HELICOPTER PERFORMANCE COMPUTER PROGRAMS

DAVID KOBUS AND STEVEN WOODS
Aircraft and Crew Systems Technology Directorate
Naval Air Development Center
Warminster, PA 18974

30 OCTOBER 1982

FINAL REPORT
AIRTASK NO. A511-5115E/001-D/2W1107-0000
WORK UNIT NO. HB320

Approved for Public Release; Distribution Unlimited

Prepared for
NAVAL AIR SYSTEMS COMMAND
Department of the Navy
Washington, DC 20361
NOTICES

REPORT NUMBERING SYSTEM - The numbering of technical project reports issued by the Naval Air Development Center is arranged for specific identification purposes. Each number consists of the Center acronym, the calendar year in which the number was assigned, the sequence number of the report within the specific calendar year, and the official 2-digit correspondence code of the Command Office or the Functional Directorate responsible for the report. For example: Report No. NADC-78015-20 indicates the fifteenth Center report for the year 1978, and prepared by the Systems Directorate. The numerical codes are as follows:

<table>
<thead>
<tr>
<th>CODE</th>
<th>OFFICE OR DIRECTORATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Commander, Naval Air Development Center</td>
</tr>
<tr>
<td>01</td>
<td>Technical Director, Naval Air Development Center</td>
</tr>
<tr>
<td>02</td>
<td>Comptroller</td>
</tr>
<tr>
<td>10</td>
<td>Directorate Command Projects</td>
</tr>
<tr>
<td>20</td>
<td>Systems Directorate</td>
</tr>
<tr>
<td>30</td>
<td>Sensors &amp; Avionics Technology Directorate</td>
</tr>
<tr>
<td>40</td>
<td>Communication &amp; Navigation Technology Directorate</td>
</tr>
<tr>
<td>50</td>
<td>Software Computer Directorate</td>
</tr>
<tr>
<td>60</td>
<td>Aircraft &amp; Crew Systems Technology Directorate</td>
</tr>
<tr>
<td>70</td>
<td>Planning Assessment Resources</td>
</tr>
<tr>
<td>80</td>
<td>Engineering Support Group</td>
</tr>
</tbody>
</table>

PRODUCT ENDORSEMENT - The discussion or instructions concerning commercial products herein do not constitute an endorsement by the Government nor do they convey or imply the license or right to use such products.

APPROVED BY: T. J. GALLAGHER
CAPT, MSC, USN

DATE: 20 Jan 1973
<table>
<thead>
<tr>
<th>Report Number</th>
<th>NADC-82136-60</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Helicopter Performance Computer Programs</td>
</tr>
<tr>
<td>Author(s)</td>
<td>David Kobus and Steven Woods</td>
</tr>
<tr>
<td>Performing Organization Name and Address</td>
<td>Naval Air Development Center Aircraft and Crew Systems Technology Directorate Warminster, PA 18974</td>
</tr>
<tr>
<td>Controlling Office Name and Address</td>
<td>Naval Air Systems Command Department of the Navy Washington, DC 20361</td>
</tr>
<tr>
<td>Period Covered</td>
<td>Final Report</td>
</tr>
<tr>
<td>Contract or Grant Number(s)</td>
<td>2M1107-0000</td>
</tr>
<tr>
<td>Program Element, Project, Task Area &amp; Work Unit Numbers</td>
<td>Airtask No. A511-5115E/001-D/ Work Unit No. HB320</td>
</tr>
<tr>
<td>Report Date</td>
<td>30 October 1982</td>
</tr>
<tr>
<td>Number of Pages</td>
<td>6</td>
</tr>
<tr>
<td>Distribution Statement (of this Report)</td>
<td>Approved for Public Release; distribution unlimited.</td>
</tr>
<tr>
<td>Key Words (Continue on reverse side if necessary and identify by block number)</td>
<td>Helicopter Performance Tactical Manual</td>
</tr>
<tr>
<td>Abstract (Continue on reverse side if necessary and identify by block number)</td>
<td>Helicopter performance computer programs have been developed by NAVAIRDEVCEN. Performance items considered are specific excess power, sustained and instantaneous turn rate, and maneuver capability. Performance output can be displayed graphically as well as numerically.</td>
</tr>
</tbody>
</table>
The Naval Air Development Center (NAVAIRDEVCEN) is supporting the Naval Air Systems Command (NAVAIR) (AIR-5301) in the development of computer aided tactical performance plots for inclusion in Navy and Marine Helo Tactical Manuals. As part of this effort, computer programs have been developed which are capable of analyzing the flight performance of typical helicopter configurations. Areas of flight performance include specific excess power, sustained and instantaneous turn rate, and maneuver capability.

These programs were written in FORTRAN and use three supporting subroutines. This report describes the analytical development and logic development for the programs. In addition, it includes a user description and complete listing.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summary</td>
<td>1</td>
</tr>
<tr>
<td>List of Figures</td>
<td>4</td>
</tr>
<tr>
<td>List of Tables</td>
<td>5</td>
</tr>
<tr>
<td>List of Symbols</td>
<td>6</td>
</tr>
<tr>
<td>Introduction</td>
<td>9</td>
</tr>
<tr>
<td>Discussion</td>
<td>10</td>
</tr>
<tr>
<td>Theoretical Development</td>
<td>10</td>
</tr>
<tr>
<td>Specific Excess Power</td>
<td>11</td>
</tr>
<tr>
<td>Sustained Turn Rate</td>
<td>16</td>
</tr>
<tr>
<td>Instantaneous Turn Rate</td>
<td>17</td>
</tr>
<tr>
<td>Maneuver Capability</td>
<td>18</td>
</tr>
<tr>
<td>Program Description</td>
<td>20</td>
</tr>
<tr>
<td>Computational Procedure</td>
<td>29</td>
</tr>
<tr>
<td>Acknowledgments</td>
<td>30</td>
</tr>
<tr>
<td>References</td>
<td>31</td>
</tr>
<tr>
<td>Appendix A: Example Cases</td>
<td>A-1</td>
</tr>
<tr>
<td>Appendix B: Program Listings</td>
<td>B-1</td>
</tr>
<tr>
<td>Appendix C: Program Logic</td>
<td>C-1</td>
</tr>
<tr>
<td>Figure No.</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>1</td>
<td>Data Input Deck Structure</td>
</tr>
<tr>
<td>2</td>
<td>Power Available Tabular Input</td>
</tr>
<tr>
<td>3</td>
<td>Torque Available Tabular Input</td>
</tr>
<tr>
<td>4</td>
<td>Power Required Coefficient Tabular Input, MUCTSW = 0</td>
</tr>
<tr>
<td>5</td>
<td>Power Required Coefficient Tabular Input, MUCTSW ≠ 0</td>
</tr>
<tr>
<td>6</td>
<td>Thrust Coefficient Tabular Input</td>
</tr>
<tr>
<td>A1</td>
<td>Sample SHF Input (Graphical)</td>
</tr>
<tr>
<td>A2</td>
<td>Sample C_p Input (Graphical)</td>
</tr>
<tr>
<td>A3</td>
<td>Sample C_T Input (Graphical)</td>
</tr>
<tr>
<td>A4</td>
<td>Sample P_g Tactical Manual Plot</td>
</tr>
<tr>
<td>A5</td>
<td>Sample TAPE6 P_g Crossplot</td>
</tr>
<tr>
<td>A6</td>
<td>Sample Sustained (\dot{\psi}) Tactical Manual Plot</td>
</tr>
<tr>
<td>A7</td>
<td>Sample TAPE6 Sustained (\dot{\psi}) Crossplot</td>
</tr>
<tr>
<td>A8</td>
<td>Sample Instantaneous Sustained (\dot{\psi}) Crossplot</td>
</tr>
<tr>
<td>A9</td>
<td>Sample TAPE6 Instantaneous (\dot{\psi}) Crossplot</td>
</tr>
<tr>
<td>A10</td>
<td>Sample Maneuver Tactical Manual Plot</td>
</tr>
<tr>
<td>A11</td>
<td>Sample TAPE6 Maneuver Crossplot</td>
</tr>
</tbody>
</table>
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table No.</th>
<th>Description</th>
<th>Page No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Scalar Input Variable List</td>
<td>28</td>
</tr>
<tr>
<td>AI</td>
<td>Sample Data Input</td>
<td>A-4</td>
</tr>
<tr>
<td>AII</td>
<td>Sample TAPE8 Matrix Output (Abbreviated)</td>
<td>A-8</td>
</tr>
<tr>
<td>AIII</td>
<td>Sample TAPE10 Diagnostic Output (Abbreviated)</td>
<td>A-9</td>
</tr>
</tbody>
</table>
LIST OF SYMBOLS

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_{DISK}$</td>
<td>Rotor disk area ($ft^2$)</td>
</tr>
<tr>
<td>$C_P$</td>
<td>Power coefficient</td>
</tr>
<tr>
<td>$C_T$</td>
<td>Thrust coefficient</td>
</tr>
<tr>
<td>$C_{T0}$</td>
<td>Thrust coefficient at one g</td>
</tr>
<tr>
<td>$F$ equiv.</td>
<td>Equivalent flat plate area ($ft^2$)</td>
</tr>
<tr>
<td>$g$</td>
<td>Acceleration due to gravity (32.1741 ft/sec$^2$)</td>
</tr>
<tr>
<td>$h$</td>
<td>Altitude (ft)</td>
</tr>
<tr>
<td>$N_r$</td>
<td>Rotor shaft rotation rate (revolution per minute)</td>
</tr>
<tr>
<td>$n$</td>
<td>Load factor (g's)</td>
</tr>
<tr>
<td>$Q$</td>
<td>Torque (ft-lbf)</td>
</tr>
<tr>
<td>$R$</td>
<td>Radius of turn (ft)</td>
</tr>
<tr>
<td>SHP</td>
<td>Shaft Horsepower</td>
</tr>
<tr>
<td>$T$</td>
<td>Ambient Temperature (degrees C)</td>
</tr>
<tr>
<td>$V$</td>
<td>True Airspeed (ft/sec)</td>
</tr>
<tr>
<td>$V_{CAS}$</td>
<td>Calibrated Airspeed (ft/sec)</td>
</tr>
<tr>
<td>$V_{TIP}$</td>
<td>Rotor tip speed (ft/sec)</td>
</tr>
<tr>
<td>$W$</td>
<td>Helicopter gross weight (lb)</td>
</tr>
<tr>
<td>$\Delta$</td>
<td>Increment</td>
</tr>
<tr>
<td>$n_c$</td>
<td>Empirical climb factor</td>
</tr>
<tr>
<td>$n_m$</td>
<td>Mechanical efficiency</td>
</tr>
<tr>
<td>$\eta_p$</td>
<td>Propulsive efficiency</td>
</tr>
<tr>
<td>$\mu$</td>
<td>Rotor advance ratio</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>\pi</td>
<td>3.14159</td>
</tr>
<tr>
<td>\rho</td>
<td>Atmospheric density (slugs/ft(^3))</td>
</tr>
<tr>
<td>\sigma</td>
<td>Blade solidity</td>
</tr>
<tr>
<td>\sigma_d</td>
<td>Atmospheric density ratio</td>
</tr>
<tr>
<td>\psi</td>
<td>Turn rate (deg/sec)</td>
</tr>
</tbody>
</table>
INTRODUCTION

In support of the tactical manual effort sponsored by the Naval Air Systems Command (AIR-5301), a methodology for predicting selected helicopter performance capability was developed by the Naval Air Development Center (6051). Performance items considered include specific excess power, sustained and instantaneous turn rate, and maneuver capability.

The programs discussed in this report utilize power required and power available tabular input along with other geometric and physical scalar input (e.g., rotor disk area and tip speed) to compute performance by means of the appropriate equation of motion. The output may be directly presented in numerical form; if desired, tactical manual plots can be created by the use of available plotting software.
DISCUSSION

THEORETICAL DEVELOPMENT

The computer programs described in this report provide a rapid, flexible and accurate means to generate tabular and graphical data expressing several important parameters which measure helicopter performance capability. The programs accept a common input format by which any conventional helicopter can be fully described through scalar namelist parameters and tabular data of power available and nondimensional power required. The programs exist as four separate routines which calculate performance data as follows:

Program #1: PSHELO - specific excess power as a function of load factor, calibrated airspeed and altitude

Program #2: SUSTURN - maximum sustained turn rate as a function of calibrated airspeed and altitude

Program #3: INSTURN - maximum instantaneous turn rate as a function of calibrated airspeed and altitude

Program #4: MANEUV - specific excess power, load factor and turn radius as a function of ambient condition, calibrated airspeed and altitude

A derivation and description of the equations used in each of the four routines follows.
SPECIFIC EXCESS POWER

Specific excess power \( P_s \) is intended as a measurement criterion for comparing relative performance capability throughout the speed-altitude envelope for any two helicopters. It is specifically a measure of the power available for maneuvering over and above that used to maintain level flight and can be expressed as shown in equation (1).

\[
P_s = \frac{(P_{\text{supplied by engine}} - P_{\text{required}})}{\text{Gross Weight}} \text{ (ft/min)}
\]  

(1)

In relating the quantity \( P_s \) to performance, the specific energy is first derived from the total energy at a point in velocity-altitude space, equation (2).

\[
E = GWh + \frac{GWV^2}{2g} + \frac{I\Omega^2}{2}
\]

(2)

The specific energy or energy height is then:

\[
h_e = \frac{E}{GW} = h + \frac{V^2}{2g} + \frac{I\Omega^2}{2GW}
\]

(3)

Specific excess power is the time rate of change of specific energy, which yields equation (4).

\[
P_s = \frac{dh}{dt} + \frac{V}{g} \frac{dV}{dt} + \frac{I\Omega}{GW} \frac{d\Omega}{dt}
\]

(4)

where

- \( E \) = total energy
- \( h \) = altitude
- \( h_e \) = specific energy or energy height
- \( t \) = time
- \( V \) = true airspeed
- \( g \) = acceleration of gravity
- \( I \) = rotor inertia
- \( \Omega \) = rotor rotational speed
- \( GW \) = helicopter gross weight
The above equations reveal that a given \( P_S \) increment may contribute to a rate of climb \( (\frac{dh}{dt}) \), level flight acceleration \( (\frac{V}{g} \frac{dv}{dt}) \), an increase in rotor speed \( (\frac{\Omega}{GW} \frac{d\Omega}{dt}) \) or a change in the direction of the flight path. The units of \( P_S \) are distance/time, which for helicopter applications is best suited by feet/minute.

The process of computing and plotting \( P_S \) as a function of airspeed and altitude for each helicopter was based on determining power available, subtracting from it the power required at a given airspeed and altitude, and dividing by gross weight, as expressed by equation (5).

\[
P_S = \frac{.875 \times (\Delta HP) \times \eta_m \times 33000}{GW}
\]

where

- \(.875 = \) empirical climb factor
- \(\Delta HP = \) power available minus power required
- \(\eta_m = \) mechanical efficiency = 0.9
- \(33000 = \) 33000 foot-lbs/minute/horsepower conversion factor
- \(GW = \) helicopter gross weight

\( P_S \) yields accurate values for the rate of climb for speeds above approximately 60 knots; however, for speeds below 60 knots, the actual rate of climb is somewhat greater than the computed \( P_S \) value. This is not reflected in the equation because of the use of a constant as opposed to a variable empirical climb factor.

The computational process is carried out using nondimensional forms of power required and power available. Nondimensional power required, or power coefficient \( (C_p) \) is found as a function of thrust coefficient \( (C_T) \) and advance ratio \( (\mu) \) for most existing helicopters. Thrust coefficient and advance ratio are defined by the following two expressions:
\[ C_T = \frac{nG}{\rho_A V_{TIP}^2}, \quad \text{and} \quad \mu = \frac{(1.689)V_T}{V_{TIP}} \]  

(6)

where

\( n \) = load factor (g)  
\( A \) = rotor disk area (ft\(^2\))  
\( V_{TIP} \) = rotor tip speed (ft/second)  
\( V_T \) = true airspeed (knots)

(also, \( C_{T_T} \) is defined as \( C_T \) when \( n = 1.0g \))

Engine power available at IRP is given as a function of outside air temperature (OAT) and altitude. Power is nondimensionalized using equation (7)

\[ C_p = \frac{SHP \times 550}{\rho_A V_{TIP}^3} \]  

(7)

where SHP can be either power required or power available. \( P_S \), based on the nondimensional coefficients \( C_{p_{req}} \), \( C_{p_{avail}} \) and \( C_{T_T} \) is then found from equation (8).

\[ .875 \times (C_{p_{avail}} - C_{p_{req}}) \times 60 \times n_M \times V_{TIP} = 47.25 \times \Delta C_p \times V_{TIP} \]  

(8)

To further explain the procedure by which \( P_S \) is calculated, we begin by determining the power available for a given flight condition (airspeed, altitude). Shaft horsepower available can be input as a function of altitude and ambient temperature, expressed by equation (9).

\[ SHP = f(h, T), \]  

(9)

or as a function of ambient temperature and altitude:

13
or the percent torque available may be supplied, where shaft horsepower is the percent torque times shaft horsepower at 100% torque. If the power available data supplies percent torque, a factor for horsepower per percent of torque (FORTRAN name TORFAC) and a maximum torque limit (FORTRAN name XMSN) if other than 100% must be input. The relationship of TORFAC and XMSN is such that TORFAC times XMSN yields the transmission limit in shaft horsepower. If the transmission limit is known, this may be input (FORTRAN name PLIMIT).

A velocity correction in the power available due to ram effects may be applied. The correction can be linear:

\[ \text{SHP} = \text{SHP} \times C_1 \times V_T \]  

(11)

or exponential,

\[ \text{SHP} = \text{SHP} \times C_2 \times e^{(C_3 \times V_T)} \]  

(12)

where \( C_1 \), \( C_2 \) and \( C_3 \) are input constants (FORTRAN names DELHP, TMAN, TCHAR). The resulting shaft horsepower available is converted to coefficient form incorporating the effect of altitude through the factor \( \rho \), air density, in equation (13).

\[ C_p^{\text{avail}} = \frac{\text{SHP} \times 550}{\rho A V_{TIP}^3} \]  

(13)

The next major step is to determine the power required. This is found in coefficient form as a function of either advance ratio and thrust coefficient:

\[ C_p^{\text{req}} = f(\mu, C_T) \]  

(14)

or as a function of thrust coefficient and advance ratio:

\[ C_p^{\text{req}} = f(C_T, \mu) \]  

(15)
If \( C_{P_{req}} = f(\mu, C_T) \), then an input switch (FORTRAN name MUCTSW) is set equal to 1, and if \( C_{P_{req}} = f(C_T, \mu) \), the switch is set equal to 0.

Recalling that \( P_S \) is presented as a function of airspeed and altitude, at this point it is noted that advance ratio is a function of speed, and \( C_T \) is a function of density \( \rho \) which incorporates the effect of altitude. Hence, we can represent this dependence as follows:

\[
\begin{align*}
\mu &= f(V_T) \\
C_P &= f(\mu, C_T) + P_S = f(C_p, C_T) + P_S = f(V_T, h)
\end{align*}
\]

(16)

arriving at the final result of specific excess power as a function of airspeed and altitude which is ultimately plotted.

An incremental drag correction can be made to the power required coefficient using an incremental equivalent flat plate area in the last term in equation (17).

\[
C_{P_{req}} = C_{P_{req}} + \frac{\Delta F}{2 \pi A \times \eta_M \times \eta_P}
\]

(17)

Acceptable values for mechanical efficiency, \( \eta_M \) (FORTRAN name ETAM) and propulsive efficiency, \( \eta_P \) (FORTRAN name ETAP) are 0.9 and 0.8, respectively. Incremental flat plate area (FORTRAN name DELFE) is used to adjust helicopter drag for variations in weapons loadings.

Specific excess power performance plots can be generated for various load factors. This is done by inputting the desired load factor, \( n \) (FORTRAN name GFAC) which is then used in the calculation of the thrust coefficient, \( C_T \).
SUSTAINED TURN RATE

The equation used to compute turn rate is based on the point mass equation, and is a function of load factor, airspeed, and flight path angle, as expressed in equation (18).

\[
\dot{\psi} = \frac{g n^2 \cos^2 \gamma}{V \cos \gamma}
\]  

(18)

where

\[
\dot{\psi} = \text{turn rate (radians/second)}
\]
\[
n = \text{load factor (g)}
\]
\[
\gamma = \text{flight path angle (radians)}
\]
\[
V = \text{true airspeed (feet/second)}
\]

Using the small angle assumption that \( \gamma = 0 \), the following classical relationship results in equation (19).

\[
\dot{\psi} = \frac{g n^2}{V}
\]  

(19)

Computations require that \( \dot{\psi} \) be in units of degrees/second, and \( V \) in knots. With appropriate conversion factors inserted, the turn rate equation becomes that shown in equation (20).

\[
\dot{\psi} = \frac{32.17 \times \frac{\sqrt{V^2 - 1}}{V_T}}{(1.689) \times \left(\frac{180}{\pi}\right)} = 1091.4388 \times \frac{\sqrt{V^2 - 1}}{V_T}
\]  

(20)

where \( V_T = \text{true airspeed in knots} \).

To arrive at maximum sustained turn rate at any point within the flight envelope, power required is set equal to power available:

\[
C_p^{\text{req}} = C_p^{\text{avail}}
\]

Having \( C_p^{\text{req}} \) and the advance ratio \( \mu \) for a given airspeed, the thrust coefficient \( C_T^{\text{req}} \) is read from the input helicopter performance data, as expressed by equation (20).
\[ C_T = f(C_p, \mu) \]  

The sustained load factor is the thrust divided by the weight, equation (21).

\[ n = \frac{C_T \times \rho \times A \times V_{TIP}^2}{GW} \]  

Finally, the sustained turn rate for a given airspeed and altitude using the sustained load factor is found from the turn rate equation (20).

**MAXIMUM INSTANTANEOUS TURN RATE**

Maximum instantaneous turn rate is derived from the maximum amount of thrust which the rotor system can attain on a transient or nonsustained basis. Maximum thrust (or rotor limit) for a given helicopter can be expressed by the quantity in equation (22).

\[ \frac{2 \ C_T^{\text{max}}}{\sigma} = f(\mu) \]  

where

\[
\begin{align*}
C_T^{\text{max}} &= \text{maximum thrust coefficient} \\
\sigma &= \text{rotor solidity} = \frac{bc}{\pi R} \\
b &= \text{number of blades} \\
c &= \text{blade chord} \\
R &= \text{rotor radius}
\end{align*}
\]

This quantity is approximate since blade stall is gradual, and its occurrence is affected by the direction of turn as well as airspeed. The program INSTURN requires $C_T^{\text{max}}$ (FORTRAN name CTMAX) to be input, where $C_T^{\text{max}}$ is either known explicitly or can be solved for from the expression

\[
\frac{2 \ C_T^{\text{max}}}{\sigma}, \quad \frac{2 \ C_T^{\text{max}}}{\sigma}
\]

Either $C_T^{\text{max}}$ or $\frac{2 \ C_T^{\text{max}}}{\sigma}$ are known or estimated for most existing helicopters.
Having $C_{T_{\max}}$ and using the following relation in equation (23),

$$\rho A V_{TIP}^2 C_{T_{\max}}$$

the maximum instantaneous load factor is computed at any point in the flight envelope. The final step yields the turn rate as a function of true airspeed and load factor (which incorporates the effect of altitude using $\rho$) using the turn rate equation (20) derived in the section on sustained turn rate.

As airspeed approaches zero, turn rate values become unrealistically large. It becomes necessary to specify a maximum turn rate attainable at zero airspeed in degrees per second (FORTRAN name TRV0), either due to control limitations, a handling qualities limit determined in testing, some value based on vehicle similarities or other reasonable assumptions. A second value required to define the maximum instantaneous turn rate versus airspeed curve is the break velocity (FORTRAN name VB). The break velocity, usually between 50 and 80 knots, is the lowest airspeed for which the instantaneous turn rate is calculated. Below the break velocity, an internal curve-fitting routine blends the calculated turn rate curve from the break velocity to the maximum instantaneous turn rate value (TRV0) at zero airspeed.

**MANEUVER CAPABILITY**

Two previous programs computed specific excess power ($P_g$) at selected load factors and maximum sustained turn rate over the helicopter's entire airspeed-altitude envelope. By introducing the parameters $P_g$ and turn rate into one graph at one ambient condition (fixed altitude and temperature), the resultant capability for a specified airspeed and selected turn radius can be shown. The maneuver capability graph is plotted against calibrated airspeed in knots and turn rate in degrees per second.
Lines of constant radius for 500, 1000, 1500 and 2000 feet are projected on the airspeed-turn rate graph according to the geometrical relation, equation (24).

\[ \dot{\psi} = \frac{V}{R} \]  

(24)

For \( R \) in feet, \( V_T \) in knots true airspeed and \( \dot{\psi} \) in degrees per second, the relation becomes that shown in equation (25).

\[ \dot{\psi} = 96.7726 \times \frac{V_T}{R} \]  

(25)

and for calibrated airspeed in knots, \( V_T \) is replaced by \( \frac{V_C}{\sqrt{\sigma_d}} \) where \( \sigma_d \) is the density ratio at the selected ambient condition so that the final expression equation (26) is

\[ \dot{\psi} = 96.7726 \times \frac{V_C}{R\sqrt{\sigma_d}} \]  

(26)

Lines of constant load factor are plotted against airspeed and turn rate, and also result from the geometrical relation of equation (20).

\[ \dot{\psi} = \frac{2\sqrt{n^2 - 1}}{V} = 1091.4388 \times \frac{\sqrt{n^2 - 1}}{V_T} \text{ (deg/sec)} \]  

(20)

Using calibrated airspeed, this becomes that shown in equation (27).

\[ \dot{\psi} = 1091.4388 \times \frac{\sqrt{\sigma_d} \sqrt{n^2 - 1}}{V_C} \text{ (deg/sec)} \]  

(27)

Superimposed over turn radius and load factor lines are the specific excess power curves of a particular helicopter. The construction of these curves begins with expressing load factor as a function of turn rate, as shown in equation (28).

\[ \sqrt{\frac{\dot{\psi} V}{g}} = \sqrt{n^2 - 1} + 1 \]  

(28)
This defines a load factor for each value of turn rate and airspeed. This load factor defines a thrust coefficient and the airspeed defines an advance ratio which yield a power required coefficient read from the helicopter performance data. Specific excess power is determined from the power required and power available coefficients shown earlier in equation (8).

An additional line generated by the Maneuver Capability program is the rotor limit line, based on the $C_{T_{\text{max}}}$ value input for each particular helicopter. The rotor limit line is parallel to the lines of constant load factor.

The main program logic, which illustrates how the above equations are implemented, is presented in Appendix C.

**PROGRAM DESCRIPTION**

In addition to the main programs, which perform the computations on the dynamic equations, there are three supporting subroutines, as described below.

**GREAD/TLOOK** - This is a three degree-of-freedom interpolation routine and has two modes of operation. In the first mode, table data representing power required and available characteristics are input and stored. Each table is assigned a predetermined reference number. In the second mode, table data are interpolated and extrapolated by employing the function SPLINR for use in the dynamic calculations. A more detailed explanation of this routine can be found in reference (a).

**SPLINR** - This function is used to interpolate or extrapolate two-dimensional data. The interpolation is calculated using a local curve fit scheme described in reference (b). Linear extrapolations are made using each end point slope of the local curve fit.
ATMOS - This is an atmosphere table which returns properties of density, pressure, temperature, and sound velocity for an input altitude and atmosphere code (i.e., 1 = standard, 2 = hot day, 3 = tropical day).

The data required for the program consist of a series of single-value fixed inputs and multiple-valued tabular inputs. The form of the computer data deck necessary to make a run is presented in Figure 1.

The tabular data include:

- Power available as a function of altitude and ambient temperature (see Figure 2).

or:

- Percent torque available as a function of ambient temperature and altitude (see Figure 3).

- Power required coefficient as a function of thrust coefficient and advance ratio. This table is used in the specific excess power and maneuver programs (see Figures 4 and 5).

- Thrust coefficient as a function of power required coefficient and advance ratio. This table is used in the sustained turn rate program (see Figure 6).

It should be noted that none of the above tabular inputs are required for the instantaneous turn rate program.

The fixed inputs consist of helicopter size and mass data, ambient conditions, efficiencies, and program switches which regulate the options available to the user. Table I contains a scalar input variable list.
EOI
$D (3rd CASE)$
$D (2nd CASE)$
BLANK
"TABULAR INPUTS"
TITLE
$
"SCALAR INPUTS" (1st CASE)
$D
EOR
LGO.
FTN4, I = PSHELO, L = 0. (SUSTURN, INSTURN, MANEUV)
LIBRARY, A530LB1.
GET, A530LB1/UN = VT17B1.
GET, PSHELO/UN = 603845. (SUSTURN, INSTURN, MANEUV)
ACCOUNT, CHARGE NUMBER, PASSWORD
JOB, CB100000, T10

FIGURE 1. Data Input Deck Structure

22
Table Reference No. 10
SHP = f (h, T)

FIGURE 2. Power Available Tabular Input
Table Reference No. 10
$Q = f(T, h)$

**FIGURE 3. Torque Available Tabular Input**
Table Reference No. 9

\[ C_p = f(C_T, \mu) \]

FIGURE 4. Power Required Coefficient Tabular Input, MUCTSW = 0
Table Reference No. 9

\[ C_p = f(u, C_T) \]

FIGURE 5. Power Required Coefficient Tabular Input, MUCTSW ≠ 0
Table Reference No. 11
\[ C_T = f(C_p, \mu) \]

FIGURE 6. Thrust Coefficient Tabular Input
### TABLE I. SCALAR INPUT VARIABLE LIST

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
<th>Units</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADISK</td>
<td>Rotor Disk Area</td>
<td>ft²</td>
<td></td>
</tr>
<tr>
<td>ALTØ</td>
<td>Maneuver Altitude</td>
<td>ft</td>
<td>0.0</td>
</tr>
<tr>
<td>ALT1</td>
<td>Initial Matrix Altitude</td>
<td>ft</td>
<td>0.0</td>
</tr>
<tr>
<td>CASI</td>
<td>Initial Matrix Calibrated Airspeed</td>
<td>knots</td>
<td>0.0</td>
</tr>
<tr>
<td>CTMAX</td>
<td>Maximum Thrust Coefficient (used for instantaneous ( \dot{\phi} ) and maneuver)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DELCAS</td>
<td>Calibrated Airspeed Matrix Increment</td>
<td>knots</td>
<td>-</td>
</tr>
<tr>
<td>DELFE</td>
<td>Equivalent Flat Plate Area Increment</td>
<td>ft²</td>
<td>0.0</td>
</tr>
<tr>
<td>DELPH</td>
<td>Linear Velocity Correction Power Constant</td>
<td>SHP</td>
<td>0.0</td>
</tr>
<tr>
<td>DELTALT</td>
<td>Altitude Matrix Increment</td>
<td>ft</td>
<td>-</td>
</tr>
<tr>
<td>DELTRT</td>
<td>Turn Rate Matrix Increment</td>
<td>deg/sec</td>
<td>-</td>
</tr>
<tr>
<td>ETAM</td>
<td>Mechanical Efficiency</td>
<td>-</td>
<td>1.0</td>
</tr>
<tr>
<td>ETAP</td>
<td>Propulsive Efficiency</td>
<td>-</td>
<td>1.0</td>
</tr>
<tr>
<td>GFAC</td>
<td>Load Factor</td>
<td>g's</td>
<td>-</td>
</tr>
<tr>
<td>IPRINT</td>
<td>If ( \neq 0 ), diagnostics will be printed</td>
<td>-</td>
<td>0.0</td>
</tr>
<tr>
<td>KATMOS</td>
<td>If ( \neq 1 ), standard day atmospheric properties; if = 2, hot day; if = 3, tropical day</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>MJCTSW</td>
<td>If = 0, ( C_p = f(C_T, \mu) ) input; if ( \neq 0 ), ( C_p = f(\mu, C_T) ) input</td>
<td>-</td>
<td>0.0</td>
</tr>
<tr>
<td>NALT</td>
<td>Number of Matrix Altitudes</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>NCAS</td>
<td>Number of Matrix Calibrated Airspeeds</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>NPRINT</td>
<td>If ( \neq 0 ), tabular input will be printed</td>
<td>-</td>
<td>0.0</td>
</tr>
<tr>
<td>NTRRRT</td>
<td>Number of Matrix Turn Rates</td>
<td>deg/sec</td>
<td>-</td>
</tr>
<tr>
<td>PLIMIT</td>
<td>Transmission Limit</td>
<td>SHP</td>
<td>10⁶</td>
</tr>
<tr>
<td>PSIMAX</td>
<td>Maximum Sustained Turn Rate</td>
<td>deg/sec</td>
<td>60.0</td>
</tr>
<tr>
<td>TCHAR</td>
<td>Exponential Velocity Correction Characteristic</td>
<td>-</td>
<td>1.0</td>
</tr>
<tr>
<td>TMAN</td>
<td>Exponential Velocity Correction Mantissa</td>
<td>-</td>
<td>0.0</td>
</tr>
<tr>
<td>TOGW</td>
<td>Helicopter Weight</td>
<td>lb.</td>
<td>-</td>
</tr>
<tr>
<td>TORFAC</td>
<td>Torque Scale Factor; if ( = 0 ), power available input</td>
<td>ft-lb</td>
<td>0.0</td>
</tr>
<tr>
<td>TRVØ</td>
<td>Maximum Instantaneous Turn Rate</td>
<td>deg/sec</td>
<td>-</td>
</tr>
<tr>
<td>VB</td>
<td>When ( V_{CAS} &lt; V_B ), sustained ( \dot{\phi} ) altered to satisfy ( \text{PSIMAX} ) ( V_{CAS} = 0 )</td>
<td>knots</td>
<td>-</td>
</tr>
<tr>
<td>VTIP</td>
<td>Rotor Tip Speed</td>
<td>ft/sec</td>
<td>-</td>
</tr>
<tr>
<td>XMSN</td>
<td>PLIMIT/TORFAC, used when torque is input</td>
<td>-</td>
<td>0.0</td>
</tr>
</tbody>
</table>
Appendix A consists of a series of example cases which show how each helicopter performance program is implemented. Reference (c) should be consulted for information concerning the generation of the tactical manual plots presented in this appendix. Appendix B contains listings of the main performance source decks.

**COMPUTATIONAL PROCEDURE**

Initially, default conditions are set (e.g., $\eta_M = \eta_P = 1.$, (DELHP=0.), then scalar and tabular input, previously described is loaded into the programs. Scalar variables are input through the FORTRAN utility NAMELIST and the tabular data is input through the subroutine TREAD. Minimal tabular input for each program was discussed in the previous section. The inputted scalar variables override the initial default values. After the input has been stored, the programs enter an altitude DO LOOP at statement #50 which concludes at statement #100. At the beginning of this loop, altitude and temperature corrections are implemented and the input data is adjusted accordingly. Nested in the altitude loop is a velocity DO LOOP which begins at statement #60 and concludes at statement #100 (statement #101 for the instantaneous turn rate program). It is in this velocity loop that the helicopter $P_S$ or $\dot{\psi}$ is determined for each velocity-altitude matrix condition. After the two main loops have been executed, the programs load the output matrix on a file designated as TAPE6. This data is in TPLOT format and can be directly plotted by utilizing the software described in reference (c). Finally, the output matrix is loaded on a file designated as TAPE8 in a format directly applicable to the tactical manual interface software. The instantaneous turn rate program incorporates low speed turn rate corrections to the output matrix before TAPE6 and TAPE8 files are generated. The maneuver program is similar in context to the other helicopter performance programs except that the main altitude DO LOOP is replaced by a turn rate input loop.
ACKNOWLEDGMENT

The authors wish to express appreciation to Michael Caddy who developed the interpolation and graphics software used in this analysis. Appreciation is also extended to Adam Petruszka for his technical concepts regarding helicopter theory.
REFERENCES


(b) Akima, Hiroshi, "Interpolation and Smooth Curve Fitting Based on Local Procedures," Institute for Telecommunications Sciences of 1 Mar 1972.


APPENDIX A
EXAMPLE CASES
Representative cases showing the numerical and graphical output of the specific excess power, sustained and instantaneous turn rate and maneuver programs are presented. An example input deck, which is compatible with all four performance cases, is displayed in Table A1. The table section of this sample input is graphically illustrated in Figures A-1, A-2, and A-3. Each program contains three output file units. The file unit designated as TAPE 10 contains a detailed input listing together with the output diagnostics (if selected by the IPRINT and NPRINT options). The file unit designated as TAPE 6 contains tabulated output in a format directly usable to the graphics package described in reference (c). The file unit designated as TAPE 8 contains the tabulated matrix output which forms the basis of the tactical manual plots. TAPE 8 data has to be transmuted by preprocessing software before the graphics routines of reference (c) can be applied to create the tactical manual plots. The nature of this intermediate software will be discussed in a future publication. The remaining figures and tables in this appendix present examples of some of these output files for each performance program along with a sample tactical manual plot.

In addition to the above output, the specific excess power program creates a file unit designated as TAPE 3 which contains the flight envelope data displayed in the instantaneous turn rate tactical manual plot. Finally, the capability to portray comparative plots between two separate helicopters is available in the previously mentioned intermediate software.

Reference (d) should be consulted for tactical manual applications involving existing helicopter weapon systems.
EXAMPLE INPUT

<table>
<thead>
<tr>
<th>Z</th>
<th>T</th>
<th>ALT</th>
<th>POWER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.</td>
<td></td>
<td>0.</td>
</tr>
<tr>
<td>3</td>
<td>-50.</td>
<td>1000.</td>
<td>50.</td>
</tr>
<tr>
<td>0</td>
<td>10000.</td>
<td>20000.</td>
<td></td>
</tr>
<tr>
<td>2500.</td>
<td>1500.</td>
<td>1000.</td>
<td>500.</td>
</tr>
<tr>
<td>2000.</td>
<td>1500.</td>
<td>1000.</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1500.</td>
<td>1000.</td>
<td></td>
</tr>
</tbody>
</table>

EXAMPLE CP=F(CT,MU)

<table>
<thead>
<tr>
<th>Z</th>
<th>MU</th>
<th>CT</th>
<th>CP</th>
<th>CP</th>
<th>CP</th>
<th>CP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td></td>
<td>0.25</td>
<td>0.50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.</td>
<td>0.01</td>
<td>0.005</td>
<td>0.01</td>
<td>0.002</td>
<td>0.001</td>
</tr>
<tr>
<td>3</td>
<td>0.003</td>
<td>0.005</td>
<td>0.002</td>
<td>0.001</td>
<td>0.008</td>
<td>0.003</td>
</tr>
<tr>
<td>3</td>
<td>0.005</td>
<td>0.008</td>
<td>0.001</td>
<td>0.009</td>
<td>0.008</td>
<td>0.003</td>
</tr>
</tbody>
</table>

EXAMPLE CT=F(CP,MU)

<table>
<thead>
<tr>
<th>Z</th>
<th>MU</th>
<th>CP</th>
<th>CT</th>
<th>CT</th>
<th>CT</th>
<th>CT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0.25</td>
<td>0.50</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.003</td>
<td>0.005</td>
<td>0.002</td>
<td>0.001</td>
<td>0.008</td>
<td>0.003</td>
</tr>
<tr>
<td>4</td>
<td>0.001</td>
<td>0.005</td>
<td>0.002</td>
<td>0.005</td>
<td>0.008</td>
<td>0.003</td>
</tr>
<tr>
<td>4</td>
<td>0.005</td>
<td>0.008</td>
<td>0.001</td>
<td>0.019</td>
<td>0.008</td>
<td>0.01</td>
</tr>
<tr>
<td>4</td>
<td>0.01</td>
<td>0.005</td>
<td>0.008</td>
<td>0.01</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

EOT
FIGURE A1. Sample SHP Input (Graphical)
A-5
FIGURE A2. Sample $C_p$ Input (Graphical)
Table AII. Sample TAPE8 Matrix Output (Abbreviated)

A-8
<table>
<thead>
<tr>
<th>Table Number</th>
<th>Reference Number</th>
<th>Array Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>160</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>160</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>160</td>
</tr>
</tbody>
</table>

Data Storage Allocation: 3000

Data Storage Not Used: 7777

Table AIII. Sample TAPE Diagnostic Output (Abbreviated)
FIGURE A4. Sample $P_s$ Tactical Manual Plot
FIGURE A5. Sample TAPE6 $P_s$ Crossplot

A-11
FIGURE A7. Sample TAPE6 Sustained $\psi$ Crossplot
FIGURE A8. Sample Instantaneous Tactical Manual Plot
FIGURE A9. Sample TAPE6 Instantaneous $\psi$ Crossplot

A-15
FIGURE A10. Sample Maneuver Tactical Manual Plot
FIGURE A11. Sample TAPE6 Maneuver Crossplot

A-17
## APPENDIX B

**PROGRAM LISTINGS**

<table>
<thead>
<tr>
<th>Program</th>
<th>Page No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specific Excess Power</td>
<td>B-3</td>
</tr>
<tr>
<td>Sustained Turn Rate</td>
<td>B-7</td>
</tr>
<tr>
<td>Instantaneous Turn Rate</td>
<td>B-11</td>
</tr>
<tr>
<td>Maneuvering Capability</td>
<td>B-15</td>
</tr>
</tbody>
</table>
CPSHELO

PROGRAM PSHELO(INPUT, OUTPUT, TAPE5=INPUT, TAPE10, TAPE6, 
1TAPE8)
COMMON/PRINT/NPRINT
COMMON/IOUNITS/IP, IR, ID
DIMENSION CTA(100), PSS(100, 100), XTAS(100), XALT(100), 
1V(100), IITLE(7), XCAS(100)
NAMELIST/D/NCAS, NALT, DELCAS, DELTALT, IATYPE, CPIRP, UTIP, UNV, 
1CAS1, ALT1, CTVAL, KATMOS, GFAC, TOGW, ADISK, TORFAC, PLIMIT, 
2ETAN, ETAP, DELFE, NPRINT, TMAN, TCHAR, MUCTSW, XMSN, IPRINT, 
3DELHP, CTMAX, TRV0, VB, NTRNRT, DELTRT, ALT0, 
4PSIMAX
DATA IP, IR, ID/10, 5, 10/
DATA TORFAC, PLIMIT, TMAN, TCHAR, MUCTSW/0., 1.E6, 0., 1., 0/
DATA PSIMAX/60.0/
DATA ETAN, ETAP, DELFE/1.1., 0.0/
DATA XMSN, IPRINT, DELHP/100.0, 0, 0/
DATA TMAN, TCHAR, NPRINT/0., 1.0,
READ(5, D)
IF(EOF(5))10, 9
10 STOP
9 CONTINUE
READ(5, 1)IITLE
1 FORMAT(7A10)
CALL TREAD(1, XD, YD, ZD, FXYZ)
11 WRITE(8, 2)IITLE
PRINT(10, D)
IF(IPRINT.NE.0)PRINT(10, *)'PS, G, MU, U, ALT, CT, CPI, RHO, CPR'
2 FORMAT(*APS01G1X, 7A10)
WRITE(8, *)NCAS, NALT, CAS1, DELCAS, ALT1, DELTALT, IATYPE 
NU=INT(UNV)
L=1
CTA(I)=CTVAL
CT=CTA(L)
CPREQ=CPREQ+DELCPR
PSS(IALT,ICAS)=(CPIRP-CPREQ)*60.*.875*.90*UTIP/CT0

C
IF(IPRINT.NE.0)PRINT(10,31)PSS(IALT,ICAS),GFAC,V(ICAS),
*XCAS(ICAS),XALT(IALT),CT,CPIRP,RHO,CPREQ
31 FORMAT(5X,5(I1H*),9G13.5)
100 CONTINUE

C
*******************************************************************
C THIS SECTION LOADS THE ARRAY IN THE
C REQUIRED FORMAT OF TREAD
C
DATA IT/100/
WRITE(6,17)IT,ITITLE
17 FORMAT(I5,T11,7A10)
WRITE(6,18)GFAC
18 FORMAT(4HGFAC,3H 1,T11,F10.0)
WRITE(6,19)NALT,(XALT(IALT),IALT=1,NALT)
19 FORMAT(4HALT ,T6,I2,(T11,7F10.0))
WRITE(6,21)NCAS,(XCAS(ICAS),ICAS=1,NCAS)
21 FORMAT(4HKTAS,T6,I2,(T11,7F10.0))

C
DO 200 IALT=1,NALT
  WRITE(6,23)NCAS,(PSS(IALT,ICAS),ICAS=1,NCAS)
23 FORMAT(4HPSBS,T6,I2,(T11,7F10.1))

C
200 CONTINUE
  WRITE(6,25)
25 FORMAT(*EOT*/80(1H ))

C
*******************************************************************

ACMB INPUT ON TAPE8
C***************************************************************************************************
IC=NALT/9+1.001
JE=0
DO 810 K=1,IC
JO=JE+1
JE=MIN(JO+8,NALT)
DO 810 I=1,NCAS
IF(I.EQ.1)WRITE(8,801)(XALT(J),J=JO,JE)
801 FORMAT(*VEL/ALT 1,9F13.0)
WRITE(8,800)(XCAS(I),(PSS(J,I),J=JO,JE))
800 FORMAT(10G13.5)
810 CONTINUE
C***************************************************************************************************
C
READ(5,D)
IF(EOF(5))10,11

END
CSUSTURN
    PROGRAM CSUSTURN(INPUT,OUTPUT,TAPE5=INPUT,TAPE10,TAPE6,
1 TAPE8)
    COMMON/PRINT/NPRINT
    COMMON/IOUNITS/IP,IR,ID
    DIMENSION CTA(100),PSIDOT(100,100),XTAS(100),XALT(100),
1 V(100),ITITLE(7),XCAS(100)
    NAMELIST/D/NCAS,NALT,DELCAS,DELTALT,IATYPE,CP1P,UTIP,UNV,
1 CAS1,ALT1,CTUAL,KATMOS,GFAC,TOGU,ADISK,TORFAC,PLIMIT,
2 ETAM,ETAP,DELFE,NPRINT,TMAN,TCHAR,MUCTSU,XMSN,
3 IPRINT,DELHP,CTMAX,QRUB,VB,NTRNRT,DELTTR,ALT0,
4 PSIMAX
    DATA IP,IR,ID/10,5,10/
    DATA TORFAC,PLIMIT,TMAN,TCHAR,MUCTSU/0.1.E6,0.1,0./
    DATA ETAM,ETAP,DELFE/1.1,0./
    DATA XMSN,IPRINT,G,DELHP/100.0,32.1741,0/
    DATA PSIMAX/60./
    READ(5,D)
10 IF(EOF(5))10,9
  10 STOP
  9 CONTINUE
    READ(5,1)ITITLE
1 FORMAT(7A10)
    CALL TREAD(1,XD,YD,ZD,FXYZ)
11 WRITE(8,2)ITITLE
    PRINT(10,D)
2 IF(IPRINT.NE.0)PRINT(10,*)'PSIDOT,G,MU,V,ALT,CT,CPI,RHO'
    FORMAT(*ACZGA12%,7A10)
    WRITE(8,*)NCAS,NALT,CAS1,DELCAS,ALT1,DELTALT,IATYPE
    NU=INT(UNV)
    L=1
    CTA(L)=CTUAL
    CT=CTA(L)
50 DO 100 IALT=1,NALT
XALT(IALT)=FLOAT(IALT-1)*DELTALT
CALL ATMOS(KATMOS,XALT(IALT),TEMPR,PRESSR,RHO,SIGMA,VSN)
XOATC=(TEMPR-491.69)*5./9.

******************************************************************************

IF(TORFAC.EQ.0.)GOTO 58
PLIMIT=XMN*TORFAC
CALL TLOOK(10,XOATC,XALT(IALT),0.,TORK)
PUR=TORK*TORFAC
GOTO 59
58 CALL TLOOK(10,XALT(IALT),XOATC,0.,PUR)
59 CONTINUE
IF(PUR.GT.PLIMIT)PUR=PLIMIT
CPIRP=(PUR**65.)/(RHO*ADISK*VTIP**3)

******************************************************************************

60 DO 100 ICAS=1,NCAS
XCAS(ICAS)=FLOAT(ICAS-1)*DELCAS+CAS1
IF(XCAS(ICAS).EQ.0.)XCAS(ICAS)=1.E-6
XTAS(ICAS)=XCAS(ICAS)/(SIGMA)**5
IF(TMAN.EQ.0.)GOTO 39
PUR1=PUR1*MAN*EXP(TCHAR*XTAS(ICAS))+DELHP*XTAS(ICAS)
IF(PUR1.GT.PLIMIT)PUR1=PLIMIT
CPIRP=(PUR1**550.)/(RHO*ADISK*VTIP**3)
39 V(ICAS)=XTAS(ICAS)**1.689/VTIP
DELCPR=.5*DELP*V(ICAS)**3/(ADISK*ETAM*ETAP)
CPR=CPIRP-DELCPR
CALL TLOOK(11,CPR,V(ICAS),0.,CT)
IF(CT.LT.0.)SIDIOT(IALT,ICAS)=-1000.
IF(CT.LT.0.)GO TO 41
GFAC=CT*RHO*ADISK*UTIP**2/TOGW
KSIGN=0
   IF(GFAC.LT.0.)KSIGN=-1
   IF(GFAC.LT.0.)GFAC=-GFAC
   IF(GFAC.GE.1.)GO TO 40
   XXXX=1.+1./(1.-GFAC**2)**.5-1.)
   PSIDOT(IALT,ICAS)=G*XXXX/(XTAS(ICAS)**1.689))**5.29578
   IF(KSIGN.EQ.1)PSIDOT(IALT,ICAS)=-PSIDOT(IALT,ICAS)
   GO TO 41
40 PSIDOT(IALT,ICAS)=(G*(GFAC**2-1)**.5/(XTAS(ICAS)**1.689))**5.29578
   IF(PSIDOT(IALT,ICAS).GE.PSIMAX)PSIDOT(IALT,ICAS)=PSIMAX
   IF(KSIGN.EQ.1)PSIDOT(IALT,ICAS)=-PSIDOT(IALT,ICAS)
41 CONTINUE

   IF(IPRINT.NE.0)PRINT(10,31)PSIDOT(IALT,ICAS),GFAC,V(ICAS),
   *XCAS(ICAS),XALT(IALT),CT,CPIRP,RHO
31 FORMAT(5X,5(1H1),8G13.5)
100 CONTINUE

C ******************************************************************************
C THIS SECTION LOADS THE ARRAY IN THE
C REQUIRED FORMAT OF TREAD
C
DATA IT/100/
WRITE(6,17)IT,ITITLE
17 FORMAT(I5,T11,7A10)
WRITE(6,18)GFAC
18 FORMAT(4HFAC,3H 1,T11,F10.0)
WRITE(6,19)HALT,(XALT(IALT),IALT=1,NALT)
19 FORMAT(4HALT ,T6,I2,(T11,7F10.0))
WRITE(6,21)MCAS,(XCAS(ICAS),ICAS=1,MCAS)
21 FORMAT(4HKTAS,T6,I2,(T11,7F10.0))
C
150 DO 200 IALT=1,NALT
WRITE(6,23)NCAS,(PSIDOT(IALT,ICAS),ICAS=1,NCAS)
23 FORMAT(9HTRTE,T6,2I2,(T11,7F10.1))

C
C
200 CONTINUE
WRITE(6,25)
25 FORMAT(601H )

C
C
C*********************************************************
CACMB INPUT ON TAPE8
C*********************************************************

IC=NALT/9+1.001
JE=0
DO 810 K=1,IC
J0=JE+1
JE=MIN0(J0+8,NALT)
DO 810 I=1,NCAS
IF(I.EQ.1)WRITE(8,801)(XALT(J),J=J0,JE)
801 FORMAT(9VEL/ALT 1,9F13.0)
WRITE(8,800)(XCAS(I),(PSIDOT(J,I),J=J0,JE))
800 FORMAT(10G13.5)
810 CONTINUE
C
C*********************************************************

C
READ(5,D)
IF(EOF(5))10,11

END
CINSTURN

PROGRAM INSTURN(INPUT,OUTPUT,TAPES=INPUT,TAPE10,TAPE6,
1TAPE8)
   COMMON/PRINT/NPRINT
   COMMON/IOUNITS/IP,IR,ID
   DIMENSION CTA(100),PSIDOT(100,100),XTAS(100),XALT(100),
   *TITLE(7),ARRAY(20),XCAS(100)
   NAMELIST/D/NCAS,NALT,DELCAS,DELTALT,IATYPE,CPIRP,UTIP,UNV,
1CAS1,ALT1,CTUAL,KATMOS,GFAC,TOQU,ADISK,TORFAC,PLIMIT,
2ETAM,ETAP,DELFU,NPRINT,TMN,TCHAR,MUCTSU,XMSN,
3IPRINT,CTMAX,TRV0,VB,DELHP,NTRNRT,DELTRT,ALT0,
4PSIMAX
   DATA IP,IR,ID/10,5,10/
   DATA TORFAC,PLIMIT,TM4N,TCHAR,MUCTSU/0.,1.E6,0.,1.,0./
   DATA ETAM,ETAP,DELFU/1.,1.,0./
   DATA XMSN,IPRINT,G/100.,0.,32.1741/
   DATA PSIMAX/60./
   READ(5,D)
   IF(EQF(5))10,9
10 STOP
9 CONTINUE
   READ(5,1)ITITLE
   1 FORMAT(7A10)
   CALL TREAD(1, XD, YD, ZD, FXYZ)
   WRITE(8,2)ITITLE
   PRINT(10,D)
   IF(IPRINT.NE.0)PRINT(10,"PSIDOT,G,CTMAX,V,ALT,CT,CPI,RHO")
   2 FORMAT(*AC2GA12*,7A10)
   WRITE(8,X)NCAS,NALT,CAS1,DELCAS,ALT1,DELTALT,IATYPE
   N=INT(UNV)
   L=1
   CTA(L)=CTUAL
   CT=CTA(L)
   ARRAY(1)=4
ARRAY(2) = -1.
ARRAY(3) = 0.
ARRAY(6) = ARRAY(?). TRUE
50 DO 100 IALT = 1, MALT
   XALT(IALT) = FLOAT(IALT - 1) * DELTALT
   KATMOS = 3.
   CALL ATMOS(KATMOS, XALT(IALT), TEMPR, PRESSR, RHO, SIGMA, VSND)
   GFAC = CTMAX * RHO * ADISK * UTIP * 2 / TOGW
60 DO 101 ICAS = 1, NCAS
   XCAS(ICAS) = FLOAT(ICAS - 1) * DELCAS + CAS1
   IF(XCAS(ICAS), EQ. 0.) XCAS(ICAS) = 1. E - 6
   XTAS(ICAS) = XCAS(ICAS) / (SIGMA)**.5
   KSIGN = 0
   IF(GFAC . LT . 0.) KSIGN = 1
   IF(GFAC . LT . 0.) GFAC = -GFAC
   IF(GFAC . GE . 1.) GO TO 40
   XXXX = 1. + (1. / ((1. - GFAC**2)**.5 - 1.))
   PSIDOT(IALT, ICAS) = (GXXX / (XTAS(ICAS) * 1.689)) * 57.29578
   IF(KSIGN . EQ . 1) PSIDOT(IALT, ICAS) = -PSIDOT(IALT, ICAS)
   GO TO 41
40 PSIDOT(IALT, ICAS) = (G*(GFAC**2 - 1)**.5 / (XTAS(ICAS) * 1.689)) * 57.29578
   IF(KSIGN . EQ . 1) PSIDOT(IALT, ICAS) = -PSIDOT(IALT, ICAS)
41 CONTINUE
C
   IF(IPRINT . NE . 0) PRINT(10,31) PSIDOT(IALT, ICAS), GFAC, CTMAX,
   * XCAS(ICAS), XALT(IALT), CT, CPIRP, RHO
31 FORMAT(5X,5(1H4),8G13.5)
101 CONTINUE
   NCAS1 = NCAS - 1
   DO 102 I = 1, NCAS1
   L = NCAS - I
   IF(XTAS(L) . LE. VB) 103, 102
103 ARRAY(10) = 0.
   ARRAY(11) = 1.
ARRAY(4)=XTAS(L)
ARRAY(5)=XTAS(L+1)
ARRAY(8)=PSIDOT(IALT,L)
ARRAY(9)=PSIDOT(IALT,L+1)
DO 105 K=1,L
105 PSIDOT(IALT,K)=SPLNQ1(1,ARRAY,XTAS(K))
     IF(XTAS(NCAS).LE.VB)PSIDOT(IALT,NCAS)=
         SPLNQ1(1,ARRAY,XTAS(NCAS))
     GOTO 100
102 CONTINUE
100 CONTINUE

********************************************************************
* THIS SECTION LOADS THE ARRAY IN THE                          *
* REQUIRED FORMAT OF TREAD                                      *
********************************************************************

DATA IT/100/
WRITE(6,17)IT,ITITLE
17 FORMAT(15,T11,7A10)
WRITE(6,18)GFAC
18 FORMAT(4HGFC,3H 1,T11,F10.0)
WRITE(6,19)NALT,(XALT(IALT),IALT=1,NALT)
19 FORMAT(4HALT ,T6,I2,(T11,7F10.0))
WRITE(6,21)NCAS,(XCAS(ICAS),ICAS=1,NCAS)
21 FORMAT(4HKTAS,T6,I2,(T11,7F10.0))

DO 200 IALT=1,NALT
200 CONTINUE
WRITE(6,25)
25 FORMAT(*EOT*/80(1H ))
C
C***************************************************************************
C ACMB INPUT ON TAPES
C***************************************************************************
IC=NALT/9+1.001
JE=0
DO 810 K=1,IC
JO=JE+1
JE=MINT(JO+8,NALT)
DO 810 I=1,NCAS
IF(I.EQ.1)WRITE(8,801)(XALT(J),J=JO,JE)
801 FORMAT(*VEL/ALT *,9F13.0)
WRITE(8,800)(XCAS(I),(PSIDOT(J,I),J=JO,JE))
800 FORMAT(10G13.5)
810 CONTINUE
C
C***************************************************************************
READ(5,D)
IF(EOF(5))10,11
END
CMANEU

PROGRAM MANEUVE (INPUT, OUTPUT, TAPE5=INPUT, TAPE10, TAPE6, TAPE8, TAPE9)
   COMMON/PRINT/NPRINT
   COMMON/IOUNITS/IP, IR, ID
   DIMENSION PSS (75, 75), XTAS (100), PSIDOT (100), TR (75),
     *V (100), ITITLE (7), XCAS (100), RAD (75, 75), GFAC (75, 75)
   NAMELIST/D/NCAS, NALT, DELCAS, DELTALT, IATYPE, CPIR, UTIP, UNV,
   1CAS1, ALT1, CTVAL, KATMOS, GFAC, TOGW, ADISK, TORFAC, PLIMIT,
   2ETAM, ETAP, DELFE, NPRINT, TMAN, TCHR, MUPTSU, XMSN, IPRINT,
   3DELHP, CMAX, TRV0, VB, ALT0, NTRNR, DELTR, TRT1
   DATA IP, IR, ID/10, 5, 10/
   DATA TORFAC, PLIMIT, TMAN, TCHR, MUPTSU/0., 1.E6, 0., 1., 0./
   DATA ETAM, ETAP, DELFE/1., 1., 0./
   DATA XMSN, IPRINT, DELHP/100., 0., 0./
   DATA G, NTRNR, DELTR, ALT0/32.1741, 24., 1., 0./
   DATA TRT1/1.
   READ(5, D)
   IF (EOF(5)) 10, 9
10 STOP
9 CONTINUE
   READ(5, 1) ITITLE
1 FORMAT (1A10)
   CALL TREAD (1, XD, YD, ZD, FXYZ)
11 WRITE (8, 2) ITITLE
   PRINT (10, D)
   IF (IPRINT .NE. 0) PRINT (10, *) 'PS, G, MU, U, ALT, CT, CPI, RHO, CPR'
2 FORMAT (*APS01G1*, 7A10)
   WRITE (8, *) NCAS, NTRNR, CAS1, DELCAS, TRT1, DELTR, IATYPE
   CALL ATOMOS (KATMOS, ALT0, TEMPR, PRESSR, RHO, SIGMA, VSND)
   X0ATC = (TEMPR - 491.69) * 5. / 9.
   CT1 = GFAC * TOGW / (RHO * ADISK * UTIP * 2)
**-----------------------------------**

```c
IF(TORFAC.EQ.0.)GO TO 58
PLIMIT=XMSN TORFAC
CALL TLOOK(10,XOATC,ALT0,0.,TORK)
PUR=TORK TORFAC
GO TO 59
58 CALL TLOOK(10,ALT0,XOATC,0.,PUR)
59 CONTINUE
IF(PUR.GT.PLIMIT)PUR=PLIMIT
CPIRP=(PUR*550.)/(RHO*ADISK*UTIP**3)
**-----------------------------------**

GMAX=CTMAX*RHO*ADISK*UTIP**2/TOGU
GTOP=GMAX+.5
50 DO 100 ITRNRT=1,NTRNRT
PSIDOT(ITRNRT)=FLOAT(ITRNRT-1)*DELTRT+TRT1
60 DO 100 ICAS=1,NCAS
   XCAS(ICAS)=FLOAT(ICAS-1)*DELCAS+CAS1
   XTAS(ICAS)=XCAS(ICAS)/SIGMA**.5
   V(ICAS)=XTAS(ICAS)*1.689/UTIP
   GFAC(I=ITRNRT,ICAS)=
   *(PSIDOT(ITRNRT)*XTAS(ICAS)**.029479/G)**2+1.**.5
   IF(TMAN.EQ.0.)GO TO 39
   PUR1=PUR*TMAN*EXP(TCHAR*XTAS(ICAS))+DELHP*XTAS(ICAS)
   IF(PUR1.GT.PLIMIT)PUR1=PLIMIT
   CPIRP=(PUR1*550.)/(RHO*ADISK*UTIP**3)
   CONTINUE
CT=CTI*GFAC(ITRNRT,ICAS)
   IF(MUCSTU.EQ.0.)CALL TLOOK(9,CT,V(ICAS),0.,CPREQ)
```
IF(MUCNTY.NE.0)CALL TLOOK(9,UCAS),CT,0.,CPREQ)
DELCPR=.5*DELPF(VICAS)**3/(ADK*ETAX)*ETAP)
CPREQ=CPREQ+DELCPR
PSS(ITNR,T,ICAS)=(CPRP-CPREQ)**875*90*UTIP*60./CT
C**************************************************************
RAD(ITNR,T,ICAS)=1.689*57.29578*XTAS(ICAS)/PSIDOT(ITNR)
C**************************************************************
IF(GFACT(FF,ICAS).GT.GTOP)GFACT(FF,ICAS)=-2.E10
C
IF(IPRINT.NE.0)PRINT(10,31)PSS(FF,ICAS),GFACT(FF,ICAS)
*V(ICAS),XTAS(ICAS),PSIDOT(FF),CT,CPRP,RHO,CPREQ
31 FORMAT(5X,5(I1%),5G13.5)
100 CONTINUE
C
C*****************************************************************************
C THIS SECTION LOADS THE ARRAY IN THE
C REQUIRED FORMAT OF TREAD
C
C DATA IT/100/
WRITE(6,17)IT,ITITLE
17 FORMAT(15,T11,7A10)
WRITE(6,18)GFAC
18 FORMAT(4HGFAC,3H 1,T11,F10.0)
WRITE(6,19)NTRNR,PSIDOT(FF),ITNR=1,NTRNR)
19 FORMAT(4HALT ,76,I2,(T11,F10.0))
WRITE(6,21)NCAS,XCAS(FCAS),CAS=1,NCAS)
21 FORMAT(4HKTAS,T6,I2,(T11,F10.0))
C
150 DO 200 ITNR=1,NTRNR
WRITE(6,23)NCAS,PSS(FF,ICAS),ICAS=1,NCAS)
23 FORMAT(4PSSBS,T6,I2,(T11,F10.1))
C
200 CONTINUE
WRITE(6,25)
   25 FORMAT(*E0T*/80(1H ))
C
C
C*******************************************************************************
C   ACM6 INPUT ON TAPE8
C*******************************************************************************
IC=NTRNRT/9+1.001
JE=0
DO 810 K=1,IC
   JO=JE+1
   JE=MIN(JO+8,NTRNRT)
   DO 810 I=1,NCAS
      IF(I.EQ.1)WRITE(8,801)(PSIDOT(J),J=JO,JE)
801 FORMAT(*VEL/ALT *,9F13.5)
      WRITE(8,800)(XCAS(I),(PSS(J,I),J=JO,JE))
800 FORMAT(10G13.5)
810 CONTINUE
C*******************************************************************************
WRITE(8,2)ITITLE
   WRITE(8,*)(NCAS,NTRNRT,CAS1,DELCAS,TRT1,DELTTRT,IATYPE)
C*******************************************************************************
IC=NTRNRT/9+1.001
JE=0
DO 860 K=1,IC
   JO=JE+1
   JE=MIN(JO+8,NTRNRT)
   DO 860 I=1,NCAS
      IF(I.EQ.1)WRITE(8,851)(PSIDOT(J),J=JO,JE)
851 FORMAT(*VEL/ALT *,9F13.5)
      WRITE(8,850)(XCAS(I),(GFACT(J,I),J=JO,JE))
850 FORMAT(10G13.5)
860 CONTINUE
C*******************************************************************************
WRITE(8,2)ITITLE
   NTRNRT=NTRNRT-1
WRITE(8,*)NCAS,NTRNRT,CAS1,DELCAS,PSIDOT(2),DELTTRT,IATYPE
C*****************************************************************************
IC=NTRNRT/9+1.001
JE=1
DO 910 K=1,IC
JO=JE+1
JE=MIN0(JO+8,NTRNRT)
DO 910 I=1,NCAS
IF(I.EQ.1)WRITE(8,901)(PSIDOT(J),J=JO,JE)
   901 FORMAT(*VEL/ALT *,9F13.5)
WRITE(8,900)(XCAS(I),(RAD(J,I),J=JO,JE))
   900 FORMAT(10G13.5)
910 CONTINUE
C*****************************************************************************
IQ=1
IF(XCAS(1).EQ.0.)IQ=2
NCAS1=NCAS+1-IQ
DO 950 ICAS=IQ,NCAS
   950 TR(ICAS)=G*((GMAX**2-1.)*.5/(XTAS(ICAS)*1.689))
   **57.29578
   ITIT=4HMACH
WRITE(9,951)ITIT,NCAS1,(XCAS(I),I=IQ,NCAS)
   951 FORMAT(A4,1X,12,%612x.(T11,7F10.5))
WRITE(9,952)XCAS(NCAS-5),TR(NCAS-5)
   952 FORMAT(*....@000*,T11.*.15*,T21,F10.5,T31,F10.5,
       */ROTOR LIMIT*)
C*****************************************************************************
C
READ(5,D)
APPENDIX C
PROGRAM LOGIC

<table>
<thead>
<tr>
<th>Program</th>
<th>Page No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specific Excess Power</td>
<td>C-3</td>
</tr>
<tr>
<td>Sustained Turn Rate</td>
<td>C-13</td>
</tr>
<tr>
<td>Instantaneous Turn Rate</td>
<td>C-25</td>
</tr>
<tr>
<td>Maneuvering Capability</td>
<td>C-36</td>
</tr>
</tbody>
</table>
PROGRAM PSHEL0(INPUT,OUTPUT,TAPE6=INPUT,TAPEIO,TAPE6,TAPE6)

COMMON/PRINT/MPRINT
COMMON/IOUNIT5/IP,IR,IO

DIMENSION CTA(100),PS6(100,100),XTAS(100),XALT(100) ; V(100),ITITLE(7),XCA8(100)

NAMELIST/O/NCAS,XALT,DELCA8,DELALT,IRTYPE,CP,RP,VV,UNV.
CAS1,ALT1,CTVAL,KATNS,DIFAC,TOON,ADISK,TORFAC,PLIMIT.
ETAN,ETAP,DELFE,MPRINT,THAN,TCHAR,NUCT6W,XSNX,PRINT,
DELHP,LMAX,Tavo,VE,NTNKT,DELRT,ALTO.
PSIMAX

DATA IP,IR,IO/10,6,10/
DATA TORFAC,PLIMIT,THAN,TCHAR,NUCT6W/0..1.EB,0..1,0/
DATA PSIMAX/60./
DATA ETAN,ETAP,DELFE/1,1,0./
DATA XSNX,PRINT,DELHP/100,0,0./
DATA THAN,TCHAR,PRINT/0..1,0./
READ(6,0)

IF(EOF(5))10,9

CONT. ON PG 2
C-3
PG 1 OF 10
NV=INT(UNV)
L=1
CTA(L)=CTVAL
CT=CTA(L)

DO 100 IALT=1,NALT

XALT(IALT)=FLOAT(IALT-1)*DELTALT

CALL ATMOS(KATMOS,XALT(IALT),TEMPR,PRESS,RHO,SIGMA,VSN0)

CT=GFC=TOGN/(RHO=ADISK*VTIP=2)
XORTC=(TEMPR-491.6S)=6./9.

IF(TORFAC.EQ.0.)

GO TO 50

PLIMIT=XSN0=TORFAC

CONT. ON PG 4
CALL TLOOK(10,XOATC,XALT(IALT),0..TORK)

PWR=TORK*TORFAC

GO TO 58

CALL TLOOK(10,XALT(IALT),XOATC,0..PWR)

58 CONTINUE

IF(PWR>PLIMIT)

F

T

PWR=PLIMIT

CPIRP=(PWR=550.)/(RHO=RDISK=VTIP=3)

CONT. ON PO 6

C-6

PO 4 OF 10
CONT. ON PG 6
NADC-82136-60

IF(NUCTSW.EQ.0)

CALL TLOOK(S.CT.V(ICAS).O.CPREG)

IF(NUCTSW.NE.0)

CALL TLOOK(S.V(ICAS).CT.O.CPREG)

DELCFR=.S=DELF=V(ICAS)=3/(AOISK=ETAM=ETAP)

CPREG=CPREG+DELCFR

F88(IALT.ICAS)=(CPHRCPREG)=80.*.875=90=VTIP/CT0

IF(IPRINT.NE.0)

PRINT(10.3)F88(IALT.ICAS).OFAC.V(ICAS).

CONT. ON PG 7
C THIS SECTION LOADS THE ARRAY IN THE
C REQUIRED FORMAT OF TREAD

DATA IT/100/
WRITE(6,17)IT,ITITLE

17
FORMAT(15,T11.7A10)

WRITE(6,18)OFAC

18
FORMAT(4HOFAC,3H 1,T11,F10.0)

WRITE(6,19)NALT,(XALT(IALT),IALT=1,NALT)

19
FORMAT(4HALT,7B12,(T11,7F10.0))

WRITE(6,21)NCAS,(XCAS(ICAS),ICAS=1,NCAS)

CONT. ON PO 8
NADC-82136-60

21

FORMAT(4HKIAS.T8.12.(T11.7F10.0))

180

D0 200 IALT=1.NALT

WRITE(8.23)NCAS,(PSS(IALT,ICAS),ICAS=1,NCAS)

23

FORMAT(4HP886.T8.12.(T11.7F10.1))

8 --> > 200

CONTINUE

WRITE(6.25)

25

FORMAT(=EOT=/BO(1H ))

C-10

ACME INPUT ON TAPE

IC=ICALT/8+1.001
JE=0

D0 810 K=1.IC

CONT. ON PG 9

C-10

PG 8 OF 10
NADC-82136-60

\[ \text{JO} = \text{JE} + 1 \]
\[ \text{JE} = \text{MINO} (\text{JO} + 8, \text{NALT}) \]

\[ \text{DO 810 I} = 1, \text{NCA6} \]

\[ \text{IF(I.EQ.1)} \]

\[ \text{WRITE}(8.001)(\text{XALT}(J)), J = \text{JO}, \text{JE} \]

\[ \text{FORMAT}(=\text{VEL/ALT}=.9F13.0) \]

\[ \text{WRITE}(8.000)(\text{XCA6}(I)), (\text{PSI}(J,I)), J = \text{JO}, \text{JE} \]

\[ \text{FORMAT}(=10013.5) \]

\[ \text{CONTINUE} \]

\[ \text{READ}(5.0) \]

CONT. ON PG 10

PG 9 OF 10

C-11
CONTINUE

READ(6,1)ITITLE

CALL TREAD(1.XD,YD.ZD.FXYZ)

WRITE(8,2)ITITLE

PRINT(10,0)

IF(IPRINT.NE.0)

PRINT(10,"*PE100T.0,HU.V,ALT,CT,CP. RHO")

WRITE(8,"NCA6,NALT,CAS1,DELCA6,ALT1,DELTALT,IA TYPE")

CONT. ON PG 3
CALL TLOOK(I0,XOATC,XALT(IALT),0,,TORK)

PWR=TORK=TORFAC

GO TO 69

CALL TLOOK(I0,XALT(IALT),XOATC,0,,PWR)

CONTINUE

IF(PWR.GT.PLIMIT)

F

T

PWR=PLIMIT

CPIRF=(PWR=5d0.)/(RHO=ADIGK*VTIP=3)

C

CONT. ON PG 5

C-16
DO 100 ICAS = 1, NCA6

XCA6(ICAS) = FLOAT(ICAS - 1) = DELCA6 + CA6

IF(XCA6(ICAS) .EQ. 0.)

XCA6(ICAS) = 1.E-6

XTAS(ICAS) = XCA6(ICAS) / (SIOMA) = .5

IF(TMAS .EQ. 0.)

GO TO 39

PWR1 = PWR * TAMAN = EXP(TCHAR * XTAS(ICAS)) + DELHP = XTAS(ICAS)

IF(PWR1 .GT. PLIMIT)

CONT. ON PG 6
QFAC=CT*RH*AO18K*VTIP=2/TOGN

KSIGN=0

IF(QFAC.LT.0.)

KSIGN=1

IF(QFAC.LT.0.)

QFAC=QFAC

IF(QFAC.GE.1.)

GO TO 40

XXXX=1.0+1.0/(1.0-QFAC=2)*.5-1.0

CONT. ON PG 8
CONTINUE

IF(IPRINT.NE.0)

PRINT(10,31)PSDOT(IALT,ICAS).OFAC.V(ICAS).
XCAS(ICAS).XALT(IALT).CT.CPIRP.RHO

FORMAT(6X,5(H=1003.6))

CONTINUE

C THIS SECTION LOADS THE ARRAY IN THE : 
C REQUIRED FORMAT OF TREAD

DATA IT/100/ :
WRITE(6.17)IT.ITITLE

17

FORMAT(15.I11,7A10)

WRITE(6.18)OFAC

CONT. ON PG 10

c-21
WRITE(8,800)(XCA8(I),PSIDOT(J,I),J=J0,JE))

100 FORMAT(10013.6)

CONTINUE

READ(5,0)

IF(EOF(5))10,11

C

END
IF(XCAR6(ICAR6).EQ.0.)

T

XCAR6(ICAR6)=1.E-6

XTR6(ICAR6)=XCAR6(ICAR6)/(SIGMA)=6

K5IGN=0

IF(GFAC.LT.0.)

F

T

K5IGN=1

IF(GFAC.LT.0.)

F

T

GFAC=-GFAC

CONT. ON PG 6

C-28
CONTINUE

IF (IPRINT.NE.0)

PRINT(10,3) PSIDOT(IALT,ICAS),GFAC,CTMAX,
XCAS(ICAS),XALT(IALT),CT,CPIRF,RHO

FORMAT(6X,6(IH=9013.6))

CONTINUE

NCAS1=NCAS-1

DO 102 I=1,NCAS1

L=NCAS-1

CONT. ON PO 7
This section loads the array in the required format of READ.

```
DATA IT/100/;
WRITE(*,17)IT,ITITLE

FORMAT(15,T11,7F10.0)
WRITE(*,18)GFAC

FORMAT(4HGFAC,3H T11,F10.0)
WRITE(*,19)NALT,(XALT,IALT),IALT=1,NALT

FORMAT(4HALT,10,I2,(T11,7F10.0))
WRITE(*,21)NCAS,(XCAS,ICAS),(CAS=1,NCAS)
```

Cont. on PG 8

C-32
NADC-82136-60

21

FORMAT(4HMTAS.T6.I2.(T11.7F10.0))

160

DO 200 IALT=1,NALT

WRITE(6,23)ICAS.(PSIDOT(IALT,ICAG),ICAG=1,NCAS)

23

FORMAT(4HTRIE.T6.I2.(T11.7F10.1))

CONT

200

CONTINUE

WRITE(6,26)

26

FORMAT(=E0T=80(1H))

C==================================================================================================
C ACME INPUT ON TAPE
C==================================================================================================

IC=NALT/8+1.001
JE=0

DO 810 K=1,IC

CONT. ON PG 10

C-33
JO=JE+1
JE=MIN(JO+9,NALT)

DO 810 I=1,NCA6

IF(I.EQ.1)

WRITE(8,801)(XALT(J).J=JO,JE)

801
FORMAT(*VEL/ALT = S13.0)

WRITE(8,800)(XCA6(I),(FS1QT(J,I).J=JO,JE))

800
FORMAT(10013.5)

10.9 -- -- P 810
CONTINUE

C

READ(5,0)

CONT. ON PG 11

PO. 10 OF 11
1 A2
   CONTINUE

READ(5,1) TITLE

FORMAT(7A10)

CALL TREAD(1, XD, YD, ZD, FXYZ)

WRITE(0,2) TITLE

PRINT(10.0)

IF (IPRINT .NE. 0) F
   T

PRINT(10.0) "PG.0, MU, V, ALT, CT, CPI, RHO, CPR"

FORMAT(=AP5010=,7A10)

WRITE(0,6) MCAS, NTANRT, CAS1, DELCAS, TRT1, DELTRT, IATYPE

CONT. ON PG 3

C-37
CALL ATMOS(KATMOS.ALTO,TEMPR,PRES&R, RHO, SIGMA, V6NO)

XORAC=(TEMPR-481.69)=6./9.
CTI=0FAC=TOOM/(RHO=ADISK*VTIP=2)

IF(TORFAC.EQ.0.)

CALL TLOOK(10, XORAC, ALTO, 0..TORK)

CONT. ON PO 4
CALL TLOOK(10, ALTO, XORAC, 0..PWR)

CONTINUE

IF (PWR >= PLIMIT)

PWR = PLIMIT

CPVR = (PWR = 550.)*(RHO = ADIGK = VTIP = 2)

OMAX = CMAX = RHO = ADIGK = VTIP = 2 / TGGN
GTOP = OMAX = 0

GO 100 ITRNRT = 1, NTRNRT

PSIODT(1TRNRT) = FLOAT(1TRNRT - 1) = DELTRT + TRT1

CONT. ON PG 5
NADC-82136-60

60

00 100 ICAS=1.0CAS

XCAS(ICAS)=FLOAT(ICAS-1)+DELCAS+CA61
XTAS(ICAS)=XCAS(ICAS)/SIGMA=.5
V(ICAS)=XTAS=ICAS=1.688/VTIP

GFACT(ITRNR,ICAS)=
((P810T=ICAS)*XTAS=ICAS)+.029479/G)1+1111=+5

IF(TMN=EQ.0.) T F

GO TO 39

PWR1=PWR1/TMN*EXP(TCHAR*XTAS=ICAS)+DELPH*XTAS=ICAS

IF(PWR1=GT.PLIMIT) T F

PWR1=PLIMIT

CPERP=(PWR1=880.)/(RHO=ROK*XTIP=3)

CONT. ON PG 6

C-40

PG 5 OF 16
IF(MUCTSW = 0)

CALL TLOOK(B.CT.V(ICA6).0..CPREG)

IF(MUCTSW = 0)

CALL TLOOK(B.V(ICA6).CT.0..CPREG)

DELCPR = 0 = DELFE = V(ICA6) = 0/(ADISK = ETAR = ETAP)

CPREG = CPREG + DELCPR

POS(ITRNRT.ICA6) = (CPRF - CPREG) = @76 = @0 = VTF = 00 / CT

CONT. ON PG 7

C-41
NADC-82136-60

\[ \text{RADO}_\text{ITRANT, ICAS} = 1.688 \times 57.29678 \times \text{XTAS(ICAS)} / \text{PSI DOT(ITRANT)} \]

\[ \text{C} = \frac{\text{OFACIT(ITORNT, ICAS)}}{0.4 \text{TOP}} \]

\[ \text{IF(OFACIT(ITORNT, ICAS)) = 0} \]

\[ \text{OFACIT(ITORNT, ICAS)} = -2.0 \times 10 \]

\[ \text{IF(IPRINT.NE.0)} \]

\[ \text{PRINT(10.3)} \times \text{PSI(ITORNT, ICAS)} \times \text{OFACIT(ITORNT, ICAS)} \]

\[ \times \text{V(ICAS)} \times \text{XTAS(ICAS)} \times \text{PSI DOT(ITORNT)} \times \text{CT,CPRT,RHO,CPREG} \]

\[ \text{FORMAT} \left( \text{EX, 6.1H}, \text{9013.5} \right) \]

\[ 5.4 \rightarrow 100 \]

\[ \text{CONTINUE} \]

\[ \text{C} \]}

\[ \text{THIS SECTION LOADS THE ARRAY IN THE} \]

\[ \text{C} \]}

\[ \text{REQUIRED FORMAT OF TREAD} \]

\[ \text{CONT. ON PO 8} \]

C-42
DATA IT/100/;
WRITE(*,17)IT,ITITLE

17
FORMAT(15,T11,7A10)

WRITE(6,18)OFAC

18
FORMAT(4HOFAC,3H1,T11,F10.0)

WRITE(6,19)NTRANRT,PSIDOT(ITRANRT),ITRANRT=1,NTRANRT

19
FORMAT(4HALT,6E12,7F10.0)

WRITE(6,21)NCAS,(XCAS(ICAS),ICAS=1,NCAS)

21
FORMAT(4HKTAB,6E12,7F10.0)

150
DO 200 ITRANRT=1,NTRANRT

WRITE(6,25)NCAS,(PSS(ITRANRT,ICAS),ICAS=1,NCAS)

CONT. ON PO 8
CONTINUE

WRITE(6,26)

FORMAT('IC=MTRNRT/8+1.001,JE=0')

DO 810 K=1,IC

DO 810 I=1,NCAS

CONT. ON 10
NADC-82136-60

IF(I.EQ.1)
  WRITE(8.801)(PSI00T(J),J=JO,JE)
  FORMAT(MVEL/ALT=M9F13.5)
  WRITE(8.800)(XCA6(I),(PS8(J,I),J=JO,JE))
  FORMAT(10013.5)
  CONTINUE

WRITE(8.801)(TITLE
WRITE(8.8)INCA6,NTMRT,CAS1,DELCA6,TRT1,DELTRT,IATYPE

IC=NTMRT/3+1.001

CONT. ON PG 11