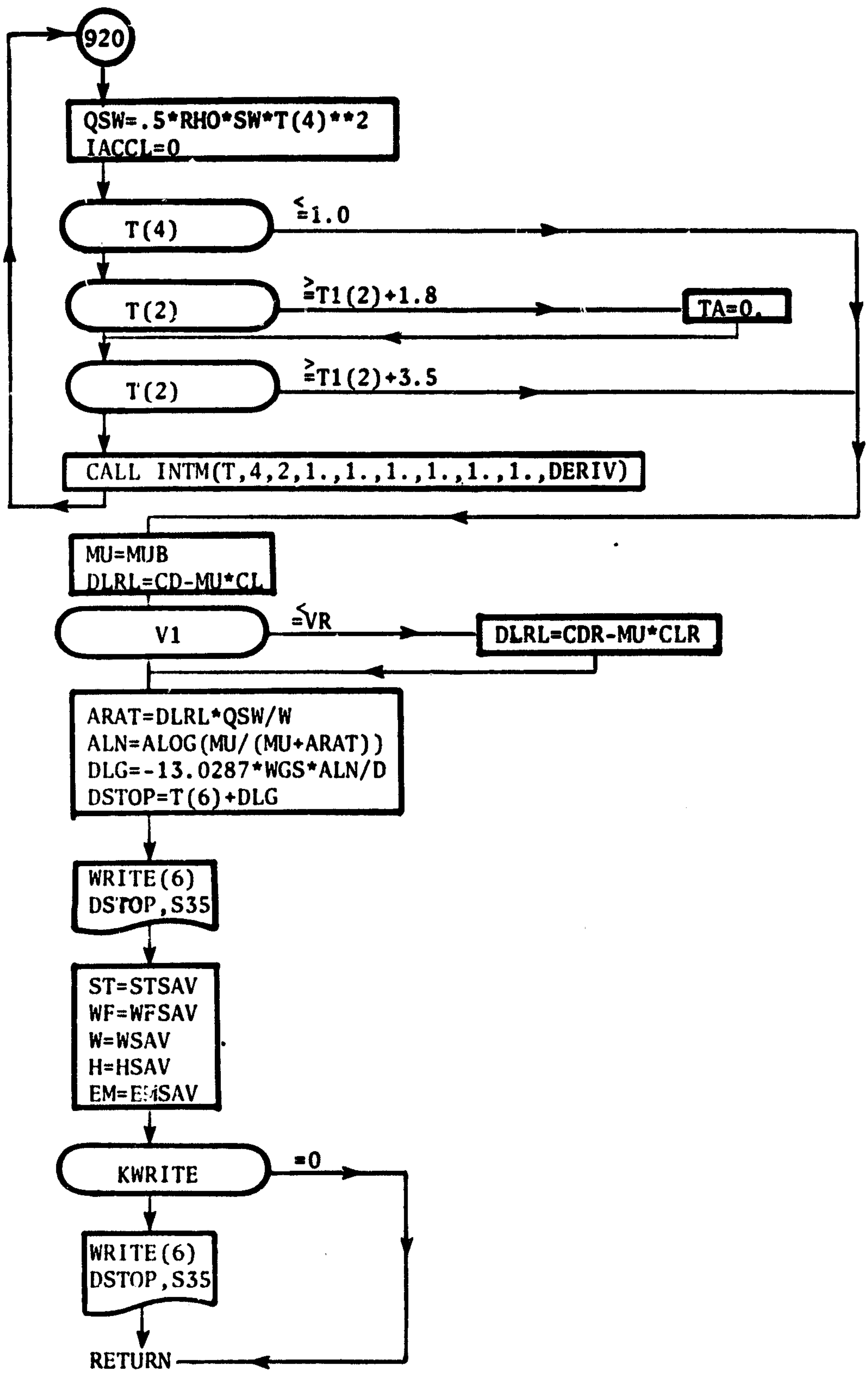

T(2)
≥TIMEND+2.
CL=((CLTO-CLR)/DELTVR)*(T(2)-TIMR)+CLR
T(2)
≥TIMEND
CL=CLTO
CALL CLIFT(5,ALPHA,CL,ALPHLO,CLALPH,HOB,AR,DLMC4,EM)
100
ALPHAR=(ALPHA-EYEW)*.01745
CL*QSW+TA*SIN(ALPHAR)
<(W*COS(T(5)))
MU=0.
VTOKTS=T(4)*.592087
KWRITE
=0
WRITE(6)T(2),
T(6),VTOKTS,EAS
DO 227 I=1,11
PT_i=T_i
227 CONTINUE
PWF=WF
PW=W
PEAS=EAS
PVKTS=VKTS
PCL=CL
PCD=CD
PTA=TA
PALPHA=ALPHA
PRHO=RHO
MU
≠0
T(2)
≥100
WRITE(6)
RETURN
CALL INTM(T,4,2,1.,1.,1.,1.,1.,1.,DERIV)
230
CALL TPALT(T(7),GALT,PO,FKALT,TOG,XKV)

C-2


TAMB=TO+TDELT
PO=PO*144.
SA=49.1*SQRT(TAMB)
RHO=PO/53.32/TAMB/G
RELI=T(4)/XKV
PAMB=PO
HOB=(T(7)+HTG-HAPT)/B
EM=T(4)/SA
VKTS=T(4)*.592087
EAS=VKTS*SQRT(RHO/.0023769
QSW=.5*RHO*SW*T(4)**2
KENG=5
CALL ENGINE(TA,TSFC,SFN,F,FAR,PAMB,TAMB,EM,T(7),KENG)
TA=ENP*TA
F=TA*TSFC
IM
=2
TA=XLOST*TA
IM
=2
F=XLOST*F
ROC=T(11)*60
GAMA=T(5)*57.29578
T(8)=(32.1b/w)*(TACOS(ALPHAR)-QSW*CD-MU
*(W-CL*QSW-TA*SIN(ALPHAR)-W*SIN(T(5)))
IACCL
=2
T(8)=0.
THETAF=T(5)*57.29578-EYEW+ALPHA
IM
≠0
240
T(1)=T1SAV
CALL INTS(T,4,2 1.,1.,1.,1.,1.,1.,DERIV)
IM=1
KWRITE
≤0
225


WRITE(6)TDELTO,HAPT,
T(2),T(6),WF,W,T(7),VKTS,EAS,EMM
T(8),CL,CD,ALPHA,GAMA,ROC,XLF,
TA,F,THETAF
WF=F*T(3)/3600.+WF
W=W-F*T(3)/3600.
IPAS
>0
(T(7)-HAPT)
< 35.
260
HHH=35.+HAPT
S35=YYY(HHH,PT(7),T(7),PT(6),T(6))
IOUT
=0
S35ALL=S35
XTO=S35
KWRITE
=0
IOUT=0
WRITE(6)S35
260
KSIZE
=0
IOUT
=1
XTORQ
=99999.
IPAS
>1
265
KSIZE
=2
RETURN
T(7)-HAPT
<50
KSIZE
>2
261
XTO
≤ 1.005*XTORQ
RETURN
KSIZE
=1
KSIZE=2


261
HHH=50.+HAPT
S50=YYY(HHH,PT(7),T(7),PT(6),T(6))
IPAS=2
265
KWRITE
≤0
M
=5
WRITE(6)T(2),WF,W
T(7),VKTS,EAS,EM,T(8),
CL,CD,ALPHA,GAMA,ROC,
XLF,TA,F,THETAF
T(7)
≥HMAX
IOUT
=1
350
M
<5
M=0
1
HSAV=HMAX+HAPT
STSAV=YYY(HSAV,PT(7),PT(2),
T(2))/3600.+ST
WFSAV=YYY(HSAV,PT(7),T(7),PWF,WF)
WSAV=YYY(HSAV,PT(7),T(7),PW,W)
EMSAV=EM
400
DO 9201 I=2,11
T_i=$T1_i$
9201 CONTINUE
T(1)=T1SAV
IACCL=0.
IOUT=2
MU=UM
W=W1
TA=TA1*XLOST
CL=CL1
CD=CD1
RHO=RHO1
ALPHA=ALPHA1
CALL INTS(T,4,2,1.,1.,1.,1.,1.,1.,DERIV)
920
350
ENP
=1
500
IOUT
=1
400
NFAIL
=1
500
IOUT=1.
EAS=0.
H=HAPT
WF=WFSTO
W=WSTO
5


920
QSW=.5*RHO*SW*T(4)**2
IACCL=0
T(4)
≤1.0
T(2)
≥T1(2)+1.8
TA=0.
T(2)
≥T1(2)+3.5
CALL INTM(T,4,2,1.,1.,1.,1.,1.,1.,DERIV)
MU=MUB
DLRL=CD-MU*CL
V1
≤VR
DLRL=CDR-MU*CLR
ARAT=DLRL*QSW/W
ALN=ALOG(MU/(MU+ARAT))
DLG=-13.0287*WGS*ALN/D
DSTOP=T(6)+DLG
WRITE(6)
DSTOP,S35
ST=STSAV
WF=WFSAV
W=WSAV
H=HSAV
EM=EMSAV
KWRITE
=0
WRITE(6)
DSTOP,S35
RETURN

VI.3.6 <u>Subroutine DERIV, Time Derivative Calculation</u>

Subroutine DERIV computes the instantaneous flight path and velocity derivatives required by subroutine TAKOFF. Routines CLIFT and DRAG provide aerodynamic characteristics. Propulsion system characteristics are computed from ENGINE.

DERIV

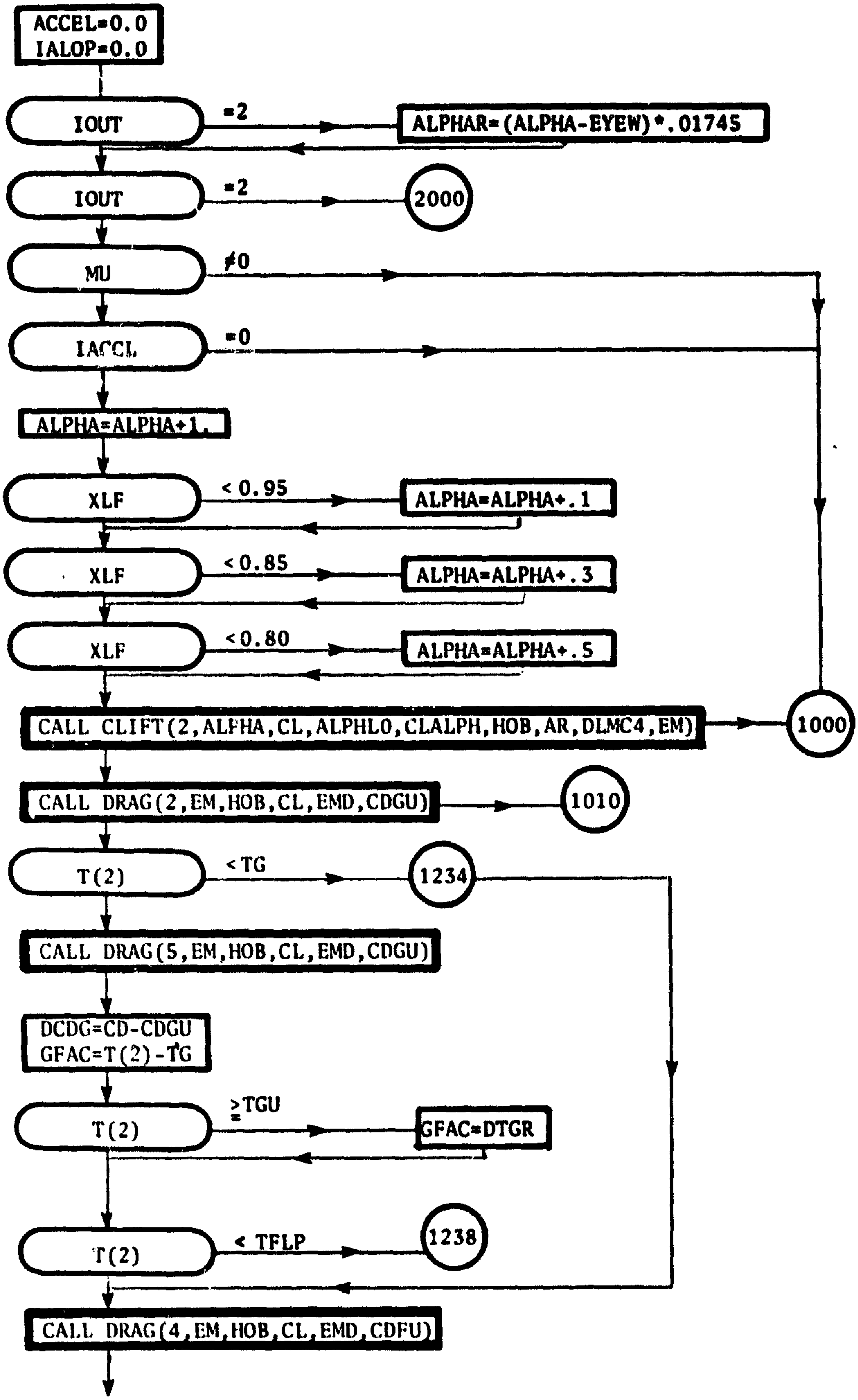
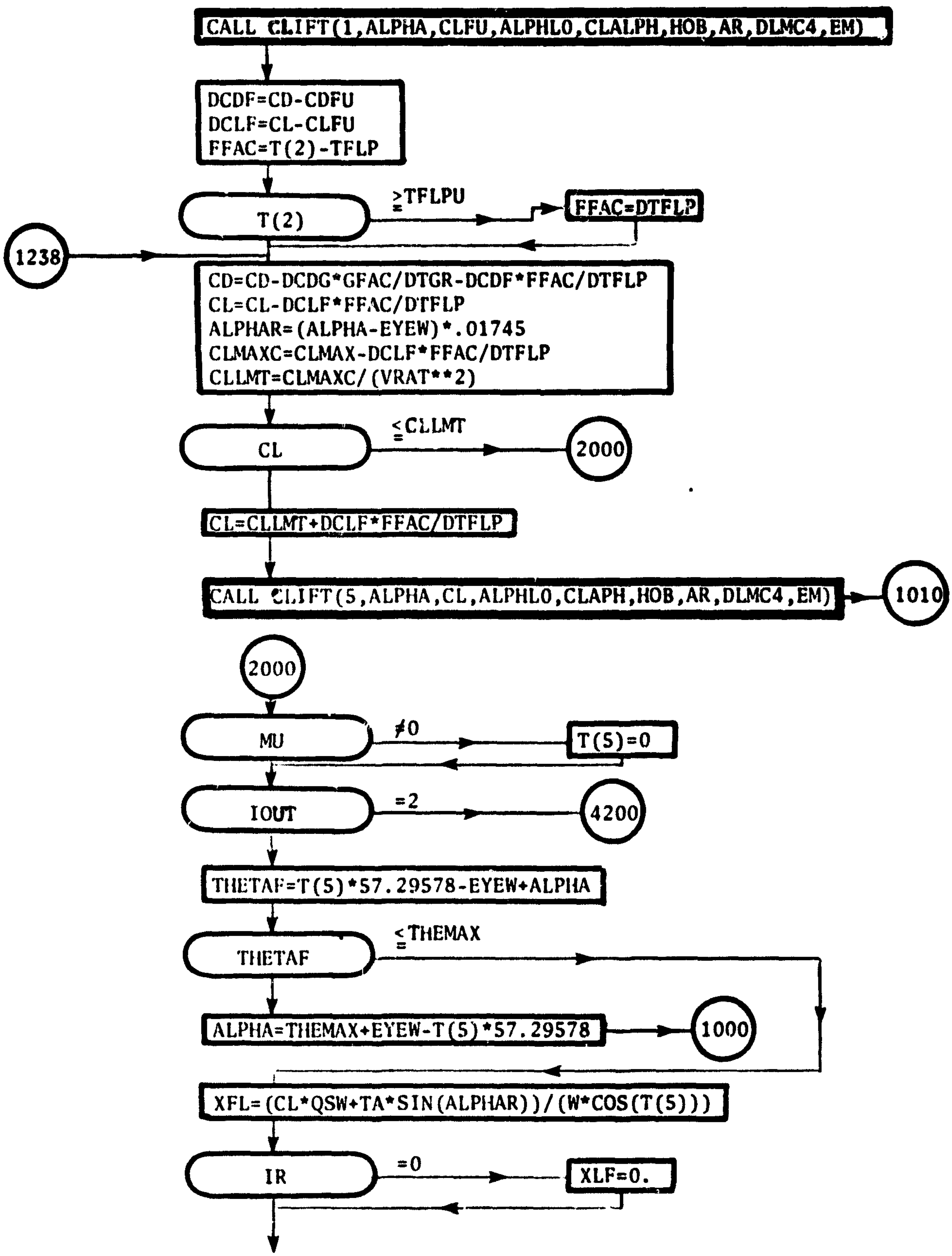
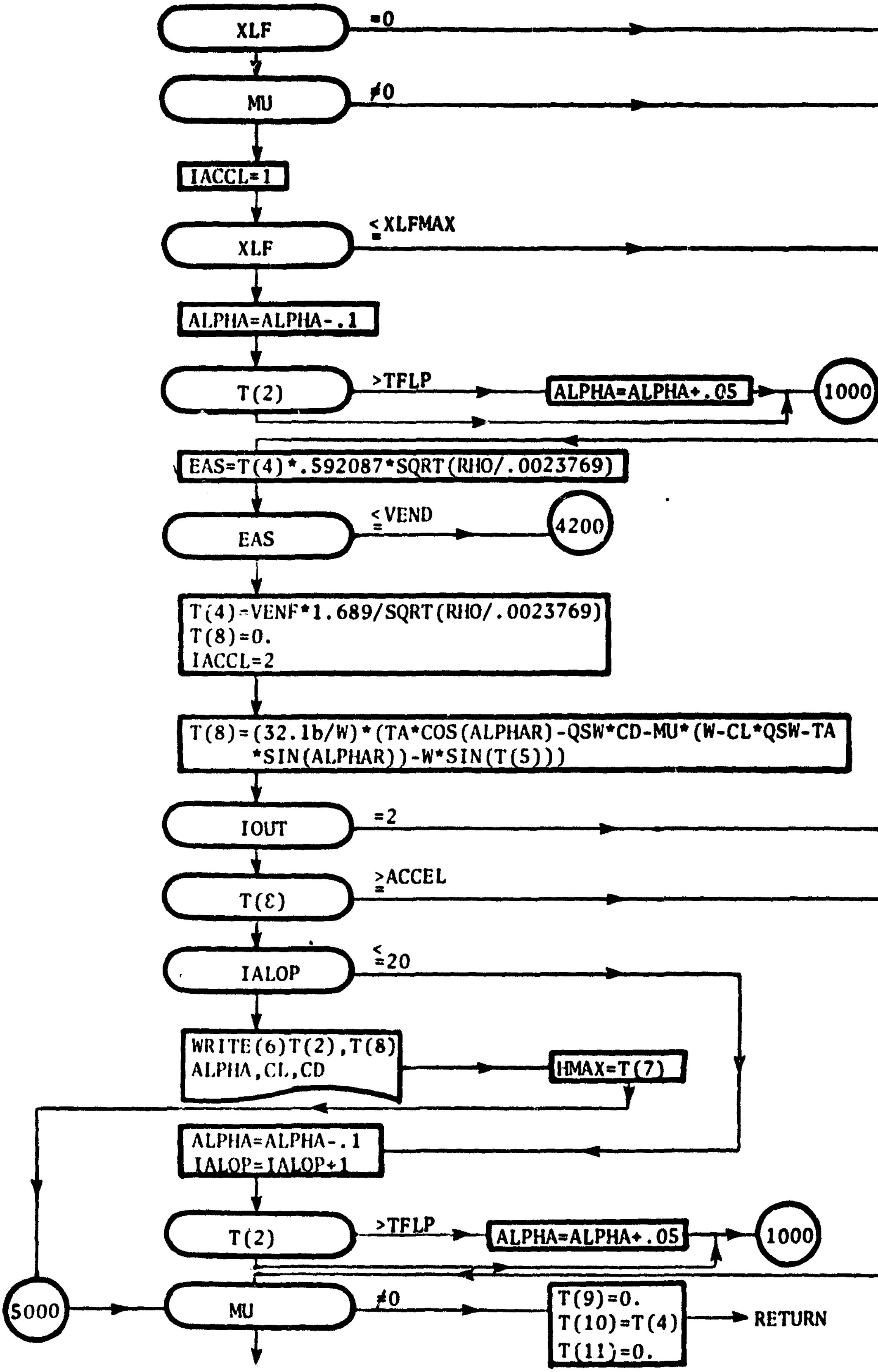


Figure VI.3.6 Subroutine DERIV, Detailed Flow Chart


CALL CLIFT(1,ALPHA,CLFU,ALPHLO,CLALPH,HOB,AR,DLMC4,EM)
DCDF=CD-CDFU
DCLF=CL-CLFU
FFAC=T(2)-TFLP
T(2)
≥TFLPU
FFAC=DTFLP
1238
CD=CD-DCDG*GFAC/DTGR-DCDF*FFAC/DTFLP
CL=CL-DCLF*FFAC/DTFLP
ALPHAR=(ALPHA-EYEW)*.01745
CLMAXC=CLMAX-DCLF*FFAC/DTFLP
CLLMT=CLMAXC/(VRAT**2)
CL
≤CLLMT
2000
CL=CLLMT+DCLF*FFAC/DTFLP
CALL CLIFT(5,ALPHA,CL,ALPHLO,CLAPH,HOB,AR,DLMC4,EM)
1010
2000
MU
≠0
T(5)=0
IOUT
=2
4200
THETAF=T(5)*57.29578-EYEW+ALPHA
THETAF
≤THEMAX
ALPHA=THEMAX+EYEW-T(5)*57.29578
1000
XFL=(CL*QSW+TA*SIN(ALPHAR))/(W*COS(T(5)))
IR
=0
XLF=0.


XLF
=0
MU
≠0
IACCL=1
XLF
≤XLFMAX
ALPHA=ALPHA-.1
T(2)
>TFLP
ALPHA=ALPHA+.05
1000
EAS=T(4)*.592087*SQRT(RHO/.0023769)
EAS
≤VEND
4200
T(4)=VENF*1.689/SQRT(RHO/.0023769)
T(8)=0.
IACCL=2
T(8)=(32.1b/W)*(TA*COS(ALPHAR)-QSW*CD-MU*(W-CL*QSW-TA
*SIN(ALPHAR))-W*SIN(T(5)))
IOUT
=2
T(8)
≥ACCEL
IALOP
≤20
WRITE(6)T(2),T(8)
ALPHA,CL,CD
HMAX=T(7)
ALPHA=ALPHA-.1
IALOP=IALOP+1
T(2)
>TFLP
ALPHA=ALPHA+.05
1000
5000
MU
≠0
T(9)=0.
T(10)=T(4)
T(11)=0.
RETURN

```
T(9)=(32.1b/W*T(4)))*(CL*QSW+TA*SIN(ALPHAR)-W**COS(T(5)))
T(10)=T(4)*COS(T(5))
T(11)=T(4)*SIN(T(5))
```

RETURN

VI.3.7 Subroutine TAXI, Taxi Calculations

Subroutine TAXI computes taxi fuel requirements. Subroutine TPALT supplies atmospheric conditions at the runway altitude. Subroutine ENGINE supplies propulsion system characteristics during ground idle.

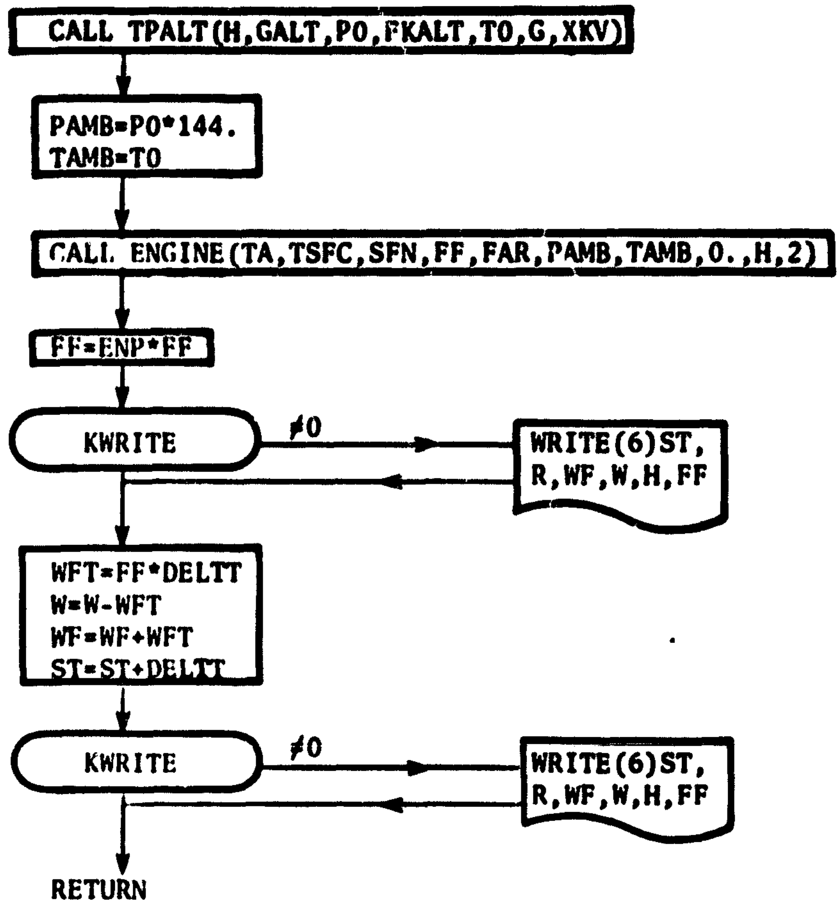


Figure VI.3.7 Subroutine TAXI, Detailed Flow Chart

VI.3.8 Subroutine TURN, Turning Performance

Subroutine TURN computes steady state turn maneuvers. The routine TPALT provides atmospheric properties during the turning maneuver. Drag coefficient is obtained from DRAG and propulsion system characteristics from ENGINE.

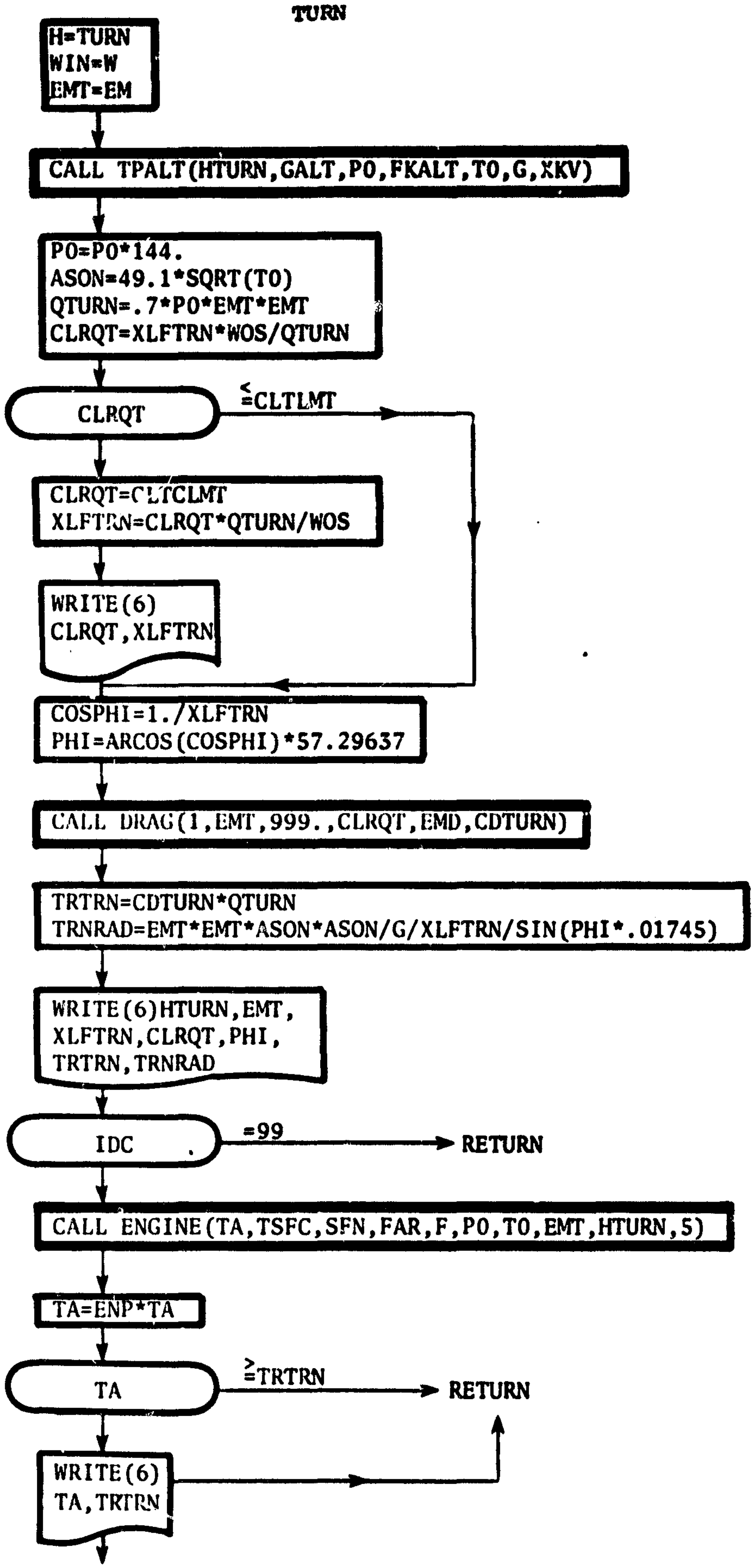


Figure VI.3.8 Subroutine TURN, Detailed Flow Chart

VI.3.9 Subroutine XRANGE, Cruise Range Calculation

Subroutine XRANGE computes cruise performance in GASP. Calculations are based on the Brequet equation. Atmospheric properties are computed by TPALT. Maximum speed is obtained from ASPEED. Cruise aerodynamics are computed by CTAER and propulsion characteristics from ENGINE. Accelerating flight paths are computed by ACCEL.

XRANGE

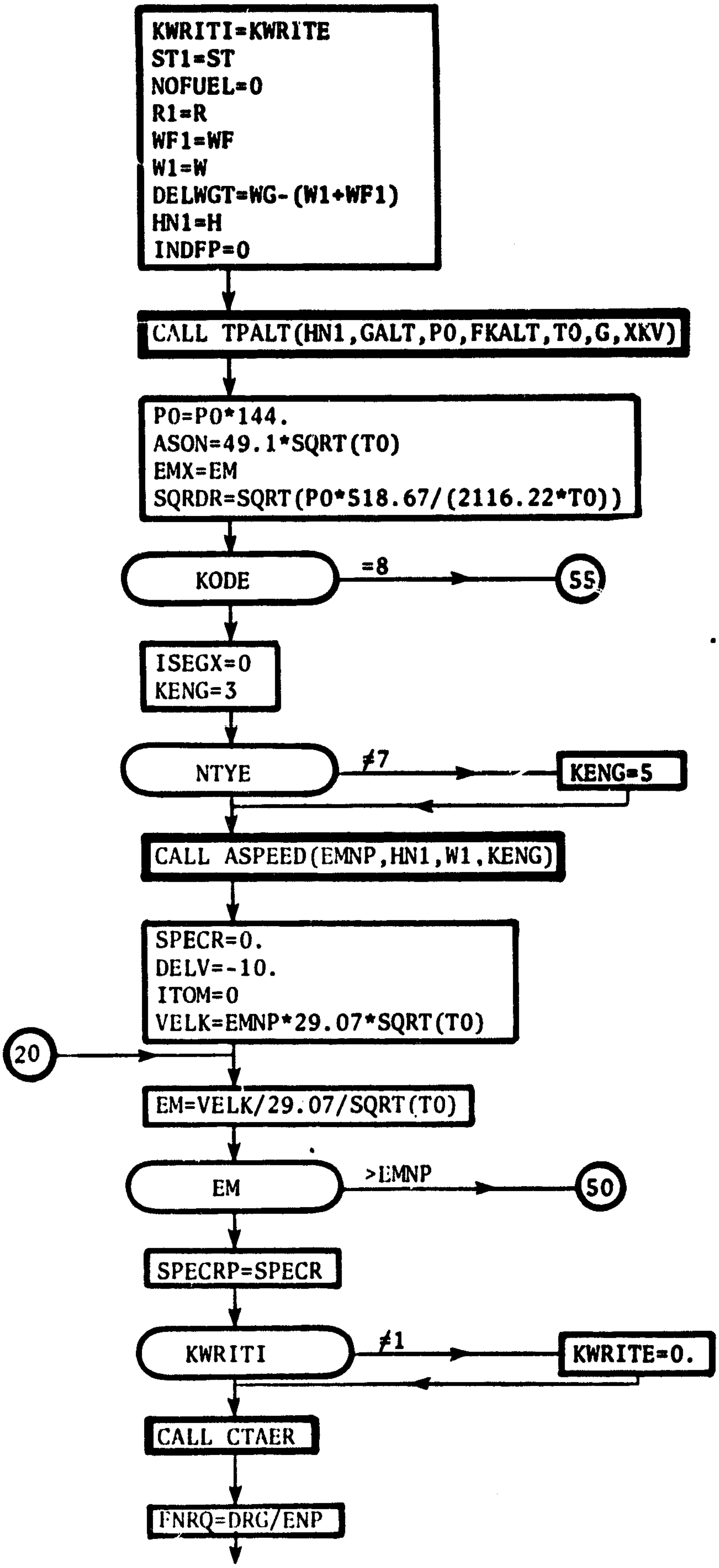
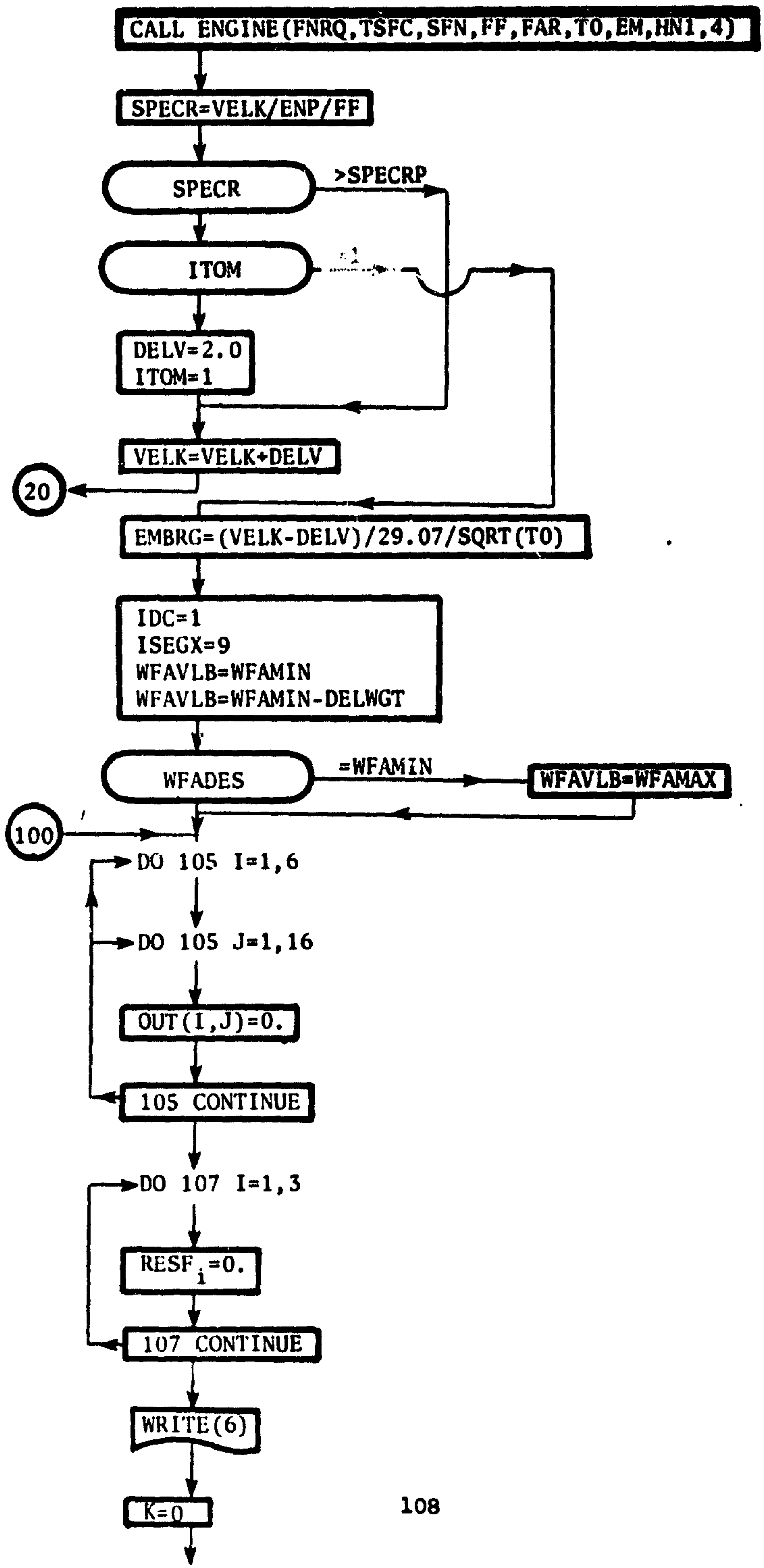
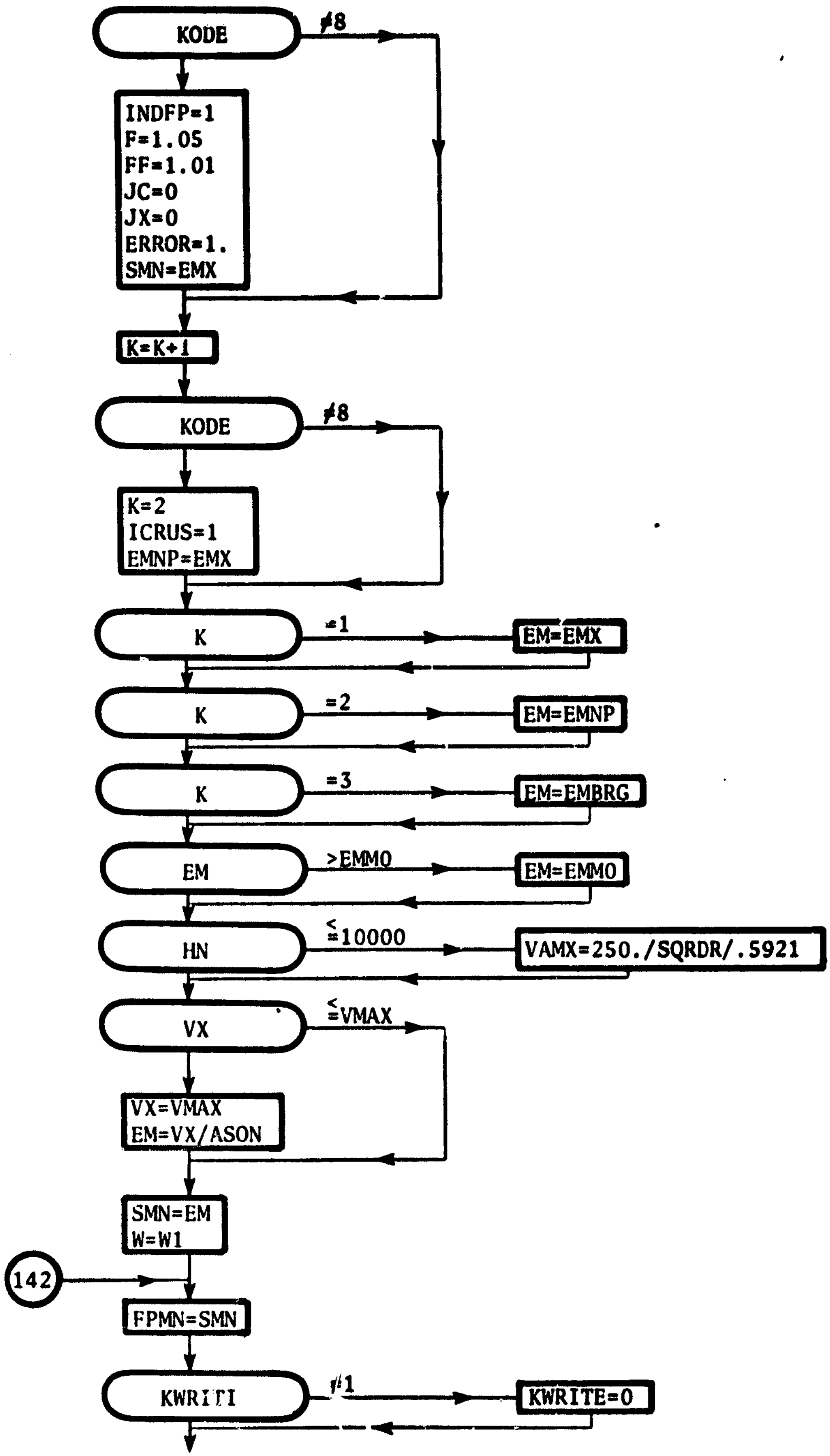
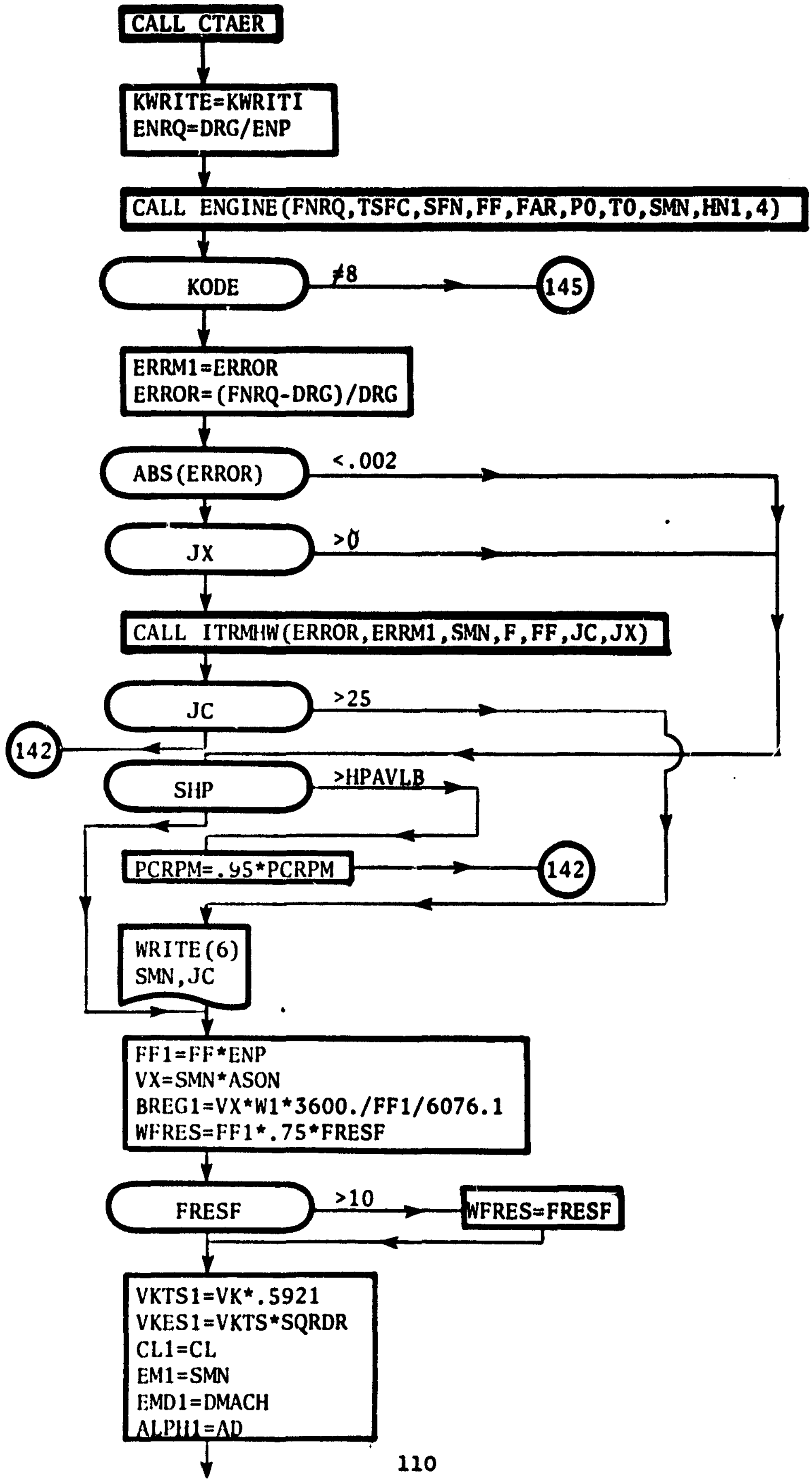
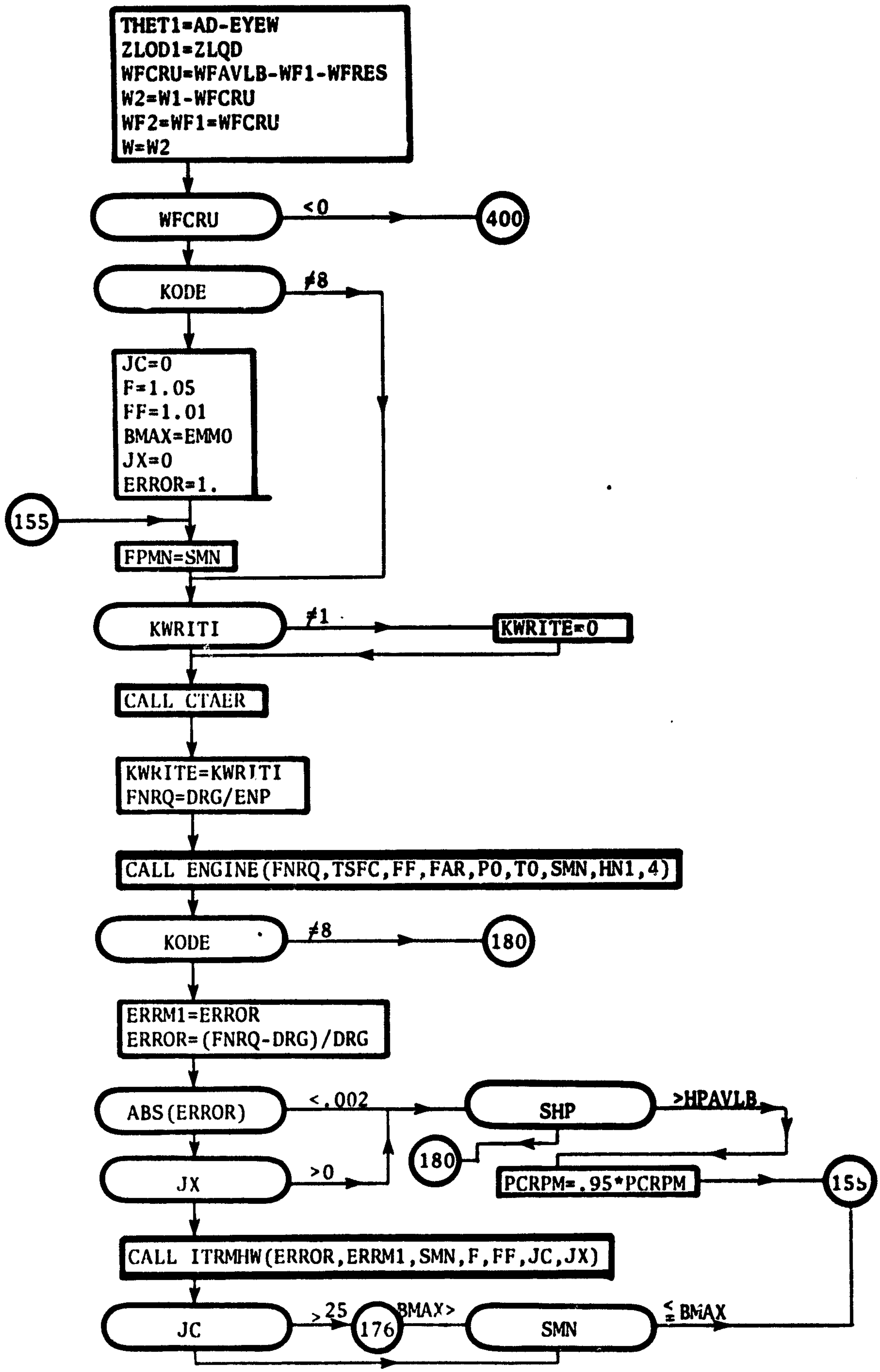


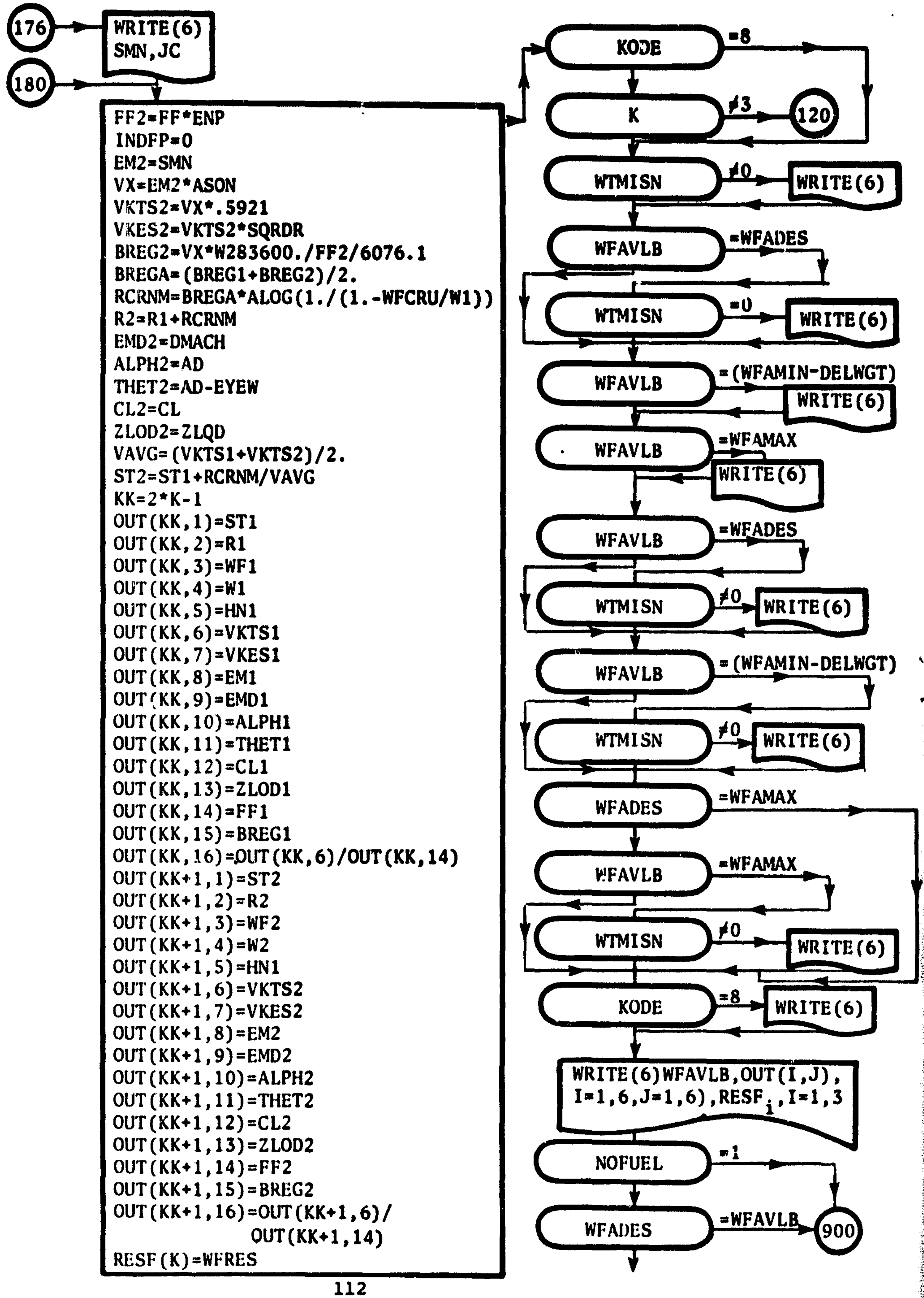
Figure VI.3.9 Subroutine XRANGE, Detailed Flow Chart

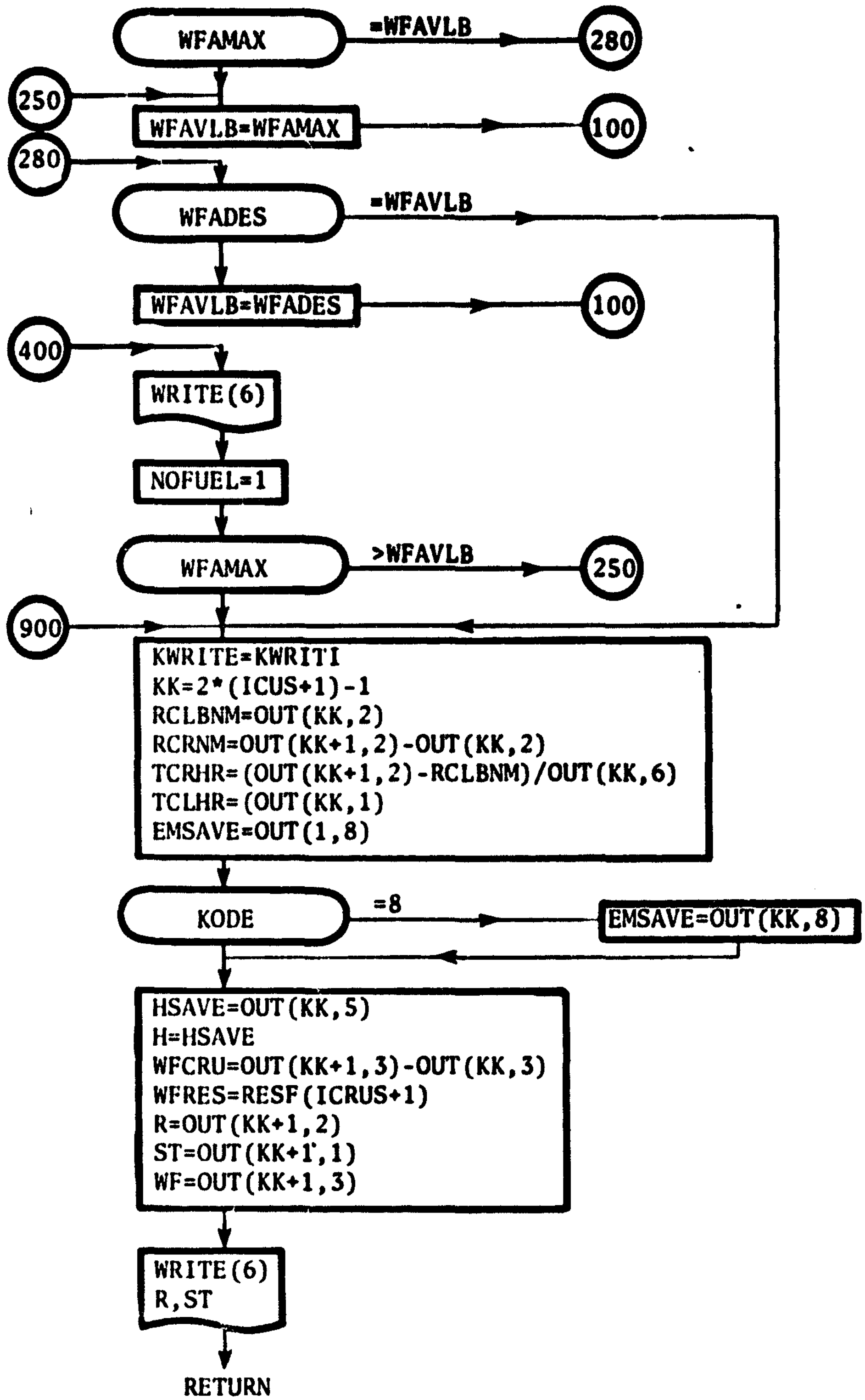

CALL ENGINE(FNRQ,TSFC,SFN,FF,FAR,TO,EM,HN1,4)
SPECR=VELK/ENP/FF
SPECR
>SPECRP
ITOM
DELV=2.0
ITOM=1
VELK=VELK+DELV
20
EMBRG=(VELK-DELV)/29.07/SQRT(TO)
IDC=1
ISEGX=9
WFAVLB=WFAMIN
WFAVLB=WFAMIN-DELWGT
WFADES
=WFAMIN
WFAVLB=WFAMAX
100
DO 105 I=1,6
DO 105 J=1,16
OUT(I,J)=0.
105 CONTINUE
DO 107 I=1,3
RESF$_i$=0.
107 CONTINUE
WRITE(6)
K=0


KODE
≠8
INDFP=1
F=1.05
FF=1.01
JC=0
JX=0
ERROR=1.
SMN=EMX
K=K+1
KODE
≠8
K=2
ICRUS=1
EMNP=EMX
K
=1
EM=EMX
K
=2
EM=EMNP
K
=3
EM=EMBRG
EM
>EMMO
EM=EMMO
HN
≤10000
VAMX=250./SQRDR/.5921
VX
≤VMAX
VX=VMAX
EM=VX/ASON
SMN=EM
W=W1
142
FPMN=SMN
KWRITI
≠1
KWRITE=0


CALL CTAER
KWRITE=KWRITI
ENRQ=DRG/ENP
CALL ENGINE(FNRQ,TSFC,SFN,FF,FAR,P0,T0,SMN,HN1,4)
KODE
≠8
145
ERRM1=ERROR
ERROR=(FNRQ-DRG)/DRG
ABS(ERROR)
<.002
JX
>0
CALL ITRMHW(ERROR,ERRM1,SMN,F,FF,JC,JX)
JC
>25
142
SHP
>HPAVLB
PCRPM=.95*PCRPM
142
WRITE(6)
SMN,JC
FF1=FF*ENP
VX=SMN*ASON
BREG1=VX*W1*3600./FF1/6076.1
WFRES=FF1*.75*FRESF
FRESF
>10
WFRES=FRESF
VKTS1=VK*.5921
VKES1=VKTS*SQRDR
CL1=CL
EM1=SMN
EMD1=DMACH
ALPH1=AD


THET1=AD-EYEW
ZLOD1=ZLQD
WFCRU=WFAVLB-WF1-WFRES
W2=W1-WFCRU
WF2=WF1=WFCRU
W=W2
WFCRU
<0
400
KODE
≠8
JC=0
F=1.05
FF=1.01
BMAX=EMMO
JX=0
ERROR=1.
155
FPMN=SMN
KWRITI
≠1
KWRITE=0
CALL CTAER
KWRITE=KWRITI
FNRQ=DRG/ENP
CALL ENGINE(FNRQ,TSFC,FF,FAR,PO,TO,SMN,HN1,4)
KODE
≠8
180
ERRM1=ERROR
ERROR=(FNRQ-DRG)/DRG
ABS(ERROR)
<.002
SHP
>HPAVLB
180
JX
>0
PCRPM=.95*PCRPM
155
CALL ITRMHW(ERROR,ERRM1,SMN,F,FF,JC,JX)
JC
>25
176
BMAX>
SMN
≤ BMAX


176
WRITE(6) SMN,JC
180
FF2=FF*ENP
INDFP=0
EM2=SMN
VX=EM2*ASON
VKTS2=VX*.5921
VKES2=VKTS2*SQRDR
BREG2=VX*W283600./FF2/6076.1
BREGA=(BREG1+BREG2)/2.
RCRNM=BREGA*ALOG(1./(1.-WFCRU/W1))
R2=R1+RCRNM
EMD2=DMACH
ALPH2=AD
THET2=AD-EYEW
CL2=CL
ZLOD2=ZLQD
VAVG=(VKTS1+VKTS2)/2.
ST2=ST1+RCRNM/VAVG
KK=2*K-1
OUT(KK,1)=ST1
OUT(KK,2)=R1
OUT(KK,3)=WF1
OUT(KK,4)=W1
OUT(KK,5)=HN1
OUT(KK,6)=VKTS1
OUT(KK,7)=VKES1
OUT(KK,8)=EM1
OUT(KK,9)=EMD1
OUT(KK,10)=ALPH1
OUT(KK,11)=THET1
OUT(KK,12)=CL1
OUT(KK,13)=ZLOD1
OUT(KK,14)=FF1
OUT(KK,15)=BREG1
OUT(KK,16)=OUT(KK,6)/OUT(KK,14)
OUT(KK+1,1)=ST2
OUT(KK+1,2)=R2
OUT(KK+1,3)=WF2
OUT(KK+1,4)=W2
OUT(KK+1,5)=HN1
OUT(KK+1,6)=VKTS2
OUT(KK+1,7)=VKES2
OUT(KK+1,8)=EM2
OUT(KK+1,9)=EMD2
OUT(KK+1,10)=ALPH2
OUT(KK+1,11)=THET2
OUT(KK+1,12)=CL2
OUT(KK+1,13)=ZLOD2
OUT(KK+1,14)=FF2
OUT(KK+1,15)=BREG2
OUT(KK+1,16)=OUT(KK+1,6)/
OUT(KK+1,14)
RESF(K)=WFRES
KODE
=8
K
≠3
120
WTMISN
≠0
WRITE(6)
WFAVLB
=WFADES
WTMISN
=0
WRITE(6)
WFAVLB
=(WFAMIN-DELWGT)
WRITE(6)
WFAVLB
=WFAMAX
WRITE(6)
WFAVLB
=WFADES
WTMISN
≠0
WRITE(6)
WFAVLB
=(WFAMIN-DELWGT)
WTMISN
≠0
WRITE(6)
WFADES
=WFAMAX
WFAVLB
=WFAMAX
WTMISN
≠0
WRITE(6)
KODE
=8
WRITE(6)
WRITE(6)WFAVLB,OUT(I,J),
I=1,6,J=1,6),RESF$_i$,I=1,3
NOFUEL
=1
WFADES
=WFAVLB
900


WFAMAX
=WFAVLB
280
250
WFAVLB=WFAMAX
100
280
WFADES
=WFAVLB
WFAVLB=WFADES
100
400
WRITE(6)
NOFUEL=1
WFAMAX
>WFAVLB
250
900
KWRITE=KWRITI
KK=2*(ICUS+1)-1
RCLBNM=OUT(KK,2)
RCRNM=OUT(KK+1,2)-OUT(KK,2)
TCRHR=(OUT(KK+1,2)-RCLBNM)/OUT(KK,6)
TCLHR=(OUT(KK,1)
EMSAVE=OUT(1,8)
KODE
=8
EMSAVE=OUT(KK,8)
HSAVE=OUT(KK,5)
H=HSAVE
WFCRU=OUT(KK+1,3)-OUT(KK,3)
WFRES=RESF(ICRUS+1)
R=OUT(KK+1,2)
ST=OUT(KK+1,1)
WF=OUT(KK+1,3)
WRITE(6)
R,ST
RETURN

VI.3.10 Subroutine RGBAL Range Balancing Calculation

Subroutine RGBAL balances the range computed by XRANGE against a desired input range value by adjusting the take-off gross weight. The following subroutines are required in this resizing procedure:

SIZE--Aircraft Geometry

FLAPS, CTAER--Aerodynamics

ENGINE, ENGWT--Engine Performance Size

WGHT--Weights

PERFRM--Performance

XRANGE--Cruise Range

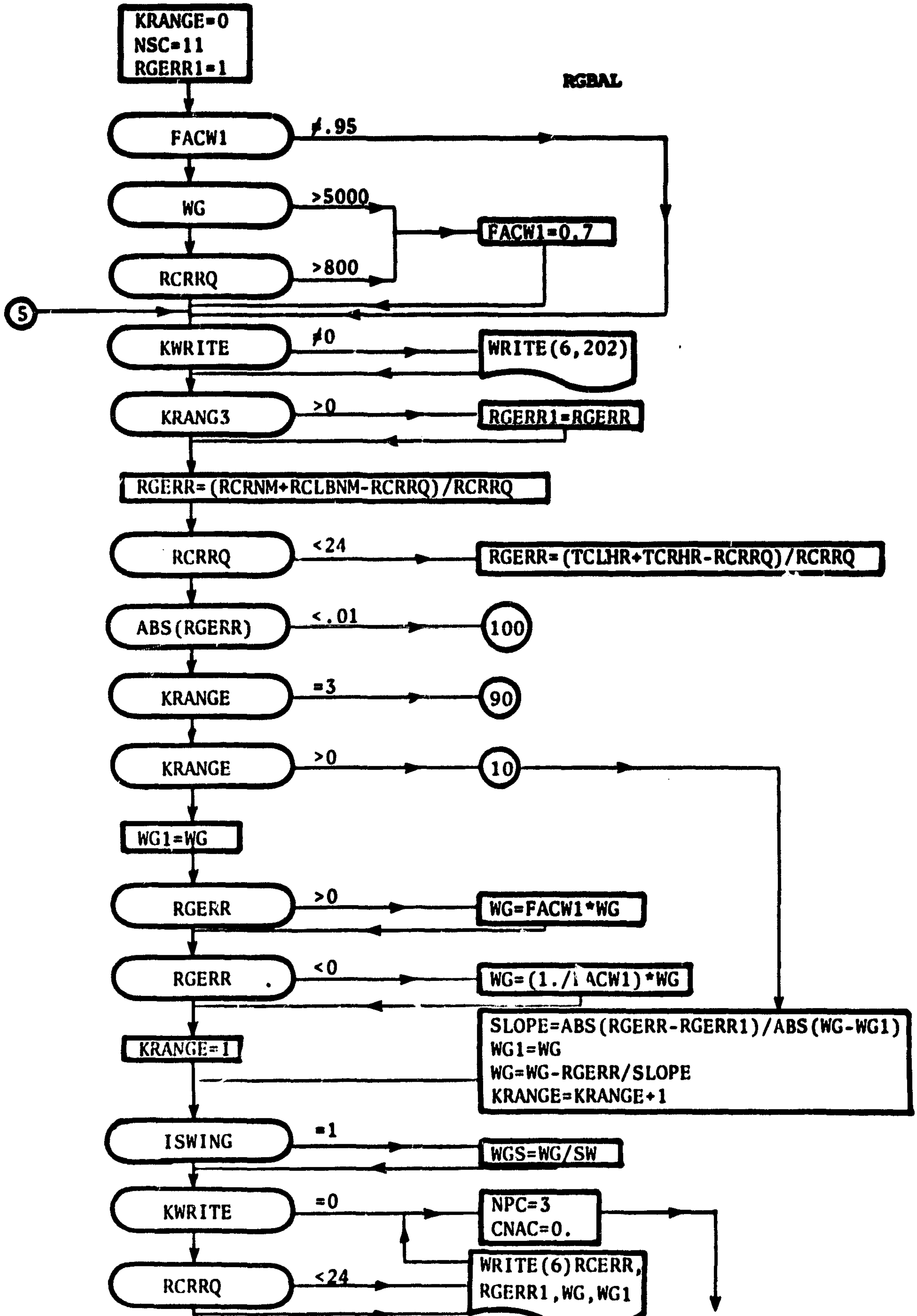
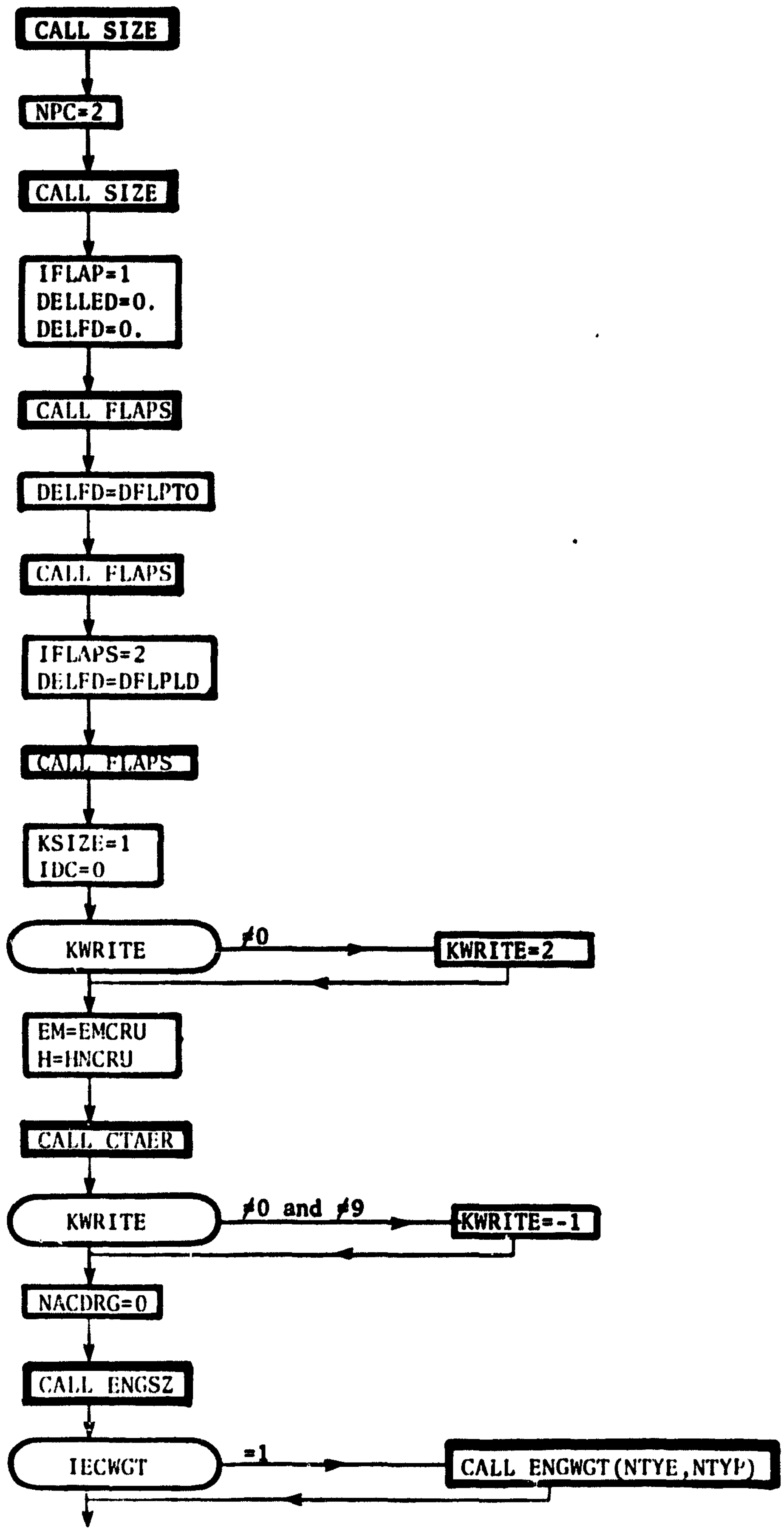
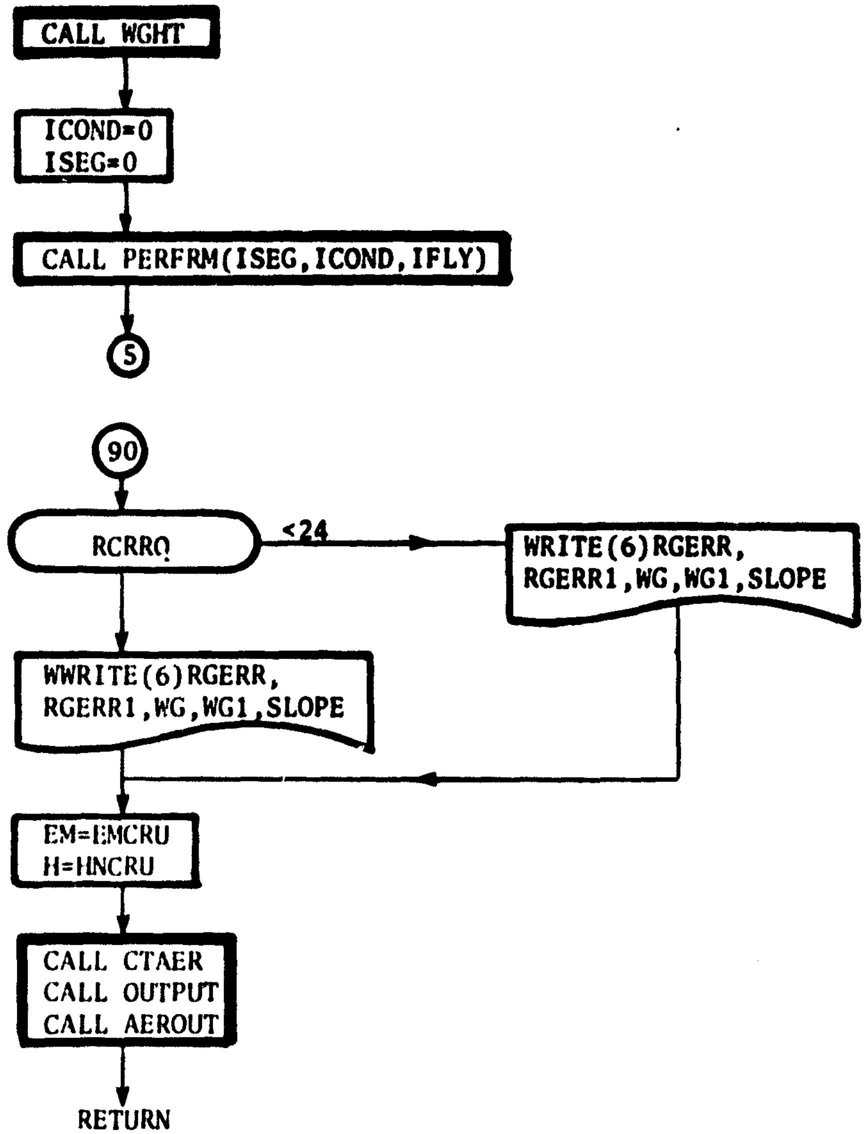


Figure VI.3.10 Subroutine RGBAL, Detailed Flow Chart

RGBAL


CALL SIZE
NPC=2
CALL SIZE
IFLAP=1
DELLED=0.
DELFD=0.
CALL FLAPS
DELFD=DFLPTO
CALL FLAPS
IFLAPS=2
DELFD=DFLPLD
CALL FLAPS
KSIZE=1
IDC=0
KWRITE
≠0
KWRITE=2
EM=EMCRU
H=HNCRU
CALL CTAER
KWRITE
≠0 and ≠9
KWRITE=-1
NACDRG=0
CALL ENGSZ
IECWGT
=1
CALL ENGWGT(NTYE,NTYP)


CALL WGHT
ICOND=0
ISEG=0
CALL PERFRM(ISEG,ICOND,IFLY)
5
90
RCRRO
<24
WRITE(6)RGERR,
RGERR1,WG,WG1,SLOPE
WWRITE(6)RGERR,
RGERR1,WG,WG1,SLOPE
EM=EMCRU
H=HNCRU
CALL CTAER
CALL OUTPUT
CALL AEROUT
RETURN