NASA
Technical
Paper
1851

August 1982

NASA-TP-1851 19820025144

Automated Procedure for Developing Hybrid Computer Simulations of Turbofan Engines

I-General Description

John R. Szuch,
Susan M. Krosel, and
William M. Bruton

NASA

NASA
Technical
Paper
1851

1982

Automated Procedure for Developing Hybrid Computer Simulations of Turbofan Engines

I - General Description

John R. Szuch,
Susan M. Krosel, and
William M. Bruton
Lewis Research Center
Cleveland, Ohio

NASA
National Aeronautics
and Space Administration

Scientific and Technical
Information Branch

Contents

Summary

Computer simulations provide a means of analyzing the behavior and interactions of complex dynamic systems such as aircraft propulsion systems. The hybrid (analog-digital) computer offers the opportunity to combine the best features of analog and digital computation to satisfy the requirements of engine simulation. Unfortunately there are a number of difficult problems to be faced when developing hybrid computer simulations. This report addresses some of those problems and offers a systematic, computer-aided, self-documenting methodology for developing hybrid computer simulations of turbofan engines. The concepts and computer codes presented are applicable to simulations of other engine types.

The proposed methodology makes use of a host program, written in Fortran, that can run on a large digital computer. The host program performs all the calculations and data manipulations that are needed to transform user-supplied engine design information to a form suitable for the hybrid computer. The companion target (hybrid) computer program is, of course, machine dependent. As presented, the target program consists of a scaled-fraction Fortran program that runs on the EAI PACER 100 digital computer and a prescribed patching arrangement for the EAI 681 analog computer. A test case is described and comparisons between hybrid and user-specified engine performance data are presented. Steady-state results are shown to be within 2 to 3 percent of the user-specified data over the entire engine operating line. A number of simulated engine transients were run on the hybrid computer to demonstrate the transient capabilities of the simulation. Finally extensions of the host-target simulation concept are discussed, including planned investigations of model simplifications, alternative analog-digital computation splits, parallel processing, and other approaches to achieving real-time operation.

Introduction

The development of aircraft propulsion systems depends, to a great extent, on being able to predict the performance of the propulsion system and its associated controls. Computer simulations provide the means for analyzing the behavior and interactions of these increasingly complex systems prior to building and testing expensive hardware. Simulations can also serve as aids in understanding and solving problems that arise after the propulsion system is developed.

Digital computer simulations with the necessary steady-state and transient capabilities have been developed. Simulations of specific engines and also generalized codes for simulating a range of engine types are available (refs. 1 to 6). Although these digital simulations offer many advantages, they do consume a great deal of computing time since they generally require iterative solutions and numerical integration. This restricts their usefulness in many applications.

Analog computer simulations can provide high-speed (faster than real time) solutions, interaction with the user, and hardware-in-the-loop capability. However, analog computers are not widely used for engine simulations because of the large amounts of computing equipment and the time required to set up that equipment, particularly if many multivariate functions have to be generated. The digital computer is, of course, better suited for the function generation task.

The hybrid (analog plus digital) computer offers the opportunity to use the best features of digital and analog computation to satisfy the requirements of engine simulation. By appropriately splitting the computation load between the analog and digital processors, steady-state and dynamic accuracy can be achieved with reasonable solution times (at or near real time) with a significant reduction in analog equipment. In addition, the user is provided "hands on" interactive control of the simulation with convenient display and recording of simulation results. Real-time, hybrid computer simulations of the Pratt & Whitney TF30-P-3 and F100-PW-100 turbofan engines have been developed (refs. 7 to 9) and used to support the development of advanced electronic engine controls (refs. 10 to 13) at the Lewis Research Center. In these applications the engine simulations served as "test beds" for evaluating new control laws and for verifying control software prior to engine testing.

Unfortunately the development of accurate hybrid computer simulations is often viewed as an "art" that requires specialists experienced in dynamic system modeling, computer programming, and computer operations. There are, in fact, a number of difficult problems that must be dealt with when developing hybrid simulations. As shown in figure 1 these include (1) the *formulation* of a mathematical model that is detailed enough for the particular application yet does not require

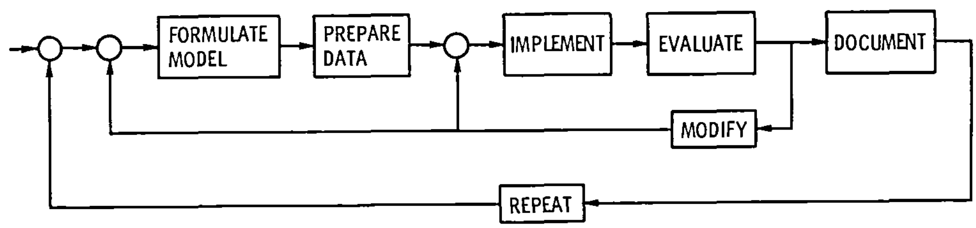


Figure 1. - Simulation development process.

excessive computing equipment or consume excessive computing time, (2) the *preparation* of simulation design data so as to put numerical values into the mathematical model, (3) the *implementation* of the simulation on the hybrid computer, (4) the *evaluation* of the simulation model relative to available design and off-design engine data, (5) the *modification* of the simulation, if necessary, to match the reference data, (6) the *documentation* of the simulation and its development to facilitate changes and extensions of the simulation design as requirements and applications change, and (7) *repetition* of the development process for each new engine to be studied.

This report addresses these problems and offers a systematic, computer-aided, self-documenting methodology for the development of a hybrid computer simulation of a typical turbofan engine. This methodology represents an extension of many of the ideas presented in reference 14. However, reference 14 concentrated on the techniques for generalizing the simulation to handle many different engine configurations. This report focuses on one engine type and concentrates on the automation and documentation of the simulation development process. It is the feeling of the authors that the concepts and computer codes presented in this report are sufficiently general to permit their adaptation to simulations of other engine types. However, the problem of generalization is not addressed in this report.

The proposed methodology is illustrated in figure 2. A host program, written in Fortran, runs on a suitable large-scale digital computer (in this case, an IBM 370/3033). The host program performs all the precalculations and data manipulations that are needed to transform user-supplied engine design information to a form suitable for the hybrid computer. The host program trims the self-contained engine model to match user-supplied "design" point performance data and can also provide a quantitative measure of each engine component model's accuracy relative to supplied "off design" point data if such data are available. This permits off-line refinement and evaluation of the models by using the host program, prior to implementation of the hybrid simulation. Finally the host program provides printouts of simulated engine parameters and punched cards containing the hybrid computer setup and control information.

The target (hybrid) simulation, as presented, runs on the Electronic Associates Inc. (EAI) PACER 600 hybrid computer system. That system consists of a 32K 16-bit digital processor, a 10-volt analog computer, and an

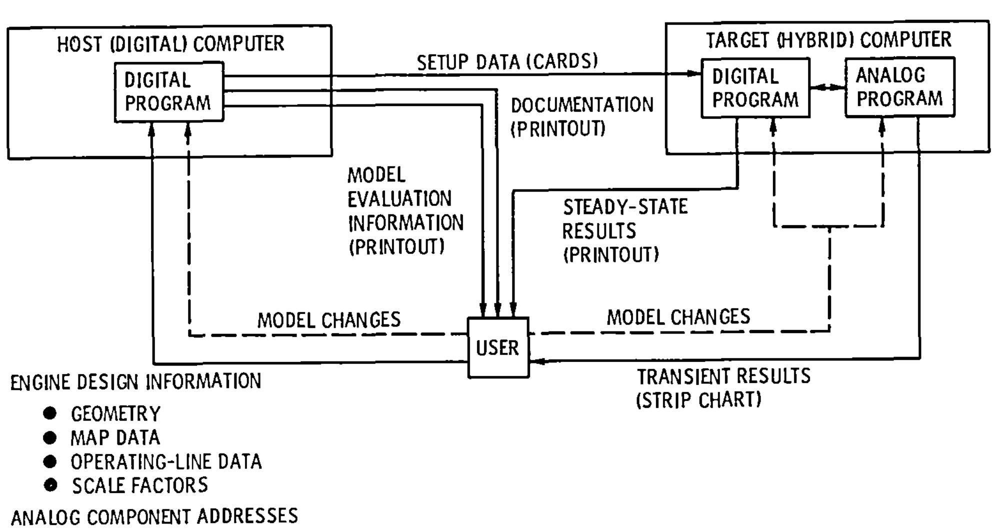


Figure 2. - Proposed simulation development methodology.

interface unit that provides communication between the analog and digital machines. The target simulation consists of digital subroutines and function routines and a prescribed analog patching arrangement. The digital routines do the following: (1) set up the analog computer components, including setting of potentiometers and integrator gains, (2) compute state variable derivatives based on sampled values of state variables and simulated engine inputs such as fuel flow, and (3) provide printouts of selected steady-state data. The analog computer is used, primarily, for integration of the state variable derivatives. Strip-chart recorders are used to monitor and record the transient behavior of the simulation. Twenty-five analog-to-digital converters (ADC's) and 21 digital-to-analog converters (DAC's) are used to transfer information between the analog and digital processors.

Part I of this report contains a detailed description of the turbofan engine model and the host and target programs. Source listings of digital routines and analog patching diagrams are provided along with program statistics. Results are presented for a test case (a turbofan engine operating at sea-level, static conditions between idle and maximum power) including time histories of simulation transients. Computer printouts of host program input and output data for the test case are presented in Part II of this report (ref. 15). However, Part II is not required to understand the concepts and computer codes presented in Part I.

The simulation, as presented, does not run in real time on the hybrid computer. This is due to the level of detail in the engine model, the extensive use of digital computation in the hybrid simulation, and the limited speed of the PACER 100 digital computer. However, the computer programs that are presented should provide an excellent baseline from which to explore model simplifications, different analog-digital computational splits, parallel processing, and other approaches to achieving real-time operation within the framework of the presented methodology. The computer-aided, self-documenting nature of the methodology should facilitate such studies. Nonetheless, the simulation, as presented, represents a significant improvement over batch running of comparable all-digital simulations.

Engine Model

The mathematical model describing the performance of a two-spool, augmented turbofan engine is patterned after the generalized engine model presented in reference 14. Overall performance maps are used to provide accurate steady-state representations of the engine's rotating components. The effects of variable fan and compressor geometry are accounted for by using baseline maps that correspond to nominally scheduled geometry and then biasing the baseline map outputs by functions of the actual geometry. Factors such as fluid momentum, mass and energy storage, and rotor inertias are included to provide transient capability. To improve steady-state accuracy, the effects of temperature and fuel-air ratio on thermodynamic gas properties are also included. The following sections describe the turbofan engine configuration and the individual component models.

Engine Configuration

Figure 3 contains a schematic representation of a two-spool, augmented turbofan engine. A single inlet is used to supply airflow to the fan. Air leaving the fan is separated into two flow streams: One stream passes

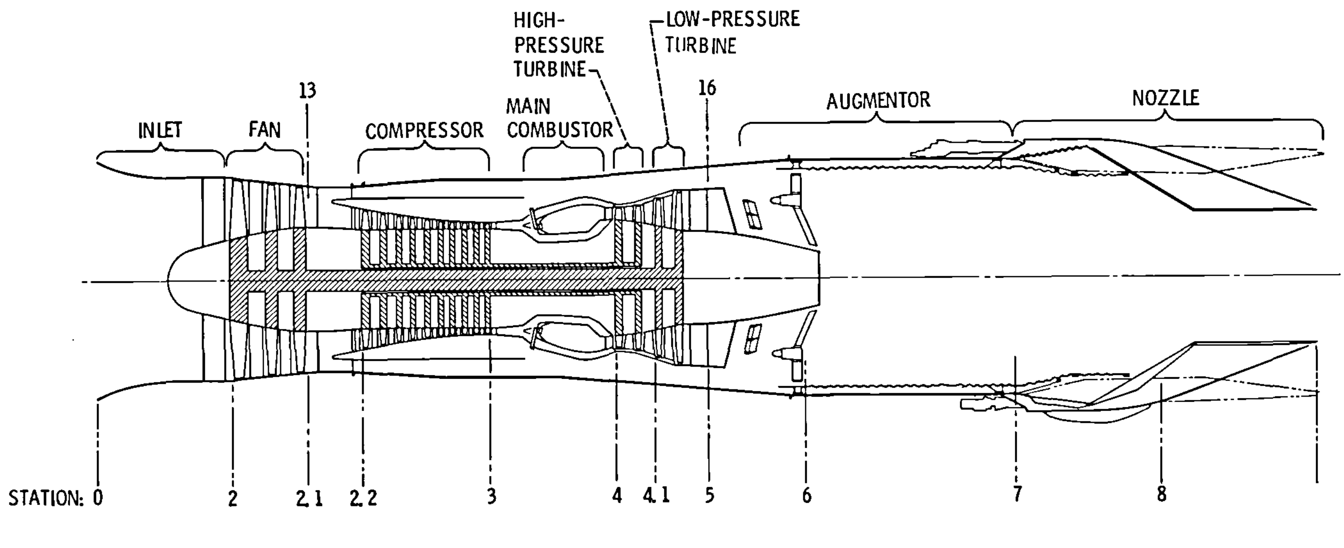


Figure 3. - Schematic representation of augmented turbofan engine.

through the engine core; the other stream passes through an annular bypass duct. The fan is driven by a low-pressure turbine. The core airflow passes through a compressor that is driven by a high-pressure turbine. Both the fan and compressor are assumed to have variable geometry (vanes) to improve stability at low speeds. Engine airflow bleeds are extracted at the compressor exit (station 3) and used for turbine cooling (flow returns to the cycle) and for accessory drives (flow lost to the cycle). Fuel flow is injected in the main combustor and burned to produce hot gas for driving the turbines. The engine core and bypass streams combine in an augmentor duct, where additional fuel is added to further increase the gas temperature (and hence thrust). The augmentor flow is discharged through a variable convergent-divergent nozzle. The nozzle throat area (station 8) and exhaust area (station 9) are varied to maintain engine airflow and to minimize drag during augmentor operation. Figure 4 contains a computational flow diagram of the engine model. All symbols are defined in appendix A. The following sections describe the steady-state and dynamic elements in the engine model.

Steady-State Models

Flight condition and inlet. – For any simulated flight condition (altitude and flight Mach number) it is necessary to compute the fan inlet conditions (total pressure and temperature) and the back-pressure on the nozzle. A steady-state, military specification inlet recovery characteristic is used together with standard atmospheric data. The following equations define the flight condition and inlet model in the simulation:

$$P_0 = f_1(a) \tag{1}$$

$$T_0 = f_2(a) + T_{am} \tag{2}$$

$$\left.\begin{aligned} \eta_I &= 1.0 \qquad \text{if } M_0 \leq 1.0 \\ &= 1.0 - 0.075\,(M_0 - 1.0)^{1.35} \qquad \text{if } M_0 > 1.0 \end{aligned}\right\} \tag{3}$$

$$T_2 = T_0\left[1.0 + \frac{(\gamma_I - 1)M_0^2}{2}\right] \tag{4}$$

$$P_2 = P_0 \eta_I \left(\frac{T_2}{T_0}\right)^{\gamma_I/(\gamma_I - 1)} \tag{5}$$

$$\gamma_I = \gamma_0 = 1.4 \tag{6}$$

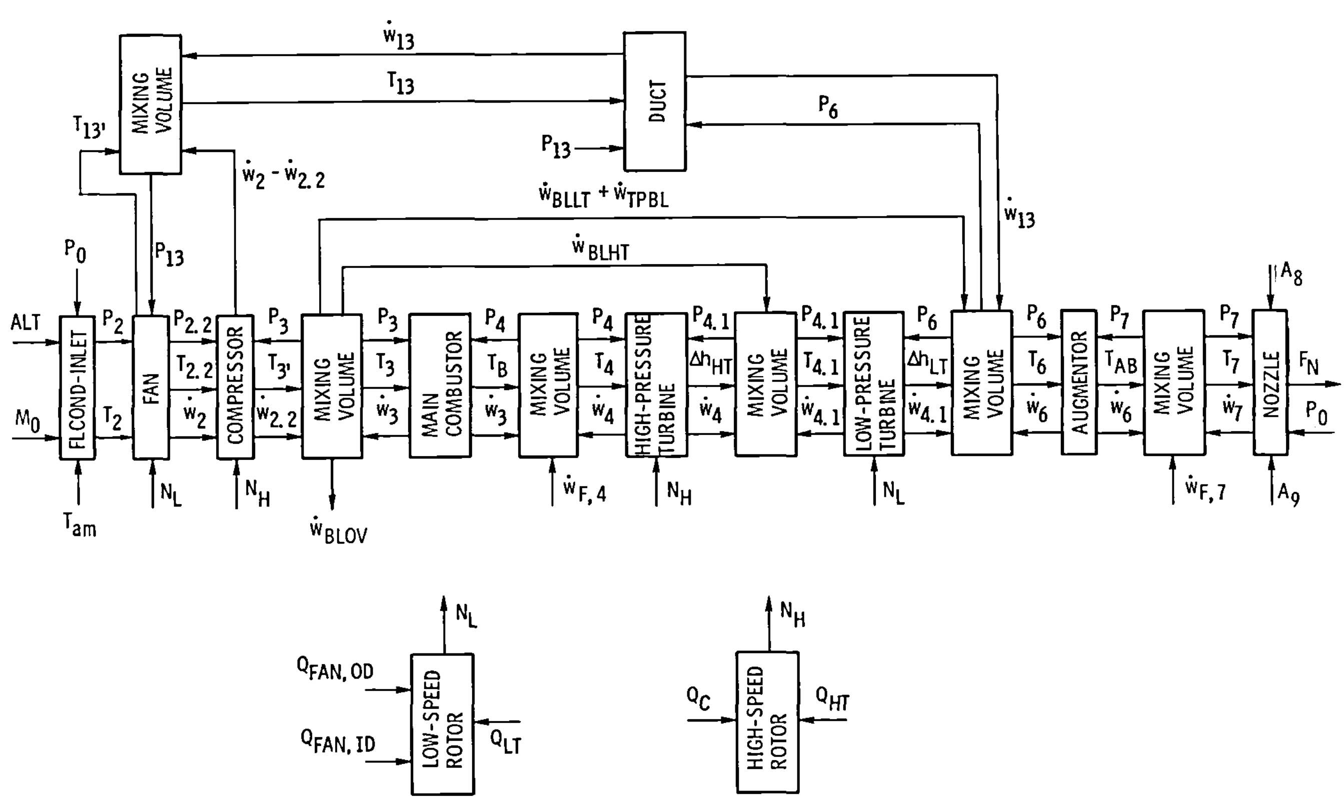


Figure 4. - Computational flow diagram of augmented turbofan engine simulation.

The functions f_1 and f_2 are curve fits of atmospheric data from reference 16.

Gas properties. – In general, the engine component models utilize variable thermodynamic gas properties. Curve fits of data found in reference 17 are used to compute specific heats, specific heat ratios, and specific enthalpies from given values of temperature and fuel-air ratio. (JP-4 is assumed as the fuel.) For each intercomponent volume, the following equations are used:

$$c_p = f_3(T, f/a) \tag{7}$$

$$R = f_4(f/a) \approx R_A \tag{8}$$

$$c_v = c_p - \frac{R}{J} \tag{9}$$

$$\gamma = \frac{c_p}{c_v} \tag{10}$$

$$h = f_5(T, f/a) \tag{11}$$

Fan. – Fan performance is represented by a set of overall performance maps. This technique does not account for interstage dynamics in the frequency range of interest ($f < 10$ Hz). Separate maps (functions) are used for the tip (bypass) and hub (core) sections of the fan to account for the pressure gradient that can occur at the fan exit. The maps are assumed to represent the fan performance with the variable geometry (inlet guide vanes) at nominal, scheduled conditions. The effects of off-schedule geometry are included in the model by adjusting the map-generated value of corrected fan airflow. These effects are assumed to be correlated with the actual vane position and the corrected fan speed. The following equations describe the fan model:

$$(\dot{w}_c)_{\mathrm{FAN},M} = f_6\left(\frac{P_{13}}{P_2}, \frac{N_L}{\theta_2^{1/2}}\right) \tag{12}$$

$$P_{2.1} = P_{2.2} = P_2 f_7\left(\frac{P_{13}}{P_2}, \frac{N_L}{\theta_2^{1/2}}\right) \tag{13}$$

$$\dot{w}_2 = \frac{(\dot{w}_c)_{\mathrm{FAN},M}\,\delta_2\left[1 + f_8(\mathrm{CIVV}, N_L/\theta_2^{1/2})\right]}{\theta_2^{1/2}} \tag{14}$$

$$\eta_{\mathrm{FAN,OD}} = f_9\left(\frac{P_{13}}{P_2}, \frac{N_L}{\theta_2^{1/2}}\right) \tag{15}$$

$$\left(\frac{\Delta T}{T}\right)_{\mathrm{FAN,OD,id}} = \left(\frac{P_{13}}{P_2}\right)^{(\gamma_{\mathrm{FAN}} - 1)/\gamma_{\mathrm{FAN}}} - 1.0 \tag{16}$$

$$T'_{13} = \left[\frac{(\Delta T/T)_{\mathrm{FAN,OD,id}}}{\eta_{\mathrm{FAN,OD}} + 1}\right] T_2 \tag{17}$$

$$\eta_{\mathrm{FAN,ID}} = f_{10}\left(\frac{P_{13}}{P_2}, \frac{N_L}{\theta_2^{1/2}}\right) \tag{18}$$

$$(\Delta T/T)_{\mathrm{FAN,ID,id}} = \left(\frac{P_{2.1}}{P_2}\right)^{(\gamma_{\mathrm{FAN}} - 1)/\gamma_{\mathrm{FAN}}} - 1.0 \tag{19}$$

$$T_{2.1} = T_{2.2} = \left[\frac{(\Delta T/T)_{\mathrm{FAN,ID,id}}}{\eta_{\mathrm{FAN,ID}} + 1}\right] T_2 \tag{20}$$

$$\gamma_{\mathrm{FAN}} = \gamma_2 \tag{21}$$

Compressor. – The mathematical model of the compressor is very similar to the fan model. That is, overall performance maps are utilized with a shift in the corrected airflow based on off-schedule values of variable vane position. However, because of the greater temperature rise through the compressor, the assumption of constant specific heat ratio based on inlet conditions could lead to a problem in matching steady-state cycle data. For this reason an "average" compressor temperature is computed and used to generate an average specific heat ratio. The following equations describe the compressor model:

$$(\dot{w}_c)_{C,M} = f_{11}\left(\frac{P_3}{P_{2.2}}, \frac{N_H}{\theta_{2.2}^{1/2}}\right) \tag{22}$$

$$\dot{w}_{2.2} = \frac{(\dot{w}_c)_{C,M}\,\delta_{2.2}\left[1 + f_{12}(\mathrm{RCVV}, N_H/\theta_{2.2}^{1/2})\right]}{\theta_{2.2}^{1/2}} \tag{23}$$

$$\eta_C = f_{13}\left(\frac{P_3}{P_{2.2}}, \frac{N_H}{\theta_{2.2}^{1/2}}\right) \tag{24}$$

$$\left(\frac{\Delta T}{T}\right)_{C,\mathrm{id}} = \left(\frac{P_3}{P_{2.2}}\right)^{(\gamma_C - 1)/\gamma_C} - 1.0 \tag{25}$$

$$T_C = \beta_C T_{2.2} + (1 - \beta_C) T_3 \tag{26}$$

$$T_3' = \left[\frac{(\Delta T/T)_{C,\mathrm{id}}}{\eta_C} + 1 \right] T_{2.2} \tag{27}$$

Bleeds. – The extraction of turbine cooling and auxiliary drive air from the compressor exit is accounted for in the turbofan engine model. Because of the high pressure at the compressor exit, flow through the bleed passages is assumed to be choked. The following equations describe the bleed model:

$$\left(\frac{\dot{w}}{A}\right)_{\mathrm{BL}} = P_3 \left(\frac{g_c \gamma_3}{R_A T_3}\right)^{1/2} \left[\frac{2}{(\gamma_3 + 1)}\right]^{(\gamma_3 + 1)/2(\gamma_3 - 1)} \tag{28}$$

$$\dot{w}_{\mathrm{BLHT}} = A_{\mathrm{BLHT}} \left(\frac{\dot{w}}{A}\right)_{\mathrm{BL}} \tag{29}$$

$$\dot{w}_{\mathrm{BLLT}} = A_{\mathrm{BLLT}} \left(\frac{\dot{w}}{A}\right)_{\mathrm{BL}} \tag{30}$$

$$\dot{w}_{\mathrm{BLOV}} = A_{\mathrm{BLOV}} \left(\frac{\dot{w}}{A}\right)_{\mathrm{BL}} \tag{31}$$

Turbines. – As in the case of the fan and compressor the overall performance of the high- and low-pressure turbines is represented by bivariate maps. Turbine flow and enthalpy drop parameters are computed as functions of pressure ratio and corrected speed. As described in the section *Intercomponent volumes,* the cooling bleed for each turbine is assumed to reenter the cycle at the turbine discharge, although a portion of each bleed flow is assumed to do turbine work. The following equations define the turbine models:

$$(\dot{w}_p)_{\mathrm{HT}} = f_{14} \left(\frac{P_{4.1}}{P_4}, \frac{N_H}{T_4^{1/2}} \right) \tag{32}$$

$$\dot{w}_4 = \frac{(\dot{w}_p)_{\mathrm{HT}} P_4 N_H}{T_4} \tag{33}$$

$$(h_p)_{\mathrm{HT}} = f_{15} \left(\frac{P_{4.1}}{P_4}, \frac{N_H}{T_4^{1/2}} \right) \tag{34}$$

$$(\Delta h)_{\mathrm{HT}} = (h_p)_{\mathrm{HT}} N_H T_4^{1/2} \tag{35}$$

$$(\dot{w}_p)_{\mathrm{LT}} = f_{16} \left(\frac{P_5}{P_{4.1}}, \frac{N_L}{T_{4.1}^{1/2}} \right) \tag{36}$$

$$\dot{w}_{4.1} = \frac{(\dot{w}_p)_{\mathrm{LT}} P_{4.1} N_L}{T_{4.1}} \tag{37}$$

$$(h_p)_{\mathrm{LT}} = f_{17} \left(\frac{P_5}{P_{4.1}}, \frac{N_L}{T_{4.1}^{1/2}} \right) \tag{38}$$

$$(\Delta h)_{\mathrm{LT}} = (h_p)_{\mathrm{LT}} N_L T_{4.1}^{1/2} \tag{39}$$

Combustors and ducts. – Total pressure losses are included in the models of the main combustor, bypass duct, mixer entrance, and augmentor. Mach numbers in these regions are assumed to be moderate, thus allowing the use of fairly simple loss equations. The losses for the main combustor, bypass duct, and augmentor are assumed to be proportional to the square of the corrected flow parameters $\dot{w}T^{1/2}/P$. The pressure drop associated with the entrance to the augmentor is assumed to be proportional to the pressure level. The heat addition associated with the burning of fuel in the main combustor and augmentor is assumed to take place in volumes V_4 and V_7, respectively. As an aid in matching steady-state cycle data, average combustor temperatures are computed and used in the dynamic energy balances (see the section *Intercomponent volumes*). The following equations describe the combustor and duct models:

$$\dot{w}_3 = \left[\frac{P_3(P_3 - P_4)}{K_B T_3} \right]^{1/2} \tag{40}$$

$$T_B = \beta_B T_3 + (1 - \beta_B) T_4 \tag{41}$$

$$\Delta h_B = \mathrm{HVF}\eta_B \tag{42}$$

$$\eta_B = f_{18}[(f/a)_4] \tag{43}$$

$$(f/a)_4 = \frac{\dot{w}_{F,4}}{\dot{w}_3} \tag{44}$$

$$P_5 = K_{\mathrm{PR5}} P_6 \tag{45}$$

$$P_7' = \frac{P_6 - K_{\mathrm{AB}} \dot{w}_6^2 T_6}{P_6} \tag{46}$$

$$T_{\mathrm{AB}} = \beta_{\mathrm{AB}} T_6 + (1 - \beta_{\mathrm{AB}}) T_7 \tag{47}$$

$$\Delta h_{\mathrm{AB}} = \mathrm{HVF}\eta_{\mathrm{AB}} \tag{48}$$

$$\eta_{AB} = f_{19}[(f/a)_7] \tag{49}$$

$$(f/a)_7 = \frac{(\dot{w}_{F,7} + \dot{w}_{F,4})}{(\dot{w}_6 - \dot{w}_{F,4})} \tag{50}$$

$$P_{16} = \frac{P_{13} - K_D \dot{w}_{13}^2 T_{13}}{P_{13}} \tag{51}$$

$$T_{16} = T_{13} \tag{52}$$

Exhaust nozzle. – A convergent-divergent nozzle configuration is assumed for the turbofan engine model. A convergent-only nozzle may be considered a subset of the more general model. A fairly detailed mathematical representation of the thermodynamics is used, including the treatment of normal shocks in the divergent section. The pressure losses associated with the shock are computed along with gross and net engine thrust. The following equations define the nozzle model and are based on material presented in reference 18:

$$\dot{w}_7 = P_7 A_E^* C_{d,N} \left(\frac{g_c \gamma_N}{R_A T_7}\right)^{1/2} \left[\frac{2}{(\gamma_N + 1)}\right]^{(\gamma_N+1)/2(\gamma_N-1)} \tag{53}$$

$$F_N = \frac{\dot{w}_7 v_E}{g_c} + A_E(P_E - P_0) \tag{54}$$

$$C_{d,N} = f_{20}\left(\frac{P_0}{P_7}\right) \tag{55}$$

$$\left(\frac{P_0}{P_7}\right)_{cr} = f_{21}\left(\frac{A_E}{A_8}\right) \tag{56}$$

If $P_0/P_7 \geq (P_0/P_7)_{cr}$, the flow is subsonic in the nozzle and

$$P_E = P_0 \tag{57}$$

$$\frac{A_E}{A_E^*} = f_{21}^{-1}\left(\frac{P_0}{P_7}\right) \tag{58}$$

$$A_E^* = \frac{A_E}{(A_E/A_E^*)} \tag{59}$$

$$M_E^* = f_{22}\left(\frac{P_0}{P_7}\right) \tag{60}$$

$$v_E = M_E^* C_{v,N} \left[\frac{2 g_c \gamma_N R_A T_7}{(\gamma_N + 1)}\right]^{1/2} \tag{61}$$

$$C_{v,N} = f_{23}\left(\frac{P_0}{P_7}\right) \tag{62}$$

Otherwise a shock may exist in the divergent portion of the nozzle. To compute the required parameters under these conditions, shock tables such as those in reference 18 must be used.

$$M_x = f_{24}\left(\frac{A_E}{A_8}\right) \tag{63}$$

$$\frac{P_y}{P_x} = f_{25}(M_x) \tag{64}$$

$$\frac{P_y}{p_x} = f_{26}(M_x) \tag{65}$$

$$\frac{p_y}{p_x} = f_{27}(M_x) \tag{66}$$

$$\left(\frac{P_0}{P_7}\right)_{es} = \frac{(P_y/P_x)(p_y/p_x)}{P_y/p_x} \tag{67}$$

If $P_0/P_7 = (P_0/P_7)_{es}$, the shock will be in the nozzle exit plane. Then

$$P_E = P_0 \tag{68}$$

$$M_x^* = f_{28}\left(\frac{A_E}{A_8}\right) \tag{69}$$

$$v_x = M_x^* \left[\frac{2 \gamma_N R_A g_c T_7}{(\gamma_N + 1)}\right]^{1/2} \tag{70}$$

$$\frac{v_x}{v_y} = f_{29}(M_x) \tag{71}$$

$$v_E = \frac{C_{v,N} v_x}{v_x v_y} \tag{72}$$

If $P_0/P_7<(P_0/P_7)_{es}$, the shock is external to the nozzle. Then

$$M_E^*=f_{28}\left(\frac{A_E}{A_8}\right) \tag{73}$$

$$\frac{P_E}{P_7}=f_{30}\left(\frac{A_E}{A_8}\right) \tag{74}$$

$$P_E=P_7\left(\frac{P_E}{P_7}\right) \tag{75}$$

$$v_E=M_E^*C_{v,N}\left[\frac{2\gamma_N R_A g_c T_7}{(\gamma_N+1)}\right]^{1/2} \tag{76}$$

If $(P_0/P_7)_{cr}>P_0/P_7>(P_0/P_7)_{es}$, the shock is in the divergent section and

$$P_E=P_0 \tag{77}$$

$$\frac{A_x^*}{A_y^*}=\frac{P_y}{P_x}=f_{25}(M_x) \tag{78}$$

$$\frac{A_E}{A_y^*}=\left(\frac{A_E}{A_8}\right)\left(\frac{A_x^*}{A_y^*}\right) \tag{79}$$

$$\frac{P_E}{P_y}=f_{21}\left(\frac{A_E}{A_y^*}\right) \tag{80}$$

$$\frac{P_E}{P_x}=\left(\frac{P_E}{P_y}\right)\left(\frac{P_y}{P_x}\right) \tag{81}$$

$$P_E=P_7\left(\frac{P_E}{P_x}\right) \tag{82}$$

To solve these equations, M_x can be varied until equations (82) and (77) produce the same values for P_E. Then

$$M_E^*=f_{28}\left(\frac{A_E}{A_y^*}\right) \tag{83}$$

$$v_E=M_E^*C_{v,N}\left[\frac{2\gamma_N R_A g_c T_7}{(\gamma_N+1)}\right]^{1/2} \tag{84}$$

The compressible flow tables in reference 18 and the functional relationships (f_{21} to f_{30}) assume a specific heat ratio of 1.4. In general, the specific heat ratio in the tailpipe of a turbofan engine will be lower than this value, particularly during augmentor operation. To compensate for this error when setting up the model to match specified cycle data, the value of γ_N in the flow rate and velocity equations is adjusted in the host program to give the desired values of flow rate $\dot{w}_7$ and gross thrust F_N. The net engine thrust is computed by subtracting the inlet ram drag from the gross thrust.

$$F_n=F_N-M_0\dot{w}_2\left(\frac{\gamma_0 R_A T_0}{g_c}\right)^{1/2} \tag{85}$$

Engine Dynamics

Intercomponent volumes. – Intercomponent volumes are assumed at engine locations where either (1) gas dynamics are considered important or (2) gas dynamics are required to avoid the need for iterative solution of equations. In these volumes the storage of mass and energy occurs. The dynamic forms of the continuity, energy, and state equations (ref. 19) are solved for the stored mass, temperature, and pressure in each volume. The following equations define the dynamic models of the intercomponent volumes. Appendix B contains a derivation of the general form of the energy equation used in the engine model.

$$W_{13}=\int_0^t(\dot{w}_2-\dot{w}_{2.2}-\dot{w}_{13})\,dt+W_{13,i} \tag{86}$$

$$T_{13}=\int_0^t\Big\{\big[(\dot{w}_2-\dot{w}_{2.2})(h_{13}'-h_{13})/c_{v,13}+T_{13}(\dot{w}_2-\dot{w}_{2.2}-\dot{w}_{13})(\gamma_{13}-1)\big]\Big/W_{13}\Big\}dt+T_{13,i} \tag{87}$$

$$P_{13}=\frac{R_A W_{13}T_{13}}{V_{13}} \tag{88}$$

$$W_3=\int_0^t(\dot{w}_{2.2}-\dot{w}_{BLHT}-\dot{w}_{BLLT}-\dot{w}_{BLOV}-\dot{w}_3)\,dt+W_{3,i} \tag{89}$$

$$T_3=\int_0^t\Big\{\big[\dot w_{2.2}(h_3'-h_3)/c_{v,3}+T_3(\dot w_{2.2}-\dot w_{\mathrm{BLHT}}-\dot w_{\mathrm{BLLT}}-\dot w_{\mathrm{BLOV}}-\dot w_3)\times(\gamma_3-1)\big]\Big/W_3\Big\}\,dt+T_{3,i} \quad (90)$$

$$P_3=\frac{R_A W_3 T_3}{V_3} \quad (91)$$

$$W_4=\int_0^t(\dot w_3+\dot w_{F,4}-\dot w_4)\,dt+W_{4,i} \quad (92)$$

$$T_4=\int_0^t\left(\Big\{\big[\dot w_3 h_B+\dot w_{F,4}\Delta h_B-h_4(\dot w_3+\dot w_{F,4})\big]\big/c_{v,4}+T_4(\dot w_3+\dot w_{F,4}-\dot w_4)(\gamma_4-1)\Big\}\Big/W_4\right)dt+T_{4,i} \quad (93)$$

$$P_4=\frac{R_A W_4 T_4}{V_4} \quad (94)$$

$$W_{4.1}=\int_0^t(\dot w_4+\dot w_{\mathrm{BLHT}}-\dot w_{4.1})dt+W_{4.1,i} \quad (95)$$

$$T_{4.1}=\int_0^t\left(\Big\{\big[\dot w_4(h_4-\Delta h_{\mathrm{HT}})+\dot w_{\mathrm{BLHT}}(h_3-K_{\mathrm{BLWHT}}\Delta h_{\mathrm{HT}})-h_{4.1}(\dot w_4+\dot w_{\mathrm{BLHT}})\big]\big/c_{v,4.1}+T_{4.1}(\dot w_4+\dot w_{\mathrm{BLHT}}-\dot w_{4.1})(\gamma_{4.1}-1)\Big\}\Big/W_{4.1}\right)dt+T_{4.1,i} \quad (96)$$

$$P_{4.1}=\frac{R_A W_{4.1} T_{4.1}}{V_{4.1}} \quad (97)$$

$$W_6=\int_0^t(\dot w_{4.1}+\dot w_{\mathrm{BLLT}}+\dot w_{13}-\dot w_6)dt+W_{6,i} \quad (98)$$

$$T_6=\int_0^t\left(\Big\{\big[\dot w_{4.1}(h_{4.1}-\Delta h_{\mathrm{LT}})+\dot w_{\mathrm{BLLT}}(h_3-K_{\mathrm{BLWLT}}\Delta h_{\mathrm{LT}})+\dot w_{13}h_{16}-h_6(\dot w_{4.1}+\dot w_{\mathrm{BLLT}}+\dot w_{13})\big]\big/c_{v,6}+T_6(\dot w_{4.1}+\dot w_{\mathrm{BLLT}}+\dot w_{13}-\dot w_6)(\gamma_6-1)\Big\}\Big/W_6\right)dt+T_{6,i} \quad (99)$$

$$P_6=\frac{R_A W_6 T_6}{V_6} \quad (100)$$

$$W_7=\int_0^t(\dot w_6+\dot w_{F,7}-\dot w_7)dt+W_{7,i} \quad (101)$$

$$T_7=\int_0^t\left(\Big\{\big[\dot w_6 h_{\mathrm{AB}}+\dot w_{F,7}\Delta h_{\mathrm{AB}}-h_7(\dot w_6+\dot w_{F,7})\big]\big/c_{v,7}+T_7(\dot w_6+\dot w_{F,7}-\dot w_7)(\gamma_7-1)\Big\}\Big/W_7\right)dt+T_{7,i} \quad (102)$$

$$P_7=\frac{R_A W_7 T_7}{V_7} \quad (103)$$

Fluid momentum. – The effects of fluid momentum on the transient behavior of the turbofan are considered in the bypass and augmentor duct models. The contribution of flow dynamics in the compressor, main combustor, and turbines is assumed to be high frequency (>10 Hz) and is ignored. The following equations define the flow dynamics in the engine model:

$$\dot w_{13}=g_c\left(\frac{A}{l}\right)_D\int_0^t(P_{16}-P_6)dt+\dot w_{13,i} \quad (104)$$

$$\dot w_6=g_c\left(\frac{A}{l}\right)_{\mathrm{AB}}\int_0^t(P_7'-P_7)dt+\dot w_{6,i} \quad (105)$$

Rotor inertias. – The most significant factors in determining the transient behavior of the turbofan are the rotor moments of inertia. Rotor speeds are computed from the dynamic forms of the angular momentum equations.

$$N_L=\left(\frac{30}{\pi}\right)^2\frac{J}{I_L}\int_0^t\Big\{\big[\Delta h_{\mathrm{LT}}(\dot w_{4.1}+K_{\mathrm{BLWLT}}\dot w_{\mathrm{BLLT}})-(\dot w_2-\dot w_{2.2})(h_{13}'-h_2)-\dot w_{2.2}(h_{2.2}-h_2)\big]\big/N_L\Big\}dt+N_{L,i} \quad (106)$$

$$N_H = \left(\frac{30}{\pi}\right)^2 \frac{J}{I_H} \int_0^t \left\{ \left[\Delta h_{\mathrm{HT}} (\dot{w}_4 + K_{\mathrm{BLWHT}} \dot{w}_{\mathrm{BLHT}}) - \dot{w}_{2.2}(h_3' - h_{2.2}) \right] / N_H \right\} dt + N_{H,i} \tag{107}$$

Thermal lags due to the storage of heat in the engine metal are not included in the turbofan model.

Hybrid Computer Program

Hybrid Computing System

The equations describing the turbofan engine model were implemented on one of the Lewis Research Center's two hybrid computing systems. Each hybrid computer consists of an EAI PACER 100 digital processor, two PACER 681 (680) analog processors, and a PACER 693 communications interface for control and data exchange between the digital and analog processors. Table I lists the salient features of the hybrid computing systems. The following sections describe the implementation of the turbofan engine simulation on the hybrid computer.

Scaling

The use of the analog computer requires the scaling of the analog variables so that no analog signal exceeds 1.0 computer unit (10 V on the EAI 681). For each variable x a scale factor SF_x is chosen so as to limit the scaled variable $X = x/\mathrm{SF}_x$ to the range $-1 < X < +1$. Also, to reduce the core requirements and computing time in the PACER 100 digital part of the simulation, it was decided to use scaled fractions throughout the digital program. Therefore all digital variables are scaled in the same way so as not to exceed unity during digital program execution. The section Host Digital Program describes how the user specifies scale factors. Although the user may specify most scale factors through the host program, certain scale factors are fixed to allow the use of general-purpose subroutines. For example, all fuel-air ratios are scaled for a maximum of 0.08. Also, certain scale factors are related to user-specified scale factors. For example, the scale factor on specific enthalpy h is fixed as the product of the scale factor on specific heat c_p and the scale factor on temperature T.

In addition to the previously described amplitude scaling, it may be necessary to time scale the analog portion of the simulation. To allow the treatment of digital outputs as continuous inputs to the analog, a sufficient number of cycles through the digital loop must occur for each cycle of the analog frequencies. Often computational stability can only be achieved by decreasing the analog frequencies (slowing down the analog). A time-scale factor SF_t is selected such that the

TABLE I. – FEATURES OF THE LEWIS HYBRID COMPUTER SYSTEMS

[ADC's are 14 bits plus sign and do not invert; DAC's are 14 bits plus sign and do not invert (all DAC's are multiplying DAM's).]

(a) PACER 100 digital system

Total memory (at 16 bits/word), words	32 768
Cycle time, μsec	1.0
Moving head disk	Two platters for 2.2 megawords
Card reader, cards/min	300
Line printer, characters/sec	165
High-speed paper tape reader, characters/sec	300
High-speed paper tape punch, characters/sec	120
Tektronix 4010 CRT terminal	
Hard copier for Tektronix 4010	

(b) 680-681 analog systems

	Consoles A1, B1, B2	Console A2
	Number per console	
Integrator summer	30	30
Track/store summer	12	12
Zero-limit summer	24	12
SJ-INV (interface)	24	24
SJ-INV (fixed DFG)	0	2
QSM inverter	60	48
QSM-HG amplifier	30	24
VDFG inverter	0	6
Total amplifiers per console	180	158
SS potentiometers (2-terminal)	64	96
SS potentiometers (3-terminal)	16	24
HS potentiometers (2-terminal)	12	12
DCA's	40	0
Total potentiometers per console	132	132
Multipliers	30	24
DCFG's (digital)	8	0
VDFG's (analog)	0	6
Fixed DFG's (analog)	0	2
Variable limiters	12	12
Comparators	24	24
Function relays	24	24
D/A switches	24	24
General-purpose registers	6	6
"AND" gates	36	36
BCD counters	3	3
Monostable timers	6	6
Logic differentiators	6	6

(c) 693 interface system

	System A	System B
	Number	
ADC's	32	48
DAC's	24	24
Control lines per console	16	16
General-purpose interrupts per console	8	8
Sense lines per console	8	8
Interface clock	1	1

computer time t' equals $SF_t t$. A time-scale factor less than 1.0 corresponds to speeding up the problem. Reference 14 provides some guidelines for selecting a suitable time-scale factor. The section Results and Discussion also covers this issue.

Having specified amplitude and time-scale factors through the host program, the user need not be concerned about the scaling of variables within the simulation. The host program computes digital coefficients and analog potentiometer settings based on the specified scale factors. The equations that are programmed on the host and target digital computers are organized in such a way as to minimize the chance of overflows. Table II lists the prescaled variables, their scale factors, and implied relations between scale factors.

TABLE II. - PRESCALED SIMULATION VARIABLES

Variable, x	Scale factor, SF_x
$C_{d,N}$	1.0
$C_{v,N}$	1.0
$c_{p,i}$	0.5 Btu/lbm °R
$c_{v,i}$	0.5 Btu/lbm °R
$(f/a)_i$	0.08
h_i	$SF_{c_{p,i}} SF_{T_i}$
T_i'	SF_{T_i}
T_{16}	$SF_{T_{13}}$
T_B	SF_{T_4}
T_C	SF_{T_3}
T_{am}	1000° R
T_{AB}	SF_{T_7}
P_7'	SF_{P_7}
P_{16}	$SF_{P_{13}}$
P_E	SF_{P_0}
$\dot{w}_{BLHT}$	$0.2\ SF_{\dot{w}_{2.2}}$
$\dot{w}_{BLLT}$	$0.02\ SF_{\dot{w}_{2.2}}$
$\dot{w}_{BLOV}$	$0.002\ SF_{\dot{w}_{2.2}}$
$\Delta T/T$	1.25
η	1.0
γ_i	2.0

Target Analog Program

The computational split between the analog and digital portions of the hybrid simulation is basically the same as that used in reference 14. That is, the analog computer is limited to doing continuous integration with respect to time and some related multiplication and division. By comparison, the real-time simulations in references 7 to 9 used extensive analog computation, with the digital limited to performing bivariate function generation.

Scaled values for the engine state variables (stored masses, temperatures, duct flow rates, and rotor speeds) and intercomponent volume pressures are computed on the analog computer by using scaled versions of equations (86) to (107). Appendix C lists all the scaled analog equations.

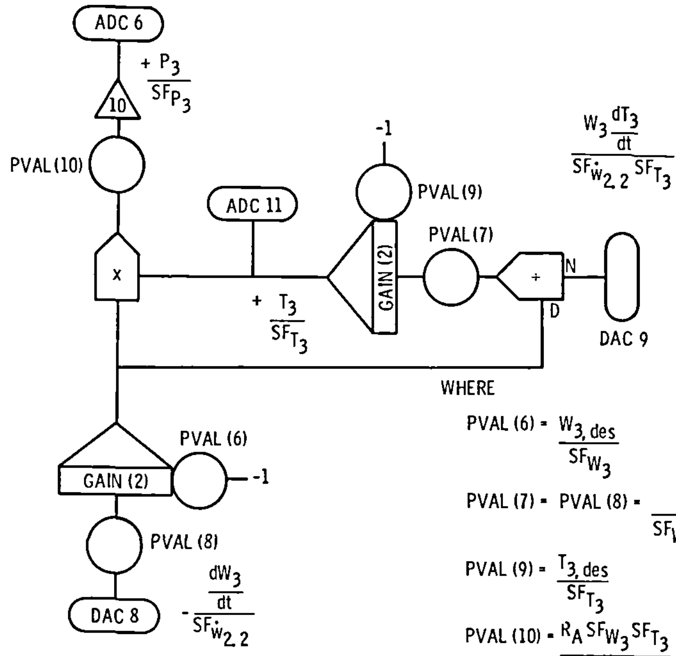
Figure 5 illustrates the analog calculation of variables in volume V_3. Identical circuitry is patched by the user for each of the intercomponent volumes. Having selected the analog components, the user can specify the addresses of the integrators and potentiometers as input to the host digital program (see the section Host Digital Program). Using design-point information, scale factors, and geometric engine data, the host program will automatically compute amplifier gains and potentiometer settings and will generate punched-card data for automatic setup of those components. The following section describes the PACER 100 software that accomplishes the automatic setup of the analog. Figures 6 and 7 illustrate the analog calculations of bypass duct flow rate and low-rotor speed, respectively. Appendix C contains patching diagrams for the entire analog portion of the simulation. In addition to computing state variables and pressures, the analog computer is used to generate (or transmit from external sources) inputs to the simulation. These inputs include scaled control inputs (fuel flow, e.g.) and scaled flight condition inputs (altitude, Mach number, and sea-level ambient temperature). Table III defines the analog patching structure to which the user must assign components.

Inputs to the analog computer from the digital computer are transmitted by 21 DAC's. They include the scaled stored mass derivatives, nonspecific temperature derivatives ($W\dot{T}$), rotor speed derivatives, and variables for display. Outputs from the analog computer to the digital computer are transmitted by 25 ADC's. They include the intercomponent pressures and temperatures, duct flow rates, rotor speeds, control inputs, and flight condition inputs. Table IV lists the DAC and ADC variables and designated channels.

Target Digital Program

The target digital processor is used to perform the bulk of the computations in the target program. That is, the digital must do all the arithmetic and function generation necessary to compute the time derivatives of the engine state variables. Other engine variables of interest, such as net thrust, are also computed in the digital and output to the analog for display and recording. The digital computer is also used for steady-state data display and for automated setup of the analog computer.

The following sections describe the target digital program that has been developed for the EAI PACER 100 digital processor and the EAI 693 hybrid interface that allows the PACER to communicate with the EAI 681



WHERE

$$PVAL(6) = \frac{W_{3,des}}{SF_{W_3}}$$

$$PVAL(7) = PVAL(8) = \frac{SF_{\dot{w}2.2}}{SF_{W_3} SF_t\, GAIN(2)}$$

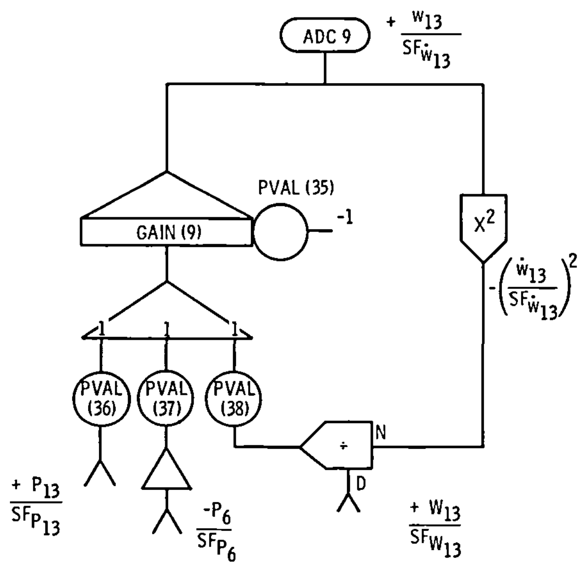
$$PVAL(9) = \frac{T_{3,des}}{SF_{T_3}}$$

$$PVAL(10) = \frac{R_A SF_{W_3} SF_{T_3}}{10\, V_3 SF_{P_3}}$$

$$GAIN(2) = 1.0 \text{ if } \frac{SF_{\dot{w}2.2}}{SF_{W_3} SF_t} \leq 1.0$$

$$= 10^{IG(2)} \text{ OTHERWISE}$$

Figure 5. - Analog computation of compressor discharge conditions.



WHERE

$$PVAL(35) = \frac{\dot{w}_{13,des}}{SF_{\dot{w}_{13}}}$$

$$PVAL(36) = \left(\frac{A}{\ell}\right)_D \frac{g_c SF_{P_{13}}}{SF_{\dot{w}_{13}} SF_t\, GAIN(9)}$$

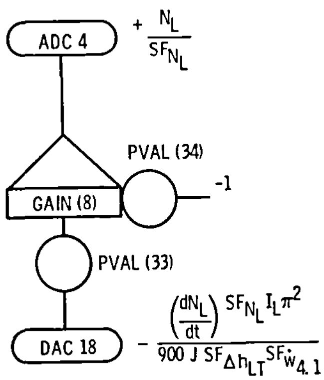
$$PVAL(37) = \left(\frac{A}{\ell}\right)_D \frac{g_c SF_{P_6}}{SF_{\dot{w}_{13}} SF_t\, GAIN(9)}$$

$$PVAL(38) = \left(\frac{A}{\ell}\right)_D \frac{g_c P_{13,des} (P_{13,des} - P_{6,des}) SF_{\dot{w}_{13}} V_{13}}{\dot{w}^2_{13,des} R_A T_{13,des} SF_{W_{13}} SF_t\, GAIN(9)}$$

$$GAIN(9) = 1.0 \text{ if } \left(\frac{A}{\ell}\right)_D \frac{g_c SF_{P_{13}}}{SF_{\dot{w}_{13}} SF_t} < 1.0$$

$$= 10^{IG(9)} \text{ OTHERWISE}$$

Figure 6. - Analog computation of bypass duct flow rate.



WHERE

$$\text{PVAL (33)} = \frac{900\,J\,SF_{\Delta h_{LT}}\,SF_{\dot{w}_{4.1}}}{\pi^2 I_L\,SF_{N_L}^2\,SF_t\,\text{GAIN (8)}}$$

$$\text{PVAL (34)} = \frac{N_{L,des}}{SF_{N_L}}$$

$$\text{GAIN (8)} = 1.0 \text{ if } \frac{900\,J\,SF_{\Delta h_{LT}}\,SF_{\dot{w}_{4.1}}}{\pi^2 I_L\,SF_{N_L}^2\,SF_t} \le 1.0$$

$$= 10^{IG(8)} \text{ OTHERWISE}$$

Figure 7. - Analog computation of fan rotor speed.

analog computer. Although the target digital program is computer dependent, it should provide a sound basis for development of target programs for other hybrid computer systems.

Input data. – The link between the host digital program and the target program is a set of punched cards containing component performance map data, digital coefficients, analog potentiometer settings, integrator gains, and other information needed to set up and run the hybrid computer simulation. Figure 8 shows the nature and ordering of the punched-card data generated by the host program.

The first six sets of data cards contain bivariate function data defining the steady-state performance of the engine's rotating components. The sets are read in the following order: (1) fan variable-geometry effects map, $f_8(\text{CIVV}, N_L/\theta_2^{1/2})$; (2) compressor variable-geometry effects map, $f_{12}(\text{RCVV}, N_H/\theta_{2.2}^{1/2})$; (3) baseline fan performance map, $f_i(P_{13}/P_2, N_L/\theta_2^{1/2})$, $i=6, 9, 7, 10$; (4) baseline compressor performance map, $f_i(P_3/P_{2.2}, N_H/\theta_{2.2}^{1/2})$, $i=11, 13$; (5) high-pressure-turbine performance map, $f_i(P_{4.1}/P_4, N_H/T_4^{1/2})$, $i=14, 15$; and (6) low-pressure-turbine performance map, $f_i(P_5/P_{4.1}, N_L/T_{4.1}^{1/2})$, $i=16, 17$. The performance map cards contain information pertaining to the size of the map arrays, the

TABLE III. – ANALOG COMPONENT ARRANGEMENT

Model element	Gain integer (IG(K)) index, K	Integrator address (AADR(K)) index, K[a]	Potentiometer address (PADR(K)) index, K[b]
V_{13}	1	1-2	1-2-3-4-5
V_3	2	3-4	6-7-8-9-10
V_4	3	5-6	11-12-13-14-15
$V_{4.1}$	4	7-8	16-17-18-19-20
V_6	5	9-10	21-22-23-24-25
V_7	6	11-12	26-27-28-29-30
I_H	7	13	31-32
I_L	8	14	33-34
$(A/l)_D$	9	15	35-36-37-38
$(A/l)_{AB}$	10	16	39-40-41-42

Model input	Potentiometer address (PADR(K)) index, K	Biased variable	Potentiometer address (PADR(K)) index, K
$\dot{w}_{F,4}$	43	P_{13}/P_2	52
$\dot{w}_{F,7}$	44	$P_3/P_{2.2}$	53
A_8	45		
A_E	46		
CIVV	47		
RCVV	48		
ALT	49		
M_0	50		
T_{am}	51		

[a]W-T integrators for volumes.

[b]Address index is same as setting index, as defined in appendix C.

TABLE IV. - ADC AND DAC VARIABLES

ADC channel	Variable
0	ALT
1	M_0
2	T_{am}
3	P_{13}
4	N_L
5	-CIVV
6	P_3
7	N_H
8	-RCVV
9	$\dot{w}_{13}$
10	T_1
11	T_3
12	P_4
13	T_4
14	$P_{4.1}$
15	$\dot{w}_{F,4}$
16	$T_{4.1}$
17	P_6
18	$\dot{w}_6$
19	T_6
20	P_7
21	T_7
22	A_8
23	A_E
24	$\dot{w}_{F,7}$

DAC channel	Variable
0	P_2
1	T_2
2	P_{13}/P_2
3	$\dot{w}_2\theta_2^{1/2}/\delta_2$
4	$P_3/P_{2.2}$
5	$\dot{w}_{2.2}\theta_{2.2}^{1/2}/\delta_{2.2}$
6	dW_{13}/dt
7	$W_{13}dT_{13}/dt$
8	dW_3/dt
9	W_3dT_3/dt
10	dW_4/dt
11	W_4dT_4/dt
12	$dW_{4.1}/dt$
13	$W_{4.1}dT_{4.1}/dt$
14	dW_6/dt
15	W_6dT_6/dt
16	dW_7/dt
17	W_7dT_7/dt
18	dN_L/dt
19	dN_H/dt
20	F_n

format of the data cards, scale factors to be used in scaling the map data, and the unscaled performance map data. The sets of performance map cards are identical to the sets input by the user to the host program.

The first card for each set of map data contains: (1) a map number to be used in the $Z_i=f_i(X,Y)$ function call, (2) the number of curves Y on the map, (3) the number of points (X,Z) on each curve, (4) the number of common functions Z_i of the same independent variables, and (5) an additional integer to be discussed later. A (513) format is used to read the first card. The second card for each map contains the formats to be used in reading the remaining map cards. A (28A2) format is used to specify the formats of (1) the X, Y, and Z_i scale factors where $i=1$ to 4, (2) the X values, (3) the Y values, and (4) the Z_i values. The third card for each map contains the X, Y, and Z_i scale factors in the specified format. The remaining cards in the set contain (in order) the Y values, X values for the first curve, Z_i values for the first curve, X values for the second curve, Z_i values for the second curve, and so forth. For those functions where each curve may be defined by identically the same X values, those X values need only be input once, immediately following the Y values. In this case a nonzero integer is added as the fifth element in the first card of the set. A blank card follows the last card of the last map. Figure 9 shows an

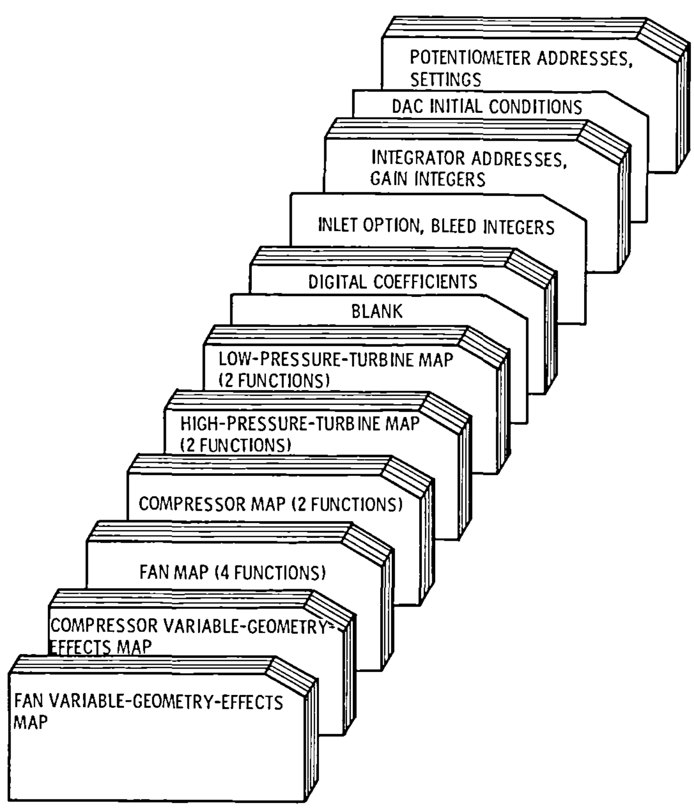


Figure 8. - Host-program-generated target program input data.

```
2  3  5   3                                                  MAP NUMBER, NCV, NPT, NFCT SCALE
(5F8.1)  (5F8.1)  (3F8.1)  (5F8.1)  (5F8.2)  (5F8.3)         FACTOR, X, Y, Z1, Z2, Z3, FORMATS
1.0      1.0      1.0      1.0      1.0                      X, Y, Z1, Z2, Z3 SCALE FACTORS
0.2      0.4      0.6                                Y   VALUES
0.0      0.2      0.3      0.4      0.5              X   VALUES - CURVE 1
0.3      0.3      0.2      0.1      0.0              Z1  VALUES - CURVE 1
0.15     0.15     0.10     0.05     0.00             Z2  VALUES - CURVE 1
0.225    0.225    0.150    0.075    0.000            Z3  VALUES - CURVE 1
0.0      0.4      0.5      0.6      0.7              X   VALUES - CURVE 2
0.6      0.6      0.4      0.2      0.0              Z1  VALUES - CURVE 2
0.30     0.30     0.20     0.10     0.00             Z2  VALUES - CURVE 2
0.450    0.450    0.300    0.150    0.000            Z3  VALUES - CURVE 2
0.0      0.6      0.7      0.8      0.9              X   VALUES - CURVE 3
0.9      0.9      0.6      0.3      0.0              Z1  VALUES - CURVE 3
0.45     0.45     0.30     0.15     0.00             Z2  VALUES - CURVE 3
0.675    0.675    0.450    0.225    0.000            Z3  VALUES - CURVE 3

NCV - NUMBER OF CURVES IN MAP
NPT - NUMBER OF POINTS PER CURVE
NFCT - NUMBER OF COMMON FUNCTIONS OF X, Y
```

Figure 9. - Example of map input data.

example of map data where there are three common functions of the independent variables.

Following the component performance map data are cards containing digital coefficients. The coefficients are computed by the host program from user-supplied design-point data, scale factors, and engine geometric information. Also, the host programs "trims" the coefficients to achieve an equilibrium condition at the design point. The digital coefficients are punched in a (3X,5(1X,F6.5,2X)) format, which permits them to be read by the target program in a (3X,5(S7,2X)) format and declared as scaled-fraction variables.

Following the digital coefficient data is a card containing four integers (INLET, KBH, KBL, and KBV) in a (1X,4(I7,2X)) format. A value of 1 for INLET denotes the use of the standard inlet pressure recovery characteristic (eq. (3)) in the simulation. Any other value will result in an inlet recovery of 1.0 (no inlet). The KBH, KBL, and KBV integers indicate whether the compressor discharge bleeds of equations (28) to (31) are to be included in the simulation. The host program determines whether positive values of $\dot{w}_{BLHT}$, $\dot{w}_{BLLT}$, and $\dot{w}_{BLOV}$ are required to match the user-supplied design point data. If so, the corresponding integers are set to 1. If negative or zero values are required, the corresponding integers are set to 0 and the bleed flows are ignored in the simulation.

The next set of cards contains analog integrator addresses and associated gain integers in a (3I4) format. The user assigns integrator addresses to the analog patching structure defined in table III and appendix C. The integrator addresses are input to the host program, and the host program computes the required integrator gains based on design-point data, scale factors, etc. The gain integers represent the power-of-10 gain required. The first six cards in the set contain three integers each. The first two integers are the addresses of the two integrators associated with a particular intercomponent volume. The third integer is the gain integer for those two integrators. The next four cards contain two integers each – the integrator address and gain integer associated with (in order) the high-speed spool, the low-speed spool, the bypass duct, and the augmentor duct.

Following the integrator data are cards containing initial conditions for the 21 DAC's used in the hybrid simulation. The DAC's must be initialized to avoid analog component overloads prior to executing the dynamic portion of the target digital program. The DAC initial conditions are punched by the host program in a (3X,5(1X,F6.5,2X)) format, which permits them to be read by the target program in a (3X,5(S7,2X)) format and declared as scaled-fraction variables.

The last set of cards contains analog potentiometer addresses and potentiometer settings. As in the case of the integrators the user assigns potentiometer addresses to the analog patching structure and inputs those addresses to the host program. The host program computes the required settings and punches the addresses and settings in a (1X,A4,1X,F5.4) format. The cards are read by the target program in a (1X,A4,1X,S5) format.

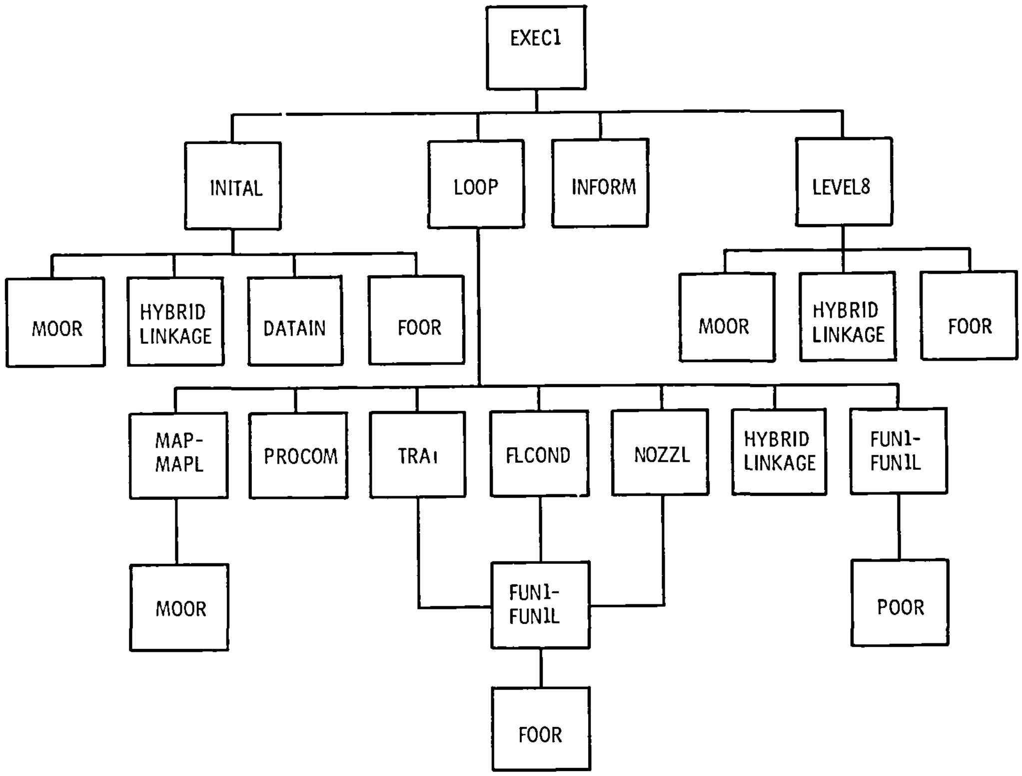


Figure 10. - Organization of target digital program.

Organization. – Figure 10 shows the structure of the target digital program. The program consists of a number of subroutines. The major subroutines are INITAL, an initialization and setup routine, and LOOP, the digital portion of the dynamic engine simulation. INITAL is executed once prior to entering the main dynamic loop. LOOP is repetitively executed at a fixed rate selected by the user on the basis of (1) the time required to perform the calculations in LOOP and (2) the maximum allowable digital delay for stable operation of the hybrid simulation. LOOP samples analog variables through 25 ADC's, performs the calculation of the engine state variable derivatives, and outputs those derivatives to the analog through 21 DAC's. Other subroutines and function routines are called by INITAL and LOOP to support the computation and the input and output. Appendix D contains a flow chart of the target program and source listings of the target program routines.

To facilitate the running of the target program on the PACER 100 digital processor, the INITAL and LOOP subroutines were run under the control of a general-purpose main program called EXEC1 (refs. 20 and 21). Although not required for running the target program, EXEC1 provided a time-shared, interrupt environment that allowed interactive control of steady-state and dynamic data displays. Subroutine INFORM is called by EXEC1 when the user depresses a sense switch at the computer console *and if* the computations in LOOP have been completed. That is, INFORM runs on a lower priority level than LOOP. If spare time is available, the user can interactively obtain displays of simulation data while the program is running. For the application reported herein, an EXEC1 subroutine, BTASK, was constructed to unscale and output tables of steady-state data upon user command. This method of obtaining steady-state results was used in lieu of WRITE statements in the target program. Also, subroutine LEVEL8, running at the next-to-lowest priority level (INFORM is the lowest), was used to obtain map- and function-out-of-range messages at the line printer (or CRT). Another EXEC1-related subroutine, described in references 20 and 21, is LEVEL0. For the application reported herein, LEVEL0 was replaced with a dummy routine. The following sections describe the various routines that make up the target program.

Subroutines. – The target program is made up of the routines INITAL, LOOP, LEVEL9, BTASK, DATAIN, FLCOND, PROCOM, TRAT, and NOZZL.

INITAL: Subroutine INITAL is a special-purpose initialization and setup routine that is called once prior to executing subroutine LOOP. INITAL selects the analog console and initializes the analog modes through calls to hybrid linkage routines. INITAL calls subroutine DATAIN to accomplish the reading in of component performance map data and the scaling of those data. The dimensions of the map arrays and the scaled map data are stored in the MAPS common block. Digital coefficients are read in by INITAL and stored in the COEF common block. The inlet option integer INLET and the bleed integers KBH, KBL, and KBV are read in and stored in the INL and BLEED common blocks, respectively. INITAL then reads the punched cards containing the integrator data. INITAL translates the integrator gain integers into specific analog patching instructions. Table V relates the desired integrator gain (1) to the selection of either a 1- or 10-input gain by positioning of a function relay and (2) to the selection of additional multiplicative gain by appropriate patching of logic signals into the "F" and "MS" time-scale inputs on the EAI 681 integrators. INITAL outputs patching instructions to the user at the line printer (or CRT). INITAL also reads in DAC initial condition data, stores them in the IDAC common block, and initializes the DAC's. Finally, INITAL reads in the potentiometer data and sets the potentiometers to their design-point values.

LOOP: Subroutine LOOP is a scaled-fraction Fortran implementation of the steady-state turbofan engine model. Scaled versions of equations (1) to (85) and the integrands in equations (86) to (107) (excluding (104) and (105)) are programmed in LOOP and are repetitively solved. For each pass through LOOP, (1) hybrid linkage routines are called to read ADC's, (2) the derivative calculations are performed, and (3) hybrid linkage routines are called to output the DAC's. To support the calculation of derivatives, LOOP calls subroutines FLCOND, PROCOM, TRAT, and NOZZL and function routines MAP, MAPL, FUN1, and FUN1L. Common blocks MAPS, COEF, INL, BLEED, and IDAC provide LOOP with the required map data, digital coefficients, integer data, and DAC initial conditions. A common block VAR is used to store all the variables computed in LOOP, making it possible to output any of the variables through INFORM. DATA statements in LOOP define univariate functions $C_{d,n} = f_{20}(P_0/P_7)$ and $C_{v,n} = f_{23}(P_0/P_7)$. LOOP represents a 1:1 correspondence to the floating-point (but scaled) Fortran code in the host program's ENGINE subroutine. The most noticeable differences are the declaration of scaled-fraction variables, the "S" prefix on the square-root function in LOOP. The ordering of calculations in each equation in LOOP is intended to minimize the chances of scaled-fraction overflows during off-design and transient operation of the simulation. To minimize the effects of digital delays on simulation accuracy and stability, the sampling of analog variables and the output of digital results are organized as shown in figure 11. References 9 and 22 indicate that this method of breaking up the digital loops can significantly reduce the dynamic errors associated with digital delays in hybrid simulations.

TABLE V. – EAI 681 ANALOG INTEGRATOR GAIN PATCHING

[Master (default) control sets F-SEC (F-MS); logic 1 denotes "high"; logic 0 denotes "low"; function relay plus position feeds 10 input to integrator; function relay minus position feeds 1 input to integrator.]

Desired integrator gain	"F" logic input	"MS" logic input	Function relay position
1	0	0	–
10	1	0	–
100	1	0	+
1000	0	1	–
10000	1	1	–
100000	1	1	+

LEVEL8: Subroutine LEVEL8 is called by EXEC1 upon user command by means of a pushbutton at the analog console. LEVEL8 calls the map- and function-out-of-range routines MOOR and FOOR to output messages to the user indicating the map or function number and the map or function input values at the time the out-of-range condition occurred. Appendix C shows the logic patching on the analog console that is required by LEVEL8.

BTASK: Subroutine BTASK, called by EXEC1, unscales and outputs simulation data upon user command by means of a sense switch at the digital console. As written, BTASK reads in cards containing scale factors and desired values for the variables to be listed. BTASK outputs tables of unscaled simulation variables and ratios of calculated-to-desired values.

DATAIN: Subroutine DATAIN is a general-purpose data input routine that is called by INITAL to read component performance map data from cards and to scale the map data. The call to DATAIN contains two arguments: an integer array associated with the first map to be read, and the scaled-fraction array into which the scaled data for the first map will be placed. DATAIN reads punched cards that contain (for each map) the map number, the dimensions of the map, scale factors for the map variables, formats for reading the map data, and the map data. DATAIN reads the cards, scales the data, and fills the integer and scaled-fraction arrays. DATAIN reads map data until a blank card is detected. Therefore all the maps can be read by one call to DATAIN. The integer array contains the map number, the search index for X (initialized to 1), the search index for Y (initialized to 1), the number of points per curve, and the number of curves defining the map. The scaled-fraction array contains the scaled Y values, the scaled X values for all Y values, and the scaled Z values for the functions in the map.

FLCOND: Subroutine FLCOND is a scaled-fraction Fortran implementation of equations (1) to (6).

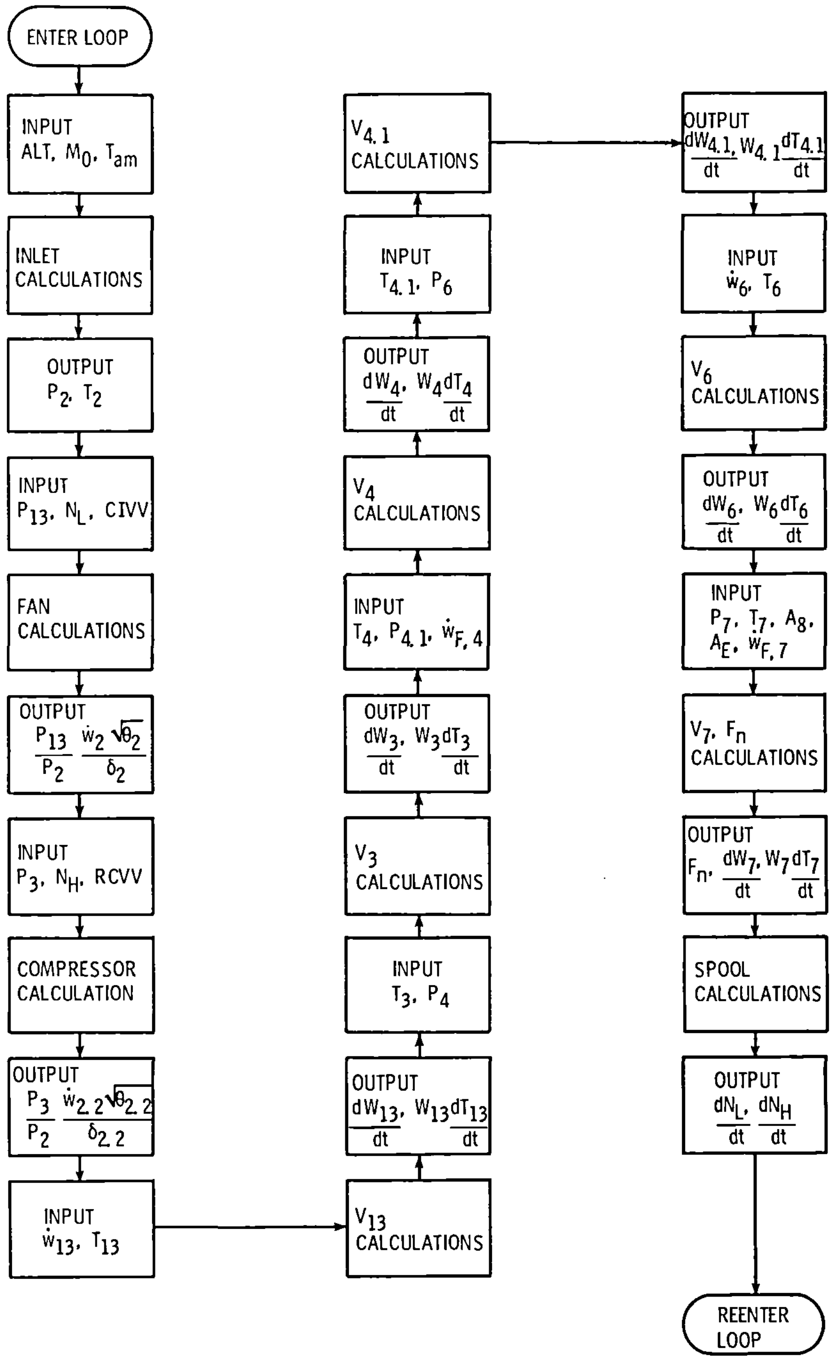


Figure 11. - Data transfer in subroutine LOOP.

FLCOND computes ambient pressure and temperature, fan inlet total pressure, and fan inlet total temperature from input values of altitude, Mach number, and sea-level ambient temperature. The inputs to FLCOND are altitude (scaled to a maximum of 80 000 ft), Mach number (scaled to a maximum of 3.0), and sea-level ambient temperature (scaled to a maximum of 1000° R). The inlet option integer INLET is input through the INL common block. A univariate function routine FUN1 is called by FLCOND to interpolate tabular data representing the functions $f_1(a)$, $(T_2/T_0)^{\gamma_l/(\gamma_l-1)}$, and $(M_0-1)^{1.35}$. These functions were scaled and fit by tabular data because of the excessive computing times associated with exponentiation. The tabular data are defined by DATA statements in FLCOND. A linear approximation to $f_2(a)$ is used in FLCOND. The outputs of FLCOND are ambient pressure (scaled to a maximum of 20-psia), ambient temperature (scaled to a maximum of 1000° R), fan inlet total pressure (scaled to a maximum of 40 psia) and fan inlet total temperature (scaled to a maximum of 1000° R).

PROCOM: Subroutine PROCOM is a scaled-fraction implementation of equations (7) to (11). PROCOM computes thermodynamic properties for air and JP-4/air mixtures. PROCOM is called for each engine station during each pass through the LOOP subroutine. Inputs to the subroutine are the gas temperature (scaled to a maximum of 5000° R) and the fuel-air ratio (scaled to a maximum of 0.08). Curve fits of the tables of reference 17 are used in PROCOM. The outputs of PROCOM are the specific heats c_p and c_v (scaled to a maximum of 0.5 Btu/lbm °R), the specific heat ratio (scaled to a maximum of 2.0), and the specific enthalpy (scaled to a maximum of 1500 Btu/lbm). Appendix D contains the source listing for the Fortran version of PROCOM. Because of the many calls to PROCOM in LOOP, PROCOM was coded in assembly language to reduce the computing time.

TRAT: Subroutine TRAT is a scaled-fraction Fortran implementation of equations (16), (19), and (25). TRAT computes the isentropic temperature rise parameter $(\Delta T/T)_{id}$ for a given pressure ratio and specific heat ratio. TRAT is called three times by LOOP (once each for the fan inside diameter, the fan outside diameter, and the compressor). The inputs to TRAT are an integer indicating the component for which the call is being made (fan inside diameter = 1, fan outside diameter = 2, compressor = 3), the pressure ratio (rescaled to a maximum of 15 prior to calling TRAT), and the average component specific heat ratio (scaled to a maximum of 2). TRAT calls FUN1 to interpolate tabular data representing the function $(P/P)^{(\gamma-1)/\gamma} - 1$ for $\gamma = 1.35$. The tabular data are defined by DATA statements in TRAT. The integer argument in the calling statement specifies which integer array (N1, N2, or N3) should be used in the call to FUN1. This allows the search index to be saved for each of the three calls to TRAT. An algebraic function of pressure ratio and specific heat ratio is used to correct the FUN1 output for different values of the specific heat ratio. The output of the TRAT subroutine is the temperature rise parameter (scaled to a maximum of 1.25).

NOZZL: Subroutine NOZZL is a scaled-fraction Fortran implementation of equations (53) to (83). NOZZL computes the exhaust nozzle flow rate and gross thrust. The required digital coefficients are input through the COEF common block. The inputs to NOZZL are the scaled values of ambient pressure, nozzle inlet total pressure, nozzle inlet total temperature, nozzle throat area, and nozzle exit area; the nozzle pressure ratio (scaled to a maximum of 1.0); the nozzle flow coefficient (scaled to a maximum of 1.0); and the nozzle velocity coefficient (scaled to a maximum of 1.0). FUN1 is called by NOZZL to interpolate tabular data representing the functions $f_{21}(A_E/A_8)$, $f_{21}^{-1}(P_0/P_7)$, $f_{22}(P_0/P_7)$, $f_{24}(A_E/A_8)$, $f_{28}(A_E/A_8)$, and $f_{30}(A_E/A_8)$. The tabular data are defined by DATA statements in NOZZL. The functions $f_{25}(M_x)$ and $f_{26}(M_x)$ are represented by quadratic functions. The iterative loop associated with shocks in the divergent section is replaced by a quadratic function of pressure ratio that is biased by a cubic function of area ratio. The result, M_E^*, is used to compute the nozzle exit velocity. The outputs of NOZZL are the nozzle flow rate, the gross thrust, and the pressure-area component of gross thrust.

Function routines. – The function routines are FUN1, FUN1L, and FOOR and MAP, MAPL, and MOOR.

MAP/MAPL/MOOR: Routine MAP is a general-purpose, scaled-fraction routine for generating functions of two variables. MAP was developed to handle functions that cannot be defined by rectilinear arrays. That is, each curve Y does not have to extend over the entire range of the variable X defining the curves. For this reason MAP is used extensively for generating fan and compressor performance maps. Because of its efficient coding, MAP is also used in the target program for generating the turbine maps. MAP performs radial interpolation of the map data and saves both X and Y search indices prior to returning to the calling program. Upon reentering MAP for the same function the search begins in the previously determined X and Y intervals, thus saving considerable time during relatively steady-state operation. MAP is described in more detail in reference 20.

If more than one function of the same two independent variables is required, a second entry point to MAP, called MAPL, may be used. However, this applies only if identical X and Y breakpoints are used. MAPL fetches and interpolates table values only. No searching is required since the search indices are used from the last entry to MAP.

Subroutine MOOR is called by MAP and is used to inform the user (via a line printer or CRT) when a map input goes out of range. This is important since MAP does not extrapolate.

The inputs to MAP are an integer array containing the map number, the X and Y search indices, the number of points per curve, and the number of curves; a scaled-fraction array (F) containing the X, Y, Z_1, Z_2, . . ., Z_n data; and the scaled-fraction input variables. The ordering of the data in the F array is handled by the DATAIN subroutine. For the case of multiple functions of common variables, only one argument (the scaled-fraction array) is needed in the second, third, etc., calls. That is,

$$Z1 = \mathrm{MAP}(N1,F1,XIN,YIN) \tag{108}$$

$Z2 = \text{MAPL(F1)}$ (109)

Note that MAP and MAPL must be declared as scaled-fraction variables in the calling program.

Appendix D contains source listings for Fortran versions of MAP, MAPL, and MOOR. As in the case of PROCOM, assembly language versions of these routines were used in the target program to reduce the computing time.

Subroutine DATAIN is used in conjunction with MAP and MAPL to read unscaled map data and scale factors into the target program, to scale the map data, and to store the results in the MAPS common block.

FUN1/FUN1L/FOOR: Routine FUN1 is a general-purpose, scaled-fraction routine for generating functions of a single variable. In the turbofan engine simulation, FUN1 is used to generate fixed functions that do not require user-specified data. The function data are defined by DATA statements in the calling routines. FUN1 performs linear interpolation of the tabular data and saves the search index for subsequent calls for the same function.

If more than one function of the same independent variable is required, a second entry point to FUN1, called FUN1L, may be used. However, this applies only if identical breakpoints are used. For multiple functions of the same variable the DATA statements must be in the following order: X, Y1, Y2, . . ., YN, where the Y's are in the order of the calls to FUN1 and FUN1L.

Subroutine FOOR is called by FUN1 and is used to inform the user (by line printer or CRT) when a function input goes out of range. This is important since FUN1 does not extrapolate.

The inputs to FUN1 are an integer array containing the function number, the search index, and the number of points defining the function; a scaled-fraction array containing the X data; and the scaled-fraction input variable. The proper ordering of the DATA statements in the calling program and the coding in FUN1 result in FUN1 using the appropriate Y values in the interpolation. For the case of multiple functions of the same variable, only one argument (the scaled-fraction array) is needed in the second, third, etc., calls. That is,

$Y1 = \text{FUN1(N1,X1,XIN)}$ (110)

$Y2 = \text{FUN1L(X1)}$ (111)

Note that FUN1 and FUN1L must be declared as scaled-fraction variables in the calling program.

Appendix D contains source listings of Fortran versions of FUN1, FUN1L, and FOOR. Assembly language versions of these routines were used in the target program to reduce the computing time.

Program statistics. – Table VI lists the significant statistics for the target program. These statistics include the required numbers of analog components, the interface requirements, the core storage requirements, and the execution times for the various routines that make up the dynamic loop. Comparing tables I(b) and VI(a) shows that the analog computing requirements of the target program do not tax the capabilities of one EAI 681 analog console. Tables I(c) and VI(b) show that the ADC and DAC requirements are also satisfied by either of the Lewis hybrid systems. The core requirement of the basic target program (LOOP plus required subroutines) is only about 7.6K words. However, the addition of the initialization routine INITAL and the EXEC1 main program and their associated subroutines (DATAIN, INFORM, etc.) expands the total target program to about 25K words. The total LOOP execution time is about 23 milliseconds with assembly language versions of PROCOM, MAP/MAPL/MOOR, and FUN1/FUN1L/FOOR used.

Host Digital Program

One of the fundamental differences between the simulation development process described in this report and the process described in reference 14 is the current use of a host digital program. This program, written in Fortran, is intended to run on a suitable, large-scale digital computer (in our case, the IBM 370/3033) and performs all the data manipulations and calculations (1) that are necessary to translate engine design information into a functioning hybrid simulation, and (2) that were previously done by hand.

In addition to performing the required calculations the host program incorporates features that can significantly reduce simulation development time on the hybrid computer. These features include (1) automatic trimming of the engine model to match user-supplied design-point data, (2) a quantitative evaluation of subsystem models relative to user-supplied off-design-point data, and (3) computer printouts that provide documentation for the hybrid simulation.

The host digital program is structured in a modular fashion to accept user-supplied input data (by means of punched cards), to perform the required operations on that data, and to output the desired information to the user (through computer printouts and punched cards). The following sections describe the required input data, the organization of the host program, and the individual subroutines and function routines that make up the host program.

Input Data

Figure 12 shows the nature and ordering of the user-supplied input data. The first card contains three integers in a (1X,8(I1,2X)) format that specify punch options JP,

TABLE VI. - TARGET PROGRAM STATISTICS

(a) Analog Program

Components	Number required
Integrators	16
Summers	19
Multipliers	6
x^2	2
Dividers	8
Potentiometers	53
Function relays	10
"AND" gates	4
BCD counter	1
Logic pushbutton	1

(b) Interface

Components	Number required
ADC's	25
DAC's	21
Control lines	2
Sense lines	3 (2 if INFORM used)
General-purpose interrupt	1 (only if INFORM used)
Real-time clock	1

(c) Digital program

Basic software	Core storage, number of words[a]	Execution time, msec
LOOP	1805	22.5
FLCOND	219	.95
PROCOM	[b]179 (266)	.28
TRAT	263	.67
NOZZL	528	1.94
MAP/MAPL	[b]213 (622)	0.28/0.08
MOOR	[b]136 (116)	
FUN1/FUN1L	[b]106 (282)	0.15/0.06
FOOR	[b]125 (102)	
LINKAGE	271	
RUN-TIME LIBRARY	591	
COMMON	3007	
ZONE0	120	
Total	[b]7563 (8202)	

INFORM-related software	Core storage, number of words
EXEC1	629
INITAL	1368
INFORM	6710
DATAIN	941
BTASK	823
LEVEL0	2
LEVEL8	36
MOOR[c]	[b]272 (232)
FOOR[c]	[b]250 (204)
LINKAGE[c]	981
RUN-TIME LIBRARY[c]	5142
ZERO0[c]	89
Total	[b]17 243 (17 154)
Grand total	[b]24 882 (25 432)

[a]16-Bit word.
[b]Assembly-language version used for test case; number in parentheses denotes number of words in Fortran version.
[c]Additional loads required by INFORM.

JPD, and JPA. JP = 1 will cause all the previously described output data cards to be punched. JPD = 1 will cause only the performance map data, digital coefficients, inlet option, and bleed integers to be punched. JPA = 1 will cause only the analog setup data to be punched. The second card contains the inlet configuration option INLET in a (1X,I1) format. INLET = 1 will cause the military specification inlet pressure recovery characteristic to be used in the host program calculations. Any other value will result in an inlet recovery of 1.0 (no inlet). INLET is transmitted to the target program through the punched cards. The third card contains two constants K_{BLWHT} and K_{BLWLT} in a (3X,2F12.5) format. These constants represent the fraction of turbine cooling bleed that performs turbine work for the high- and low-pressure turbines, respectively.

The next six sets of cards contain bivariate function data related to the engine components' performance. The contents and order of the component performance map cards are the same as described in the preceding input data section under Target Digital Program. The last set of map data is followed by a blank card. Following the blank card are six cards containing duplicates of the map scale factor cards. This repeated input of map scale factors is necessary because the scale factors are needed to unscale certain map variables in the host program. The scale factors, read as part of the map data set, are not returned to the MAIN program from the DATAIN subroutine.

Following the component map scale factors are nine cards containing scale factors for the key simulation variables. Table VII lists the variables (in order) that require user-supplied scale factors. The scale factor data are read in a 5(F12.5) format.

Following the scale factor data is a card containing two integers, NDRY and NAUG, in a (1X,2(I2,2X)) format. These integers specify the number of nonaugmented (dry) and augmented operating points to be input to the host program by the user. That is, a total of NDRY + NAUG sets of engine cycle data (pressures, temperatures, etc.) are input by the user. By definition, the first operating point read in is the dry "design" point. It would usually be the maximum nonaugmented thrust condition. The NDRY + 1 point is assumed to be the augmented design point and would usually be the maximum thrust condition.

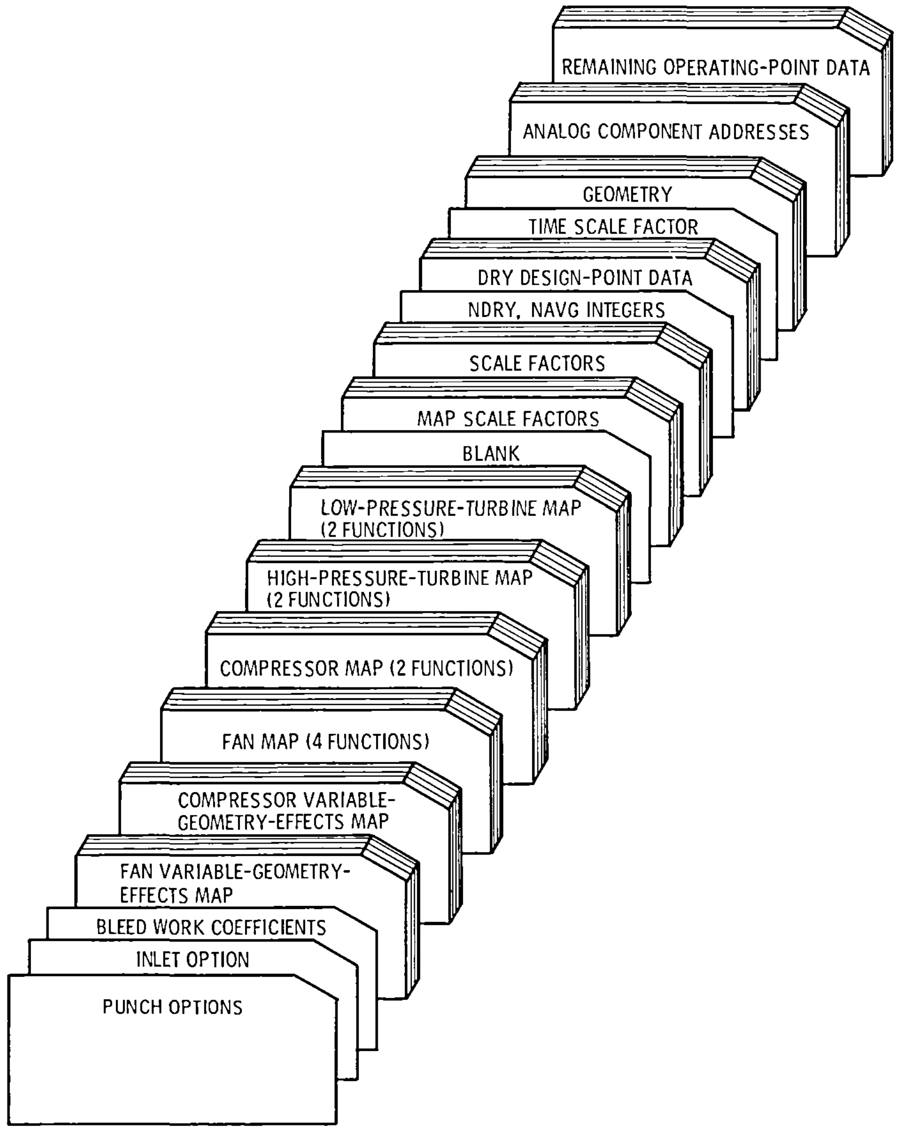


Figure 12. - User-supplied host program input data.

Following the NDRY,NAUG specification card are cards containing the dry design-point data. The first card contains a power lever angle (PLA) indicator or label in a (9X,I3) format. The following nine cards contain, in a (5F12.5) format, steady-state values for the engine variables listed (in order) in table VIII.

Following the dry design-point data are cards containing the desired time-scale factor (in a (1X,F5.0) format) and engine geometric data. The geometric parameters, discussed in the section Engine Dynamics are volumes V_{13}, V_3, V_4, $V_{4.1}$, V_6, and V_7, reactances $(A/l)_D$, and $(A/l)_{AB}$, and rotor inertias I_H and I_L. The volumes are read in a (6F12.5) format. The reactances and inertias are read in a (4F12.5) format.

Following the geometric data are cards containing the user-assigned analog potentiometer and integrator addresses. The analog component addresses would be selected by the user from the available complement of analog equipment. A total of 53 potentiometers and 16 integrators must be selected and patched in the prescribed manner (table III and appendix C). The 53 potentiometer addresses are read in a (1X,8(A4,1X)) format. The 16 integrator addresses are read in an (8I4) format. The analog component addresses are subsequently transmitted to the target program along with the potentiometer settings and integrator gain integers computed by the host program.

Following the analog component addresses are cards containing additional dry and augmented operating-point data. As previously stated, the NDRY+1 point is assumed to be the augmented design point. Note that each set of operating-point data is preceded by a PLA

TABLE VII. - SIMULATION VARIABLES REQUIRING USER-SUPPLIED SCALE FACTORS

Order of input of scale factor	Variable	Order of input of scale factor	Variable
1	P_0	23	$\dot{w}_4$
2	P_2	24	$\dot{w}_{4.1}$
3	P_{13}	25	$\dot{w}_6$
4	$P_{2.2}$	26	$\dot{w}_7$
5	P_3	27	Δh_{HT}
6	P_4	28	Δh_{LT}
7	$P_{4.1}$	29	F_n
8	P_5	30	N_L
9	P_6	31	N_H
10	P_7	32	$\dot{w}_{F,4}$
11	T_2	33	$\dot{w}_{F,7}$
12	T_{13}	34	A_8
13	$T_{2.2}$	35	A_E
14	T_3	36	a
15	T_4	37	M_0
16	$T_{4.1}$	38	CIVV[a]
17	T_6	39	CIVV[b]
18	T_7	40	RCVV[a]
19	$\dot{w}_2$	41	RCVV[b]
20	$\dot{w}_{13}$	42	v_E
21	$\dot{w}_{2.2}$	43	T_0
22	$\dot{w}_3$	44	F_M

[a]Scale factor is bias to be subtracted from variable.
[b]Scale factor is scale factor on biased variable.

TABLE VIII. - SIMULATION VARIABLES REQUIRING USER-SUPPLIED STEADY-STATE DESIGN AND OFF-DESIGN VALUES

Order of input of scale factor	Variable	Order of input of scale factor	Variable
1	P_0	24	$\dot{w}_4$
2	P_2	25	$\dot{w}_{4.1}$
3	P_{13}	26	$\dot{w}_6$
4	$P_{2.2}$	27	$\dot{w}_7$
5	P_3	28	Δh_{HT}
6	P_4	29	Δh_{LT}
7	$P_{4.1}$	30	η_B
8	P_5	31	η_{AB}
9	P_6	32	F_n
10	P_7	33	N_L
11	T_{am}	34	N_H
12	T_2	35	$\dot{w}_{F,4}$
13	T_{13}	36	$\dot{w}_{F,7}$
14	$T_{2.2}$	37	A_8
15	T_3	38	A_8
16	T_4	39	a
17	$T_{4.1}$	40	M_0
18	T_6	41	$C_{d,N}$
19	T_7	42	$C_{v,N}$
20	$\dot{w}_2$	43	CIVV
21	$\dot{w}_{13}$	44	RCVV
22	$\dot{w}_{2.2}$	45	F_n
23	$\dot{w}_3$		

indicator card. Reference 15 contains listings of user-supplied input data for the turbofan engine test case.

Organization

Figure 13 shows the modular structure of the host digital program. The program flow is controlled by the MAIN program. The MAIN program and subroutines called by MAIN are used to read the input data cards. The subroutines and function routines operate on the input data and output to the user the information needed for setup of the target program on the hybrid computer. The following sections describe the various host program routines. A flow chart of the host program and source listings of the host program routines are provided in appendix E.

MAIN Program

The principal function of the MAIN program is to control the flow of the host program through calls to subroutines MAPIN, PRINT, DCOEF, ENGINE, and ANALOG. These subroutines perform the bulk of the data input and output, the data manipulation, and the calculations. The MAIN program also does a limited amount of data input and output and scaling.

The punch options, inlet option, and bleed work coefficients are read by the MAIN program and stored in common blocks IVARS and INL.

The MAIN program calls subroutine MAPIN to accomplish the reading in and scaling of the component performance map data. The resulting integer and scaled-function data arrays are stored in the NMAPS common block. Computer printouts of unscaled map data are generated for each map. The call to MAPIN also produces values for some of the digital coefficients. They are stored in the AVARS common block. MAPIN also reads in the map scale factor data and stores them in the AVARS common block. The MAIN program then reads in the NDRY and NAUG integer data. These integers are used to control a DO loop in which the user-supplied operating-point data are read in and used. After each set of operating-point data is read in, a call to subroutine PRINT is made to obtain a computer printout of the operating-point data.

For each operating point the MAIN program computes values of the compressor discharge bleed flows $\dot{w}_{\mathrm{BLHT}}$, $\dot{w}_{\mathrm{BLLT}}$, and $\dot{w}_{\mathrm{BLOV}}$ required to match the user-supplied cycle data. If nonpositive bleed values are computed at the dry design point (IF = 1), zero bleed will be assumed at all operating points and the corresponding bleed

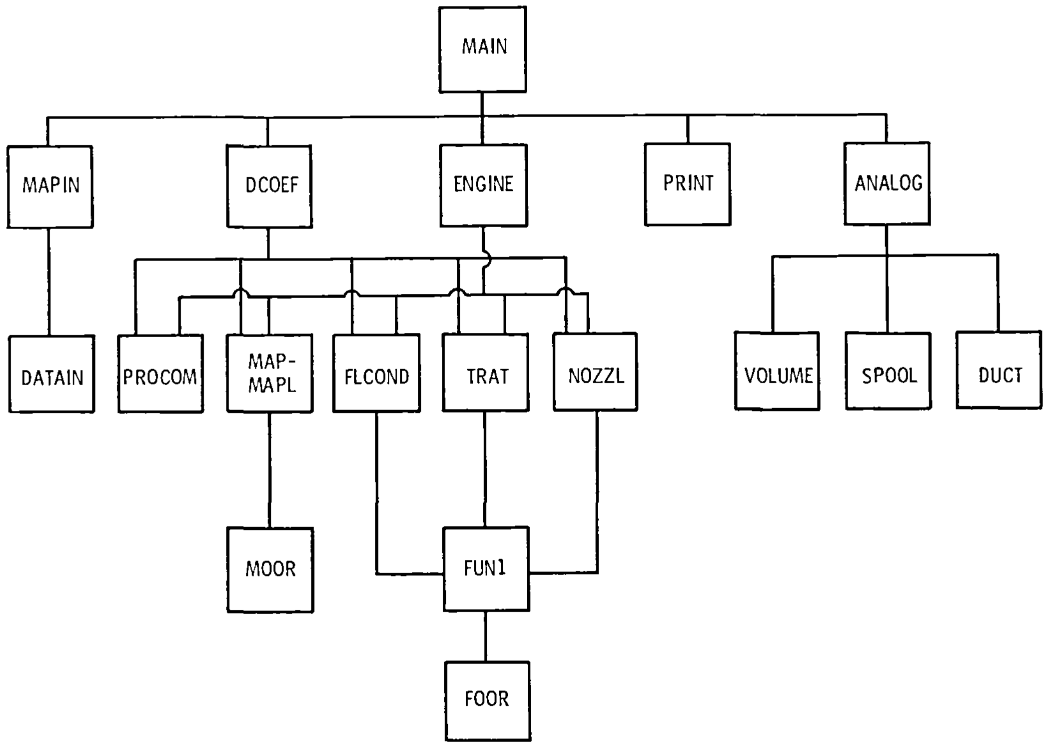


Figure 13. - Organization of host digital program.

integers (KBH, KBL, and KBV) will be set to zero. The computed bleed flows are scaled in the MAIN program by using scale factors that are fixed proportions of the compressor flow scale factor.

$$SF_{\dot{w}_{BLHT}} = 0.2\, SF_{\dot{w}_{2.2}} \tag{110}$$

$$SF_{\dot{w}_{BLLT}} = 0.02\, SF_{\dot{w}_{2.2}} \tag{111}$$

$$SF_{\dot{w}_{BLOV}} = 0.002\, SF_{\dot{w}_{2.2}} \tag{112}$$

Each set of user-supplied operating point data is scaled by the MAIN program from the user-supplied scale factors read in by MAPIN and stored in the SVARS common block.

After the input and scaling of the IP = 1 and IP = NDRY + 1 design-point data, the MAIN program calls subroutine DCOEF to compute additional digital coefficients, correction (trim) factors, and corrected (trimmed) digital coefficients. The correction factors, CC(i), uncorrected digital coefficients, UDC(i), and corrected digital coefficients DC(i) are stored in the AVARS common block.

For each operating point the MAIN program calls subroutines ENGINE, ANALOG, and PRINT. Subroutine ENGINE uses the corrected digital coefficients and floating-point Fortran versions of the scaled equations in the target subroutine LOOP to compute scaled values of the engine state variable derivatives. At the IP = 1 point the corrected digital coefficients will produce (essentially) zero derivatives and hence an equilibrium condition at the user-specified operating point. However, at the other operating points no such guarantee of a balanced condition exists. Therefore calculations are performed in ENGINE for each operating point to obtain ratios of individual component model outputs to the output values required to achieve the desired equilibrium condition. Nonunity values for these ratios indicate the need for modifications to the models to match the operating-point data. The model evaluation ratios are stored in the RVARS common block.

The IP = 1 call to ANALOG causes time scale factor, geometric, and analog component address data to be read into the host program. Potentiometer settings and integrator gains are calculated in ANALOG. For IP = 1 the call to ANALOG produces model evaluation ratios DP6R and DP13R that indicate how well the augmentor and bypass duct models (implemented on the analog in the target program) match the user-supplied cycle data. These ratios are also stored in the RVARS common block. After the calls to ANALOG, calls to PRINT result in computer printouts of all scaled and unscaled simulation variables, including such thermodynamic parameters are specific heat ratios and enthalpies. The scaled variables are stored in the XVARS common block.

After all the operating points have been treated in the host program, the MAIN program calls subroutines

ANALOG and PRINT to produce printouts of analog information (potentiometer settings, integrator gains, etc.) and digital information (digital coefficients, bleed integers, etc.) and to punch the desired data on cards (if JP, JPD, or JPA equals 1). The final call to PRINT also produces printouts of the model evaluation ratios.

Subroutines

MAPIN. – The MAPIN subroutine performs two main functions. The first is to control the reading in of component performance map data through a call to the DATAIN subroutine. The second is to read in user-supplied scale factors and to compute those digital coefficients that are functions of the scale factors. The integer and scaled-map data arrays are filled by DATAIN and stored in the NMAPS common block. The scale factors are stored in the SVARS common block. The computed digital coefficients are stored in the AVARS common block.

DCOEF. – The DCOEF subroutine is used to compute digital coefficients and correction factors from design-point data. DCOEF is called by MAIN for the IP = 1 and IP = NDRY + 1 operating points only. At the IP = 1 point the digital coefficients are computed from the input values of pressures, temperatures, flow rates, etc., and the scale factors. The correction factors are then determined and applied to the uncorrected digital coefficients. The corrections are intended to compensate for interpolation errors, etc., and will produce (essentially) zero derivatives at the IP = 1 design point. At the IP = NDRY + 1 point, additional coefficients and correction factors are computed such that a balanced condition exists in the augmentor at the maximum thrust point. To do this, DCOEF uses scaled, floating-point Fortran code that is equivalent to the code in the ENGINE subroutine. The uncorrected digital coefficients and correction factors are stored in the AVARS common block along with the corrected digital coefficients.

ENGINE. – The ENGINE subroutine represents a scaled, floating-point Fortran equivalent to the scaled-fraction Fortran subroutine LOOP in the target program. The scaled equations that describe the turbofan engine model are solved in ENGINE by using the corrected digital coefficients determined in MAPIN and DCOEF.

ENGINE is called once for each user-specified operating point, IP = 1, 2, . . ., NDRY + NAUG. Scaled values of the engine pressures, temperatures, duct flow rates, rotor speeds, and controlled inputs (fuel flows, nozzle areas, etc.) represent ADC inputs to the LOOP subroutine in the target program. The scaled derivatives, computed in ENGINE, represent variables to be output by LOOP through the DAC's. It should be emphasized that no numerical integrations are performed in the host program. This is analogous to holding the target program's analog integrators in their initial condition (IC) mode. Nonzero derivatives, computed in ENGINE, indicate a nonequilibrium condition that would, if not corrected by model changes, result in a different steady-state operating point than the one specified by the user. To aid the user in evaluating the off-design performance of the model and determining what (if any) changes need to be made to the model, ENGINE computes individual model evaluation ratios at each operating point. The model evaluation ratios represent multipliers that, if applied to the corresponding model's output, will produce the desired equilibrium condition. By analyzing the evaluation ratios over an operating line, correlations with engine parameters can be developed, incorporated into the model, and reevaluated prior to going to the hybrid. The model evaluation ratios are stored in the RVARS common block. The XVARS common block is used to store all the scaled variables computed in ENGINE.

ANALOG. – The ANALOG subroutine is used by the host program (1) to read in engine geometric data and analog component addresses, (2) to compute analog potentiometer settings and integrator gain integers at the dry design point, (3) to compute bypass and augmentor duct model evaluation ratios at all operating points, and (4) to generate computer printouts of all pertinent analog setup information.

The functions to be performed by ANALOG are governed by the value of the calling argument APRINT and the operating-point index IP (input through the IVARS common block). APRINT is set to zero in MAIN prior to the call for each operating point. For APRINT = 0 and IP = 1, ANALOG reads user-supplied values of the time scale factor, engine geometric parameters, potentiometer addresses, and integrator addresses. ANALOG then calls subroutines VOLUME, SPOOL, and DUCT to compute the potentiometer settings and the integrator gain integers for each element in the engine model. Each gain integer represents the power-of-10 gain required for the integrator(s) associated with the particular element. The VOLUME subroutine also computes the stored mass value and the stored mass scale factor for each intercomponent volume in the model. Although DUCT is called for each operating point, the potentiometer settings and integrator gain integers associated with each duct are not updated for IP = 1. For these points the computed values of the feedback potentiometers (PVAL(4) in fig. 6) are ratioed to the IP = 1 values. The ratios DP13R and DP6R indicate how well the duct model matches the off-design-point data. ANALOG also computes potentiometer settings corresponding to simulation input variables (fuel flows, etc.) at the dry design point.

After the ENGINE, ANALOG, and PRINT calls for the IP = NDRY + NAUG point, a final call to ANALOG is made with APRINT = 1. At this time computer printouts of al the pertinent analog setup data are generated. The ANVARS common block is used to store the analog data. The computed duct model evaluation ratios are stored in the RVARS common block.

PRINT. – Subroutine PRINT is a multipurpose output routine that performs one of three functions, depending on the value of the calling argument IPRINT. IPRINT is initialized to zero in MAIN and is incremented in PRINT just after each call. For the first call PRINT merely lists the user-supplied operating-point data. For the second call PRINT lists all the scaled variables computed in ENGINE. PRINT proceeds to unscale selected variables by using scale factors stored in the SVARS common block. The unscaled variables are then listed. For the third call to PRINT digital coefficients (uncorrected and corrected) and correction factors are printed out. The coefficients are tested to ensure that they are less than 1.0. If they exceed 1.0, they are flagged in the printout to indicate the need for rescaling of variables. Digital and analog setup data are punched on cards according to the selected punch options (JP, JPD, and JPA). All the model evaluation ratios are printed out after the third call to PRINT.

VOLUME. – The VOLUME subroutine is called by ANALOG for each intercomponent volume in the engine model. The subroutine calculates potentiometer settings and integrator gain integers based on user-supplied design-point data, scale factors, and geometric parameters. The inputs to the subroutine are the unscaled pressure and temperature in the volume, the volume, the time scale factor, and the scale factors for the flow rate through the volume and for the pressure and the temperature. The outputs of the subroutine are an array of five potentiometer settings, a gain integer representing the required integrator gain, the unscaled stored mass in the volume, and the stored mass scale factor as determined by VOLUME. Logic is included in the subroutine to determine the stored mass scale factor that will result in a scaled value between 0.5 and 0.8 at the design point. Also, logic is included to ensure that the computed potentiometer settings are less than 1.0.

SPOOL. – The SPOOL subroutine is called by ANALOG for each of the two spools in the engine model. The subroutine calculates potentiometer settings and an integrator gain integer based on user-supplied design-point data, scale factors, and geometric parameters. The inputs to the subroutine are the unscaled rotor speed, the rotor inertia, the time scale factor, and the scale factors for rotor speed, turbine enthalpy drop, and turbine flow rate. The outputs of the subroutine are an array of two potentiometer settings and an integrator gain integer. As in the case of VOLUME, logic is included to ensure that the potentiometer settings are less than 1.0.

DUCT. – Subroutine DUCT is called by ANALOG for each of the two ducts in the engine model. The subroutine calculates potentiometer settings and an integrator gain integer based on user-supplied design-point data, scale factors, and geometric parameters. The inputs to the subroutine are the unscaled values of inlet pressure, exit pressure, inlet temperature, and flow rate; the duct A/l; the inlet volume (includes one-half of the duct volume); the time scale factor; and scale factors for inlet pressure, exit pressure, inlet temperature, and flow rate. The outputs of the subroutine are the array of four potentiometer settings and an integrator gain integer. As in the case of VOLUME, logic is included to ensure that the potentiometer settings are less than 1.0.

DATAIN. – DATAIN is a floating-point Fortran version of the scaled-fraction Fortran subroutine in the target program. In the host program, DATAIN is called by MAPIN to read component performance map data from user-supplied cards. DATAIN scales the map data and fills the appropriate integer and scaled-data arrays. The host program version of DATAIN contains a provision for punching unscaled map data cards for input to the target program. The map data are punched if the punch options JP = 1 or JPD = 1 are selected by the user.

FLCOND. – FLCOND is a floating-point Fortran version of the scaled-fraction version in the target program. In the host program, FLCOND is called by ENGINE and DCOEF to compute the scaled values of the ambient pressure and fan inlet total pressure and temperature from specified scaled values of altitude, Mach number, and sea-level ambient temperature.

PROCOM. – PROCOM is a floating-point Fortran version of the scaled-fraction version in the target program. In the host program, PROCOM is called by ENGINE and DCOEF to compute scaled values of JP-4/air thermodynamic properties based on specified scaled values of temperature and fuel-air ratio. The thermodynamic properties are the specific heats c_p and c_v, the specific heat ratio, and the specific enthalpy.

TRAT. – TRAT is a floating-point Fortran version of the scaled-fraction version in the target program. In the host program, TRAT is called by ENGINE and DCOEF to compute the scaled isentropic temperature rise parameter $(\Delta T/T)_{id}$ based on specified scaled values of pressure ratio and specific heat ratio.

NOZZL. – NOZZL is a floating-point Fortran version of the scaled-fraction version in the target program. In the host program, NOZZL is called by ENGINE and DCOEF to compute the scaled values of nozzle flow rate and nozzle gross thrust based on specified scaled values of ambient pressure, nozzle inlet pressure and temperature, nozzle throat area, nozzle exit area, flow coefficient, and velocity coefficient.

Function Routines

MAP/MAPL/MOOR. – Function routine MAP is a floating-point Fortran version of the scaled-fraction assembly language version in the target program. In the host program, MAP is called by ENGINE and DCOEF to interpolate tables of scaled, bivariate function data based on specified scaled values of the independent variables. As in the case of the target program a second entry point (MAPL) is used when multiple functions of the same independent variables are to be generated. Subroutine MOOR is called by MAP to flag out-of-range inputs to MAP.

FUN1/FUN1L/FOOR. – Function routine FUN1 is a floating-point Fortran version of the scaled-fraction assembly language version in the target program. In the host program, FUN1 is called by ENGINE, DCOEF, FLCOND, TRAT, and NOZZL to interpolate tables of univariate function data based on specified scaled values of the independent variable. In general, FUN1 is used in the host program to generate fixed functions (defined by DATA statements in the calling subroutines). As in the case of the target program, a second entry point (FUN1L) is used when multiple functions of the same independent variable are to be generated. Subroutine FOOR is called by FUN1 to flag out-of-range inputs to FUN1.

Program Statistics

Table IX lists the core storage requirements for the host program. Approximately 24K words are required.

TABLE IX. – HOST PROGRAM STATISTICS

Software	Core storage,[a] number of words
MAIN	1088
MAPIN	1242
DCOEF	1628
ENGINE	1463
ANALOG	1581
PRINT	6802
VOLUME	249
SPOOL	157
DUCT	258
DATAIN	1672
FLCOND	234
PROCOM	205
TRAT	246
NOZZL	471
MAP	292
MAPL	73
MOOR	136
FUN1	158
FUN1L	71
FOOR	108
COMMON	5662
Total	23 796

[a]32-Bit words.

Turbofan Engine Test Case

To demonstrate the computer-aided simulation development process, a test case was run. The test case involved simulating an engine in the 111.1-kilonewton (25 000-lbf) thrust class operating at sea-level, static conditions from idle to maximum thrust. The following section describes the test case engine.

Engine Description

The test case engine is an augmented turbofan engine, as illustrated in figure 3. The engine is assumed to have the standard inlet pressure recovery as defined by equation (3). The turbine cooling bleeds are configured in such a way as to have bleed work coefficients K_{BLWHT} and K_{BLWLT} of 0.63412 and 0.17058, respectively. Table X lists these and other salient parameters for the test case engine.

TABLE X. – TEST-CASE ENGINE PARAMETERS

K_{BLWHT}	0.63412
K_{BLWLT}	0.17058
V_3	0.0467 m^3 (2850 in.^{3})
V_4	0.0467 m^3 (2850 in.^{3})
$V_{4.1}$	0.6555 m^3 (40 000 in.^{3})
V_6	0.8473 m^3 (51 705 in.^{3})
V_7	0.7046 m^3 (43 000 in.^{3})
V_{13}	1.426 m^3 (87 050 in.^{3})
$(A/l)_D$	13.15 cm (5.1778 in.)
$(A/l)_{AB}$	13.15 cm (5.1778 in.)
I_H	515.2 N cm sec^2 (45.6 lbf in. sec^2)
I_L	610.1 N cm sec^2 (54.0 lbf in. sec^2)

The test case engine is assumed to have variable fan inlet guide vanes (CIVV) and variable compressor stator vanes (RCVV). Figures 14 and 15 show the effects of variable vanes on the steady-state performance of the fan and compressor, respectively. Figure 14 corresponds to the function f_8 in the engine model and to map 1 in the host and target programs. Figure 15 corresponds to the function f_{12} and to map 2. Figures 16 to 19 show the steady-state performance characteristics of the fan, compressor, high-pressure turbine, and low-pressure turbine, respectively. Figures 16 and 17 represent the nominal fan and compressor performance with scheduled vane angles (outputs of maps 1 and 2 equal to zero). Figures 16(a) to (d) correspond to functions f_6, f_7, f_9, and f_{10} and to map 3. Figures 17(a) and (b) correspond to functions f_{11} and f_{13} and to map 4. For the turbines figures 18(a) and (b) correspond to functions f_{14} and f_{15} and to map 5; figures 19(a) and (b) correspond to functions f_{16} and f_{17} and to map 6. Note that the component performance maps are normalized. That is, the map variables are expressed as "fractions of design."

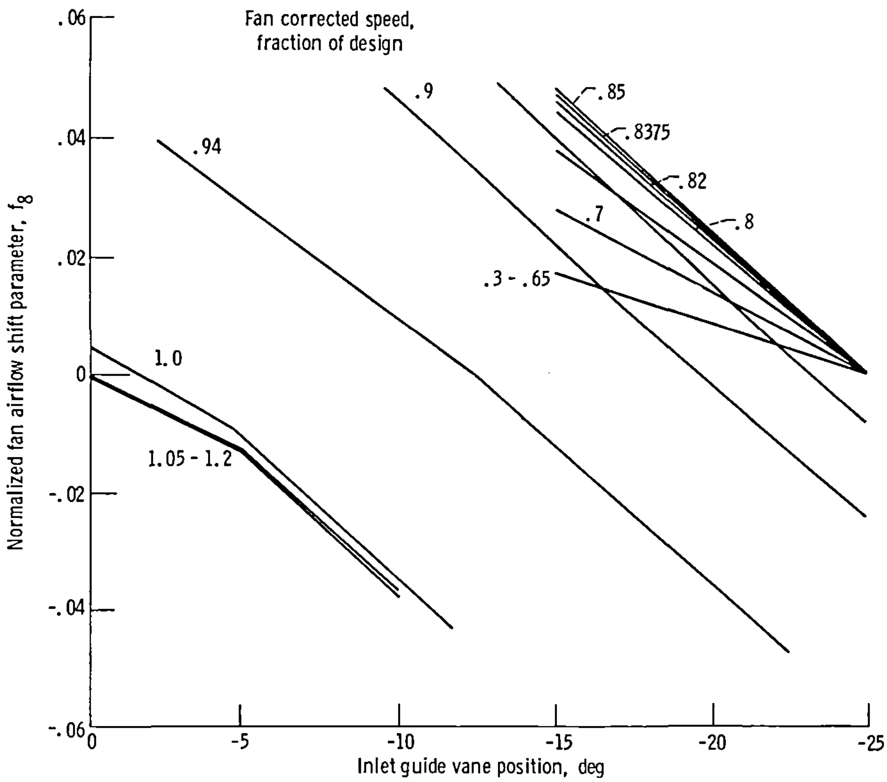


Figure 14. - Effect of variable inlet guide vane position on fan performance - test case engine.

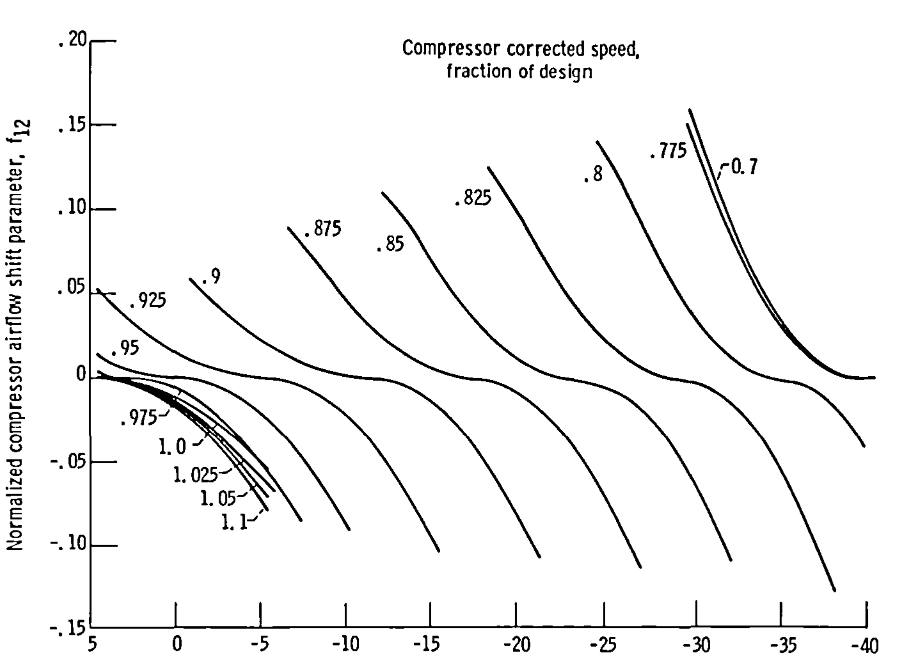


Figure 15. - Effect of variable stator vane position on compressor performance - test case engine.

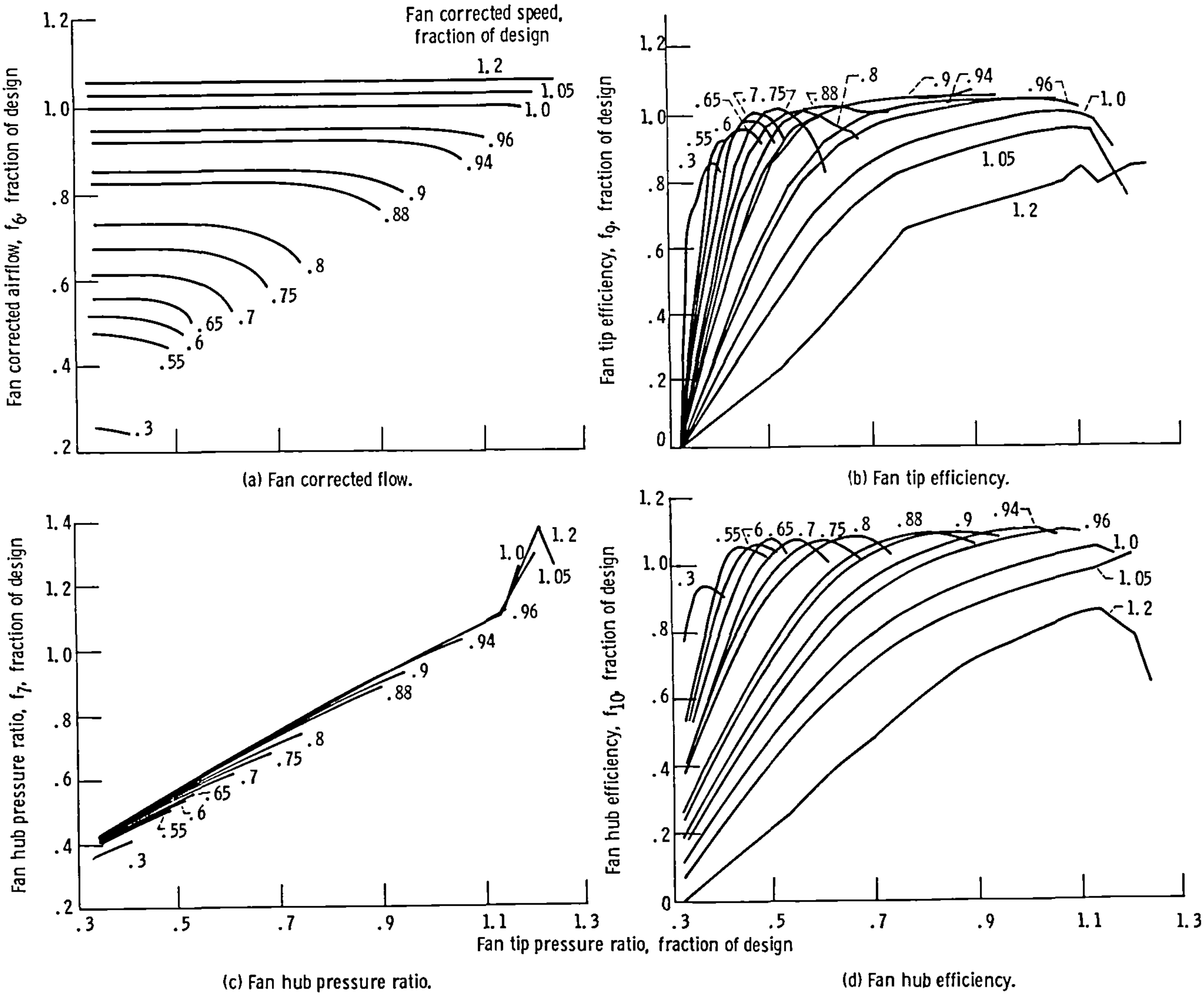


Figure 16. - Fan performance maps with inlet guide vanes at their nominally scheduled position - test case engine.

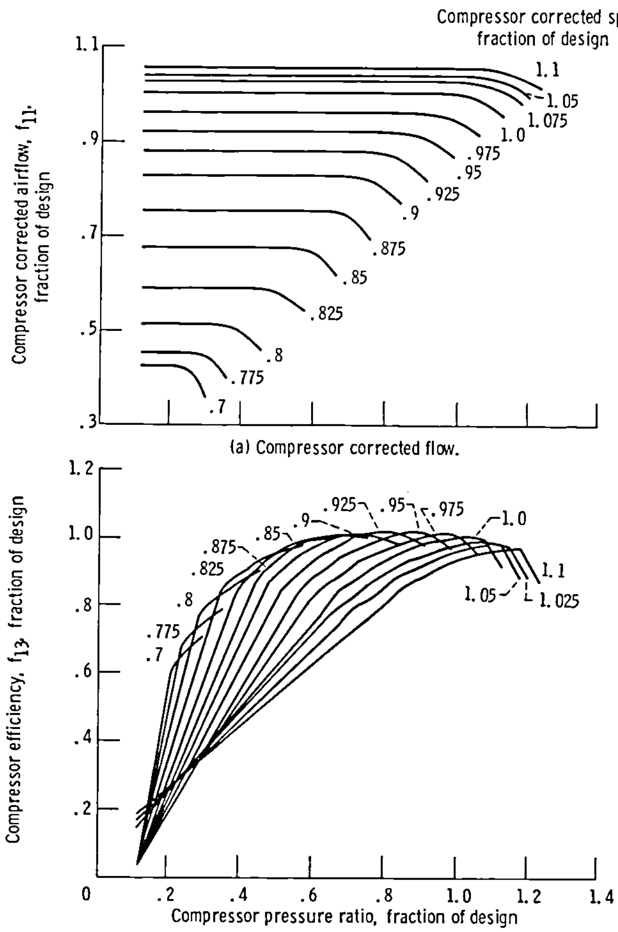
The component design points are assumed to match the engine dry design point.

Table VIII lists the user-supplied engine variables that define the steady-state performance of the engine. Values of these variables were input to the host program for the design and off-design points. The design point in this case was the maximum nonaugmented thrust point at the sea-level, static, standard-day condition. All the relevant design- and off-design-point data for the test case engine are contained in reference 15.

Hybrid Computer Specification

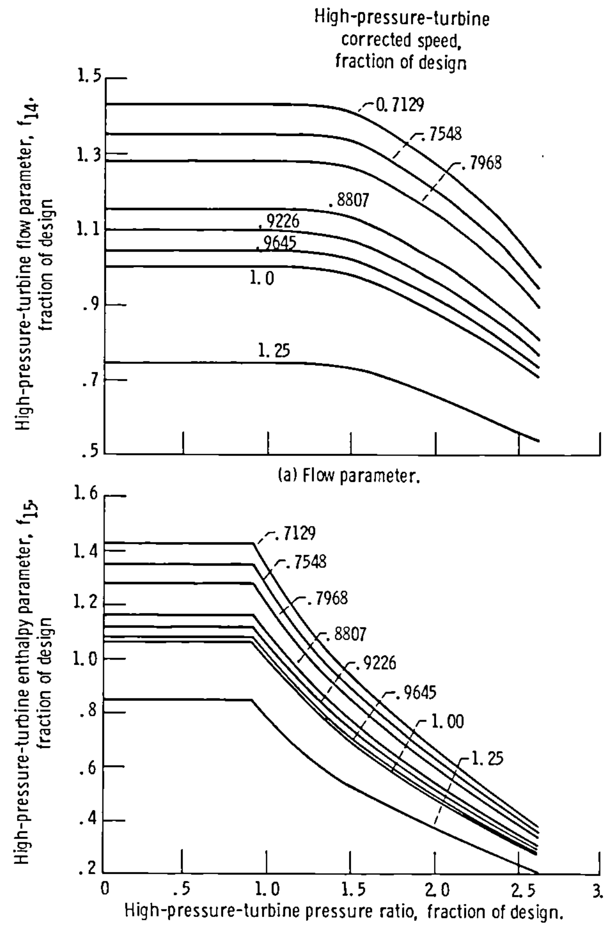
Having defined the engine geometry and steady-state performance, the user must specify scale factors and analog component addresses. The scale factors are used by the host program to calculate digital coefficients, analog potentiometer settings, and integrator gains. The scaling of simulation variables, although not mandatory for the host program, is needed for the target program to permit the use of scaled-fraction software in the digital and to limit analog variables to ± 1 computer unit (± 10 V on the EAI 681). On the basis of the expected range of variables, the scale factors listed in table XI were chosen for the test case. The time scale factor, which is directly related to the integrator gains, was set to 50 (i.e., 50 times slower than real time) to ensure stable operation. The effect of the time scale factor on the stability of the test case simulation is discussed in the section Results and Discussion.

The specification of analog component addresses by the user permits the host program to generate data that can be used to either automatically set up or instruct the



(b) Compressor efficiency.

Figure 17. - Compressor performance map with stator vanes at their nominally scheduled position - test case engine.



(b) Enthalpy parameter.

Figure 18. - High-pressure-turbine performance maps - tests case engine.

user to set up the analog computer. Table XII lists the analog components that were patched according to the guidelines provided in table III and appendix C.

Host Program Input Data

The ordering of the user-supplied input data is shown in figure 12. Reference 15 contains a listing of the input data for the test case. The first card contained the punch options JP = 1, JPD = 1, and JPA = 1. This caused the host program to punch all the host program output data. The second card contained the inlet option INLET = 1. The third card contained the bleed work coefficients $K_{BLWHT} = 0.63412$ and $K_{BLWLT} = 0.17058$.

Following the bleed work coefficient data were the cards defining the test case engine component performance maps. Numerical data corresponding to the maps shown in figures 14 to 19 were read in along with information pertaining to map dimensions, data formats, and scale factors. The section Host Digital Program contains detailed descriptions of the map data cards. A blank card followed the map 6 data. This was followed by cards containing the scale factors for the input and output variables for maps 1 to 6.

Following the map scale factors were cards containing user-defined scale factors for the engine variables that are listed in table VII. Table XI lists the engine variables and the corresponding scale factors.

Following the scale factor data was the card containing the integers NDRY and NAUG. For the test case six dry operating points and four augmented operating points were input to the host program. These points covered the sea-level, static, standard-day operating line from idle to maximum thrust. The dry design-point (PLA = 83) data followed the specification of NDRY and NAUG. These data were followed by the specification of the time scale factor. For the test case a scale factor of 50 was selected. Following the time scale factor were cards containing the engine geometric data corresponding to table X. These data were followed by user-specified analog potentiometer and integrator addresses. Table XII lists the component addresses selected for the test case.

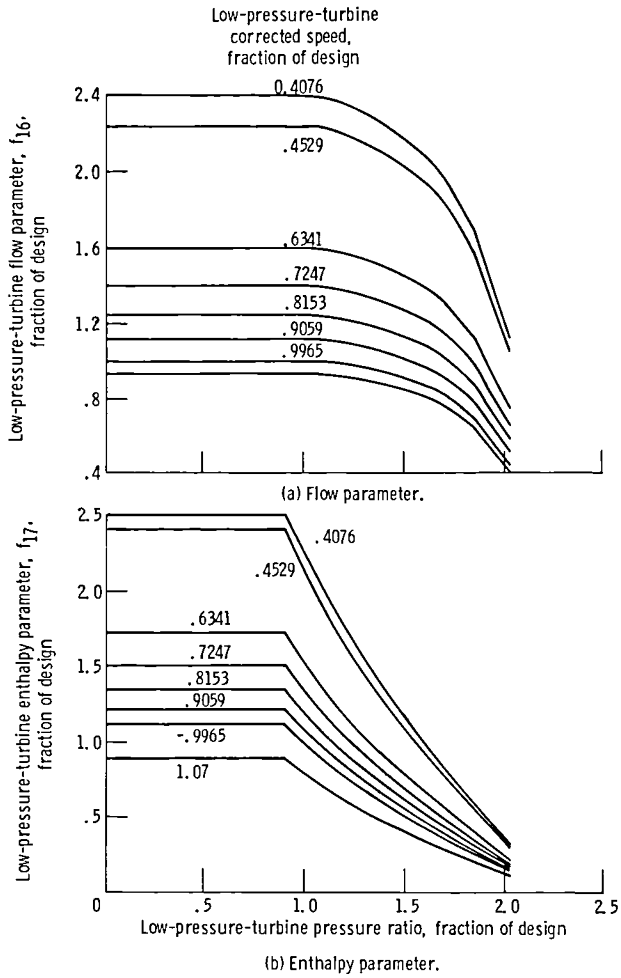


Figure 19. - Low-pressure-turbine performance maps - test case engine.

Following the component addresses were cards defining the remaining NDRY + NAUG − 1 operating points.

Host Program Output Data

The ordering of the host-program-generated output data cards is shown in figure 8. Reference 15 contains a listing of the host program output data for the test case. The first six sets of data were reproductions of the component performance map data cards that were input to the host program. A blank card was generated by the host program after the last card for map 6. Following the map data cards were cards containing the 125 corrected digital coefficients. These were followed by a card containing the inlet option integer INLET and the bleed integers KBH, KBL, and KVB. For the test case, all four integers were set to 1 by the host program. That is, the standard inlet recovery was to be used in the target program and positive bleeds were needed at the dry design point to match the user-supplied steady-state data. Following the inlet and bleed integers were cards

TABLE XI. - SCALE FACTORS FOR TEST CASE

Variable, x	Scale factor, SF_x	Variable, x	Scale factor, SF_x
P_0	20 psia/c.u.[a]	$\dot{w}_4$	200 lbm/sec-c.u.
P_2	40 psia/	$\dot{w}_{4.1}$	200 lbm/sec
P_{13}	100 psia/	$\dot{w}_6$	450 lbm/sec
$P_{2.2}$	100 psia/	$\dot{w}_7$	450 lbm/sec
P_3	600 psia/	Δh_4	300 Btu/lbm-c.u.
P_4	600 psia/	$\Delta h_{4.1}$	200 Btu/lbm
$P_{4.1}$	150 psia/	F_n	50 000 lbf/c.u.
P_5	100 psia/	N_L	15 000 rpm/c.u.
P_6	100 psia/	N_H	15 000 rpm/
P_7	100 psia/	$\dot{w}_{F,4}$	4.5833 lbm/sec-c.u.
T_2	1000° R/c.u.	$\dot{w}_{F,7}$	20 lbm/sec
T_{13}	1500° R/	A_8	1000 in.^{2}/c.u.
$T_{2.2}$	1500° R/	A_E	2000 in.^{2}/
T_3	2000° R/	a	80 000 ft/c.u.
T_4	4000° R/	M_0	3 c.u.$^{-1}$
$T_{4.1}$	3000° R/	CIVV[b]	0 deg
T_6	2000° R/	CIVV[b]	25 deg/c.u.
T_7	5000° R/	RCVV[c]	4 deg
$\dot{w}_2$	450 lbm/sec-c.u.	RCVV[b]	44 deg/c.u.
$\dot{w}_{13}$	250 lbm/sec	V_E	5000 in./sec-c.u.
$\dot{w}_{2.2}$	225 lbm/sec	T_0	1000° R/c.u.
$\dot{w}_3$	200 lbm/sec	F_N	50 000 lbf/c.u.

[a] c.u. denotes computer unit.
[b] Scale factor is bias subtracted from variable.
[c] Scale factor is applied to biased variable.

containing the analog integrator addresses and gain integers. This information was used in the target program's subroutine INITAL to set (or instruct the user to set) the proper integrator gains. The gain integers indicated that gains of 1 were to be set for integrators 00, 02, 15, 17, 30, and 35; gains of 10 for integrators 05, 07, 10, 12, 20, 22, 25, 27, and 95; and a gain of 100 for integrator 90. Following the integrator data were cards containing initial condition values for the 21 DAC's being used. The last set of cards contained potentiometer addresses and settings for the 53 potentiometers being used. This information was also used by subroutine INITAL to automatically set the potentiometers.

Reference 15 contains a complete printout from the execution of the host program for the test case. In addition to listing the input and output data, the host program printed out scaled and unscaled values of calculated engine variables at each operating point. At the dry design point, small ($\sim 10^{-6}$) values of the scaled derivatives (DAC outputs) were observed. This was caused by the trim procedure in the DCOEF subroutine. Larger values did result at off-design points (DXNH = 0.93447×10^{-3} at PLA = 20, e.g.) and indicated the need for modifications to the engine model to match specified operating data when the loops are closed through the analog integrators.

TABLE XII. – ANALOG COMPONENT SELECTION FOR TEST CASE

Model element	Potentiometer address index, k	Potentiometer address	Integrator address index, k	Integrator address
V_{13}	1	P000	1	00
	2	P001	2	02
	3	P002		
	4	P003		
	5	P004		
V_3	6	P005	3	05
	7	P006	4	07
	8	P007		
	9	P008		
	10	P009		
V_4	11	P010	5	10
	12	P011	6	12
	13	P012		
	14	P013		
	15	P014		
$V_{4.1}$	16	P015	7	15
	17	P016	8	17
	18	P017		
	19	P018		
	20	P019		
V_6	21	P020	9	20
	22	P021	10	22
	23	P022		
	24	P023		
	25	P024		
V_7	26	P025	11	25
	27	P026	12	27
	28	P027		
	29	P028		
	30	P029		
I_H	31	P032	13	30
	32	P030		
I_L	33	P037	14	35
	34	P035		
$(A/I)_D$	35	P090	15	90
	36	P091		
	37	P092		
	38	P093		
	39	P095		
$(A/I)_{AB}$	40	P096		
	41	P097		
	42	P098		

Model input	Potentiometer address index, k	Potentiometer address
$w_{F,4}$	43	P081
$w_{F,7}$	44	P087
A_8	45	P085
A_E	46	P086
CIVV[a]	47	P038
RCVV[a]	48	P080
ALT	49	P031
M_0	50	P033
T_{am}	51	P036

[a]Biased input.

Model output bias	Potentiometer address index, x	Potentiometer address
P_{13}/P_2	52	P100
$P_3/P_{2.2}$	53	P101

After the printout of the DAC initial conditions the host-program-determined values of the stored mass scale factors (and corresponding scaled and unscaled values of the stored masses) were listed.

In addition to printing out the corrected digital coefficients, the host program also printed out the uncorrected digital coefficients and correction factors. Digital coefficients that were zero (such as unused coefficients) or that equaled or exceeded unity were flagged to aid the user in detecting situations that might require rescaling of the simulation or redefinition of coefficients.

Finally the host program printed out tables of the model evaluation ratios for each operating point. As expected, the ratios were equal to 1.0 at the dry design point. Nonunity values at other operating points indicated the need for modifying the individual component models to produce zero derivatives at the off-design points. Subsequently, modifications were made to various subsystem models. In particular, the bleed flow, augmentor efficiency, nozzle flow, and gross thrust models were adjusted to achieve near-unity values for CBLHR(k), ETAABR(k), WG7R(k), and FGR(k), respectively. Nonunity values for other ratios such as

P0	P2	P13	P22	P3
P4	P41	P5	P6	P7
TAM	T2	T13	T22	T3
T4	T41	T6	T7	WA2
WA13	WA22	WA3	WG4	WG41
WG6	WG7	DH4	DH41	ETAB
ETAAB	FNET	XNL	XNH	WF4
WF7	A8	AE	ALT	XMN
CDN	CVN	CIVV	RCVV	FG

14693 02	14629 02	19354 02	21024 02	10624 03
10140 03	29343 02	18634 02	18176 02	17743 02
51868 03	51846 03	57962 03	58713 03	10065 04
17317 04	12398 04	84204 03	84229 03	10679 03
55679 02	51038 02	42511 02	42926 02	50720 02
10703 03	10706 03	11101 03	31854 02	99997 00
58252 00	23239 04	56818 04	98209 04	45066 00
00000 00	43201 03	48499 03	00000 00	00000 00
88507 00	95322 00	-24993 02	-21839 02	23239 04

STEADY-STATE ERROR RATIOS

.99980	.99543	.96923	.98765	1.01735
1.01615	1.02044	1.04198	1.01873	1.01801
1.00013	.99984	.99298	.99569	1.00134
.99597	.99600	.98166	.98209	1.07015
1.13542	1.00780	1.01703	1.01600	1.01676
1.06981	1.06816	1.00369	.97728	.99997
.98984	1.11041	1.06119	1.00321	1.00068
.00000	.99987	1.00032	.00000	.00000
1.00029	.99947	.99973	.99997	1.11553

Figure 20. - Target simulation steady-state results - baseline model; point 5 (PLA = 24).

DP13R(k) and DP6R(k) were initially judged to have lesser effects on the steady-state accuracy of the simulation and were not adjusted. The section Results and Discussion treats the impact of this.

Results and Discussion

The target simulation was implemented and run on the PACER 600 hybrid computer system. The analog portion of the simulation was prepatched as shown in appendix C. The digital portion of the simulation, consisting of the EXEC1 main program, subroutines including INITAL and LOOP, function routines, and library routines, was compiled (or assembled) and loaded by using the procedures described in reference 20. Execution of the INITAL subroutine resulted in the reading in of the host-program-generated output data and the setting up and initialization of the analog (potentiometers, integrator gains, etc.) at the dry design point.

Stable closed-loop operation of the hybrid simulation (analog computer in OPERATE and repetitive operation of the LOOP subroutine) was not possible with time scale factors much below 50. After some study it was concluded that this was due to relatively high loop gains in the "hot" sections of the engine model and the effective time delays associated with those loops. Although it was felt that the time scale factor could probably be reduced by modifying the structure of the LOOP subroutine (update the high-gain loops more often than the others), we decided not to attempt this and, rather, to concentrate on demonstrating the basic simulation methodology. The host and target programs could then be the basis for later work aimed at reducing the digital frame time and achieving real-time operation.

Steady State

With the simulation inputs (potentiometers) fixed at the design-point values, the turbofan engine simulation was allowed to run until an equilibrium (steady state) condition was reached. The INFORM subroutine was then used to obtain a tabular listing of unscaled steady-state data. Variables to be displayed and their scale factors were defined prior to run time by using the procedures described in reference 21. The selected variables and the format of the tables were chosen to closely match the tables of host program input data (ref. 15). Figure 20 shows the steady-state hybrid results at the IA = 5 (PLA = 24) point. Reference 15 contains all the operating line data. Note that the unscaled data were displayed as XXXXX EE, which represents 0.XXXXX times 10 to the EE power. Also note that steady-state error ratios (observed value/desired value) were listed for each variable. This provided a convenient comparison between hybrid and host program input data.

At the design point the steady-state errors were generally less than 1.0 percent. This was attributed to the trim procedure in the host program, which leads to extremely small state variable derivatives at the design point. As one moved further away from the design point, larger errors were observed. At point 5 (PLA = 24) the largest error was found to be 13.5 percent in $\dot{w}_{13}$ (see WA13 in fig. 20). These results were not completely unexpected since the host program had produced relatively large derivatives at these points along with nonunity values for the model evaluation ratios. However, the lack of convergence capability in the host program made it necessary to run the hybrid to determine the actual equilibrium conditions produced by the model.

The steady-state results, coupled with the model evaluation ratios listed in the host program printout, indicated that modifications to the basic engine model would be needed if the steady-state errors were to be reduced to acceptable (3 to 5 percent) levels. In particular, the duct models and the turbine enthalpy drop parameter maps were identified as possible causes for the observed high values of fan speed (and hence flow rates and pressures). The model evaluation ratios DP13R, DP6R, DH4QR, and DH41TR were used to guide modifications to the duct and turbine models.

The bypass duct loss characteristic (eq. 51)) was changed to reflect a linear relationship between the pressure drop and the corrected flow. That is,

$$P_{16}=P_{13}-\frac{K_{D1}\dot{w}_{13}T_{13}^{1/2}}{P_{13}}-K_{D2} \qquad (113)$$

Figure 21 illustrates the modified implementation of the bypass duct model.

The augmentor duct model was modified primarily to reduce the observed steady-state errors at the augmented operating points. To do this, the fixed value of K_{AB} in equation (46) was replaced with a nonlinear function of afterburner fuel flow. Figure 22 shows the modified implementation of the augmentor duct model.

In addition to the analog changes, it was determined that the steady-state errors could be further reduced by employing speed-sensitive shifts of the high- and low-pressure-turbine enthalpy drop parameter maps. These shifts were accomplished in the LOOP subroutine by the following steps:

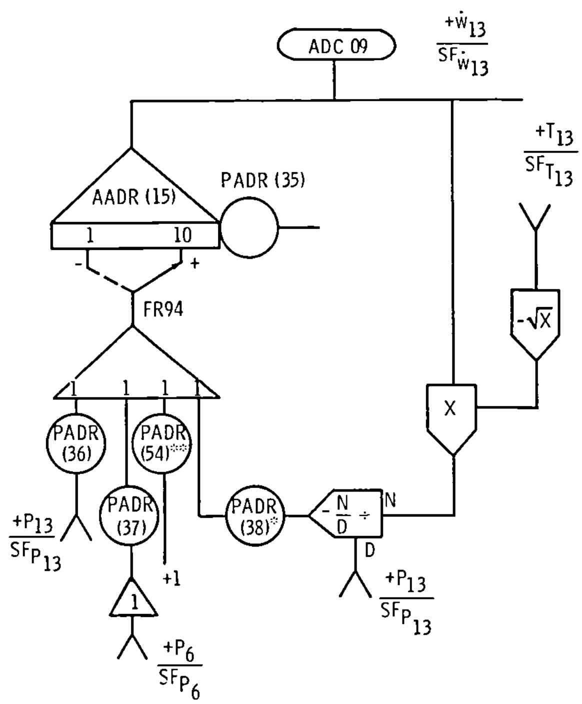
(1) Defining and declaring variables for two additional table lookups as follows:

DIMENSION N8(3),N9(3)

SCALED FRACTION X8(8),Y8(8),X9(8),Y9(8), NHSHFT,NLSHFT

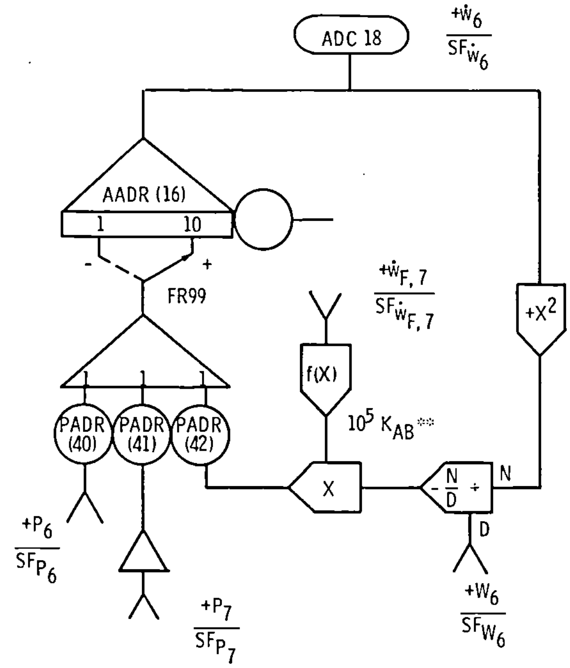
(2) Defining tables of data for the two shift functions as follows:

DATA N8/18,1,8/



* PVAL (38) CHANGED TO 0.0157.
** NEW POTENTIOMETER ADDED - PVAL (54) = 0.0060.

Figure 21. - Modified analog computation of bypass duct flow rate.



* PVAL (42) CHANGED TO 0.4154.
** FUNCTION DEFINED BY POINTS:

X	F(X)
0	0.0852
.0750	.1480
.2200	.2062
.3800	.2496
.5250	.2732
.9999	.3500

Figure 22. - Modified analog computation of augmentor duct flow rate.

DATA X8/.00000S,.61321S,.65269S,.69753S,.74127S, .79527S,.87567S,.99999S/

DATA Y8/.99999S,.98860S,.98766S,.98766S,.99307S, .99660S,.99960S,.99999S/

DATA N9/19,1,8/

DATA X9/.00000S,.25977S,.35683S,.47077S,.53519S, .61229S,.68580S,.99999S/

DATA Y9/.50000S,.50993S,.49499S,.50557S.,49447S, .49707S,.49811S,.50000S/

(3) Computing and applying the shift functions as follows:

NHSHFT = FUN1(N8,X8,XNH)

P0	P2	P13	P22	P3
P4	P41	P5	P6	P7
TAM	T2	T13	T22	T3
T4	T41	T6	T7	WA2
WA13	WA22	WA3	WG4	WG41
WG6	WG7	DH4	DH41	ETAB
ETAAB	FNET	XNL	XNH	WF4
WF7	A8	AE	ALT	XMN
CON	CVN	CIVV	RCVV	FG

14693 02	14629 02	19952 02	21222 02	10532 03
10068 03	28967 02	18274 02	17822 02	17426 02
51874 03	51859 03	58640 03	58896 03	10055 04
17371 04	12473 04	86462 03	86426 03	99564 02
48935 02	50688 02	42041 02	42554 02	50293 02
99783 02	99838 02	11017 03	32068 02	99997 00
58844 00	20889 04	53677 04	97998 04	45039 00
00000 00	43201 03	48511 03	00000 00	00000 00
88480 00	95367 00	-24995 02	-21844 02	20889 04

STEADY-STATE ERROR RATIOS

.99980	.99543	1.00013	.99784	1.00858
1.00734	1.00738	1.02049	.99788	1.00049
1.00001	.99960	1.00458	.99957	1.00037
.99920	1.00203	1.00785	1.00770	.99828
.99668	.99913	1.00579	1.00690	1.00808
.99816	.99788	.99566	.98667	.99997
1.00041	1.00208	1.00287	1.00087	1.00192
.00000	1.00030	1.00032	.00000	.00000
.99998	.99995	.99973	.99997	1.00208

Figure 23. - Target simulation steady-state results - modified model; point 5 (PLA = 24).

DH4 = (HP4*SSQRT(T4)*XNH*NHSHFT)/DC(87)

.

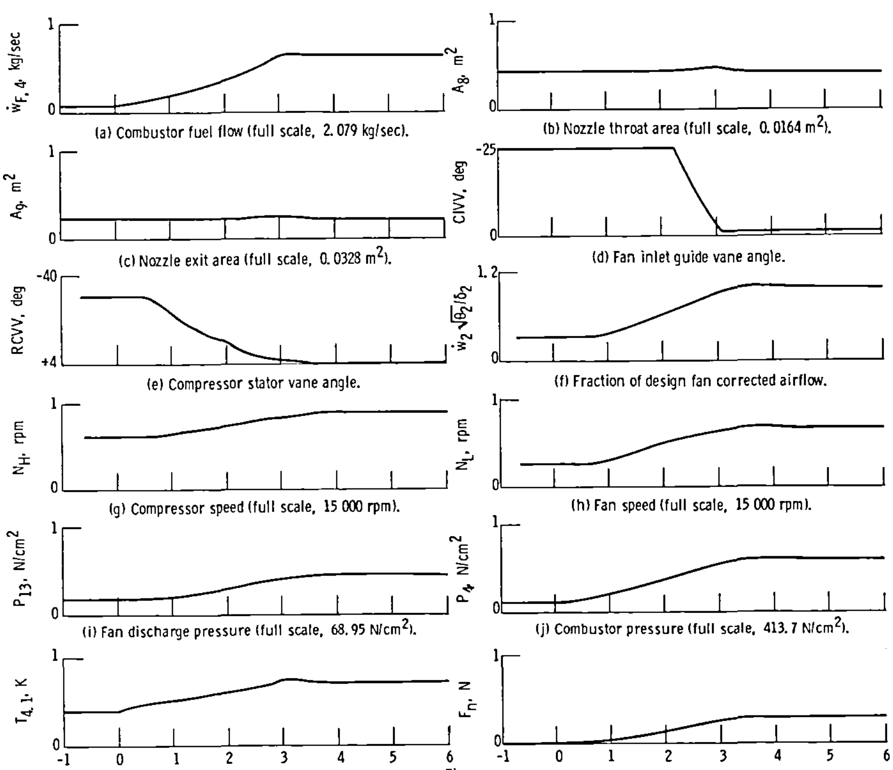
.

.

NLSHFT = FUN1(N9,X9,XNL)

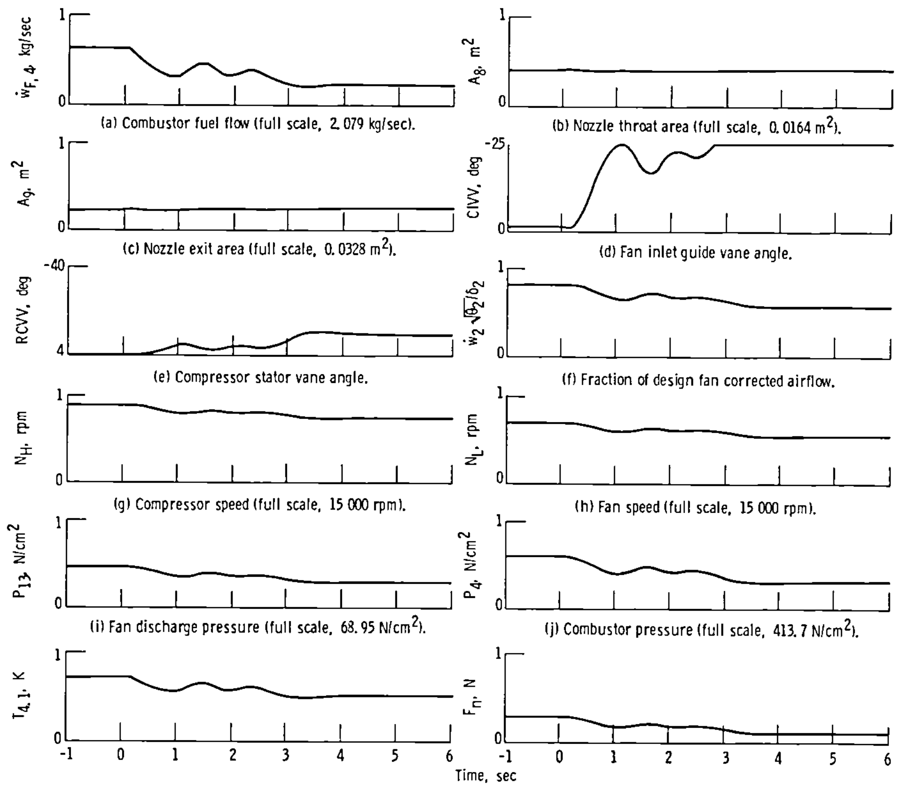
DH41 = (HP41*SSQRT(T41)*XNL*NLSHFT)/(.5S*DC(88))

After these changes had been made to the target program, the simulation was rerun to determine the steady-state errors relative to the host program input data. Reference 15 contains all the resulting data. Figure 23 shows the results at the IP = 5 (PLA = 24) point. In general the modified simulation produced steady-state errors of less than 2.5 percent over the entire operating line. The maximum error in pressures was 2.5 percent (P_5 at PLA = 20). The maximum error in temperatures was 0.99 percent (T_7 at PLA = 20). The maximum error in flows was 0.81 percent ($\dot{w}_{4.1}$ at PLA = 24). The maximum error in rotor speeds was 0.30 percent (N_L at PLA = 20). This level of accuracy should be acceptable for almost any application.



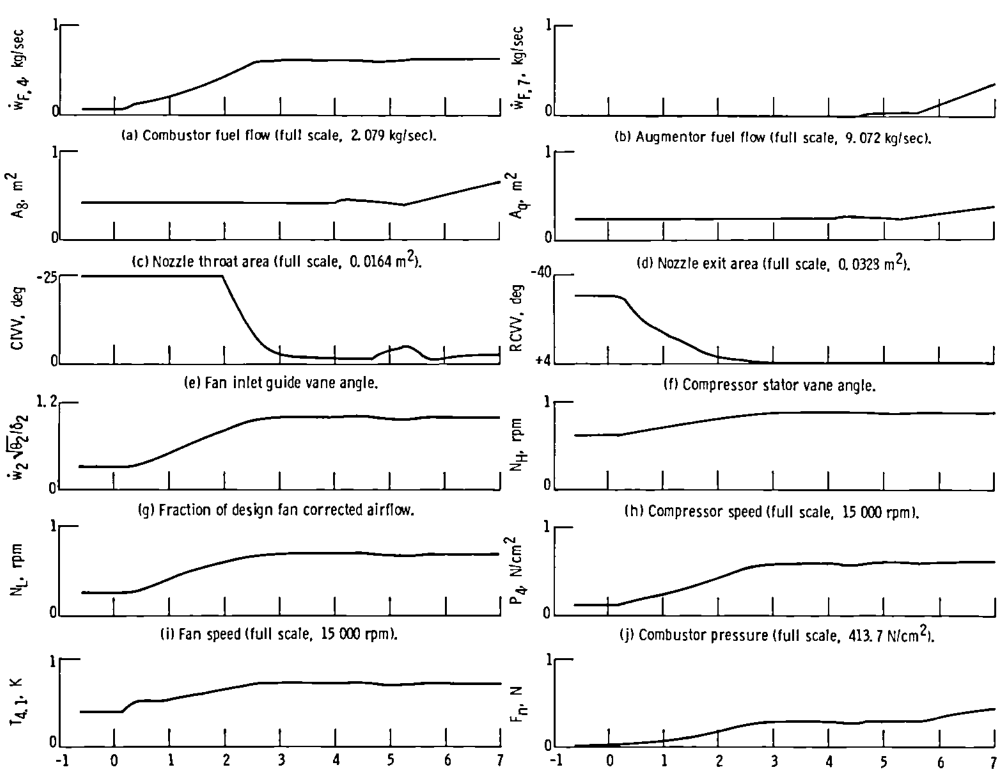
(a) Combustor fuel flow (full scale, 2.079 kg/sec).
(b) Nozzle throat area (full scale, 0.0164 m^2).
(c) Nozzle exit area (full scale, 0.0328 m^2).
(d) Fan inlet guide vane angle.
(e) Compressor stator vane angle.
(f) Fraction of design fan corrected airflow.
(g) Compressor speed (full scale, 15 000 rpm).
(h) Fan speed (full scale, 15 000 rpm).
(i) Fan discharge pressure (full scale, 68.95 N/cm^2).
(j) Combustor pressure (full scale, 413.7 N/cm^2).
(k) Low-pressure-turbine inlet temperature (full scale, 1666.6 K).
(l) Net thrust (full scale, 2.224×10^5 N).

Figure 24. - Target simulation response to inputs representing throttle slam from idle (PLA = 20) to dry design (PLA = 83) power - sea-level static, standard-day conditions.



(a) Combustor fuel flow (full scale, 2.079 kg/sec).
(b) Nozzle throat area (full scale, 0.0164 m^2).
(c) Nozzle exit area (full scale, 0.0328 m^2).
(d) Fan inlet guide vane angle.
(e) Compressor stator vane angle.
(f) Fraction of design fan corrected airflow.
(g) Compressor speed (full scale, 15 000 rpm).
(h) Fan speed (full scale, 15 000 rpm).
(i) Fan discharge pressure (full scale, 68.95 N/cm^2).
(j) Combustor pressure (full scale, 413.7 N/cm^2).
(k) Low-pressure-turbine inlet temperature (full scale, 1666.6 K).
(l) Net thrust (full scale, 2.224x10^5 N).

Figure 25. - Target simulation response to inputs representing cyclic throttle movements between dry design (PLA = 83) and PLA = 40 power - sea-level, static, standard-day conditions.



(a) Combustor fuel flow (full scale, 2.079 kg/sec).
(b) Augmentor fuel flow (full scale, 9.072 kg/sec).
(c) Nozzle throat area (full scale, 0.0164 m^2).
(d) Nozzle exit area (full scale, 0.0323 m^2).
(e) Fan inlet guide vane angle.
(f) Compressor stator vane angle.
(g) Fraction of design fan corrected airflow.
(h) Compressor speed (full scale, 15 000 rpm).
(i) Fan speed (full scale, 15 000 rpm).
(j) Combustor pressure (full scale, 413.7 N/cm^2).
(k) Low-pressure-turbine inlet temperature (full scale, 1666.6 K).
(l) Net thrust (full scale, 2.224x10^5 N).

Figure 26. - Target simulation response to inputs representing throttle slam from idle (PLA = 20) to maximum augmented (PLA = 130) power - sea-level, static, standard-day conditions.

Transients

Although no reference or basis for evaluating the transient performance of the simulation was available, it was felt that it was important to demonstrate the transient operation of the simulation over the entire thrust range. To avoid the need for simulating an engine controller, it was decided to test the simulation dynamics in an "open loop" fashion. That is, time histories of the simulation inputs were constructed from representative engine data and implemented by using analog function generators driven by a ramp signal denoting time. Time histories were generated for three typical engine transients—an acceleration from idle to dry design thrust, a cyclic movement of the throttle, and an acceleration from idle to maximum thrust. Figures 24 to 26 show the simulation responses for these cases. The simulation responses were stable and exhibited reasonable response times, overshoots, etc. No analog component overloads or scaled-fraction overflows were observed, an indication that the selected scale factors and the organization of the target program computations were satisfactory.

Concluding Remarks

The value of hybrid computation as a simulation tool has been aptly demonstrated in a variety of applications, including gas turbine engine controls development. Despite the tremendous technological advances in digital computation (microprocessors, array processors, etc.) it is expected that hybrid computers will continue to play a significant role in simulation because of the speed of the analog computer and the "hands on" interaction available to the user. Still, it is recognized that problems exist when developing hybrid computer simulations and that these problems, if not solved, can significantly reduce the effectiveness of the hybrid approach. In particular, programming aids are needed that can make it easier to formulate models of dynamic systems; that can support the development, implementation, and documentation of the simulation; and that can ensure acceptable levels of steady-state and dynamic accuracy.

This report has focused on the gas turbine engine simulation problem and has presented a systematic, computer-aided, self-documenting methodology for developing a hybrid computer simulation of an augmented turbofan engine. The proposed simulation development process has been exercised, demonstrated, and documented for a typical turbofan engine design. The results indicate that the process does satisfy most of the desired objectives.

Lewis Research Center
National Aeronautics and Space Administration
Cleveland, Ohio, July 14, 1981

Appendix A

Symbols

A	cross-sectional area, cm^2 (in^2)
AADR(i)	address of i^{th} analog integrator, $i=1$ to 16
a	altitude, m (ft)
B_x	bias on variable x, appropriate units
C_d	nozzle flow coefficient
C_v	nozzle velocity coefficient
CIVV	fan variable-geometry parameter, deg
c_p	specific heat at constant pressure, J/kg K (Btu/lbm °R)
c_v	specific heat at constant volume, J/kg K (Btu/lbm °R)
DTQW$\underline{j}$	scaled specific temperature derivative at station j
dt	differential time, sec
dt'	scaled differential time, sec
DW$\underline{j}$	scaled stored mass derivative at station j
DXNH	scaled high-rotor-speed derivative
DXNL	scaled low-rotor-speed derivative
F	thrust, N (lbf)
FR$\underline{jj}$	function relay having address jj
f_i	functional relation, $i=1$ to 30
f/a	local fuel-air ratio
GAIN(i)	analog integrator gain for i^{th} model element, $i=1$ to 10
g_c	gravitational conversion factor, 100 cm kg/N sec^2 (386.3 lbm in/lbf sec^2)
H	heat, J (Btu)
HVF	heating value of fuel, J/kg (Btu/lbm)
h	specific enthalpy, J/kg (Btu/lbm)
Δh	enthalpy change, J/kg (Btu/lbm)
h_p	turbine enthalpy drop parameter, J/kg $K^{1/2}$ rpm (Btu/lbm $°R^{1/2}$ rpm)
I	polar moment of inertia, N cm sec^2 (lbf in sec^2)
J	mechanical equivalent of heat, 100 N cm/J (9339.6 lbf in/Btu)
K_{AB}	augmentor pressure loss coefficient, N^2 sec^2/kg^2 cm^4 K (lbf^2 sec^2/lbm^2 in^4 °R)
K_B	main combustor pressure loss coefficient, N^2 sec^2/kg^2 cm^4 K (lbf^2 sec^2/lbm^2 in^4 °R)
K_{BLWHT}	fraction of high-pressure-turbine cooling bleed doing work
K_{BLWLT}	fraction of low-pressure-turbine cooling bleed doing work
K_D	duct pressure loss coefficient, N^2 sec^2/kg^2 cm^4 K (lbf^2 sec^2/lbm^2 in^4 °R)
K_{D1}	modified duct pressure loss coefficient, N^2 sec/kg cm^4 $K^{1/2}$ (lbf^2 sec/lbm in^4 $°R^{1/2}$)
K_{D2}	modified duct pressure loss coefficient, N/cm^2 (psia)
K_{PR5}	low-pressure-turbine discharge pressure loss coefficient
l	length, cm (in.)
M	Mach number
N	rotational speed, rpm
P	total pressure, N/cm^2 (psia)
PADR(i)	address of i^{th} analog potentiometer, $i=1$ to 53
P/P	pressure ratio
PVAL(i)	setting for potentiometer PADR(i), $i=1$ to 53
p	static pressure, N/cm^2 (psia)
Q	torque, N cm (in lbf)
R	gas constant, N cm/kg K (in lbf/lbm °R)
RCVV	compressor variable-geometry parameter, deg
SF_x	scale factor on variable x, appropriate units
T	total temperature, K (°R)
T/T	temperature ratio
$\Delta T/T$	temperature rise parameter
t	time, sec
t'	scaled time, sec
u	internal energy, J/kg (Btu/lbm)
V	volume, cm^3 (in^3)
v	velocity, cm/sec (in/sec)
W	stored mass, kg (lbm)
$\dot{w}$	mass flow rate, kg/sec (lbm/sec)
$\dot{w}_c$	corrected mass flow rate, kg/sec (lbm/sec)
$\dot{w}_p$	turbine flow parameter, kg K cm^2/N rpm sec (lbm °R in^2/lbf rpm sec)
X	scaled variable, computer units
(X, Y)	map inputs, appropriate units
β	temperature interpolation constant
γ	specific heat ratio
δ	ratio of total pressure to sea-level pressure
η	efficiency ($0 \leq \eta \leq 1.0$)

θ	ratio of total temperature to standard-day temperature

Subscripts (Note that subscript may be combined, e.g., $\dot{w}_{F,4}$):

A	air
AB	augmentor
am	ambient
B	main combustor
BL	bleed
BLHT	high-pressure-turbine cooling bleed
BLLT	low-pressure-turbine cooling bleed
BLOV	overboard bleed
C	compressor
cr	critical flow
des	design point
E	nozzle exit plane
es	expelled nozzle shock
F	fuel
FAN	fan
H	high-pressure spool
HT	high-pressure turbine
I	inlet
ID	fan hub region
i	initial condition
id	ideal
in	into volume
j	station (fig. 3), $j=0$, 2, 2.1, 2.2, 3, 4, 4.1, 5, 6, 7, 8, 9, 13, 16
j'	entrance to volume at station j, $j=3$, 7, 13
k	index on flows into volume
L	low-pressure spool
LT	low-pressure turbine
M	map
m	index on flows out of volume
N	nozzle
n	net
OD	fan tip region
out	out of volume
x	upstream side of shock
y	downstream side of shock

Superscripts:

$()^*$	sonic flow condition
$(\dot{\ })$	time derivative

Appendix B

Derivation of Dynamic Energy Equation

For transient flow through a control volume V, the first law of thermodynamics can be written in the following dynamic form (ref. 18):

$$\frac{d}{dt(Wu)} = \sum_k \dot{w}_{\text{in},k} h_{\text{in},k} - \sum_m \dot{w}_{\text{out},m} h_{\text{out},m} + H_{\text{in}} - \frac{P}{J}\frac{dV}{dt} \qquad \text{(B1)}$$

All symbols are defined in appendix A. Equation (B1) assumes k sources of flow into the control volume and m flows discharging from the control volume. Heat H_{in} is being added to the fluid in the volume and work may be done by the fluid. Taking the derivative of the product Wu and noting that

$$\frac{du}{dt} = c_v \frac{dT}{dt} \qquad \text{(B2)}$$

$$\frac{dW}{dt} = \sum_k \dot{w}_{\text{in},k} - \sum_m \dot{w}_{\text{out},m} \qquad \text{(B3)}$$

yield the following equation:

$$Wc_v \frac{dT}{dt} = \sum_k \dot{w}_{\text{in},k} h_{\text{in},k} - \sum_m \dot{w}_{\text{out},m} h_{\text{out},m} - u\left(\sum_k \dot{w}_{\text{in},k} - \sum_m \dot{w}_{\text{out},m}\right) \qquad \text{(B4)}$$

If we assume perfect mixing within the control volume and a small temperature gradient across the control volume, we can substitute h (the average enthalpy in the volume, corresponding to temperature T) for each of the $h_{\text{out},m}$. Doing this and noting that

$$u = h - \frac{RT}{J} \qquad \text{(B5)}$$

yield the following equation:

$$\frac{dT}{dt} = \left[\sum_k \dot{w}_{\text{in},k} h_{\text{in},k} - h \sum_k \dot{w}_{\text{in},k} + \frac{RT}{J} \times \left(\sum_k \dot{w}_{\text{in},k} - \sum_m \dot{w}_{\text{out},m}\right) + H_{\text{in}} - \frac{P}{J}\frac{dV}{dt}\right] \Bigg/ Wc_v \qquad \text{(B6)}$$

Finally we can substitute $\gamma - 1$ for R/Jc_v to obtain the following form of the dynamic energy equation:

$$\frac{dT}{dt} = \left[\sum_k \dot{w}_{\text{in},k}(h_{\text{in},k} - h) + H_{\text{in}} - \frac{P}{J}\frac{dV}{dt}\right] \Bigg/ Wc_v + (\gamma - 1)T\left(\sum_k \dot{w}_{\text{in},k} - \sum_m \dot{w}_{\text{out},m}\right) \qquad \text{(B7)}$$

Appendix C

Target Analog Program

Scaled Analog Equations

$$\frac{W_{13}}{SF_{W_{13}}} = GAIN(1) \int_0^{t'} PVAL(3)DW13\, dt' + PVAL(1) \tag{C1}$$

$$\frac{T_{13}}{SF_{T_{13}}} = GAIN(1) \int_0^{t'} \frac{PVAL(2)DTQW13}{W_{13}/SF_{W_{13}}} dt' + PVAL(4) \tag{C2}$$

$$\frac{P_{13}}{SF_{P_{13}}} = 10\, PVAL(5) \frac{W_{13}}{SF_{W_{13}}} \frac{T_{13}}{SF_{T_{13}}} \tag{C3}$$

$$\frac{W_3}{SF_{W_3}} = GAIN(2) \int_0^{t'} PVAL(8)DW3\, dt' + PVAL(6) \tag{C4}$$

$$\frac{T_3}{SF_{T_3}} = GAIN(2) \int_0^{t'} \frac{PVAL(7)DTQW3}{W_3/SF_{W_3}} dt' + PVAL(9) \tag{C5}$$

$$\frac{P_3}{SF_{P_3}} = 10\, PVAL(10) \frac{W_3}{SF_{W_3}} \frac{T_3}{SF_{T_3}} \tag{C6}$$

$$\frac{W_4}{SF_{W_4}} = GAIN(3) \int_0^{t'} PVAL(13)DW4\, dt' + PVAL(11) \tag{C7}$$

$$\frac{T_4}{SF_{T_4}} = GAIN(3) \int_0^{t'} \frac{PVAL(12)DTQW4}{W_4/SF_{W_4}} dt' + PVAL(14) \tag{C8}$$

$$\frac{P_4}{SF_{P_4}} = 10\, PVAL(15) \frac{W_4}{SF_{W_4}} \frac{T_4}{SF_{T_4}} \tag{C9}$$

$$\frac{W_{4.1}}{SF_{W_{4.1}}} = GAIN(4) \int_0^{t'} PVAL(18)DW41\, dt' + PVAL(16) \tag{C10}$$

$$\frac{T_{4.1}}{SF_{T_{4.1}}} = GAIN(4) \int_0^{t'} \frac{PVAL(17)DTQW41}{W_{4.1}/SF_{W_{4.1}}} dt' + PVAL(19) \tag{C11}$$

$$\frac{P_{4.1}}{SF_{P_{4.1}}} = 10\, PVAL(20) \frac{W_{4.1}}{SF_{W_{4.1}}} \frac{T_{4.1}}{SF_{T_{4.1}}} \tag{C12}$$

$$\frac{W_6}{SF_{W_6}} = GAIN(5) \int_0^{t'} PVAL(23)DW6\, dt' + PVAL(21) \tag{C13}$$

$$\frac{T_6}{SF_{T_6}} = GAIN(5) \int_0^{t'} \frac{PVAL(22)DTQW6}{W_6/SF_{W_6}} dt' + PVAL(24) \tag{C14}$$

$$\frac{P_6}{SF_{P_6}} = 10\, PVAL(25) \frac{W_6}{SF_{W_6}} \frac{T_6}{SF_{T_6}} \tag{C15}$$

$$\frac{W_7}{SF_{W_7}} = GAIN(6) \int_0^{t'} PVAL(28)DW7\, dt' + PVAL(26) \tag{C16}$$

$$\frac{T_7}{SF_{T_7}} = GAIN(6) \int_0^{t'} \frac{PVAL(27)DTQW7}{W_7/SF_{W_7}}\, dt' + PVAL(29) \tag{C17}$$

$$\frac{P_7}{SF_{P_7}} = 10\, PVAL(30) \frac{W_7}{SF_{W_7}} \frac{T_7}{SF_{T_7}} \tag{C18}$$

$$\frac{N_H}{SF_{N_H}} = GAIN(7) \int_0^{t'} PVAL(31)DXNH\, dt' + PVAL(32) \tag{C19}$$

$$\frac{N_L}{SF_{N_L}} = GAIN(8) \int_0^{t'} PVAL(33)DXNL\, dt' + PVAL(34) \tag{C20}$$

$$\left(\frac{\dot{w}_{13}}{SF_{\dot{w}_{13}}}\right) = GAIN(9) \int_0^{t'} \left[PVAL(36)\left(\frac{P_{13}}{SF_{P_{13}}}\right) - PVAL(37)\left(\frac{P_6}{SF_{P_6}}\right) - PVAL(38)\left(\frac{\dot{w}_{13}}{SF_{\dot{w}_{13}}}\right)^2 \Big/ \left(\frac{W_{13}}{SF_{W_{13}}}\right)\right] dt' + PVAL(35) \tag{C21}$$

$$\left(\frac{\dot{w}_{6}}{SF_{\dot{w}_{6}}}\right) = GAIN(10) \int_0^{t'} \left[PVAL(40)\left(\frac{P_{6}}{SF_{P_{6}}}\right) - PVAL(41)\left(\frac{P_7}{SF_{P_7}}\right) - PVAL(42)\left(\frac{\dot{w}_{6}}{SF_{\dot{w}_{6}}}\right)^2 \Big/ \left(\frac{W_{6}}{SF_{W_{6}}}\right)\right] dt' + PVAL(39) \tag{C22}$$

Potentiometer Settings

$$PVAL(1) = \frac{W_{13,des}}{SF_{W_{13}}}$$

$$PVAL(2) = PVAL(3) = \frac{SF_{\dot{w}_2}}{SF_{W_{13}} SF_t GAIN(1)}$$

$$PVAL(4) = \frac{T_{13,des}}{SF_{T_{13}}}$$

$$PVAL(5) = \frac{R_A SF_{W_{13}} SF_{T_{13}}}{10 V_{13} SF_{P_{13}}}$$

$$PVAL(6) = \frac{W_{3,des}}{SF_{W_3}}$$

$$PVAL(7) = PVAL(8) = \frac{SF_{\dot{w}_{2.2}}}{SF_{W_3} SF_t GAIN(2)}$$

$$PVAL(9) = \frac{T_{3,des}}{SF_{T_3}}$$

$$PVAL(10) = \frac{R_A SF_{W_3} SF_{T_3}}{10 V_3 SF_{P_3}}$$

$$PVAL(11) = \frac{W_{4,des}}{SF_{W_4}}$$

$$PVAL(12) = PVAL(13) = \frac{SF_{\dot{w}_4}}{SF_{W_4} SF_t GAIN(3)}$$

$$PVAL(14) = \frac{T_{4,des}}{SF_{T_4}}$$

$$PVAL(15) = \frac{R_A SF_{W_4} SF_{T_4}}{10 V_4 SF_{P_4}}$$

$$PVAL(16) = \frac{W_{4.1,des}}{SF_{W_{4.1}}}$$

$$PVAL(17) = PVAL(18) = \frac{SF_{\dot{w}_{4.1}}}{SF_{W_{4.1}} SF_t GAIN(4)}$$

$$\mathrm{PVAL}(19) = \frac{T_{4.1,\mathrm{des}}}{\mathrm{SF}_{T_{4.1}}}$$

$$\mathrm{PVAL}(20) = \frac{R_A \mathrm{SF}_{W_{4.1}} \mathrm{SF}_{T_{4.1}}}{10 V_{4.1} \mathrm{SF}_{P_{4.1}}}$$

$$\mathrm{PVAL}(21) = \frac{W_{6,\mathrm{des}}}{\mathrm{SF}_{W_6}}$$

$$\mathrm{PVAL}(22) = \mathrm{PVAL}(23) = \frac{\mathrm{SF}_{\dot{w}_6}}{\mathrm{SF}_{W_6} \mathrm{SF}_t \mathrm{GAIN}(5)}$$

$$\mathrm{PVAL}(24) = \frac{T_{6,\mathrm{des}}}{\mathrm{SF}_{T_6}}$$

$$\mathrm{PVAL}(25) = \frac{R_A \mathrm{SF}_{W_6} \mathrm{SF}_{T_6}}{10 V_6 \mathrm{SF}_{P_6}}$$

$$\mathrm{PVAL}(26) = \frac{W_{7,\mathrm{des}}}{\mathrm{SF}_{W_7}}$$

$$\mathrm{PVAL}(27) = \mathrm{PVAL}(28) = \frac{\mathrm{SF}_{\dot{w}_7}}{\mathrm{SF}_{W_7} \mathrm{SF}_t \mathrm{GAIN}(6)}$$

$$\mathrm{PVAL}(29) = \frac{T_{7,\mathrm{des}}}{\mathrm{SF}_{T_7}}$$

$$\mathrm{PVAL}(30) = \frac{R_A \mathrm{SF}_{W_7} \mathrm{SF}_{T_7}}{10 V_7 \mathrm{SF}_{P_7}}$$

$$\mathrm{PVAL}(31) = \frac{900 J\, \mathrm{SF}_{\Delta h_{\mathrm{HT}}} \mathrm{SF}_{\dot{w}_4}}{\pi^2 I_H \mathrm{SF}^2_{N_H} \mathrm{SF}_t \mathrm{GAIN}(7)}$$

$$\mathrm{PVAL}(32) = \frac{N_{H,\mathrm{des}}}{\mathrm{SF}_{N_H}}$$

$$\mathrm{PVAL}(33) = \frac{900 J\, \mathrm{SF}_{\Delta h_{\mathrm{LT}}} \mathrm{SF}_{\dot{w}_{4.1}}}{\pi^2 I_L \mathrm{SF}^2_{N_L} \mathrm{SF}_t \mathrm{GAIN}(8)}$$

$$\mathrm{PVAL}(34) = \frac{N_{L,\mathrm{des}}}{\mathrm{SF}_{N_L}}$$

$$\mathrm{PVAL}(35) = \frac{\dot{w}_{13,\mathrm{des}}}{\mathrm{SF}_{\dot{w}_{13}}}$$

$$\mathrm{PVAL}(36) = \frac{(A/l)_D g_c \mathrm{SF}_{P_{13}}}{\mathrm{SF}_{\dot{w}_{13}} \mathrm{SF}_t \mathrm{GAIN}(9)}$$

$$\mathrm{PVAL}(37) = \frac{(A/l)_D g_c \mathrm{SF}_{P_6}}{\mathrm{SF}_{\dot{w}_{13}} \mathrm{SF}_t \mathrm{GAIN}(9)}$$

$$\mathrm{PVAL}(38) = \frac{(A/l)_D g_c P_{13,\mathrm{des}} (P_{13,\mathrm{des}} - P_{6,\mathrm{des}}) \mathrm{SF}_{\dot{w}_{13}} V_{13}}{\dot{w}^2_{13,\mathrm{des}} R_A T_{13,\mathrm{des}} \mathrm{SF}_{W_{13}} \mathrm{SF}_t \mathrm{GAIN}(9)}$$

$$\mathrm{PVAL}(39) = \frac{\dot{w}_{6,\mathrm{des}}}{\mathrm{SF}_{\dot{w}_6}}$$

$$\mathrm{PVAL}(40) = \frac{(A/l)_{\mathrm{AB}} g_c \mathrm{SF}_{P_6}}{\mathrm{SF}_{\dot{w}_6} \mathrm{SF}_t \mathrm{GAIN}(10)}$$

$$\mathrm{PVAL}(41) = \frac{(A/l)_{\mathrm{AB}} g_c \mathrm{SF}_{P_7}}{\mathrm{SF}_{\dot{w}_6} \mathrm{SF}_t \mathrm{GAIN}(10)}$$

$$\mathrm{PVAL}(42) = \frac{(A/l)_{\mathrm{AB}} g_c P_{6,\mathrm{des}} (P_{6,\mathrm{des}} - P_{7,\mathrm{des}}) \mathrm{SF}_{\dot{w}_6} V_6}{\dot{w}^2_{6,\mathrm{des}} R_A T_{6,\mathrm{des}} \mathrm{SF}_{W_6} \mathrm{SF}_t \mathrm{GAIN}(10)}$$

$$\mathrm{PVAL}(43) = \frac{\dot{w}_{F,4}}{\mathrm{SF}_{\dot{w}_{F,4}}}$$

$$\mathrm{PVAL}(44) = \frac{\dot{w}_{F,7}}{\mathrm{SF}_{\dot{w}_{F,7}}}$$

$$\mathrm{PVAL}(45) = \frac{A_8}{\mathrm{SF}_{A_8}}$$

$$\mathrm{PVAL}(46) = \frac{A_E}{\mathrm{SF}_{A_E}}$$

$$\mathrm{PVAL}(47) = \frac{\mathrm{ALT}}{AF_a}$$

$$\mathrm{PVAL}(48) = \frac{M_0}{\mathrm{SF}_{M_0}}$$

$$\mathrm{PVAL}(49) = \frac{-(\mathrm{CIVV} - B_{\mathrm{CIVV}})}{\mathrm{SF}_{\mathrm{CIVV}}}$$

$$\mathrm{PVAL}(50) = \frac{-(\mathrm{RCVV} - B_{\mathrm{RCVV}})}{\mathrm{SF}_{\mathrm{RCVV}}}$$

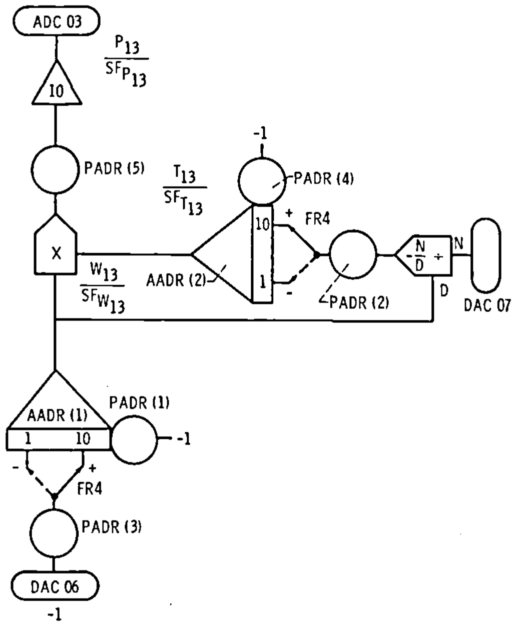
$$\mathrm{PVAL}(51) = \frac{T_{\mathrm{am}}}{\mathrm{SF}_{T_{\mathrm{am}}}}$$

$$\mathrm{PVAL}(52) = \frac{P_{2,\mathrm{des}}}{P_{13,\mathrm{des}} \mathrm{SF}_{(P_{13}/P_2)}}$$

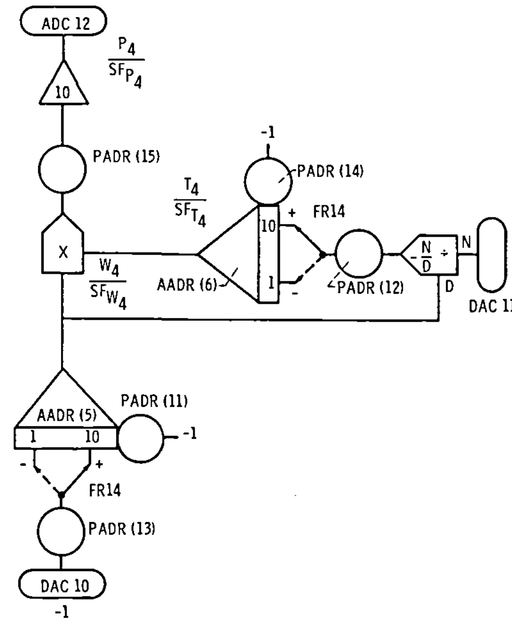
$$\mathrm{PVAL}(53) = \frac{P_{2.2,\mathrm{des}}}{P_{3,\mathrm{des}} \mathrm{SF}_{(P_3/P_{2.2})}}$$

Analog Patching Diagrams

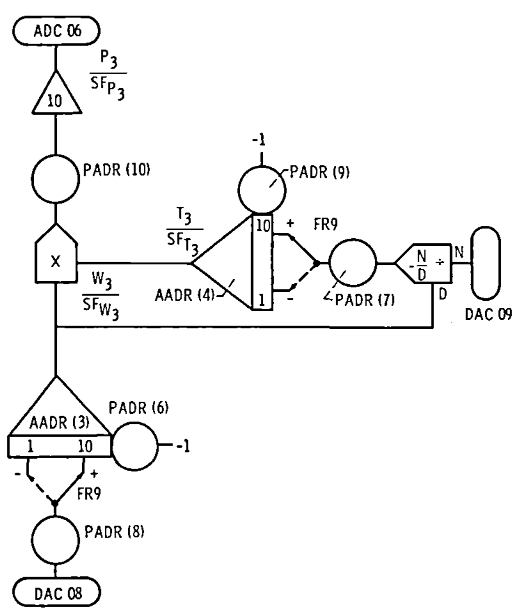
Volume V_{13}



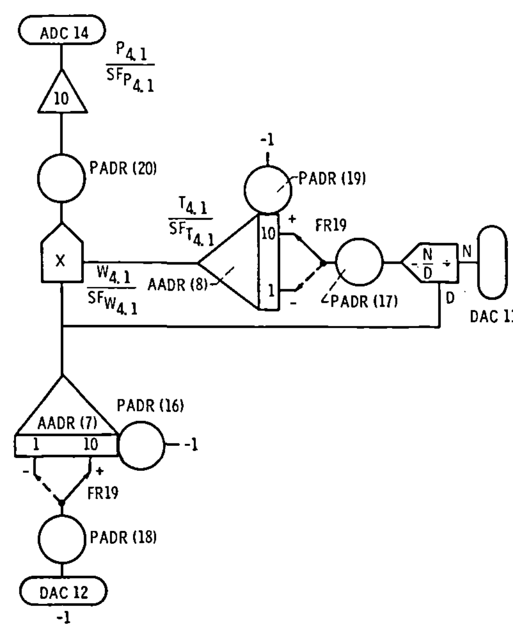
Volume V_4



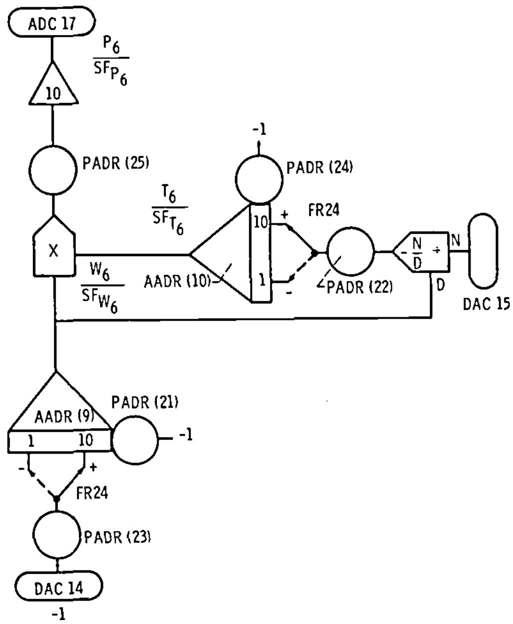
Volume V_3



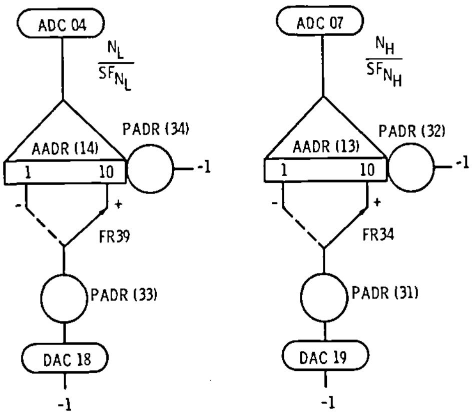
Volume 4.1



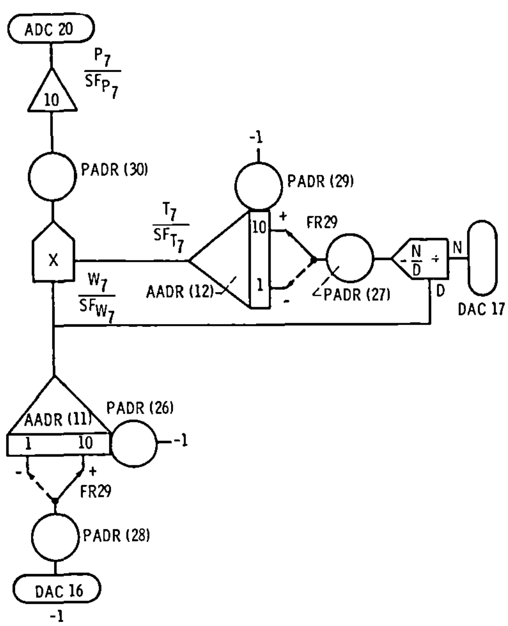
Volume V_6



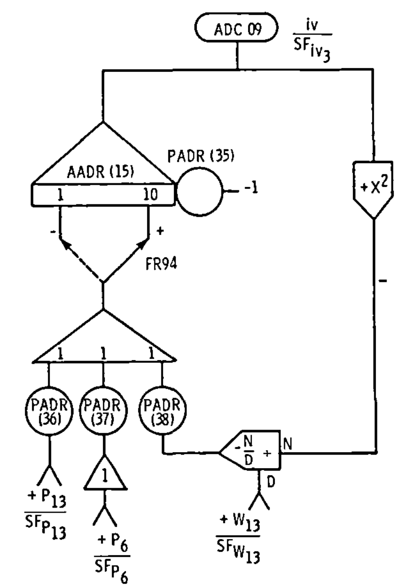
Rotors



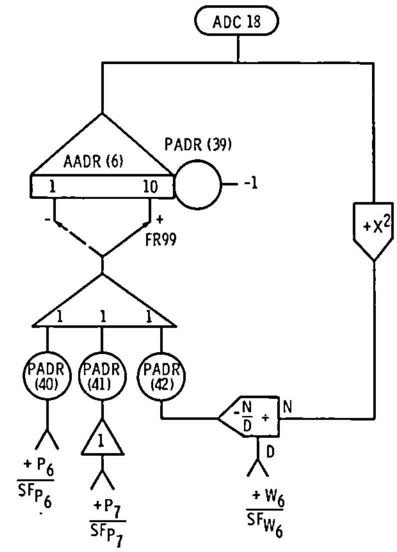
Volume V_7



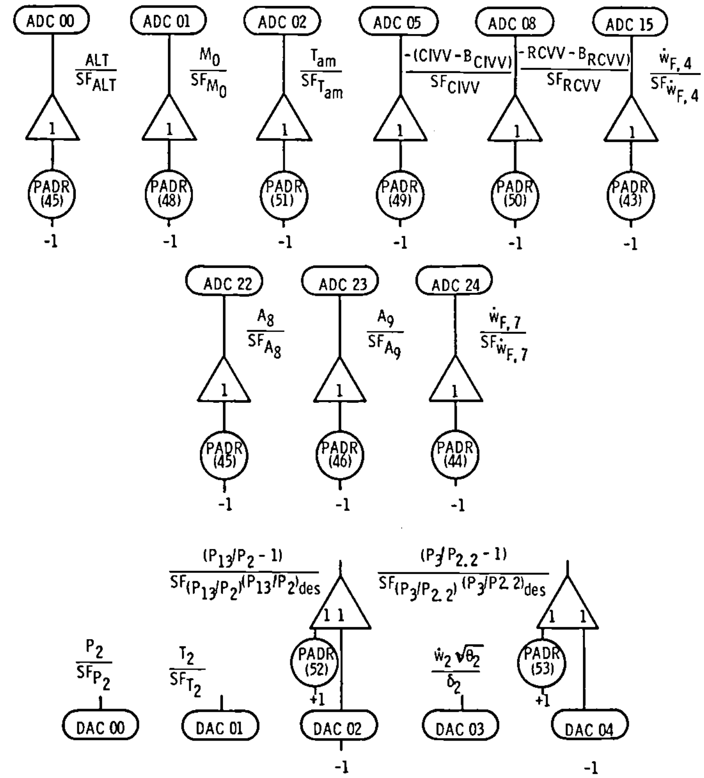
Bypass Duct



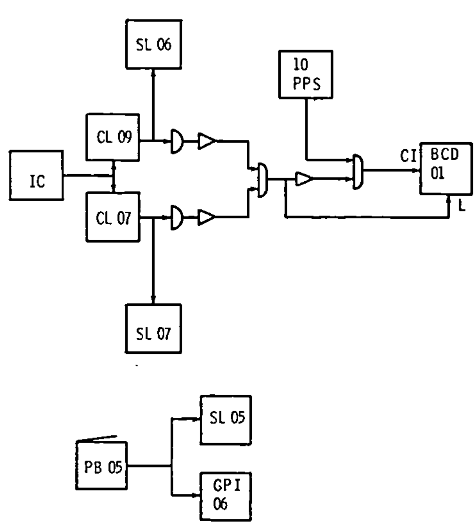
Augmentor Duct



Input and Output

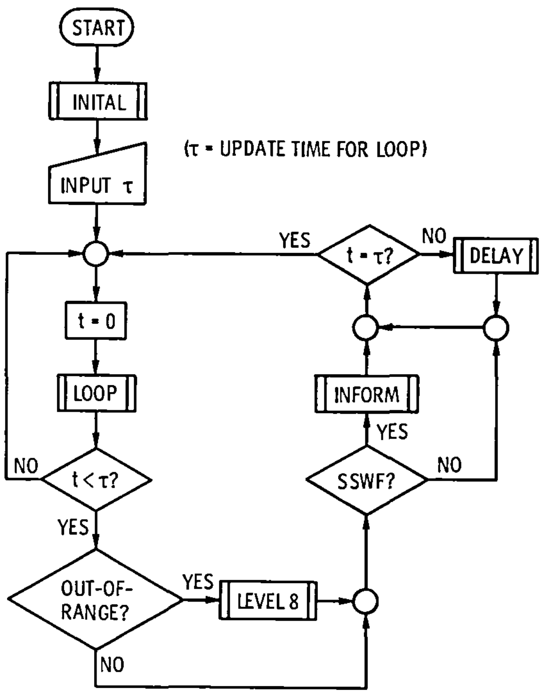


Map and Function Out-of-Range Logic



Appendix D

Target Digital Program



Flowchart of target digital program EXEC1

Source Listings

```
      SUBROUTINE BTASK
      DIMENSION X(45),SF(45),Y(45),ERR(45)
      LOGICAL SENSW
      SCALED FRACTION P0,P2,P13,P22,P3,P4,P41,P5,P6,P7,TAM,T2,T13,T22,
     1 T3,T4,T41,T6,T7,WA2,WA13,WA22,WA3,WG4,WG41,WG6,WG7,DH4,DH41,ETAB,
     2 ETAAB,FNET,XNL,XNH,WF4,WF7,A8,AE,ALT,XMN,CDN,CVN,CIVV,RCVV,FNM,
     3 J1(29),J2(7),J3(4),J4(26),J5(20),J6(14),J7(4),J8(8),J9(8),J10,
     4 J11(14),J12(9),J13(18)
      COMMON/VAR/ALT,XMN,TAM,P13,XNL,CIVV,P3,XNH,RCVV,WA13,T13,T3,P4,T4,
     1 P41,WF4,T41,P6,WG6,T6,P7,T7,A8,AE,WF7,P2,T2,J1,WA2,P22,J2,T22,J3,
     2 WA22,J4,WA3,J5,WG4,J6,ETAB,P5,J7,WG41,DH4,J8,DH41,J9,P0,CDN,CVN,
     3 J10,FNM,WG7,J11,ETAAB,J12,FNET,J13
      IF(.NOT.SENSW(5)) RETURN
      READ(4,400)(X(I),I=1,45)
      READ(4,400)(SF(I),I=1,45)
400   FORMAT(5F12.5)
      Y(1)=P0
      Y(2)=P2
```

```
      Y(3)=P13
      Y(4)=P22
      Y(5)=P3
      Y(6)=P4
      Y(7)=P41
      Y(8)=P5
      Y(9)=P6
      Y(10)=P7
      Y(11)=TAM
      Y(12)=T2
      Y(13)=T13
      Y(14)=T22
      Y(15)=T3
      Y(16)=T4
      Y(17)=T41
      Y(18)=T6
      Y(19)=T7
      Y(20)=WA2
      Y(21)=WA13
      Y(22)=WA22
      Y(23)=WA3
      Y(24)=WG4
      Y(25)=WG41
      Y(26)=WG6
      Y(27)=WG7
      Y(28)=DH4
      Y(29)=DH41
      Y(30)=ETAB
      Y(31)=ETAAB
      Y(32)=FNET
      Y(33)=XNL
      Y(34)=XNH
      Y(35)=WF4
      Y(36)=WF7
      Y(37)=A8
      Y(38)=AE
      Y(39)=ALT
      Y(40)=XMN
      Y(41)=CDN
      Y(42)=CVN
      Y(43)=-CIVV
      Y(44)=-RCVV
      Y(45)=FNM
      DO 10 I=1,45
   10 Y(I)=Y(I)*SF(I)
      Y(44)=Y(44)+4.0
      DO 20 I=1,45
      IF(X(I).EQ.0.) X(I)=.00001
   20 ERR(I)=Y(I)/X(I)
      TYPE 410
  410 FORMAT(3X,25HSTEADY-STATE ERROR RATIOS/)
      TYPE 420,(ERR(I),I=1,45)
  420 FORMAT(5F10.5)
      RETURN
      END
```

```
      SUBROUTINE DATAIN(N,F)
      DIMENSION N(1),VALS(25),XVALS(25),ZSC(4)
      DIMENSION ISC(4),IX(4),IY(4),IZ1(4),IZ2(4),IZ3(4),IZ4(4)
      SCALED FRACTION F(1),TEST
      K=1
      JF=0
    5 READ(6,400)M,NCV,NPT,NFCT,NCOM
  400 FORMAT(5I3)
      IF(M.EQ.0)RETURN
      INC=NCV*NPT
      N(K)=M
      N(K+1)=1
      N(K+2)=1
      N(K+3)=NPT
      N(K+4)=NCV
      K=K+5
      READ(6,410)(ISC(I),I=1,28)
  410 FORMAT(28A2)
      READ(6,ISC)XSC,YSC,(ZSC(1),I=1,NFCT)
      READ(6,IY)(VALS(I),I=1,NCV)
      JS=JF+1
      JF=JF+NCV
      I=1
      DO 10 J=JS,JF
      F(J)=VALS(I)/YSC
      CHECK CONSECUTIVE VALUES OF Y
      IF(J.NE.JS) TEST=F(J)-F(J-1)
   10 I=I+1
      JS=JF+1
      JF=JF+NPT
      DO 90 L=1,NCV
      IF(NCOM.EQ.0) GO TO 14
      IF(NCOM.EQ.32767) GO TO 16
      NCOM=32767
   14 READ(6,IX)(XVALS(1),I=1,NPT)
   16 I=1
      DO 20 J=JS,JF
      F(J)=XVALS(I)/XSC
      CHECK CONSECUTIVE VALUES OF X
      IF(J.NE.JS) TEST=F(J)-F(J-1)
      IF(L.EQ.1) GO TO 20
      JMNPT=J-NPT
      TEST=F(J)-F(JMNPT)
   20 I=I+1
      DO 80 NF=1,NFCT
      GO TO (30,40,50,60),NF
   30 READ(6,IZ1)(VALS(I),I=1,NPT)
      GO TO 70
   40 READ(6,IZ2)(VALS(I),I=1,NPT)
      GO TO 70
   50 READ(6,IZ3)(VALS(I),I=1,NPT)
      GO TO 70
      JF=JF+INC
      I=1
      DO 80 J=JS,JF
```

```
      F(J)=VALS(I)/ZSC(NF)
C ....CHECK CONSECUTIVE VALUES OF Z
      IF(J.NE.JS) TEST=F(J)-F(J-1)
      IF(L.EQ.1) GO TO 80
      JMNPT=J-NPT
      TEST=F(J)-F(JMNPT)
   80 I=I+1
      IF(L.EQ.NCV)GO TO 5
      JS=JS+NPT-INC*NFCT
   90 JF=JS+NPT-1
      END
```

```
      SUBROUTINE FLCOND(HT,XM0,TAS,PT,TT,PS,TS)
      DIMENSION N1(3),N2(3),N3(3)
      SCALED FRACTION X1(11),Y1(11),X2(10),Y2(10),X3(7),Y3(7),FUN1
      SCALED FRACTION HT,XM0,PT,TT,PS,TS,TTQS,PTQS,ETAI,TAS,TR35,XMP,XPP
      COMMON/INL/INLET
      DATA N1/2,1,11/,N2/3,1,10/,N3/4,1,7/
      DATA X1  /.00000S,.05000S,.10000S,.17500S,.27500S,.35000S,.42500S,
     1  .52500S,.62500S,.75000S,.99999S/
      DATA Y1  /.73480S,.63460S,.54585S,.43180S,.31062S,.23922S,.18176S,
     1  .12404S,.08458S,.05244S,.02018S/
      DATA X2  /.00000S,.11111S,.22222S,.33333S,.44444S,.55555S,.66666S,
     1  .77777S,.88888S,.99999S/
      DATA Y2  /.00000S,.00046S,.00517S,.02138S,.05853S,.12780S,.24192S,
     1  .41493S,.66214S,.99999S/
      DATA X3  /.00000S,.11111S,.22222S,.33333S,.44444S,.55555S,.66666S/
      DATA Y3  /.00000S,.05149S,.13126S,.27692S,.33461S,.45224S,.57846S/
      TS= TAS-.2848S*HT
      IF(TS .LT. .38999S)  TS= .38999S
      IF(HT .LT. .00000S)  HT= .00000S
      PS=FUN1(N1,X1,HT)
  100 TTQS= .33333S + .60000S*(XM0*XM0)
      TR35=FUN1(N2,X2,TTQS)
      PTQS= TR35/.85533S
      ETAI= .99999S
      IF(INLET.NE.1) GO TO 110
      IF(XM0 .LE. .33333S) GO TO 110
      XMP=XM0-.33333S
      XPP=FUN1(N3,X3,XMP)
      ETAI= .99999S -.33050S*XPP
  110 TT= (TTQS*TS)/.33333S
      PT= (ETAI*PTQS*PS)/.05S
      RETURN
      END
```

```
      SUBROUTINE FOOR(N,XIN)
      SCALED FRACTION XIN,X
      COMMON/FUNVAL/MN,X
      LOGICAL CL7
      IF(N.EQ.0) GO TO 100
      CALL QRSLL(7,CL7,IER)
```

```
      CALL QRSLL(7,CL7,IER)
      IF(CL7) RETURN
      CALL QWCLL(7,.TRUE.,IER)
      MN=N
      X=XIN
      RETURN
  100 WRITE(1,400)MN,X
  400 FORMAT(/12HFUNCTION NO.,I3,19H INPUT OUT OF RANGE/6HXIN = ,S7/)
      CALL QWCLL(7,.FALSE.,IER)
      RETURN
      END


      SCALED FRACTION FUNCTION FUN1(N,F,XIN)
      SCALED FRACTION F(1),XIN,X1,X2,XFRAC,FUN1
      DIMENSION N(1)
      IF(N(1).EQ.0) GO TO 200
      I=N(2)
      NXP=N(3)
  100 X1=XIN-F(I)
      IF(X1.GT..0S) GO TO 110
      IF(X1.EQ..0S) GO TO 120
      IF(I.LE.1) GO TO 140
      I=I-1
      GO TO 100
  110 X2=XIN-F(I+1)
      IF(X2.LT..0S) GO TO 180
      IF(X2.EQ..0S) GO TO 130
      I=I+1
      IF(I.GE.NXP) GO TO 150
      X1=X2
      GO TO 110
  120 XFRAC=.0S
      GO TO 190
  130 XFRAC=.99999S
      GO TO 190
  140 XFRAC=.0S
      GO TO 160
  150 XFRAC=.99999S
      I=I-1
  160 CALL FOOR(N,XIN)
      GO TO 190
  180 XFRAC=X1/(X1-X2)
  190 N(2)=I
  200 I=I+NXP
      FUN1=F(I)+XFRAC*(F(I+1)-F(I))
      RETURN
      END


      SCALED FRACTION FUNCTION FUN1L(F)
      SCALED FRACTION FUN1,FUN1L,F,XIN
C.....XIN IS DUMMY ARGUMENT AND DOES NOT AFFECT RESULTS
      N=0
```

```
      FUN1L=FUN1(N,F,XIN)
      RETURN
      END


      SUBROUTINE INITAL
C.....MAP ARRAYS
      DIMENSION N1(5),N2(5),N3(5),N4(5),N5(5),N6(5)
      SCALED FRACTION F1(322),F2(322),F3(854),F4(518),F5(224),F6(224)
C.....OTHER VARIABLES
      SCALED FRACTION DC(125),DACI(24),PVAL(80),PSET,ERR
      DIMENSION IAADR(16),IG(10),PADR(80)
      DIMENSION EXTRA(74)
      COMMON/MAPS/N1,N2,N3,N4,N5,N6,F1,F2,F3,F4,F5,F6
      COMMON/INL/INLET/IDAC/DACI/COEF/DC,EXTRA/BLEED/KBH,KBL,KBV
      DATA IBELL/"103640/
C.....INITIALIZE ANALOG CONSOLE
      CALL QSC(0,IER)
      CALL QSC(1,IER)
      CALL QSCLR(IER)
      CALL QSRUN(IER)
      TYPE 300,(IBELL,I=1,10)
  300 FORMAT(10A1)
      TYPE 400
  400 FORMAT(/3X,38HTURN ON CARD READER. THEN TYPE RETURN./)
      ACCEPT 500,J
  500 FORMAT(I3)
C.....READ AND SCALE MAP DATA
      CALL DATAIN(N1,F1)
C.....READ DIGITAL COEFFICIENTS
      READ(6,510)(DC(I),I=1,125)
  510 FORMAT((3X,5(S7,2X)))
C.....READ INLET AND BLEED INTEGERS
      READ(6,520)INLET,KBH,KBL,KBV
  520 FORMAT((1X,4(I7,2X)))
      WRITE(6,310)
  310 FORMAT(1H1)
C.....READ INTEGRATOR GAIN DATA AND PRINT INSTRUCTIONS
      DO 20 K=1,6
      I=2*K-1
      J=2*K
      READ(6,530)IAADR(I),IAADR(J),IG(K)
  530 FORMAT(3I4)
      IF(IG(K).GE.3)WRITE(6,540)IAADR(I),IAADR(J)
  540 FORMAT(3X,12HINTEGRATORS ,I3,5H AND ,I3,33H MS INPUTS SHOULD BE PA
     1TCHED HIGH)
      IF((IG(K).EQ.0).OR.(IG(K).EQ.3))WRITE(6,550)IAADR(I),IAADR(J)
  550 FORMAT(3X,12HINTEGRATORS ,I3,5H AND ,I3,31H F INPUTS SHOULD BE PAT
     1CHED LOW)
      IFRADR=IAADR(I)+4
      IF((IG(K).EQ.2).OR.(IG(K).EQ.5))WRITE(6,560)IFRADR
  560 FORMAT(3X,15HFUNCTION RELAY ,I3,19H SHOULD BE SET HIGH)
   20 CONTINUE
      DO 30 K=7,10
      I=K+6
```

```
      READ(6,530)IAADR(I),IG(K)
      IF(IG(K).GE.3)WRITE(6,570)IAADR(I)
  570 FORMAT(3X,11HINTEGRATOR ,I3,32H MS INPUT SHOULD BE PATCHED HIGH)
      IF((IG(K).EQ.0).OR.(IG(K).EQ.3))WRITE(6,580)IAADR(I)
  580 FORMAT(3X,11HINTEGRATOR ,I3,30H F INPUT SHOULD BE PATCHED LOW)
      IFRADR=IAADR(I)+4
      IF((IG(K).EQ.2).OR.(IG(K).EQ.5))WRITE(6,560)IFRADR
   30 CONTINUE
      WRITE(6,590)
  590 FORMAT(3X,31HRESET ALL OTHER FUNCTION RELAYS//)
C.....READ DAM DATA AND INITIALIZE DAMS
      WRITE(6,600)
  600 FORMAT(3X,10HDAM VALUES/)
      READ(6,510)(DACI(I),I=1,24)
      CALL QWBDAS(DACI,0,24,IER)
      CALL QSTDA
      WRITE(6,510)(DACI(I),I=1,24)
      TYPE 300,(IBELL,I=1,10)
      TYPE 405
  405 FORMAT(3X,66HBE SURE ALL FUNCTION RELAYS ARE SET PROPERLY BEFORE T
     1YPING RETURN./)
      ACCEPT 500,J
C.....READ POT DATA, SET AND CHECK POTS, AND PRINT RESULTS
   35 READ(6,610)N
  610 FORMAT(I4)
      IF(N.LE.80) GO TO 37
      TYPE 407
  407 FORMAT(3X,27HTOO MANY POTS, LIMIT AT 80.)
      CALL MONOUT
   37 WRITE(6,620)
  620 FORMAT(1H1,6X,4HADDR,4X,4HCOEF,4X,3HSET,4X,5HERROR/)
      DO 40 I=1,N
      READ(6,630)PADR(I),PVAL(I)
  630 FORMAT(1X,A4,1X,S5)
      CALL QWPS(PADR(I),PVAL(I),IER)
   40 CONTINUE
      CALL QSPC(IER)
      DO 50 I=1,N
      CALL QRAS(PADR(I),PSET,IER)
      ERR=SABS(PVAL(I)-PSET)
      IF(ERR.GE..00025) GO TO 45
      WRITE(6,640)PADR(I),PVAL(I),PSET
  640 FORMAT(6X,A4,3(2X,S6))
      GO TO 50
   45 WRITE(6,640)PADR(I),PVAL(I),PSET,ERR
   50 CONTINUE
   60 TYPE 300,(IBELL,I=1,10)
C.....INITIALIZE OUT OF RANGE ROUTINES
      TYPE 410
  410 FORMAT(/3X,39HIGNORE FOLLOWING OUT OF RANGE MESSAGES.)
      CALL MOOR(99,.0S,.0S)
      CALL MOOR(0,.0S,.0S)
      CALL FOOR(99,.0S)
      CALL FOOR(0,.0S)
      CALL QSIC(IER)
```

```
      TYPE 420
  420 FORMAT(/3X,53HSET SENSE SWITCH F TO ENTER INFORM, THEN TYPE RETURN
     1./)
      ACCEPT 500,J
      RETURN
      END


      SUBROUTINE LEVEL8
      LOGICAL PB5
C.....PULSE PB5 FOR MAP OR FUNCTION OUT OF RANGE MESSAGE
      CALL QRSLL(6,PB5,IER)
C.....DUMMY CALL TO MAP OUT OF RANGE ROUTINE
      IF(PB5) CALL MOOR(0,.0S,.0S)
      CALL QRSLL(7,PB5,IER)
C.....DUMMY CALL TO FUNCTION OUT OF RANGE ROUTINE
      IF(PB5) CALL FOOR(0,.0S)
      RETURN
      END


      SUBROUTINE LOOP
C.....ADC VARIABLES
      SCALED FRACTION ALT,XMN,TAM,P13,XNL,CIVV,P3,XNH,RCVV,WA13,T13,T3,
     1 P4,T4,P41,WF4,T41,P6,WG6,T6,P7,T7,A8,AE,WF7
C.....DAC VARIABLES
      SCALED FRACTION P2,T2,X3,WAR2,X4,WAR22,DW13,DTQW13,DW3,DTQW3,DW4,
     1 DTQW4,DW41,DTQW41,DW6,DTQW6,DW7,DTQW7,DXNL,DXNH,FNET
C.....MAP ARRAYS
      DIMENSION N1(5),N2(5),N3(5),N4(5),N5(5),N6(5)
      SCALED FRACTION F1(322),F2(322),F3(854),F4(518),F5(224),F6(224)
C.....FUNCTION ARRAYS
      DIMENSION N7(3)
      SCALED FRACTION X7(15),Y71(15),Y72(15)
C.....OTHER VARIABLES
      SCALED FRACTION DC(125),MAP,MAPL,FUN1,FUN1L,ALTM,XMNM,P2A,T2A,P0A
      SCALED FRACTION Y3,WAR2M,ETHOFM,P2202M,ETAIFM,FSHIFT,WA2,P22,PRIF
      SCALED FRACTION T2M,CP2,CV2,GM2,H2M,TRIFM1,T22,Y4,WAR22M,ETAHCM
      SCALED FRACTION CSHIFT,WA22,PROF,TROFM1,T13P,T13PM,CP13P,CV13P
      SCALED FRACTION GM13P,H13PM,H13P,T13M,CP13,CV13,GM13,H13M,H13
      SCALED FRACTION GM13M,T3M,CP3,CV3,GM3,H3M,FGM3,FGPT3,WBLHT,WALLT
      SCALED FRACTION WBLOV,WA3,H3,PRHC,TAVHC,CPHC,CVHC,GMHC,HHCM,TRHCM1
      SCALED FRACTION T3P,T3PM,CP3P,CV3P,GM3P,H3PM,H3P,GM3M,X5,Y5,WP4
      SCALED FRACTION HP4,WG4,TAVB,CPB,CVB,GMB,HBM,HB,FAR4M,T4M,CP4,CV4
      SCALED FRACTION GM4,H4M,H4,GM4M,ETAB,P5,X6,Y6,WP41,HP41,WG41,DH4
      SCALED FRACTION FAR41M,T41M,CP41,CV41,GM41,H41M,H41,GM41M,DH41,T6M
      SCALED FRACTION FAR6M,CP6,CV6,GM6,H6M,H6,GM6M,P0,CDN,CVN,WG7M,FNM
      SCALED FRACTION WG7,TAVAB,CPAB,CVAB,GMAB,HABM,HAB,T7M,FAR7M,CP7
      SCALED FRACTION CV7,GM7,H7M,H7,GM7M,ETAAB,H?,T22M,CP22,CV22,GM22
      SCALED FRACTION H22M,H22,P0QT7,T0A,UNUSED(18)
      DIMENSION EXTRA(74)
      COMMON/MAPS/N1,N2,N3,N4,N5,N6,F1,F2,F3,F4,F5,F6
      COMMON/INL/INLET/COEF/DC,EXTRA/BLEED/KBH,KBL,KBV
      COMMON/VAR/ALT,XMN,TAM,P13,XNL,CIVV,P3,XNH,RCVV,WA13,T13,T3,P4,T4,
```

```
     1 P41,WF4,T41,P6,WG6,T6,P7,T7,A8,AE,WF7,P2,T2,X3,WAR2,X4,WAR22,
     2 DW13,DTQW13,DW3,DTQW3,DW4,DTQW4,DW41,DTQW41,DW6,DTQW6,DW7,DTQW7,
     3 DXNL,DXNH,ALTM,XMNM,P2A,T2A,P0A,Y3,WAR2M,ETAOFM,P22Q2M,ETAIFM,
     4 FSHIFT,WA2,P22,PRIF,T2M,CP2,CV2,GM2,H2M,TRIFM1,T22,Y4,WAR22M,
     5 ETAHCM,CSHIFT,WA22,PROF,TROFM1,T13P,T13PM,CP13P,CV13P,GM13P,
     6 H13PM,H13P,T13M,CP13,CV13,GM13,H13M,H13,GM13M,T3M,CP3,CV3,GM3,
     7 H3M,FGM3,FGPT3,WBLHT,WBLLT,WBLOV,WA3,H3,PRHC,TAVHC,CPHC,CVHC,
     8 GMHC,HHCM,TRHCM1,T3P,T3PM,CP3P,CV3P,GM3P,H3PM,H3P,GM3M,X5,Y5,WP4,
     9 HP4,WG4,TAVB,CPB,CVB,GMB,HBM,HB,FAR4M,T4M,CP4,CV4,GM4,H4M,H4
      COMMON/VAR/GM4M,ETAB,P5,X6,Y6,WP41,HP41,WG41,DH4,FAR41M,T41M,CP41,
     1 CV41,GM41,H41M,H41,GM41M,DH41,T6M,FAR6M,CP6,CV6,GM6,H6M,H6,GM6M,
     2 P0,CDN,CVN,WG7M,FNM,WG7,TAVAB,CPAB,CVAB,GMAB,HABM,HAB,T7M,FAR7M,
     3 CP7,CV7,GM7,H7M,H7,GM7M,ETAAB,H2,T22M,CP22,CV22,GM22,H22M,H22,
     4 P0QT7,T0A,FNET,UNUSED
      DATA N7/1,1,15/
      DATA X7  /.05000S,.34732S,.40345S,.43196S,.45522S,.48029S,.51127S,
     1 .54541S,.57747S,.61735S,.66017S,.72709S,.84377S,.91296S,.95000S/
      DATA Y71 /.96610S,.96610S,.95366S,.95208S,.95019S,.95172S,.95369S,
     1 .95709S,.95519S,.94861S,.93186S,.88707S,.88481S,.90661S,.92250S/
      DATA Y72 /.97034S,.97034S,.97184S,.97431S,.97727S,.98022S,.98406S,
     1 .98933S,.99500S,.99999S,.99999S,.95010S,.95372S,.96547S,.97600S/
C.....COMPUTE FAN INLET CONDITIONS AND FAN PERFORMANCE PARAMETERS
  100 CALL QRBADS(ALT,0,3,IER)
      ALTM=DC(1)*ALT
      XMNM=DC(2)*XMN
      CALL FLCOND(ALTM,XMNM,TAM,P2A,T2A,P0A,T0A)
      P2=(.5S*P2A)/DC(3)
      T2=(.5S*T2A)/DC(4)
      CALL QWJDAS(P2,0,IER)
      CALL QWJDAS(T2,1,IER)
  105 CALL QRBADS(P13,3,3,IER)
      X3=(DC(58)*P13)/P2
      Y3=(DC(59)*XNL)/SSQRT(T2)
      WAR2M=MAP(N3,F3,X3,Y3)
      ETAOFM=MAPL(F3)
      P22Q2M=MAPL(F3)
      ETAIFM=MAPL(F3)
      IF(CIVV.LT..0S)CIVV=.0S
  110 FSHIFT=MAP(N1,F1,CIVV,Y3)
      WAR2=(WAR2M*(.5S+.5S*FSHIFT))/.5S
      CALL QWJDAS(X3,2,IER)
      CALL QWJDAS(WAR2,3,IER)
      WA2=(WAR2*P2)/(SSQRT(T2)*DC(57))
      P22=(P22Q2M*P2)/DC(60)
      PRIF=(DC(10)*P22)/P2
      T2M=DC(6)*T2
  115 CALL PROCOM(T2M,.0S,CP2,CV2,GM2,H2M)
      CALL TRAT(1,PRIF,GM2,TRIFM1)
      T22=(DC(11)*T2*(.2S+DC(91)*TRIFM1/ETAIFM))/.2S
C.....COMPUTE COMPRESSOR PERFORMANCE PARAMETERS
  120 CALL QRBADS(P3,6,3,IER)
      X4=(DC(62)*P3)/P22
      Y4=(DC(63)*XNH)/SSQRT(T22)
      WAR22M=MAP(N4,F4,X4,Y4)
      ETAHCM=MAPL(F4)
```

```
      IF(RCVV.LT..0S)RCVV=.0S
      CSHIFT=MAP(N2,F2,RCVV,Y4)
      WAR22=(WAR22M*(.5S+.5S*CSHIFT))/.5S
      CALL QWJDAS(X4,4,IER)
      CALL QWJDAS(WAR22,5,IER)
      WA22=WAR22*P22/SSQRT(T22)/DC(61)
C.....COMPUTE DERIVATIVES AT STATION 13
  125 CALL QRBADS(WA13,9,2,IER)
      DW13=WA2-DC(18)*WA22-DC(56)*WA13
      PROF=(DC(8)*P13)/P2
      CALL TRAT(2,PROF,GM2,TROFM1)
      T13P=(DC(9)*T2*(.2S+DC(96)*TROFM1/ETAOFM))/.2S
      T13PM=DC(34)*T13P
      CALL PROCOM(T13PM,.0S,CP13P,CV13P,GM13P,H13PM)
      H13P=H13PM/DC(15)
      T13M=DC(34)*T13
  130 CALL PROCOM(T13M,.0S,CP13,CV13,GM13,H13M)
      H13=H13M/DC(15)
      GM13M=(GM13-.5S)/.5S
      DTQW13=((WA2-DC(18)*WA22)*(H13P-H13))/CV13+T13*DW13*GM13M
      CALL QWJDAS(DW13,6,IER)
      CALL QWJDAS(DTQW13,7,IER)
C.....COMPUTE BLEEDS AND COMBUSTOR AIRFLOW
  135 CALL QRBADS(T3,11,2,IER)
      T3M=DC(13)*T3
      CALL PROCOM(T3M,.0S,CP3,CV3,GM3,H3M)
      FGM3=(.66572S-.23396S*GM3)*GM3+.33337S
      FGPT3=.5S*FGM3*P3/SSQRT(T3)
      IF(KBH.GT.0) GO TO 140
      WBLHT=.0S
      GO TO 145
  140 WBLHT=FGPT3/DC(98)
  145 IF(KBL.GT.0) GO TO 150
      WBLLT=.0S
      GO TO 155
  150 WBLLT=FGPT3/DC(99)
  155 IF(KBV.GT.0) GO TO 160
      WBLOV=.0S
      GO TO 165
  160 WBLOV=FGPT3/DC(100)
  165 WA3=SSQRT(P3*(P3-P4)/T3)/DC(64)
C.....COMPUTE DERIVATIVES AT STATION 3
      DW3=WA22-.2S*WBLHT-.02S*WBLLT-.002S*WBLOV-DC(43)*WA3
      H3=H3M/DC(17)
      PRHC=(DC(14)*P3)/P22
      TAVHC=DC(83)*T22+DC(84)*T3
  170 CALL PROCOM(TAVHC,.0S,CPHC,CVHC,GMHC,HHCM)
      CALL TRAT(3,PRHC,GMHC,TRHCM1)
      T3P=(DC(12)*T22*(.2S+DC(92)*TRHCM1/ETAHCM))/.2S
      T3PM=DC(13)*T3P
      CALL PROCOM(T3PM,.0S,CP3P,CV3P,GM3P,H3PM)
      H3P=H3PM/DC(17)
      GM3M=(GM3-.5S)/.5S
      DTQW3=WA22*(H3P-H3)/CV3+T3*DW3*GM3M
      CALL QWJDAS(DW3,8,IER)
```

```
      CALL  QWJDAS(DTQW3,9,IER)
C.....COMPUTE HP TURBINE PERFORMANCE PARAMETERS
  175 CALL QRBADS(T4,13,3,IER)
      X5=(DC(66)*P41)/P4
      Y5=(DC(67)*XNH)/SSQRT(T4)
      WP4=MAP(N5,F5,X5,Y5)
      HP4=MAPL(F5)
      WG4=(.5S*WP4*P4*XNH)/(T4*DC(65))
C.....COMPUTE DERIVATIVES AT STATION 4
      DW4=DC(23)*WA3-WG4+DC(44)*WF4
      TAVB=DC(85)*T3+DC(86)*T4
  180 CALL PROCOM(TAVB,.0S,CPB,CVB,GMB,HBM)
      HB=HBM*DC(22)
      FAR4M=(DC(20)*WF4)/WA3
      T4M=DC(21)*T4
  185 CALL PROCOM(T4M,FAR4M,CP4,CV4,GM4,H4M)
      H4=H4M*DC(22)
      GM4M=(GM4-.5S)/.5S
      ETAB=.99999S
      DTQW4=(DC(23)*WA3*HB-H4*(DC(23)*WA3+DC(44)*WF4)+DC(24)*ETAB*WF4)
     1 /CV4+T4*DW4*GM4M
      CALL QWJDAS(DW4,10,IER)
      CALL QWJDAS(DTQW4,11,IER)
C.....COMPUTE LP TURBINE PERFORMANCE PARAMETERS
  190 CALL QRBADS(T41,16,2,IER)
      P5=P6/DC(76)
      X6=(DC(74)*P5)/P41
      Y6=(DC(75)*XNL)/SSQRT(T41)
      WP41=MAP(N6,F6,X6,Y6)
      HP41=MAPL(F6)
      WG41=(WP41*P41*XNL)/(T41*DC(73))
C.....COMPUTE DERIVATIVES AT STATION 4.1
      DW41=DC(30)*WG4+DC(72)*WBLHT-WG41
      DH4=(HP4*SSQRT(T4)*XNH)/DC(87)
      FAR41M=(FAR4M*.5S)/(.5S+(.5S+.04S*FAR4M)*DC(71)*WBLHT/WG4)
      T41M=DC(28)*T41
  195 CALL PROCOM(T41M,FAR41M,CP41,CV41,GM41,H41M)
      H41=H41M*DC(29)
      GM41M=(GM41-.5S)/.5S
      DTQW41=(WG4*H4/DC(26)-(DC(30)*WG4+DC(72)*WBLHT)*H41+DC(68)*
     1 WBLHT*H3-DC(27)*DH4*(WG4+DC(70)*WBLHT))/CV41+T41*DW41*GM41M
      CALL QWJDAS(DW41,12,IER)
      CALL QWJDAS(DTQW41,13,IER)
C.....COMPUTE DERIVATIVES AT STATION 6
  200 CALL QRBADS(WG6,18,2,IER)
      DW6=DC(35)*WA13-WG6+DC(80)*WBLLT+DC(36)*WG41
      DH41=(HP41*SSQRT(T41)*XNL)/DC(88)
      T6M=DC(38)*T6
      FAR6M=(FAR41M*.25S)/(.25S+(.5S+.04S*FAR41M)*DC(37)*(WA13+DC(81)*
     1 WBLLT)/WG41)
  205 CALL PROCOM(T6M,FAR6M,CP6,CV6,GM6,H6M)
      H6=H6M/DC(39)
      GM6M=(GM6-.5S)/.5S
      DTQW6=(DC(31)*WA13*H13-(DC(35)*WA13+DC(80)*WBLLT+DC(36)*WG41)*
     1 H6+DC(32)*WG41*H41+DC(77)*WBLLT*H3-DC(33)*DH41*(WG41+DC(79)*
```

```
     2 WBLLT))/CV6+T6*DW6*GM6M
      CALL QWJDAS(DW6,14,IER)
      CALL QWJDAS(DTQW6,15,IER)
C.....COMPUTE NOZZLE PERFORMANCE PARAMETERS
  210 CALL QRBADS(P7,20,5,IER)
      P0=(.5S*P0A)/DC(5)
      P0QT7=DC(111)*P0/P7
      CDN=FUN1(N7,X7,P0QT7)
      CVN=FUN1L(X7)
  213 CALL NOZZL(P0,P7,P0QT7,T7,A8,AE,CDN,CVN,WG7M,FNM)
      FNET=FNM-XMNM*SSQRT(T0A)*WA2/DC(45)
      CALL QWJDAS(FNET,20,IER)
      WG7=(WG7M*DC(95))/.5S
C.....COMPUTE DERIVATIVES AT STATION 7
      DW7=DC(49)*WG6-WG7+DC(55)*WF7
      TAVAB=DC(89)*T6+DC(90)*T7
  215 CALL PROCOM(TAVAB,FAR6M,CPAB,CVAB,GMAB,HABM)
      HAB=HABM*DC(52)
      T7M=DC(51)*T7
      FAR7M=FAR6M+(DC(50)*(.04S*FHR6M+.5S)*WF7)/(.5S*WG6)
  220 CALL PROCOM(T7M,FAR7M,CP7,CV7,GM7,H7M)
      H7=H7M*DC(52)
      GM7M=(GM7-.5S)/.5S
      IF(FAR7M.GT..39762S) GO TO 225
      ETAAB=((.81226S-.56042S*FAR7M)*FAR7M+.25S)/.5S
      GO TO 230
  225 ETAAB=((.75331S-.72512S*FAR7M)*FAR7M+.29948S)/.5S
  230 DTQW7=(DC(49)*WG6*HAB-H7*(DC(49)*WG6+DC(55)*WF7)+DC(53)*WF7*
     1 ETAAB)/CV7+T7*DW7*GM7M
      CALL QWJDAS(DW7,16,IER)
      CALL QWJDAS(DTQW7,17,IER)
C.....COMPUTE SPEED DERIVATIVES
      H2=H2M/DC(7)
      T22M=DC(19)*T22
  235 CALL PROCOM(T22M,.0S,CP22,CV22,GM22,H22M)
      H22=H22M/DC(16)
      DXNL=(.5S*DH41*(WG41+DC(79)*WBLLT)/DC(93)-(WA2-DC(18)*WA22)*(H13P-
     1 DC(9)*H2)/DC(41)-WA22*(H22-DC(11)*H2)/DC(42))/XNL
      DXNH=(.5S*DH4*(WG4+DC(70)*WBLHT)/DC(94)-WA22*(H3P-DC(12)*
     1 H22)/DC(40))/XNH
      CALL QWJDAS(DXNL,18,IER)
      CALL QWJDAS(DXNH,19,IER)
      RETURN
      END


      SCALED FRACTION FUNCTION MAP(N,F,XIN,YIN)
      SCALED FRACTION F(1),XIN,YIN,YINCR,XHI,XLO,XFRAC,ZL,ZR,Y1,Y2,MAP
      DIMENSION N(1)
      IF(N(1).EQ.0) GO TO 400
      I=N(2)
      J=N(3)
      NXP=N(4)
      NYC=N(5)
```

```
      NPROD=NYC*NXP
  100 Y1=YIN-F(J)
      IF(Y1.GT..05) GO TO 110
      IF(Y1.EQ..05) GO TO 120
      IF(J.LE.1) GO TO 140
      J=J-1
      GO TO 100
  110 Y2=YIN-F(J+1)
      IF(Y2.LT..05) GO TO 180
      IF(Y2.EQ..05) GO TO 130
      J=J+1
      IF(J.GE.NYC) GO TO 150
      Y1=Y2
      GO TO 110
  120 YINCR=.05
      GO TO 190
  130 YINCR=.99995
      GO TO 190
  140 YINCR=.05
      GO TO 160
  150 YINCR=.99995
      J=J-1
  160 CALL MOOR(N,XIN,YIN)
      GO TO 190
  180 YINCR=Y1/(Y1-Y2)
  190 KX=J*NXP+NYC+I
      LX=KX-NXP
  200 XLO=F(LX)+YINCR*(F(KX)-F(LX))
      IF(XIN.GT.XLO) GO TO 210
      IF(XIN.EQ.XLO) GO TO 220
      IF(I.LE.1) GO TO 240
      I=I-1
      LX=LX-1
      KX=KX-1
      GO TO 200
  210 XHI=F(LX+1)+YINCR*(F(KX+1)-F(LX+1))
      IF(XIN.LT.XHI) GO TO 280
      IF(XIN.EQ.XHI) GO TO 230
      I=I+1
      IF(I.GE.NXP) GO TO 250
      LX=LX+1
      KX=KX+1
      XLO=XHI
      GO TO 210
  220 XFRAC=.05
      GO TO 300
  230 XFRAC=.99995
      GO TO 300
  240 XFRAC=.05
      GO TO 260
  250 XFRAC=.99995
      I=I-1
  260 CALL MOOR(N,XIN,YIN)
      GO TO 300
  280 XFRAC=(XIN-XLO)/(XHI-XLO)
```

```
  300 N(3)=J
      N(2)=I
      LZ=LX+NPROD
      KZ=LZ+NXP
  350 ZL=F(LZ)+YINCR*(F(KZ)-F(LZ))
      ZH=F(LZ+1)+YINCR*(F(KZ+1)-F(LZ+1))
      MAP=ZL+XFRAC*(ZH-ZL)
      RETURN
  400 LZ=LZ+NPROD
      KZ=LZ+NXP
      GO TO 350
      END


      SCALED FRACTION FUNCTION MAPL(F)
      SCALED FRACTION MAP, MAPL, F, XIN, YIN
C.....XIN AND YIN ARE DUMMY ARGUMENTS AND DO NOT AFFECT RESULTS
      N=0
      MAPL=MAP(N, F, XIN, YIN)
      RETURN
      END


      SUBROUTINE MOOR(N, XIN, YIN)
      SCALED FRACTION XIN, YIN, X, Y
      COMMON/MAPVAL/MN, X, Y
      LOGICAL CL9
      IF(N EQ. 0) GO TO 100
      CALL QRSLL(6, CL9, IER)
      CALL QRSLL(6, CL9, IER)
      IF(CL9) RETURN
      CALL QWCLL(9, .TRUE., IER)
      MN=N
      X=XIN
      Y=YIN
      RETURN
  100 WRITE(1, 400)MN, X, Y
  400 FORMAT(/7HMAP NO. , I3, 20H INPUTS OUT OF RANGE/6HXIN = , S7,
     1 8H   YIN = , S7/)
      CALL QWCLL(9, .FALSE., IER)
      RETURN
      END


      SUBROUTINE NOZZL(P0, P7, P0QT7, T7, A7, A8, CD7, CV8, W7, F8)
      DIMENSION N1(3), N2(3), N3(3)
      SCALED FRACTION X1(15), Y1(15), X2(15), Y21(15), Y22(15), X3(15),
     1 Y31(15), Y32(15), Y33(15)
      SCALED FRACTION P0, P7, P0QT7, T7, A7, A8, CD7, CV8, W7, F8
      SCALED FRACTION XF, A8Q7, PEQ7, ATXX, PTOL, AEQTXX, XMX, XMN, P0YQX, PYQX,
     1 P0YQ0X, PYQT, PE, XSHFT, VE, FUN1, FUN1L, DC(125)
      COMMON /COEF/ DC
      DATA N1/6, 1, 15/, N2/7, 1, 15/, N3/8, 1, 15/
      DATA X1  /.50000S, .50035S, .50155S, .50645S, .51185S, .52350S, .53405S,
```

```
     1  .54715S,.56780S,.59410S,.64320S,.70090S,.75035S,.91145S,.99999S/
      DATA Y1  /.52828S,.54691S,.56578S,.60412S,.63000S,.66905S,.69507S,
     1  .72092S,.75283S,.78400S,.82589S,.85958S,.88065S,.92312S,.93611S/
      DATA X2  /.52828S,.54691S,.56578S,.60412S,.63000S,.66905S,.69507S,
     1  .72092S,.75283S,.78400S,.82589S,.85958S,.88065S,.92312S,.93611S/
      DATA Y21 /.50000S,.48740S,.47462S,.44851S,.43070S,.40338S,.38479S,
     1  .36589S,.34187S,.31740S,.28246S,.25192S,.23128S,.18410S,.16698S/
      DATA Y22 /.50000S,.50035S,.50155S,.50645S,.51185S,.52350S,.53405S,
     1  .54715S,.56780S,.59410S,.64320S,.70090S,.75035S,.91145S,.99999S/
      DATA X3  /.50000S,.50395S,.50870S,.51520S,.52335S,.53315S,.54450S,
     1  .55745S,.58810S,.60575S,.64610S,.71950S,.77760S,.88000S,.99999S/
      DATA Y31 /.20000S,.22000S,.23000S,.24000S,.25000S,.26000S,.27000S,
     1  .28000S,.30000S,.31000S,.33000S,.36000S,.38000S,.41000S,.43942S/
      DATA Y32 /.52828S,.46835S,.43983S,.41238S,.38606S,.36092S,.33697S,
     1  .31424S,.27240S,.25320S,.21839S,.17404S,.14924S,.11823S,.09396S/
      DATA Y33 /.50000S,.54062S,.56015S,.57915S,.59760S,.61555S,.63300S,
     1  .64995S,.68230S,.69775S,.72720S,.76800S,.79305S,.82765S,.85835S/
      XF=.0S
C  ***CALC AREA RATIO
      A8Q7 = DC(110) * A8/A7
      PEQ7=FUN1(N1,X1,A8Q7)
C  ***SUBSONIC FLOW
      ATXX =A7
      IF (P0QT7 .LT. PEQ7) GO TO 10
      XMX=FUN1(N2,X2,P0QT7)
      PTOL = SABS(P0QT7 - PEQ7)
      IF(PTOL.LE .0005S) GO TO 100
      AEQTXX=FUN1L(X2)
      ATXX = A8/AEQTXX
      GO TO 100
C  ***SUPERSONIC FLOW AND SONIC FLOW FOR CONV-ONLY NOZZLE
   10 IF (A8Q7 .LE. 0.5S) GO TO 30
      XMN=FUN1(N3,X3,A8Q7)
      PEQ7=FUN1L(X3)
      POYQX = (XMN*(XMN/.72217S-.07958S)/.4S)+.08956S
      PYQX = (XMN*(XMN/.85714S)/.4S)-.01667S
      POYQOX = (XMN*(-XMN/.87600S+0.40163S)/.2S)+0.83626S
      PYQT = (PYQX * POYQOX)/POYQX
      IF (P0QT7 .GT. PYQT) GO TO 40
      XMX=FUN1L(X3)
C    *REGIME V
   20 PE = P7 * PEQ7 / DC(111)
      XF = (A8*(PE-P0))*DC(112)
      GO TO 100
   30 XMX = 0.5S
      GO TO 20
C    *REGIME III
   40 IF(SABS(A8Q7-.56345S) .LE. .0005S) GO TO 50
      XMX= .85585S + (P0QT7*((-.41254S) + .17418S * P0QT7))/ .4S
      XSHFT = 0.0S
      IF (A8Q7 .GT. .6S) XSHFT = .61755S + A8Q7 *(A8Q7*(A8Q7*(-.12575S)
     1 +.38380S)+(-.44243S))/.25S
      IF (A8Q7 .LT. .6S) XSHFT = (-.53378S*A8Q7)+.31910S
      XMX=XMX+XSHFT
      GO TO 100
```

```
   50 XMX= .88080S+(P0QT7*((-.41488S)+.17255S*P0QT7))/.45
C    ***VELOCITY,FLOW,AND THRUST CALCULATIONS
  100 VE = XMX*SSQRT(T7)*CV8/DC(113)
  105 W7 = (ATXX*P7)/SSQRT(T7) * CD7 / DC(115)
      F8 = (W7*VE)/DC(116)+XF
  500 RETURN
      END



      SUBROUTINE PROCOM(T,FA,CP,CV,GAM,H)
      SCALED FRACTION T,FA,CP,CV,GAM,H,CPA,HA,AMW,R,TD,CPF,HF
      IF(T.GE..46000S) GO TO 50
      IF(T.GE..24000S) GO TO 40
      CPA=.48068S-(.12464S-T/.98246S)*T
      HA=.00176S+(.56558S+.14075S*T)*T
      GO TO 60
   40 CPA=.40528S+(.52545S-.38182S*T)*T
      HA=.00119S+(.56298S+.16150S*T)*T
      GO TO 60
   50 CPA=.46063S+(.30024S-.15378S*T)*T
      HA=-.01870S+(.64968S+.06700S*T)*T
   60 AMW=.57940S-.00144S*FA
      R=.07945S/AMW
      TD=.70000S-T
      CPF=.93330S-(.29350S+.81750S*TD)*TD
      HF=-.03305S+(.63624S+.38625S*T)*T
      H=(.66667S*HA+.10667S*HF*FA)/(.50000S+.04000S*FA)
      CP=(.80000S*CPA+.12800S*CPF*FA)/(.80000S+.06400S*FA)
      CV=CP-R
      GAM=.50000S*CP/CV
      RETURN
      END



      SUBROUTINE TRAT(N,PRC,GAM,TR)
      DIMENSION N1(3),N2(3),N3(3)
      SCALED FRACTION GAM,TR,PRC,S,C,TRC,X1(25),Y1(25),FUN1
      DATA N1/5,1,25/,N2/5,1,25/,N3/5,1,25/
      DATA X1  /.06667S,.07333S,.08000S,.09333S,.10667S,.12000S,.13333S,
     1  .14667S,.16667S,.20000S,.23333S,.26667S,.30000S,.33333S,.36667S,
     2  .40000S,.46667S,.53333S,.60000S,.66667S,.73333S,.80000S,.86667S,
     3  .93333S,.99999S/
      DATA Y1  /.00000S,.02001S,.03872S,.07292S,.10367S,.13169S,.15749S,
     1  .18145S,.21452S,.26362S,.30699S,.34599S,.38152S,.41424S,.44462S,
     2  .47301S,.52492S,.57159S,.61412S,.65328S,.68964S,.72362S,.75557S,
     3  .78575S,.81437S/
  150 IF(GAM.GE..67500S) GO TO 300
      IF(PRC.GE..33333S) GO TO 200
      S=.50526S+.77229S*PRC-.95164S*PRC*PRC
      GO TO 700
  200 S=.58745S+.23850S*PRC-.09000S*PRC*PRC
      GO TO 700
  300 IF(PRC.GE..33333S) GO TO 400
      S=.45911S+.89475S*PRC-(PRC*PRC)/.92895S
```

```
      GO TO 700
  400 S=.55215S+.29319S*PRC-.10919S*PRC*PRC
  700 C=((.67500S-GAM)*S)/.02500S
      IF(N NE. 1) GO TO 750
      TRC=FUN1(N1,X1,PRC)
      GO TO 900
  750 IF(N.NE.2) GO TO 800
      TRC=FUN1(N2,X1,PRC)
      GO TO 900
  800 TRC=FUN1(N3,X1,PRC)
  900 TR=(TRC*(.80000S-.16000S*C))/.80000S
      RETURN
      END
```

Fortran Symbols

BTASK. – All variables are scaled fractions unless otherwise noted.

AE	exhaust nozzle exit area
ALT	altitude
A8	exhaust nozzle throat area
CDN	exhaust nozzle flow coefficient
CIVV	fan variable-geometry parameter
CVN	exhaust nozzle velocity coefficient
DH4	high-pressure-turbine enthalpy drop
DH41	low-pressure-turbine enthalpy drop
ERR(R)	error ratio array of the ratios of the individual elements of the Y array to the corresponding element in the X array, $k=1$ to 45
ETAAB	augmentor efficiency
ETAB	combustor efficiency
FNET	net thrust
FNM	gross thrust
I	integer index
JI(k)	dummy arrays to fill in VARS common block locations not used in BTASK, $I=1$ to 13
PI	total pressure at station I
RCVV	compressor variable-geometry parameter
SF(k)	scale factor array for selected engine variables, corresponds to X array, $k=1$ to 45
TAM	sea-level ambient temperature
TI	total temperature at station I
WAI	airflow rate leaving station I
WF4	combustor fuel flow rate
WF7	augmentor fuel flow rate
WGI	gas flow rate leaving station I
X(k)	array of the desired (actual) value of operating point data for selected engine variables, $k=1$ to 45
XMN	Mach number
XNH	high-spool rotor speed
XNL	low-spool rotor speed
Y(k)	array of the calculated value of operating point data for selected engine variables, $k=1$ to 45

DATAIN

F(k)	map data array (scaled fraction), k determined by calling program
I	integer index
INC	number of points defining map (integer)
ISC(k)	map scale factor format array (alphanumeric), $k=1$ to 4
IX(k)	map X input data format array (alphanumeric), $k=1$ to 4
IY(k)	map Y input data format array (alphanumeric), $k=1$ to 4
IZ1(k)	map Z1 output data format array (alphanumeric), $k=1$ to 4
IZ2(k)	map Z2 output data format array (alphanumeric), $k=1$ to 4
IZ3(k)	map Z3 output data format array (alphanumeric), $k=1$ to 4
IZ4(k)	map Z4 output data format array (alphanumeric), $k=1$ to 4
J	integer index
JF	integer index
JMNPT	integer index
JS	integer index
K	integer index
L	integer index

M	map number (integer)
N(k)	map integer array, k determined by calling program
NCOM	integer indicating rectilinear map data (nonzero for rectilinear map data)
NCV	number of curves defining map
NFCT	number of functions in map
NF	integer index
NPT	number of points per curve
TEST	map data increment used for detecting overflows (scaled fraction)
VALS(k)	array used to store unscaled Y and Z_1 map data (floating point), $k=1$ to 25
XSC	map scale factor for X input variable (floating point)
XVALS(k)	array used to store unscaled X map data (floating point), $k=1$ to 25
YSC	map scale factor for Y input variable (floating point)
ZSC(k)	map scale factor for Z_i output variables (floating point), $k=1$ to 4

FLCOND. – All variables are scaled fractions unless otherwise specified.

ETAI	inlet efficiency
HT	altitude
INLET	inlet configuration option (integer)
N1(k)	ambient pressure function integer array, $K=1$ to 3
N2(k)	$(T_2/T_0)^{\gamma_l/(\gamma_l-1)}$ function array integer array, $k=1$ to 3
N3(k)	$(M_0-1)^{1.35}$ function integer array, $k=1$ to 3
PS	ambient pressure
PT	fan inlet total pressure
PTQS	isentropic inlet pressure ratio
TAS	sea-level ambient pressure
TR35	$(T_2/T_0)^{\gamma_l/(\gamma_l-1)}$
TS	ambient pressure
TTQS	inlet temperature ratio
TT	fan inlet total temperature
XMO	Mach number
XMP	supersonic Mach number minus 1
XPP	$(M_0-1)^{1.35}$
X1(k)	altitude data array, $k=1$ to 11
X2(k)	inlet temperature ratio data array, $k=1$ to 10
X3(k)	Mach-number-minus-1 data array, $k=1$ to 7
Y1(k)	ambient pressure data array, $k=1$ to 11
Y2(k)	$(T_2/T_0)^{\gamma_l/(\gamma_l-1)}$ data array, $k=1$ to 10
Y3(k)	$(M_0-1)^{1.35}$ data array, $k=1$ to 7

FUN1/FUN1L/FOOR. – All variables are scaled fractions unless otherwise specified.

F(k)	function data array, k determined by calling program
FUN1/FUN1L	function output
I	X variable search index (integer)
MN	saved function number for out-of-range message (integer)
N(k)	function integer array, k determined by calling program
NXP	number of points defining function (integer)
X	saved value of XIN for out-of-range message
XFRAC	fraction of X1–X2 interval covered by XIN
XIN	X input
X1	difference between XIN and Ith value of X
X2	difference between XIN and $(I+1)^{th}$ value of X

INITAL

C(k)	correction factor array, $k=1$ to 50
DACI(k)	DAC initial condition array (scaled fraction), $k=1$ to 24
DC(k)	corrected digital coefficient array (scaled fraction), $k=1$ to 125
ERR	potentiometer setting error (scaled fraction)
F1(k)	fan variable-geometry-effects map data array (scaled fraction), $k=1$ to 322
F2(k)	compressor variable-geometry-effects map data array (scaled fraction), $k=1$ to 322
F3(k)	fan map data array (scaled fraction), $k=1$ to 854
F4(k)	compressor map data array (scaled fraction), $k=1$ to 518
F5(k)	high-pressure-turbine map data array (scaled fraction), $k=1$ to 224
F6(k)	low-pressure-turbine map data array (scaled fraction), $k=1$ to 224
I	integer index
IAADR(k)	integrator address array (integer), $k=1$ to 16

IBELL terminal bell integer

IER hybrid linkage routine error flag (integer)

IFRADR function relay address (integer)

IG(k) integrator gain integer array, $k = 1$ to 10

INLET inlet configuration option (integer)

J integer index

K integer index

KBH high-pressure-turbine bleed flow indicator (integer)

KBL low-pressure-turbine bleed flow indicator (integer)

KBV overboard bleed flow indicator (integer)

N1(k) fan variable-geometry-effects map integer array, $k = 1$ to 5

N2(k) compressor variable-geometry-effects map integer array, $k = 1$ to 5

N3(k) fan map integer array, $k = 1$ to 5

N4(k) compressor map integer array, $k = 1$ to 5

N5(k) high-pressure-turbine map integer array, $k = 1$ to 5

N6(k) low-pressure-turbine map integer array, $k = 1$ to 5

PADR(k) potentiometer address array (alphanumeric), $k = 1$ to 53

PSET actual potentiometer setting (scaled fraction)

PVAL(k) calculated potentiometer setting (scaled fraction), $k = 1$ to 53

XC(k) map scale factor array for X input variables, $k = 1$ to 6

YC(k) map scale factor array for Y input variables, $k = 1$ to 6

ZC(k) map scale factor array for Z output variables, $k = 1$ to 12

LEVEL8

IER hybrid linkage routine error flag (integer)

PB5 user-set pushbutton number 5 (logical)

LOOP. – All variables are scaled fractions unless otherwise noted.

AE exhaust nozzle exit area

ALT altitude

ALTM FLCOND-rescaled altitude

A8 exhaust nozzle throat area

CC(k) correction factor array (floating point), $k = 1$ to 50

CDN exhaust nozzle flow coefficient

CIVV fan variable-geometry parameter

CPAB average specific heat at constant pressure in augmentor

CPB average specific heat at constant pressure in combustor

CPHC average specific heat at constant pressure in compressor

CPI specific heat at constant pressure at station I

CSHIFT compressor variable-geometry effect on corrected flow

CVAB average specific heat at constant volume in augmentor

CVB average specific heat at constant volume in combustor

CVHC average specific heat at constant volume in compressor

CVN exhaust nozzle velocity coefficient

CVI specific heat at constant volume at station I

DACI(k) DAC initial condition array, $k = 1$ to 24

DC(k) corrected digital coefficient array, $k = 1$ to 125

DH4 high-pressure-turbine enthalpy drop

DH41 low-pressure-turbine enthalpy drop

DTQWI specific temperature derivative at station I

DXNH high-rotor-speed derivative

DXNL low-rotor-speed derivative

DWI stored mass derivative at station I

ETAAB augmentor efficiency

ETAB combustor efficiency

ETAHCM compressor efficiency

ETAIFM fan inside-diameter efficiency

ETAOFM fan outside-diameter efficiency

FARIM fuel-air ratio at station I

FGM3 function of specific heat ratio at station 3

FGPT3 compressor discharge bleed flow parameter

FNET net thrust

FSHIFT fan variable-geometry effect on corrected flow

F1(k) fan variable-geometry-effects map data array, $k = 1$ to 322

F2(k) compressor variable-geometry-effects map data array, $k = 1$ to 322

F3(k) fan map data array, $k = 1$ to 854

F4(k) compressor map data array, $k = 1$ to 518

F5(k) high-pressure-turbine map data array, $k=1$ to 224

F6(k) low-pressure-turbine map data array, $k=1$ to 224

GMAB average specific heat ratio in augmentor

GMB average specific heat ratio in combustor

GMHC average specific heat ratio in compressor

GMI specific heat ratio at station I

GMIM specific heat ratio minus 1 at station I

HA average specific enthalpy in augmentor

HABM PROCOM-rescaled augmentor specific enthalpy

HB average specific enthalpy in combustor

HBM PROCOM-rescaled combustor specific enthalpy

HHCM PROCOM-rescaled compressor specific enthalpy

HP4 high-pressure-turbine enthalpy drop parameter

HP41 low-pressure-turbine enthalpy drop parameter

HI specific enthalpy at station I

HIM PROCOM-rescaled specific enthalpy at station I

IER hybrid linkage routine error flag (integer)

INLET inlet configuration option (integer)

KBH high-pressure-turbine bleed flow indicator (integer)

KBL low-pressure-turbine bleed flow indicator (integer)

KBV overboard bleed flow indicator (integer)

N1(k) fan variable-geometry effects map integer array, $k=1$ to 5

N2(k) compressor variable-geometry-effects map integer array, $k=1$ to 5

N3(k) fan map integer array, $k=1$ to 5

N4(k) compressor map integer array, $k=1$ to 5

N5(k) high-pressure-turbine map integer array, $k=1$ to 5

N6(k) low-pressure-turbine map integer array, $k=1$ to 5

N7(k) nozzle functions integer array, $k=1$ to 3

PRHC TRAT-rescaled compressor pressure ratio

PRIF TRAT-rescaled fan inside-diameter pressure ratio

PROF TRAT-rescaled fan outside-diameter pressure ratio

PI total pressure at station I

POA FLCOND-rescaled ambient pressure

POQT7 exhaust nozzle pressure ratio

P2A FLCOND-rescaled fan inlet total pressure

P22Q2M fan inside-diameter pressure ratio

RCVV compressor variable-geometry parameter

TAM sea-level ambient temperature

TAVAB average total temperature in augmentor

TAVB average total temperature in combustor

TAVHC average total temperature in compressor

TRHCM1 compressor temperature rise parameter

TRIFM1 fan inside-diameter temperature rise parameter

TROFM1 fan outside-diameter temperature rise parameter

TOA FLCOND-rescaled ambient temperature

TI total temperature at station I

TIM PROCOM-rescaled total temperature at station I

T2A FLCOND-rescaled fan inlet total temperature

WAR2 fan corrected flow rate

WAR2M nominal fan corrected flow rate

WAR22 compressor corrected flow rate

WAI airflow rate leaving station I

WBLHT high-pressure-turbine cooling bleed flow rate

WBLLT low-pressure-turbine cooling bleed flow rate

WBLOV overboard bleed flow rate

WF4 combustor fuel flow rate

WF7 augmentor fuel flow rate

WGI gas flow rate leaving station I

WG7M NOZZL-rescaled exhaust nozzle flow rate

WP4 high-pressure-turbine flow parameter

WP41 low-pressure-turbine flow parameter

XC(k) map scale factor array for X input variables (floating point), $k=1$ to 6

XMN Mach number

XMNM FLCOND-rescaled Mach number

XNH high-spool rotor speed

XNL	low-spool rotor speed
XIX	input for map number I
YC(k)	map scale factor array for Y input variables (floating point), $k=1$ to 6
YI	Y input for map number I
Y71(k)	exhaust nozzle flow coefficient function data array, $k=1$ to 15
Y72(k)	exhaust nozzle velocity coefficient function data array, $k=1$ to 15
ZC(k)	map scale factor array for Z output variables (floating point), $k=1$ to 12

MAP/MAPL/MOOR. – All variables are scaled fractions unless otherwise specified.

F(k)	map data array, k determined by calling program
I	X variable search index (integer)
J	Y variable search index (integer)
KX	map data array index corresponding to $X(I, J+1)$(integer)
KZ	map data array index corresponding to $Z(I, J+1)$(integer)
LZ	map data array index corresponding to $Z(I, J)$(integer)
MAP/MAPL	map output
MN	saved map number for out-of-range message (integer)
N(k)	map integer array, k determined by calling program
NPROD	number of points defining map
NXP	number of points per curve (integer)
NYC	number of curves in map (integer)
X	saved value of XIN for out-of-range message
XFRAC	fraction of XHI – XLO interval covered by XIN
XHI	$(I+1)^{th}$ breakpoint on interpolated YIN curve
XIN	x input
XLO	I^{th} breakpoint on interpolated YIN curve
Y	saved value of YIN for out-of-range message
Y1	difference between YIN and J^{th} value of Y in map
Y2	difference between YIN and $(J+1)^{th}$ value of Y in map
YIN	y input
YINCR	fraction on Y(J+1) – Y(J) interval covered by YIN
ZH	map output at XHI,YIN
ZL	MAP output at XLO,YIN

NOZZL. – All variables are scaled fractions unless otherwise specified.

AEQTXX	ratio of exit area to area at which sonic flow is reached
ATXX	reduced throat area that results in sonic flow
A7	throat area
A8	exit area
A8Q7	expansion ratio
CD7	flow coefficient
CV8	velocity coefficient
DC(k)	corrected digital coefficient, $k=1$ to 125
F8	gross thrust
N1(k)	critical pressure ratio function integer array, $k=1$ to 3
N2(k)	subsonic functions integer array, $k=1$ to 3
N3(k)	supersonic functions integer array, $k=1$ to 3
PE	exit plane pressure
PEQ7	critical pressure ratio
PTOL	difference between computed exit plane pressure and ambient pressure for subconic exit flow
PYQT	pressure ratio at which shock is in exit plane
PYQX	static pressure ratio across shock
PO	ambient pressure
POQT7	ambient to total pressure ratio
POYQX	total to static pressure ratio across shock
POYQOX	total pressure ratio across shock
T7	inlet total temperature
VE	exit velocity
W7	mass flow rate
XF	pressure-area term in gross thrust equation
XMN	Mach number upstream of shock
XMX	dimensionless velocity
XSHFT	area ratio shift on dimensionless velocity–pressure ratio fit
X1(k)	subsonic flow area ratio data array, $k=1$ to 15
X2(k)	subsonic flow pressure ratio data array, $k=1$ to 15
X3(k)	supersonic flow area ratio data array, $k=1$ to 15

Y1(k)	critical pressure ratio data array, $k=1$ to 15
Y21(k)	subsonic dimensionless velocity data array, $k=1$ to 15
Y22(k)	AEQTXX data array, $k=1$ to 15
Y31(k)	XMN data array, $k=1$ to 15
Y33(k)	supersonic dimensionless velocity data array, $k=1$ to 15

PROCOM. – All variables are scaled fractions.

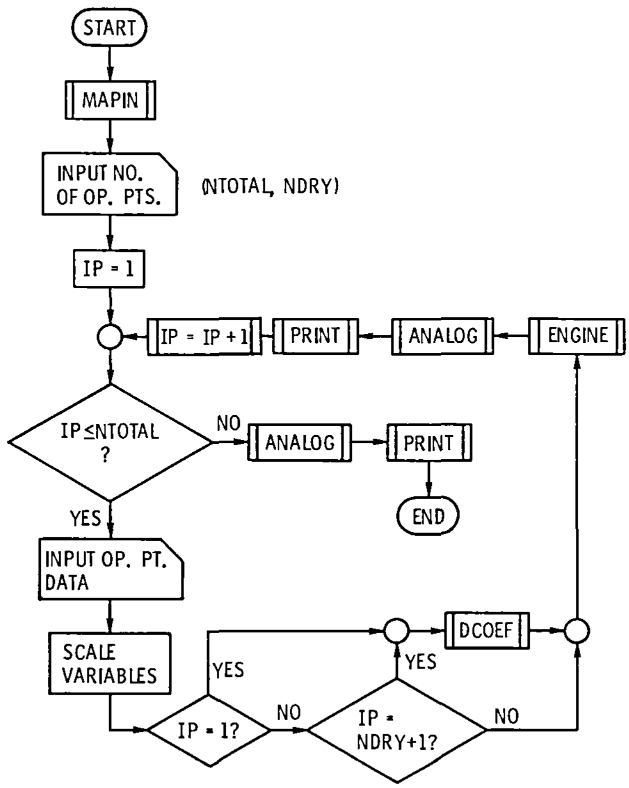
AMW	molecular weight
CP	specific heat at constant pressure
CPA	specific heat at constant pressure of air
CPF	specific heat at constant pressure of fuel
CV	specific heat at constant volume
FA	fuel-air ratio
GAM	specific heat ratio
H	specific enthalpy
HA	specific enthalpy of air
HF	specific enthalpy of fuel
R	gas constant
T	total temperature
TD	3500° R minus T

TRAT. – All variables are scaled fractions unless otherwise specified.

C	variable specific heat ratio effect on $(\Delta T/T)_{id}$
GAM	average component specific heat ratio
N	component identifying integer, N = 1 to 3
N1(k)	fan inside-diameter function integer array, $k=1$ to 3
N2(k)	fan outside-diameter function integer array, $k=1$ to 3
N3(k)	compressor function integer array, $k=1$ to 3
PRC	component pressure ratio
S	pressure ratio effect on specific heat ratio shift
TR	$(\Delta T/T)_{id}$
TRC	$(\Delta T/T)_{id}$ for specific heat ratio of 1.35
X1(k)	component pressure ratio data array, $k=1$ to 25
Y1(k)	function data array, $k=1$ to 25

Appendix E

Host Digital Program



Flowchart of host digital program

Source Listings

```
C***********MAIN*********
C
C
C      MAIN PROGRAM FOR GENERALIZED TURBOFAN SIMULATION
       INTEGER PLA,AADR,APRINT
       REAL KBLWHT,KBLWLT
       COMMON /IO/IR,IW,IPNCH
       COMMON /INL/INLET
       COMMON /NMAPS/F1(322),F2(322),F3(854),F4(518),F5(224),F6(224),   -
      1              N1(5),N2(5),N3(5),N4(5),N5(5),N6(5)
       COMMON /IVARS/ IP,JP,JPA,JPD,KBH,KBL,KBV,NAUG,NDRY,NTOTAL,PLA
       COMMON /AVARS/ CC(50),DACI(24),DC(125),XC(6),YC(6),ZC(12),UDC(125)
       COMMON /ANVARS/ AQL13,AQL6,SFT,SFW13,SFW3,SFW4,SFW41,SFW6,SFW7,      -
      1                W13,W3,W4,W41,W6,W7,SW13,SW3,SW4,SW41,SW6,SW7,       -
      A                V13,V3,V4,V41,V6,V7,XIH,XIL,PADR(53),PVAL(53),       -
      B                IG(10),AADR(16)
       COMMON /RVARS/ CBLHR(50),CBLLR(50),CBLVR(50),DH4QR(50),DH4TR(50),-
      A               DH41QR(50),DH41TR(50),DP13R(50),DP6R(50),ETAABR(50),-
      B               ETABR(50),FGR(50),P0R(50),P2R(50),P22R(50),-
      C               P5R(50),T13PR(50),T2R(50),T22R(50),T3PR(50),-
```

```
     D               WA2R(50),WA22R(50),WA3R(50),WG4R(50),WG41R(50),-
     E               WG7R(50)
      COMMON /SVARS/ BSCIVV,BSRCVV,SFA8,SFAE,SFALT,SFCIVV,SFDH4,SFDH41,-
     A               SFFG,SFFN,SFP0,SFP13,SFP2,SFP22,SFP3,SFP4,SFP41,-
     B               SFP5,SFP6,SFP7,SFRCVV,SFT0,SFT13,SFT2,SFT22,SFT3,-
     C               SFT4,SFT41,SFT6,SFT7,SFVEL,SFWA13,SFWA2,SFWA22,-
     D               SFWA3,SFWF4,SFWF7,SFWG4,SFWG41,SFWG6,SFWG7,SFXMN, -
     E               SFXNH,SFXNL
      COMMON /XVARS/ A1(50),A2(50),A3(50),A4(50),A5(50),A6(50),A7(50),  -
     1               A81(50),A9(50),A10(50),A11(50),A12(50),A13(50),    -
     2               A14(50),A15(50),A16(50),A17(50),A18(50),A19(50),   -
     3               A20(50),A21(50),A22(50),A23(50),A24(50),A25(50),   -
     4               A26(50)
      EQUIVALENCE (A8    , A1(  1)),(AE    , A1(  2)),(ALT   , A1(  3)),  -
     1            (ALTM  , A1(  4))
      EQUIVALENCE (CD7   , A3(  1)),(CDN   , A3(  2)),(CIVV  , A3(  3)),  -
     2            (CP13  , A3(  4)),(CP13P , A3(  5)),(CP2   , A3(  6)),  -
     2            (CP22  , A3(  7)),(CP3   , A3(  8)),(CP3P  , A3(  9)),  -
     3            (CP4   , A3( 10)),(CP41  , A3( 11)),(CP6   , A3( 12)),  -
     4            (CP7   , A3( 13)),(CPAB  , A3( 14)),(CPB   , A3( 15)),  -
     5            (CPHC  , A3( 16)),(CSHIFT, A3( 17)),(CV13  , A3( 18))
      EQUIVALENCE (CV13P , A3( 19)),(CV2   , A3( 20)),(CV22  , A3( 21)), -
     7            (CV3   , A3( 22)),(CV3P  , A3( 23)),(CV4   , A3( 24)), -
     8            (CV41  , A3( 25)),(CV6   , A3( 26)),(CV7   , A3( 27)), -
     9            (CV8   , A3( 28)),(CVAB  , A3( 29)),(CVB   , A3( 30)), -
     A            (CVHC  , A3( 31)),(CVN   , A3( 32))
      EQUIVALENCE (DH4   , A4(  1)),(DH41  , A4(  2)),(DTQW13, A4(  3)), -
     1            (DTQW3 , A4(  4)),(DTQW4 , A4(  5)),(DTQW41, A4(  6)), -
     2            (DTQW6 , A4(  7)),(DTQW7 , A4(  8)),(DW13  , A4(  9)), -
     3            (DW3   , A4( 10)),(DW4   , A4( 11)),(DW41  , A4( 12)), -
     4            (DW6   , A4( 13)),(DW7   , A4( 14)),(DXNH  , A4( 15)), -
     5            (DXNL  , A4( 16))
      EQUIVALENCE (ETAAB , A5(  1)),(ETAB  , A5(  2)),(ETAHCM, A5(  3)), -
     1            (ETAIFM, A5(  4)),(ETAOFM, A5(  5))
      EQUIVALENCE (FAR4M , A6(  1)),(FAR41M, A6(  2)),(FAR6M , A6(  3)), -
     1            (FAR7M , A6(  4)),(FG    , A6(  5)),(FGM3  , A6(  6)), -
     2            (FGPT3 , A6(  7)),(FN    , A6(  8)),(FNET  , A6(  9)), -
     3            (FNM   , A6( 10)),(FSHIFT, A6( 11))
      EQUIVALENCE (GM13  , A7(  1)),(GM13M , A7(  2)),(GM13P , A7(  3)), -
     1            (GM2   , A7(  4)),(GM22  , A7(  5)),(GM3   , A7(  6)), -
     2            (GM3M  , A7(  7)),(GM3P  , A7(  8)),(GM4   , A7(  9)), -
     3            (GM41  , A7( 10)),(GM4M  , A7( 11)),(GM41M , A7( 12)), -
     4            (GM6   , A7( 13)),(GM6M  , A7( 14)),(GM7   , A7( 15)), -
     5            (GM7M  , A7( 16)),(GMAB  , A7( 17)),(GMB   , A7( 18)), -
     6            (GMHC  , A7( 19))
      EQUIVALENCE (H13   ,A81(  1)),(H13M  ,A81(  2)),(H13P  ,A81(  3)), -
     1            (H13PM ,A81(  4)),(H2    ,A81(  5)),(H22   ,A81(  6)), -
     2            (H2M   ,A81(  7)),(H22M  ,A81(  8)),(H3    ,A81(  9)), -
     3            (H3M   ,A81( 10)),(H3P   ,A81( 11)),(H3PM  ,A81( 12)), -
     4            (H4    ,A81( 13)),(H41   ,A81( 14)),(H4M   ,A81( 15)), -
     5            (H41M  ,A81( 16)),(H6    ,A81( 17)),(H6M   ,A81( 18)), -
     6            (H7    ,A81( 19)),(H7M   ,A81( 20)),(HAB   ,A81( 21)), -
     7            (HABM  ,A81( 22)),(HB    ,A81( 23)),(HBM   ,A81( 24)), -
     8            (HHCM  ,A81( 25)),(HP4   ,A81( 26)),(HP41  ,A81( 27))
      EQUIVALENCE (KBLWHT,A11(  1)),(KBLWLT,A11(  2))
      EQUIVALENCE (PE    ,A16(  1)),(P0    ,A16(  2)),(P0A   ,A16(  3)), -
     1            (P0QT7 ,A16(  4)),(P13   ,A16(  5)),(P2    ,A16(  6)), -
     2            (P2A   ,A16(  7)),(P22   ,A16(  8)),(P22Q2M,A16(  9)), -
     3            (P3    ,A16( 10)),(P4    ,A16( 11)),(P41   ,A16( 12)), -
     4            (P5    ,A16( 13)),(P6    ,A16( 14)),(P7    ,A16( 15)), -
     5            (PRHC  ,A16( 16)),(PRIF  ,A16( 17)),(PROF  ,A16( 18))
      EQUIVALENCE (RCVV  ,A18(  1)),(RTT2  ,A18(  2)),(RTT22 ,A18(  3)), -
     1            (RTT4  ,A18(  4)),(RTT41 ,A18(  5))
      EQUIVALENCE (T0A   ,A20(  1)),(T13   ,A20(  2)),(T13M  ,A20(  3)), -
     1            (T13P  ,A20(  4)),(T13PM ,A20(  5)),(T2    ,A20(  6)), -
     2            (T2A   ,A20(  7)),(T2M   ,A20(  8)),(T22   ,A20(  9)),-
     3            (T22M  ,A20( 10)),(T3    ,A20( 11)),(T3M   ,A20( 12)),-
     4            (T3P   ,A20( 13)),(T3PM  ,A20( 14)),(T4    ,A20( 15)), -
     5            (T4M   ,A20( 16)),(T41   ,A20( 17)),(T41M  ,A20( 18)), -
     6            (T6    ,A20( 19)),(T6M   ,A20( 20)),(T7    ,A20( 21)), -
     7            (T7M   ,A20( 22)),(TAM   ,A20( 23)),(TAVAB ,A20( 24)), -
     8            (TAVB  ,A20( 25)),(TAVHC ,A20( 26)),(TRHCM1,A20( 27)), -
     9            (TRIFM1,A20( 28)),(TROFM1,A20( 29))
```

```
      EQUIVALENCE (WA13  ,A23(  1)),(WA2   ,A23(  2)),(WA22  ,A23(  3)), -
     1            (WA3   ,A23(  4)),(WAR2  ,A23(  5)),(WAR2M ,A23(  6)), -
     2            (WAR22 ,A23(  7)),(WAR22M,A23(  8)),(WBLHT ,A23(  9)), -
     3            (WBLLT ,A23( 10)),(WBLOV ,A23( 11)),(WF4   ,A23( 12)),-
     4            (WF7   ,A23( 13)),(WG4   ,A23( 14)),(WG41  ,A23( 15)), -
     5            (WG6   ,A23( 16)),(WG7   ,A23( 17)),(WG7M  ,A23( 18)), -
     6            (WP4   ,A23( 19)),(WP41  ,A23( 20))
      EQUIVALENCE (X3    ,A24(  1)),(X4    ,A24(  2)),(X5    ,A24(  3)), -
     1            (X6    ,A24(  4)),(XF    ,A24(  5)),(XMN   ,A24(  6)), -
     2            (XMNM  ,A24(  7)),(XNH   ,A24(  8)),(XNL   ,A24(  9))
      EQUIVALENCE (Y3    ,A25(  1)),(Y4    ,A25(  2)),(Y5    ,A25(  3)),-
     1            (Y6    ,A25(  4))
      CALL CPUTIM(ITIM1)
C******SPECIFY I/O DEVICES AND PUNCH OPTION
      IR=5
      IW=6
      IPNCH=7
      READ(IR,500)JP,JPD,JPA
      WRITE(IW,501)
      WRITE(IW,500) JP,JPD,JPA
C******INLET OPTION AND READ MAP DATA
      READ(IR,500) INLET
      WRITE(IW,502)
      WRITE(IW,503)
      WRITE(IW,500)INLET
      READ(IR,504) KBLWHT,KBLWLT
      WRITE(IW,502)
      WRITE(IW,505)
      WRITE(IW,506) KBLWHT,KBLWLT
      WRITE(IW,502)
      CALL MAPIN
C******READ OPERATING POINT DATA (IP=1 DRY DESIGN,IP=NDRY+1 WET DESIGN)
      READ(IR,508) NDRY,NAUG
      NTOTAL=NDRY+NAUG
      DO 100   IP=1,NTOTAL
      READ(IR,507) PLA
      READ(IR,504) P0,P2,P13,P22,P3,P4,P41,P5,P6,P7,TAM,T2,T13,T22,T3,  -
     1 T4,T41,T6,T7,WA2,WA13,WA22,WA3,WG4,WG41,WG6,WG7,DH4,DH41,ETAB,   -
     2 ETAAB,FN,XNL,XNH,WF4,WF7,A8,AE,ALT,XMN,CDN,CVN,CIVV,RCVV,FG
      IPRINT=0
      CALL PRINT(IPRINT)
      IF (IP .NE. 1) GO TO 10
      KBH=1
      KBL=1
      KBV=1
   10 WBLHT=WG41-WG4
      WBLLT=WG6-WG41-WA13
      IF(KBH.EQ.0) WBLHT=0.
      IF(KBL.EQ.0) WBLLT=0.
      IF(IP.NE.1) GO TO 20
      IF(WBLHT.GT.0.) GO TO 15
      KBH=0
   15 IF(WBLLT.GT.0.) GO TO 20
      KBL=0
   20 WBLOV=WA22-WA3-WBLHT-WBLLT
      IF(KBV.EQ.0) WBLOV=0.
      IF(IP.NE.1) GO TO 25
      IF(WBLOV.GT.0.) GO TO 25
      KBV=0
   25 WBLHT=WBLHT/(.2*SFWA22)
      WBLLT=WBLLT/(.02*SFWA22)
      WBLOV=WBLOV/(.002*SFWA22)
   30 P0= P0/SFP0
      P2= P2/SFP2
      P13= P13/SFP13
      P22= P22/SFP22
      P3= P3/SFP3
      P4=P4/SFP4
      P41=P41/SFP41
      P5=P5/SFP5
      P6=P6/SFP6
      P7=P7/SFP7
      TAM=TAM/1000.
```

```
      T2=T2/SFT2
      RTT2=SQRT(T2)
      T13=T13/SFT13
      T22=T22/SFT22
      RTT22=SQRT(T22)
      T3=T3/SFT3
      T4=T4/SFT4
      RTT4=SQRT(T4)
      T41=T41/SFT41
      RTT41=SQRT(T41)
      T6=T6/SFT6
      T7=T7/SFT7
      WA2=WA2/SFWA2
      WA13=WA13/SFWA13
      WA22 = WA22/SFWA22
      WA3  = WA3/SFWA3
      DH4=DH4/SFDH4
      DH41=DH41/SFDH41
      WG4  = WG4/SFWG4
      WG41 = WG41/SFWG41
      WG6  = WG6/SFWG6
      WG7  = WG7/SFWG7
      FN   =FN/SFFN
      FG = FG/SFFG
      XNL  =XNL/SFXNL
      XNH  =XNH/SFXNH
      WF4  = WF4/SFWF4
      WF7  = WF7/SFWF7
      A8   = A8/SFA8
      AE   = AE/SFAE
      ALT  = ALT/SFALT
      XMN  = XMN/SFXMN
      CIVV  = -(CIVV-BSCIVV)/SFCIVV
      RCVV  = -(RCVV-BSRCVV)/SFRCVV
      IF ((IP .EQ. 1) .OR. (IP .EQ. NDRY+1)) CALL DCOEF
      CALL ENGINE
      APRINT=0
      CALL ANALOG(APRINT)
      CALL PRINT(IPRINT)
  100 CONTINUE
      CALL ANALOG(APRINT)
      CALL PRINT(IPRINT)
      CALL CPUTIM(ITIM2)
      TIME1 = ITIM1
      TIME2 = ITIM2
      TOTAL = .001*(TIME2 - TIME1)
      WRITE(2,600) TOTAL
  500 FORMAT((1X,8(I1,2X)))
  501 FORMAT(3X,37HPUNCH OPTION OVERALL, DIGITAL, ANALOG/)
  502 FORMAT(/)
  503 FORMAT(3X,12HINLET OPTION/)
  504 FORMAT(5F12.5)
  505 FORMAT(3X,36HFRACTION OF BLEED THAT PROVIDES WORK/,3X,25HHIGH TURBINE
      LOW TURBINE/)
  506 FORMAT(3X,2F12.5)
  507 FORMAT(9X,I3)
  508 FORMAT((1X,2(I2,2X)))
  600 FORMAT(/,' TOTAL CPU TIME IS: ',G12.5,' SECONDS',/)
      STOP
      END
C
C
C**********ANALOG*********
C
C
      SUBROUTINE ANALOG(APRINT)
C     THIS ROUTINE READS ALL ANALOG COMPONENT INFORMATION
C                   CALCULATES POT SETTINGS AND INTEGRATOR
C                   GAINS AND PRINTS THE NECESSARY ANALOG
C                   INFORMATION
      DIMENSION POT(5)
      INTEGER PLA,AADR,GAIN(10),APRINT,KBLWHT,KBLWLT
      COMMON /IO/IR,IW,IPNCH
```

```
      COMMON /IVARS/ IP,JP,JPA,JPD,KBH,KBL,KBV,NAUG,NDRY,NTOTAL,PLA
      COMMON /RVARS/ CBLHR(50),CBLLR(50),CBLVR(50),DH4QR(50),DH4TR(50),-
     A               DH41QR(50),DH41TR(50),DP13R(50),DP6R(50),ETAABR(50),-
     B               ETABR(50),FGR(50),P0R(50),P2R(50),P22R(50),-
     C               P5R(50),T13PR(50),T2R(50),T22R(50),T3PR(50),-
     D               WA2R(50),WA22R(50),WA3R(50),WG4R(50),WG41R(50),-
     E               WG7R(50)
      COMMON /AVARS/ CC(50),DACI(24),DC(125),XC(6),YC(6),ZC(12),UDC(125)
      COMMON /SVARS/ BSCIVV,BSRCVV,SFA8,SFAE,SFALT,SFCIVV,SFDH4,SFDH41,-
     A               SFFG,SFFN,SFP0,SFP13,SFP2,SFP22,SFP3,SFP4,SFP41,-
     B               SFP5,SFP6,SFP7,SFRCVV,SFT0,SFT13,SFT2,SFT22,SFT3,-
     C               SFT4,SFT41,SFT6,SFT7,SFVEL,SFWA13,SFWA2,SFWA22,-
     D               SFWA3,SFWF4,SFWF7,SFWG4,SFWG41,SFWG6,SFWG7,SFXMN, -
     E               SFXNH,SFXNL
      COMMON /ANVARS/ AQL13,AQL6,SFT,SFW13,SFW3,SFW4,SFW41,SFW6,SFW7,     -
     1                W13,W3,W4,W41,W6,W7,SW13,SW3,SW4,SW41,SW6,SW7,      -
     A                V13,V3,V4,V41,V6,V7,XIH,XIL,PADR(53),PVAL(53),      -
     B                IG(10),AADR(16)
      COMMON /XVARS/ A1(50),A2(50),A3(50),A4(50),A5(50),A6(50),A7(50),  -
     1               A81(50),A9(50),A10(50),A11(50),A12(50),A13(50),     -
     2               A14(50),A15(50),A16(50),A17(50),A18(50),A19(50),   -
     3               A20(50),A21(50),A22(50),A23(50),A24(50),A25(50),  -
     4               A26(50)
      EQUIVALENCE (A8     , A1(  1)),(AE     , A1(  2)),(ALT    , A1(  3)),  -
     1            (ALTM   , A1(  4))
      EQUIVALENCE (CD7    , A3(  1)),(CDN    , A3(  2)),(CIVV   , A3(  3)),  -
     2            (CP13   , A3(  4)),(CP13P  , A3(  5)),(CP2    , A3(  6)),  -
     2            (CP22   , A3(  7)),(CP3    , A3(  8)),(CP3P   , A3(  9)),  -
     3            (CP4    , A3( 10)),(CP41   , A3( 11)),(CP6    , A3( 12)),  -
     4            (CP7    , A3( 13)),(CPAB   , A3( 14)),(CPB    , A3( 15)),  -
     5            (CPHC   , A3( 16)),(CSHIFT, A3( 17)),(CV13   , A3( 18))
      EQUIVALENCE (CV13P  , A3( 19)),(CV2    , A3( 20)),(CV22   , A3( 21)), -
     7            (CV3    , A3( 22)),(CV3P   , A3( 23)),(CV4    , A3( 24)), -
     8            (CV41   , A3( 25)),(CV6    , A3( 26)),(CV7    , A3( 27)), -
     9            (CV8    , A3( 28)),(CVAB   , A3( 29)),(CVB    , A3( 30)), -
     A            (CVHC   , A3( 31)),(CVN    , A3( 32))
      EQUIVALENCE (DH4    , A4(  1)),(DH41   , A4(  2)),(DTQW13, A4(  3)), -
     1            (DTQW3  , A4(  4)),(DTQW4  , A4(  5)),(DTQW41, A4(  6)), -
     2            (DTQW6  , A4(  7)),(DTQW7  , A4(  8)),(DW13   , A4(  9)), -
     3            (DW3    , A4( 10)),(DW4    , A4( 11)),(DW41   , A4( 12)), -
     4            (DW6    , A4( 13)),(DW7    , A4( 14)),(DXNH   , A4( 15)), -
     5            (DXNL   , A4( 16))
      EQUIVALENCE (ETAAB  , A5(  1)),(ETAB   , A5(  2)),(ETAHCM, A5(  3)), -
     1            (ETAIFM, A5(  4)),(ETAOFM, A5(  5))
      EQUIVALENCE (FAR4M  , A6(  1)),(FAR41M, A6(  2)),(FAR6M  , A6(  3)), -
     1            (FAR7M  , A6(  4)),(FG     , A6(  5)),(FGM3   , A6(  6)), -
     2            (FGPT3  , A6(  7)),(FN     , A6(  8)),(FNET   , A6(  9)), -
     3            (FNM    , A6( 10)),(FSHIFT, A6( 11))
      EQUIVALENCE (GM13   , A7(  1)),(GM13M  , A7(  2)),(GM13P  , A7(  3)), -
     1            (GM2    , A7(  4)),(GM22   , A7(  5)),(GM3    , A7(  6)), -
     2            (GM3M   , A7(  7)),(GM3P   , A7(  8)),(GM4    , A7(  9)), -
     3            (GM41   , A7( 10)),(GM4M   , A7( 11)),(GM41M  , A7( 12)), -
     4            (GM6    , A7( 13)),(GM6M   , A7( 14)),(GM7    , A7( 15)), -
     5            (GM7M   , A7( 16)),(GMAB   , A7( 17)),(GMB    , A7( 18)), -
     6            (GMHC   , A7( 19))
      EQUIVALENCE (H13    ,A81(  1)),(H13M   ,A81(  2)),(H13P   ,A81(  3)), -
     1            (H13PM  ,A81(  4)),(H2     ,A81(  5)),(H22    ,A81(  6)), -
     2            (H2M    ,A81(  7)),(H22M   ,A81(  8)),(H3     ,A81(  9)), -
     3            (H3M    ,A81( 10)),(H3P    ,A81( 11)),(H3PM   ,A81( 12)), -
     4            (H4     ,A81( 13)),(H41    ,A81( 14)),(H4M    ,A81( 15)), -
     5            (H41M   ,A81( 16)),(H6     ,A81( 17)),(H6M    ,A81( 18)), -
     6            (H7     ,A81( 19)),(H7M    ,A81( 20)),(HAB    ,A81( 21)), -
     7            (HABM   ,A81( 22)),(HB     ,A81( 23)),(HBM    ,A81( 24)), -
     8            (HHCM   ,A81( 25)),(HP4    ,A81( 26)),(HP41   ,A81( 27))
      EQUIVALENCE (KBLWHT,A11(  1)),(KBLWLT,A11(  2))
      EQUIVALENCE (PE     ,A16(  1)),(P0     ,A16(  2)),(P0A    ,A16(  3)), -
     1            (P0QT7  ,A16(  4)),(P13    ,A16(  5)),(P2     ,A16(  6)), -
     2            (P2A    ,A16(  7)),(P22    ,A16(  8)),(P22Q2M,A16(  9)), -
     3            (P3     ,A16( 10)),(P4     ,A16( 11)),(P41    ,A16( 12)), -
     4            (P5     ,A16( 13)),(P6     ,A16( 14)),(P7     ,A16( 15)), -
     5            (PRHC   ,A16( 16)),(PRIF   ,A16( 17)),(PROF   ,A16( 18))
      EQUIVALENCE (RCVV   ,A18(  1)),(RTT2   ,A18(  2)),(RTT22  ,A18(  3)), -
     1            (RTT4   ,A18(  4)),(RTT41  ,A18(  5))
```

```
      EQUIVALENCE (T0A   ,A20(  1)),(T13   ,A20(  2)),(T13M  ,A20(  3)), -
     1            (T13P  ,A20(  4)),(T13PM ,A20(  5)),(T2    ,A20(  6)), -
     2            (T2A   ,A20(  7)),(T2M   ,A20(  8)),(T22   ,A20(  9)),-
     3            (T22M  ,A20( 10)),(T3    ,A20( 11)),(T3M   ,A20( 12)),-
     4            (T3P   ,A20( 13)),(T3PM  ,A20( 14)),(T4    ,A20( 15)), -
     5            (T4M   ,A20( 16)),(T41   ,A20( 17)),(T41M  ,A20( 18)), -
     6            (T6    ,A20( 19)),(T6M   ,A20( 20)),(T7    ,A20( 21)), -
     7            (T7M   ,A20( 22)),(TAM   ,A20( 23)),(TAVAB ,A20( 24)), -
     8            (TAVB  ,A20( 25)),(TAVHC ,A20( 26)),(TRHCM1,A20( 27)), -
     9            (TRIFM1,A20( 28)),(TROFM1,A20( 29))
      EQUIVALENCE (WA13  ,A23(  1)),(WA2   ,A23(  2)),(WA22  ,A23(  3)), -
     1            (WA3   ,A23(  4)),(WAR2  ,A23(  5)),(WAR2M ,A23(  6)), -
     2            (WAR22 ,A23(  7)),(WAR22M,A23(  8)),(WBLHT ,A23(  9)), -
     3            (WBLLT ,A23( 10)),(WBLOV ,A23( 11)),(WF4   ,A23( 12)),-
     4            (WF7   ,A23( 13)),(WG4   ,A23( 14)),(WG41  ,A23( 15)), -
     5            (WG6   ,A23( 16)),(WG7   ,A23( 17)),(WG7M  ,A23( 18)), -
     6            (WP4   ,A23( 19)),(WP41  ,A23( 20))
      EQUIVALENCE (X3    ,A24(  1)),(X4    ,A24(  2)),(X5    ,A24(  3)), -
     1            (X6    ,A24(  4)),(XF    ,A24(  5)),(XMN   ,A24(  6)), -
     2            (XMNM  ,A24(  7)),(XNH   ,A24(  8)),(XNL   ,A24(  9))
      EQUIVALENCE (Y3    ,A25(  1)),(Y4    ,A25(  2)),(Y5    ,A25(  3)),-
     1            (Y6    ,A25(  4))
C******ANALOG SETUP
      APRINT=APRINT+1
      GO TO (10,100),APRINT
   10 IF (IP .NE. 1) GO TO 30
   20 READ(IR,502)SFT
      READ(IR,503) V13,V3,V4,V41,V6,V7,AQL13,AQL6,XIH,XIL
      READ(IR,504)(PADR(I),I=1,53)
      READ(IR,505)(AADR(I),I=1,16)
      CALL VOLUME(P13,T13,V13,W13,SW13,SFT,SFWA2,SFP13,SFT13,IK,POT,SFW13)
      PVAL(1)=POT(1)
      PVAL(2)=POT(2)
      PVAL(3)=POT(3)
      PVAL(4)=POT(4)
      PVAL(5)=POT(5)
      IG(1)=IK
      CALL VOLUME(P3,T3,V3,W3,SW3,SFT,SFWA22,SFP3,SFT3,IK,POT,SFW3)
      PVAL(6)=POT(1)
      PVAL(7)=POT(2)
      PVAL(8)=POT(3)
      PVAL(9)=POT(4)
      PVAL(10)=POT(5)
      IG(2)=IK
      CALL VOLUME(P4,T4,V4,W4,SW4,SFT,SFWG4,SFP4,SFT4,IK,POT,SFW4)
      PVAL(11)=POT(1)
      PVAL(12)=POT(2)
      PVAL(13)=POT(3)
      PVAL(14)=POT(4)
      PVAL(15)=POT(5)
      IG(3)=IK
      CALL VOLUME(P41,T41,V41,W41,SW41,SFT,SFWG41,SFP41,SFT41,IK,POT,SFW41)
      PVAL(16)=POT(1)
      PVAL(17)=POT(2)
      PVAL(18)=POT(3)
      PVAL(19)=POT(4)
      PVAL(20)=POT(5)
      IG(4)=IK
      CALL VOLUME(P6,T6,V6,W6,SW6,SFT,SFWG6,SFP6,SFT6,IK,POT,SFW6)
      PVAL(21)=POT(1)
      PVAL(22)=POT(2)
      PVAL(23)=POT(3)
      PVAL(24)=POT(4)
      PVAL(25)=POT(5)
      IG(5)=IK
      CALL VOLUME(P7,T7,V7,W7,SW7,SFT,SFWG7,SFP7,SFT7,IK,POT,SFW7)
      PVAL(26)=POT(1)
      PVAL(27)=POT(2)
      PVAL(28)=POT(3)
      PVAL(29)=POT(4)
      PVAL(30)=POT(5)
      IG(6)=IK
      CALL SPOOL(XIH,XNH,SFWG4,SFDH4,SFXNH,SFT,IK,POT)
```

```
      PVAL(31)=POT(1)
      PVAL(32)=POT(2)
      IG(7)=IK
      CALL SPOOL(XIL,XNL,SFWG41,SFDH41,SFXNL,SFT,IK,POT)
      PVAL(33)=POT(1)
      PVAL(34)=POT(2)
      IG(8)=IK
   30 CALL DUCT(AQL13,V13,SFP13,SFP6,SFT13,SFWA13,SFT,P13,P6,T13,-
     1 WA13,POT,IK,SFW13)
      IF(IP.NE.1) GO TO 31
      PVAL(35)=POT(1)
      PVAL(36)=POT(2)
      PVAL(37)=POT(3)
      PVAL(38)=POT(4)
      IG(9)=IK
      DP13R(IP)=1.
      GO TO 40
   31 DP13R(IP)=POT(4)/PVAL(38)
   40 CALL DUCT(AQL6,V6,SFP6,SFP7,SFT6,SFWG6,SFT,P6,P7,T6,WG6,POT,IK, -
     1 SFW6)
      IF(IP.NE.1)  GO TO 41
      PVAL(39)=POT(1)
      PVAL(40)=POT(2)
      PVAL(41)=POT(3)
      PVAL(42)=POT(4)
      IG(10)=IK
      DP6R(IP)=1.
      GO TO 50
   41 DP6R(IP)=POT(4)/PVAL(42)
   50 IF(IP.NE.1) GO TO 60
      PVAL(43)=WF4
      PVAL(44)=WF7
      PVAL(45)=A8
      PVAL(46)=AE
      PVAL(47)=CIVV
      PVAL(48)=RCVV
      PVAL(49)=ALT
      PVAL(50)=XMN
      PVAL(51)=TAM
      PVAL(52)=(P2*SFP2)/(P13*SFP13*XC(3))
      PVAL(53)=(P22*SFP22)/(P3*SFP3*XC(4))
   60 RETURN
  100 CONTINUE
C*****PRINT OUT OF ANALOG SPECIFICATIONS
      WRITE(IW,500)
      WRITE(IW,506)
      WRITE(IW,502)SFT
      WRITE(IW,501)
      WRITE(IW,507)
      WRITE(IW,508)
      WRITE(IW,509)
      WRITE(IW,503) V13,V3,V4,V41,V6,V7,AQL13,AQL6,XIH,XIL
      WRITE(IW,501)
      WRITE(IW,510)
      WRITE(IW,504)(PADR(I),I=1,53)
      WRITE(IW,501)
      WRITE(IW,511)
      WRITE(IW,505) (AADR(I),I=1,16)
      WRITE(IW,501)
      WRITE(IW,512)
      WRITE(IW,524)
      DO 105 I=1,53
      IF(PVAL(I).LE.0.) PVAL(I)=.0001
      IF(PVAL(I).GE.1.0) PVAL(I)=.9999
      WRITE(IW,513) PADR(I),PVAL(I)
  105 CONTINUE
      WRITE(IW,501)
      WRITE(IW,525)
      DO 110 K=1,10
      N=IG(K)
      GAIN(K)=10**N
      IF(K.GT.6) GO TO 106
      J=2*K
```

```
      JJ=J-1
      WRITE(IW,514) AADR(JJ),GAIN(K)
      WRITE(IW,514) AADR(J),GAIN(K)
      GO TO 107
  106 J=K+6
      WRITE(IW,514) AADR(J),GAIN(K)
  107 CONTINUE
  110 CONTINUE
      WRITE(IW,500)
      WRITE(IW,515)
      DO 120  I=22,24
      DACI(I)=0.
  120 CONTINUE
      WRITE(IW,516)(DACI(I),I=1,24)
      WRITE(IW,501)
      WRITE(IW,517)
      WRITE(IW,518) W13,SFW13,SW13
      WRITE(IW,519) W3,SFW3,SW3
      WRITE(IW,520) W4,SFW4,SW4
      WRITE(IW,521) W41,SFW41,SW41
      WRITE(IW,522) W6,SFW6,SW6
      WRITE(IW,523) W7,SFW7,SW7
      WRITE(IW,501)
  200 CONTINUE
  500 FORMAT(1H1)
  501 FORMAT(/)
  502 FORMAT(1X,F5.0)
  503 FORMAT(6F12.5)
  504 FORMAT((1X,8(A4,1X)))
  505 FORMAT(8I4)
  506 FORMAT(3X,17HANALOG TIME SCALE/)
  507 FORMAT(3X,21HENGINE GEOMETRIC DATA/)
  508 FORMAT(6X,3HV13,9X,2HV3,10X,2HV4,10X,3HV41,9X,2HV6,10X,2HV7)
  509 FORMAT(6X,5HAQL13,7X,4HAQL6,8X,3HXIH,9X,3HXIL/)
  510 FORMAT(3X,20HANALOG POT ADDRESSES/)
  511 FORMAT(3X,27HANALOG INTEGRATOR ADDRESSES/)
  512 FORMAT(3X,33HANALOG TIME SCALE SHOULD BE F-SEC/)
  513 FORMAT(3X,4HPOT ,A4,8H SET TO F7.4)
  514 FORMAT(3X,11HINTEGRATOR ,I4,3X,15HGAIN SHOULD BE  ,I6)
  515 FORMAT(3X,38HDACS INITIALIZED WITH FOLLOWING VALUES/)
  516 FORMAT(5(F8.5,2X))
  517 FORMAT(3X,16HSTORED MASS DATA/)
  518 FORMAT(3X,6HW13 = ,F8.4,3X,8HSFW13 = ,F6.1,3X,7HSW13 = ,F8.4)
  519 FORMAT(3X,6HW3  = ,F8.4,3X,8HSFW3  = ,F6.1,3X,7HSW3  = ,F8.4)
  520 FORMAT(3X,6HW4  = ,F8.4,3X,8HSFW4  = ,F6.1,3X,7HSW4  = ,F8.4)
  521 FORMAT(3X,6HW41 = ,F8.4,3X,8HSFW41 = ,F6.1,3X,7HSW41 = ,F8.4)
  522 FORMAT(3X,6HW6  = ,F8.4,3X,8HSFW6  = ,F6.1,3X,7HSW6  = ,F8.4)
  523 FORMAT(3X,6HW7  = ,F8.4,3X,8HSFW7  = ,F6.1,3X,7HSW7  = ,F8.4)
  524 FORMAT(3X,12HPOT SETTINGS/)
   525 FORMAT(3X,16HINTEGRATOR GAINS/)
      RETURN
      END
C
C
C***********DATAIN*********
C
C
      SUBROUTINE DATAIN(N,F)
C     MAP DATA INPUT ROUTINE
      COMMON /IO/ IR,IW,IPNCH
      COMMON /IVARS/ IP,JP,JPA,JPD,KBH,KBL,KBV,NAUG,NDRY,NTOTAL,PLA
      DIMENSION F(1),N(1)
      DIMENSION VALS(25),XVALS(25),ZSC(4)
      DIMENSION ISC(4),IX(4),IY(4),IZ1(4),IZ2(4),IZ3(4),IZ4(4)
  500 FORMAT(1H1,/)
  501 FORMAT(5I3)
  502 FORMAT(7(4A2))
  503 FORMAT(1X,7(4A2))
      KPNCH=0
      IF((JP.EQ.1).OR.(JPD.EQ.1)) KPNCH=1
      K=1
      JF=0
    5 READ(IR,501) M,NCV,NPT,NFCT,NCOM
```

```
      IF(M.EQ.0) RETURN
      WRITE(IW,500)
      WRITE(IW,501) M,NCV,NPT,NFCT,NCOM
      IF(KPNCH.EQ.1) WRITE(IPNCH,501) M,NCV,NPT,NFCT,NCOM
      INC=NCV*NPT
      N(K)=M
      N(K+1)=1
      N(K+2)=1
      N(K+3)=NPT
      N(K+4)=NCV
      K=K+5
      READ(IR,502) (ISC(I),I=1,4),(IX(I),I=1,4),
     A             (IY(I),I=1,4),(IZ1(I),I=1,4),
     1             (IZ2(I),I=1,4),(IZ3(I),I=1,4),(IZ4(I),I=1,4)
      WRITE(IW,503) (ISC(I),I=1,4),(IX(I),I=1,4),(IY(I),I=1,4),
     *    (IZ1(I),I=1,4),(IZ2(I),I=1,4),(IZ3(I),I=1,4),(IZ4(I),I=1,4)
      IF(KPNCH.EQ.1) WRITE(IPNCH,502) (ISC(I),I=1,4),(IX(I),I=1,4),(IY(I),I=1,4),
     *    (IZ1(I),I=1,4),(IZ2(I),I=1,4),(IZ3(I),I=1,4),(IZ4(I),I=1,4)
      READ(IR,ISC) XSC,YSC,(ZSC(I),I=1,NFCT)
      WRITE(IW,ISC) XSC,YSC,(ZSC(I),I=1,NFCT)
      IF(KPNCH.EQ.1) WRITE(IPNCH,ISC) XSC,YSC,(ZSC(I),I=1,NFCT)
      READ(IR,IY) (VALS(I),I=1,NCV)
      WRITE(IW,IY) (VALS(I),I=1,NCV)
      IF(KPNCH.EQ.1) WRITE(IPNCH,IY) (VALS(I),I=1,NCV)
      JS=JF+1
      JF=JF+NCV
      I=1
      DO 10 J=JS,JF
      F(J)=VALS(I)/YSC
C     CHECK CONSECUTIVE VALUES OF Y
      IF(J.NE.JS) TEST=F(J)-F(J-1)
   10 I=I+1
      JS=JF+1
      JF=JF+NPT
      DO 90 L=1,NCV
      IF(NCOM.EQ.0) GO TO 14
      IF(NCOM.EQ.32767) GO TO 16
      NCOM=32767
   14 READ(IR,IX) (XVALS(I),I=1,NPT)
      WRITE(IW,IX) (XVALS(I),I=1,NPT)
      IF(KPNCH.EQ.1) WRITE(IPNCH,IX) (XVALS(I),I=1,NPT)
   16 I=1
      DO 20 J=JS,JF
      F(J)=XVALS(I)/XSC
C     CHECK CONSECUTIVE VALUES OF X
      IF(J.NE.JS) TEST=F(J)-F(J-1)
      IF(L.EQ.1) GO TO 20
      JMNPT=J-NPT
      TEST=F(J)-F(JMNPT)
   20 I=I+1
      DO 80 NF=1,NFCT
      GO TO (30,40,50,60),NF
   30 READ(IR,IZ1) (VALS(I),I=1,NPT)
      WRITE(IW,IZ1) (VALS(I),I=1,NPT)
      IF(KPNCH.EQ.1) WRITE(IPNCH,IZ1) (VALS(I),I=1,NPT)
      GO TO 70
   40 READ(IR,IZ2) (VALS(I),I=1,NPT)
      WRITE(IW,IZ2) (VALS(I),I=1,NPT)
      IF(KPNCH.EQ.1) WRITE(IPNCH,IZ2) (VALS(I),I=1,NPT)
      GO TO 70
   50 READ(IR,IZ3) (VALS(I),I=1,NPT)
      WRITE(IW,IZ3) (VALS(I),I=1,NPT)
      IF(KPNCH.EQ.1) WRITE(IPNCH,IZ3) (VALS(I),I=1,NPT)
      GO TO 70
   60 READ(IR,IZ4) (VALS(I),I=1,NPT)
      WRITE(IW,IZ4) (VALS(I),I=1,NPT)
      IF(KPNCH.EQ.1) WRITE(IPNCH,IZ4) (VALS(I),I=1,NPT)
   70 JS=JS+INC
      JF=JF+INC
      I=1
      DO 80 J=JS,JF
      F(J)=VALS(I)/ZSC(NF)
C     CHECK CONSECUTIVE VALUES OF Z
```

```
      IF(J.NE.JS) TEST=F(J)-F(J-1)
      IF(L.EQ.1) GO TO 80
      JMNPT=J-NPT
      TEST=F(J)-F(JMNPT)
   80 I=I+1
      IF(L.EQ.NCV) GO TO 5
      JS=JS+NPT-INC*NFCT
   90 JF=JS+NPT-1
      RETURN
      END
C
C
C***********DCOEF**********
C
C
      SUBROUTINE DCOEF
C     THIS ROUTINE CALCULATES TRIMMED DIGITAL COEFFICIENTS
      REAL MAP,MAPL,KBLWHT,KBLWLT
      INTEGER PLA
      COMMON /IO/IR,IW,IPNCH
      COMMON /IVARS/ IP,JP,JPA,JPD,KBH,KBL,KBV,NAUG,NDRY,NTOTAL,PLA
      COMMON /NMAPS/F1(322),F2(322),F3(854),F4(518),F5(224),F6(224),    -
     1             N1(5),N2(5),N3(5),N4(5),N5(5),N6(5)
      COMMON /AVARS/ CC(50),DACI(24),DC(125),XC(6),YC(6),ZC(12),UDC(125)
      COMMON /SVARS/ BSCIVV,BSRCVV,SFA8,SFAE,SFALT,SFCIVV,SFDH4,SFDH41,-
     A               SFFG,SFFN,SFP0,SFP13,SFP2,SFP22,SFP3,SFP4,SFP41,-
     B               SFP5,SFP6,SFP7,SFRCVV,SFT0,SFT13,SFT2,SFT22,SFT3,-
     C               SFT4,SFT41,SFT6,SFT7,SFVEL,SFWA13,SFWA2,SFWA22,-
     D               SFWA3,SFWF4,SFWF7,SFWG4,SFWG41,SFWG6,SFWG7,SFXMN, -
     E               SFXNH,SFXNL
      COMMON /XVARS/ A1(50),A2(50),A3(50),A4(50),A5(50),A6(50),A7(50),  -
     1               A81(50),A9(50),A10(50),A11(50),A12(50),A13(50),    -
     2               A14(50),A15(50),A16(50),A17(50),A18(50),A19(50),   -
     3               A20(50),A21(50),A22(50),A23(50),A24(50),A25(50),   -
     4               A26(50)
      EQUIVALENCE (A8     , A1(  1)),(AE     , A1(  2)),(ALT    , A1(  3)),  -
     1            (ALTM   , A1(  4))
      EQUIVALENCE (CD7    , A3(  1)),(CDN    , A3(  2)),(CIVV   , A3(  3)),  -
     2            (CP13   , A3(  4)),(CP13P  , A3(  5)),(CP2    , A3(  6)),  -
     2            (CP22   , A3(  7)),(CP3    , A3(  8)),(CP3P   , A3(  9)),  -
     3            (CP4    , A3( 10)),(CP41   , A3( 11)),(CP6    , A3( 12)),  -
     4            (CP7    , A3( 13)),(CPAB   , A3( 14)),(CPB    , A3( 15)),  -
     5            (CPHC   , A3( 16)),(CSHIFT, A3( 17)),(CV13   , A3( 18))
      EQUIVALENCE (CV13P  , A3( 19)),(CV2    , A3( 20)),(CV22   , A3( 21)), -
     7            (CV3    , A3( 22)),(CV3P   , A3( 23)),(CV4    , A3( 24)), -
     8            (CV41   , A3( 25)),(CV6    , A3( 26)),(CV7    , A3( 27)), -
     9            (CV8    , A3( 28)),(CVAB   , A3( 29)),(CVB    , A3( 30)), -
     A            (CVHC   , A3( 31)),(CVN    , A3( 32))
      EQUIVALENCE (DH4    , A4(  1)),(DH41   , A4(  2)),(DTQW13, A4(  3)), -
     1            (DTQW3  , A4(  4)),(DTQW4  , A4(  5)),(DTQW41, A4(  6)), -
     2            (DTQW6  , A4(  7)),(DTQW7  , A4(  8)),(DW13   , A4(  9)), -
     3            (DW3    , A4( 10)),(DW4    , A4( 11)),(DW41   , A4( 12)), -
     4            (DW6    , A4( 13)),(DW7    , A4( 14)),(DXNH   , A4( 15)), -
     5            (DXNL   , A4( 16))
      EQUIVALENCE (ETAAB  , A5(  1)),(ETAB   , A5(  2)),(ETAHCM, A5(  3)), -
     1            (ETAIFM, A5(  4)),(ETAOFM, A5(  5))
      EQUIVALENCE (FAR4M  , A6(  1)),(FAR41M, A6(  2)),(FAR6M  , A6(  3)), -
     1            (FAR7M  , A6(  4)),(FG     , A6(  5)),(FGM3   , A6(  6)), -
     2            (FGPT3  , A6(  7)),(FN     , A6(  8)),(FNET   , A6(  9)), -
     3            (FNM    , A6( 10)),(FSHIFT, A6( 11))
      EQUIVALENCE (GM13   , A7(  1)),(GM13M  , A7(  2)),(GM13P  , A7(  3)), -
     1            (GM2    , A7(  4)),(GM22   , A7(  5)),(GM3    , A7(  6)), -
     2            (GM3M   , A7(  7)),(GM3P   , A7(  8)),(GM4    , A7(  9)), -
     3            (GM41   , A7( 10)),(GM4M   , A7( 11)),(GM41M  , A7( 12)), -
     4            (GM6    , A7( 13)),(GM6M   , A7( 14)),(GM7    , A7( 15)), -
     5            (GM7M   , A7( 16)),(GMAB   , A7( 17)),(GMB    , A7( 18)), -
     6            (GMHC   , A7( 19))
      EQUIVALENCE (H13    ,A81(  1)),(H13M   ,A81(  2)),(H13P   ,A81(  3)), -
     1            (H13PM  ,A81(  4)),(H2     ,A81(  5)),(H22    ,A81(  6)), -
     2            (H2M    ,A81(  7)),(H22M   ,A81(  8)),(H3     ,A81(  9)), -
     3            (H3M    ,A81( 10)),(H3P    ,A81( 11)),(H3PM   ,A81( 12)), -
     4            (H4     ,A81( 13)),(H41    ,A81( 14)),(H4M    ,A81( 15)), -
     5            (H41M   ,A81( 16)),(H6     ,A81( 17)),(H6M    ,A81( 18)), -
```

```
     6          (H7    ,A81( 19)),(H7M   ,A81( 20)),(HAB   ,A81( 21)), -
     7          (HABM  ,A81( 22)),(HB    ,A81( 23)),(HBM   ,A81( 24)), -
     8          (HHCM  ,A81( 25)),(HP4   ,A81( 26)),(HP41  ,A81( 27))
      EQUIVALENCE (KBLWHT,A11(  1)),(KBLWLT,A11(  2))
      EQUIVALENCE (PE    ,A16(  1)),(P0    ,A16(  2)),(P0A   ,A16(  3)), -
     1          (P0QT7 ,A16(  4)),(P13   ,A16(  5)),(P2    ,A16(  6)), -
     2          (P2A   ,A16(  7)),(P22   ,A16(  8)),(P22Q2M,A16(  9)), -
     3          (P3    ,A16( 10)),(P4    ,A16( 11)),(P41   ,A16( 12)), -
     4          (P5    ,A16( 13)),(P6    ,A16( 14)),(P7    ,A16( 15)), -
     5          (PRHC  ,A16( 16)),(PRIF  ,A16( 17)),(PROF  ,A16( 18))
      EQUIVALENCE (RCVV  ,A18(  1)),(RTT2  ,A18(  2)),(RTT22 ,A18(  3)), -
     1          (RTT4  ,A18(  4)),(RTT41 ,A18(  5))
      EQUIVALENCE (T0A   ,A20(  1)),(T13   ,A20(  2)),(T13M  ,A20(  3)), -
     1          (T13P  ,A20(  4)),(T13PM ,A20(  5)),(T2    ,A20(  6)), -
     2          (T2A   ,A20(  7)),(T2M   ,A20(  8)),(T22   ,A20(  9)),-
     3          (T22M  ,A20( 10)),(T3    ,A20( 11)),(T3M   ,A20( 12)),-
     4          (T3P   ,A20( 13)),(T3PM  ,A20( 14)),(T4    ,A20( 15)), -
     5          (T4M   ,A20( 16)),(T41   ,A20( 17)),(T41M  ,A20( 18)), -
     6          (T6    ,A20( 19)),(T6M   ,A20( 20)),(T7    ,A20( 21)), -
     7          (T7M   ,A20( 22)),(TAM   ,A20( 23)),(TAVAB ,A20( 24)), -
     8          (TAVB  ,A20( 25)),(TAVHC ,A20( 26)),(TRHCM1,A20( 27)), -
     9          (TRIFM1,A20( 28)),(TROFM1,A20( 29))
      EQUIVALENCE (WA13  ,A23(  1)),(WA2   ,A23(  2)),(WA22  ,A23(  3)), -
     1          (WA3   ,A23(  4)),(WAR2  ,A23(  5)),(WAR2M ,A23(  6)), -
     2          (WAR22 ,A23(  7)),(WAR22M,A23(  8)),(WBLHT ,A23(  9)), -
     3          (WBLLT ,A23( 10)),(WBLOV ,A23( 11)),(WF4   ,A23( 12)),-
     4          (WF7   ,A23( 13)),(WG4   ,A23( 14)),(WG41  ,A23( 15)), -
     5          (WG6   ,A23( 16)),(WG7   ,A23( 17)),(WG7M  ,A23( 18)), -
     6          (WP4   ,A23( 19)),(WP41  ,A23( 20))
      EQUIVALENCE (X3    ,A24(  1)),(X4    ,A24(  2)),(X5    ,A24(  3)), -
     1          (X6    ,A24(  4)),(XF    ,A24(  5)),(XMN   ,A24(  6)), -
     2          (XMNM  ,A24(  7)),(XNH   ,A24(  8)),(XNL   ,A24(  9))
      EQUIVALENCE (Y3    ,A25(  1)),(Y4    ,A25(  2)),(Y5    ,A25(  3)),-
     1          (Y6    ,A25(  4))
      IF(IP.NE.1) GO TO 40
      DC(57)= P2/(ZC(3)*WA2*SQRT(T2))
      DC(58)= P2/(P13* XC(3))
      DC(59)= SQRT(T2)/(XNL*YC(3))
      DC(60)= P2/(ZC(5)*P22)
      DC(61)= P22/(ZC(7)*WA22*SQRT(T22))
      DC(62)= P22/(P3*XC(4))
      DC(63)= SQRT(T22)/(XNH*YC(4))
      DC(64)= SQRT(P3*(P3-P4)/T3)/WA3
      DC(65)= (P4*XNH*.5)/(ZC(9)*WG4*T4)
      DC(66)= P4/(P41*XC(5))
      DC(67)= SQRT(T4)/(XNH*YC(5))
      DC(68)=(.2*SFT3*SFWA22)/(SFT41*SFWG41)
      DC(69)=KBLWHT
      DC(70)=(DC(69)*.2*SFWA22)/SFWG4
      DC(71)=.2*SFWA22/SFWG4
      DC(72)=.2*SFWA22/SFWG41
      DC(73)=(P41*XNL)/(ZC(11)*WG41*T41)
      DC(74)= P41/(P5*XC(6))
      DC(75)= SQRT(T41)/(XNL*YC(6))
      DC(76)= P6/P5
      DC(77)=(.02*SFT3*SFWA22)/(SFT6*SFWG6)
      DC(78)=KBLWLT
      DC(79)=(DC(78)*.02*SFWA22)/SFWG41
      DC(80)=.02*SFWA22/SFWG6
      DC(81)=.02*SFWA22/SFWA13
      DC(89)=DC(54)*DC(51)
      DC(90)=0.
      DC(91)= .25
      DC(92)= .25
      DC(93)= .5
      DC(94)= .5
      DC(95)= .5
      DC(96)= .25
      DC(110) = SFAE/(2.*SFA8)
      DC(111) = SFP0/SFP7
      DC(112) = (SFAE*SFP0)/SFFG
      DC(113) = SFVEL/(44.75 * 2. * SQRT(SFT7))
      DC(114)=.2
```

```
      DC(115)=(SQRT(SFT7)*SFWG7)/(0.53218*SFA8*SFP7)
      DC(116)=(32.17405*SFFG)/(SFWG7*SFVEL)
C******COMPUTE DIGITAL COEFFICIENTS REQUIRING LOOKUPS
C******COMPUTE CORRECTION FACTORS AT DRY DESIGN POINT
      DO 5 I=1,50
      CC(I)= 1.
    5 CONTINUE
      ALTM = DC(1)*ALT
      XMNM = DC(2)*XMN
      CALL FLCOND(ALTM,XMNM,TAM,P2A,T2A,P0A,T0A)
      CC(1)=(DC(3)*P2)/(.5*P2A)
      CC(2)=(DC(4)*T2)/(.5*T2A)
      CC(3)=(DC(5)*P0)/(.5*P0A)
      X3= 1./XC(3)
      Y3= 1./YC(3)
      WAR2M = MAP(N3,F3,X3,Y3)
      ETAOFM = MAPL(F3)
      P22Q2M = MAPL(F3)
      ETAIFM = MAPL(F3)
      FSHIFT=MAP(N1,F1,CIVV,Y3)
      WAR2=(WAR2M*(.5+.5*FSHIFT))/.5
      CC(4)= 1./(ZC(3)*WAR2)
      T2M= DC(6)*T2
      CALL PROCOM(T2M,0.,CP2,CV2,GM2,H2M)
      H2= H2M/DC(7)
      PROF =(DC(8)*P13)/P2
      CALL TRAT(1,PROF,GM2,TROFM1)
      CC(5)=(DC(96)*TROFM1)/((.2*T13/(DC(9)*T2)-.2)*ETAOFM)
      CC(6)= 1./(ZC(5)*P22Q2M)
      PRIF = (DC(10)*P22)/P2
      CALL TRAT(2,PRIF,GM2,TRIFM1)
      CC(7)= (DC(91)*TRIFM1)/((.2*T22/(DC(11)*T2)-.2)*ETAIFM)
      T22M= T22*DC(19)
      CALL PROCOM(T22M,0.,CP22,CV22,GM22,H22M)
      H22=H22M/DC(16)
      X4=1./XC(4)
      Y4=1./YC(4)
      WAR22M= MAP(N4,F4,X4,Y4)
      ETAHCM= MAPL(F4)
      CSHIFT=MAP(N2,F2,RCVV,Y4)
      WAR22=(WAR22M*(.5+CSHIFT*.5))/.5
      TRHC1D= ((.2*T3/(DC(12)*T22)-.2)*ETAHCM)/DC(92)
      CC(8)= 1./(ZC(7)*WAR22)
      PRHC= (DC(14)*P3)/P22
      BETAHC=1.
   10 DC(83)=BETAHC*DC(12)*DC(13)
      DC(84)=(1.-BETAHC)*DC(13)
      TAVHC= DC(83)*T22 + DC(84)*T3
      CALL PROCOM(TAVHC,0.,CPHC,CVHC,GMHC,HHCM)
      CALL TRAT(3,PRHC,GMHC,TRHCM1)
      IF((BETAHC.GE.1.).AND.(TRHCM1.LT.TRHC1D))   GO TO 20
      IF((BETAHC.LE.0.).AND.(TRHCM1.GT.TRHC1D))   GO TO 25
      ERTRHC= TRHCM1 - TRHC1D
      IF(ABS(ERTRHC).LE..01*TRHC1D)   GO TO 30
      IF(ERTRHC.GT.0.)  GO TO 15
      BETAHC=BETAHC + .001
      GO TO 10
   15 BETAHC=BETAHC- .001
      GO TO 10
   20 DC(83)=DC(12)*DC(13)
      DC(84)=0.
      WRITE(IW,500)
      TAVHC=DC(83)*T22
      CALL PROCOM(TAVHC,0.,CPHC,CVHC,GMHC,HHCM)
      CALL TRAT(3,PRHC,GMHC,TRHCM1)
      GO TO 30
   25 DC(83)=0.
      DC(84)=DC(13)
      WRITE(IW,501)
      TAVHC=DC(84)*T3
      CALL PROCOM(TAVHC,0.,CPHC,CVHC,GMHC,HHCM)
      CALL TRAT(3,PRHC,GMHC,TRHCM1)
   30 CC(9)= (DC(92)*TRHCM1)/((.2*T3/(DC(12)*T22)-.2)*ETAHCM)
   35 T3M=DC(13)*T3
```

```
      CALL PROCOM(T3M,0.,CP3,CV3,GM3,H3M)
      H3=H3M/DC(17)
      FGM3=-.23396*GM3*GM3+.66572*GM3+.33337
      DC(97)=.5*FGM3*P3/SQRT(T3)
      IF(KBH.GT.0) DC(98)=DC(97)/WBLHT
      IF(KBL.GT.0) DC(99)=DC(97)/WBLLT
      IF(KBV.GT.0) DC(100)=DC(97)/WBLOV
      T4M = DC(21)*T4
   40 FAR4M = (DC(20)*WF4)/WA3
      IF(IP.NE.1) GO TO 75
      CALL PROCOM(T4M,FAR4M,CP4,CV4,GM4,H4M)
      H4 = H4M*DC(22)
      HBD = (H4*WG4/DC(23)-ETAB*DC(24)*WF4/DC(23))/WA3
      BETAB= 1.
   45 DC(85)= BETAB*DC(25)*DC(21)
      DC(86)= (1.-BETAB)*DC(21)
      TAVB = DC(85)*T3 + DC(86)*T4
      CALL PROCOM(TAVB,0.,CPB,CVB,GMB,HBM)
      HB=HBM*DC(22)
      IF((BETAB.GE.1.).AND.(HB.GT.HBD))  GO TO 55
      IF((BETAB.LE.0.).AND.(HB.LT.HBD))  GO TO 60
      ERRHB= HB-HBD
      IF(ABS(ERRHB).LE..01*HBD)    GO TO 65
      IF(ERRHB.GT.0.)  GO TO 50
      BETAB= BETAB - .001
      GO TO 45
   50 BETAB = BETAB + .001
      GO TO 45
   55 DC(85)=DC(25)*DC(21)
      DC(86)=0.
      TAVB=DC(85)*T3
      CALL PROCOM(TAVB,0.,CPB,CVB,GMB,HBM)
      HB=HBM*DC(22)
      WRITE(IW,502)
      GO TO 65
   60 DC(85)=0.
      DC(86)=DC(21)
      WRITE(IW,503)
      TAVB=DC(86)*T4
      CALL PROCOM(TAVB,0.,CPB,CVB,GMB,HBM)
      HB=HBM*DC(22)
   65 CC(10)= (H4*WG4/DC(23)-HB*WA3)/(WF4*ETAB*DC(24)/DC(23))
   70 X5=1./XC(5)
      Y5=1./YC(5)
      WP4 = MAP(N5,F5,X5,Y5)
      HP4 = MAPL(F5)
      DC(87)=(HP4*SQRT(T4)*XNH)/DH4
      CC(11)= 1./(ZC(9)*WP4)
      T41M= DC(28)*T41
   75 FAR41M=(FAR4M*.5)/(.5+(.5+.04*FAR4M)*DC(71)*WBLHT/WG4)
      IF(IP.NE.1)GO TO 80
      CALL PROCOM(T41M,FAR41M,CP41,CV41,GM41,H41M)
      H41= H41M*DC(29)
      CC(12)=(WG4*H4/DC(26)-WG41*H41+DC(68)*WBLHT*H3)/
     1 (DC(27)*DH4*(WG4+DC(70)*WBLHT))
      X6 = 1./XC(6)
      Y6 = 1./YC(6)
      WP41= MAP(N6,F6,X6,Y6)
      HP41= MAPL(F6)
      DC(88)=(HP41*SQRT(T41)*XNL)/DH41
      CC(13)= 1./(ZC(11)*WP41)
      T13M= DC(34)*T13
      CALL PROCOM(T13M,0.,CP13,CV13,GM13,H13M)
      H13= H13M/DC(15)
   80 FAR6M =(FAR41M*.25)/(.25+(.5+.04*FAR41M)*DC(37)*(WA13+DC(81)* -
     1 WBLLT)/WG41)
      IF(IP.NE.1) GO TO 100
      T6M= DC(38)*T6
      CALL PROCOM(T6M,FAR6M,CP6,CV6,GM6,H6M)
      H6= H6M/DC(39)
      CC(14)= (DC(31)*WA13*H13 - WG6*H6 + DC(32)*WG41*H41+DC(77)*WBLLT* -
     1 H3)/(DC(33)*DH41*(WG41+DC(79)*WBLLT))
      FAR7M = FAR6M
      T7M = T7*DC(51)
```

```
      CALL PROCOM(T7M,FAR7M,CP7,CV7,GM7,H7M)
      H7= DC(52)*H7M
      POQT7=DC(111)*P0/P7
      CALL NOZZL (P0,P7,P0QT7,T7,A8,AE,CDN,CVN,WG7M,FNM,XF)
      CC(19)=(FG-XF)/(FNM-XF)
      FNET=FNM-XMNM*SQRT(T0A)*WA2/DC(45)
      CC(17)= WG7/WG7M
      CC(15)=WA22*(H3-DC(12)*H22)*DC(94)/((WG4+DC(70)*WBLHT)*DC(40)*     -
     1 .5*DH4)
      CC(16)=(WA2-DC(18)*WA22)*(H13-DC(9)*H2)*DC(93)/((WG41+DC(79)*      -
     1WBLLT)*.5*DC(41)*DH41)+WA22*(H22-DC(11)*H2)*DC(93)/((WG41+DC(79)* -
     2WBLLT)*DC(42)*.5*DH41)
C******STORE UNMODIFIED COEFFICIENTS
      DO 85 I=1,125
      UDC(I)=DC(I)
   85 CONTINUE
C******MODIFY THOSE COEFFICIENTS THAT REQUIRE ADJUSTMENT
   90 DC(3)= DC(3)/CC(1)
      DC(4)= DC(4)/CC(2)
      DC(5)= DC(5)/CC(3)
      DC(57)= DC(57)/CC(4)
      DC(96)= DC(96)/CC(5)
      DC(60)= DC(60)/CC(6)
      DC(91)= DC(91)/CC(7)
      DC(61)= DC(61)*CC(8)
      DC(92)= DC(92)/CC(9)
      DC(24)= DC(24)*CC(10)
      DC(65)= DC(65)/CC(11)
      DC(27)= DC(27)*CC(12)
      DC(73)= DC(73)/CC(13)
      DC(33)= DC(33)*CC(14)
      DC(94)= DC(94)/CC(15)
      DC(93)= DC(93)/CC(16)
      DC(95)= DC(95)*CC(17)
      DC(116)=DC(116)*CC(19)
C******COMPUTE CORRECTION FACTORS AT WET DESIGN POINT
   95 IF(IP.NE.NDRY+1) GO TO 150
  100 FAR7M= FAR6M + (DC(50)*(.04*FAR6M + .5)*WF7)/(.5*WG6)
  105 T7M = T7*DC(51)
      CALL PROCOM(T7M,FAR7M,CP7,CV7,GM7,H7M)
      H7= DC(52)*H7M
      HABD= (H7*WG7/DC(49)-ETAAB*DC(53)*WF7/DC(49))/WG6
      BETAAB=1.
  110 DC(89)= BETAAB*DC(54)*DC(51)
      DC(90)= (1.-BETAAB)*DC(51)
      TAVAB= DC(89)*T6 + DC(90)*T7
      CALL PROCOM(TAVAB,FAR6M,CPAB,CVAB,GMAB,HABM)
      HAB= HABM*DC(52)
      IF((BETAAB.GE.1.).AND.(HAB.GT.HABD)) GO TO 120
      IF((BETAAB.LE.0.).AND.(HAB.LT.HABD)) GO TO 125
      ERRHAB = HAB - HABD
      IF(ABS(ERRHAB).LE..01*HABD)   GO TO 130
      IF(ERRHAB.GT.0.)  GO TO 115
      BETAAB = BETAAB - .001
      GO TO 110
  115 BETAAB = BETAAB + .001
      GO TO 110
  120 DC(89)=DC(54)*DC(51)
      DC(90)=0.
      WRITE(IW,504)
      TAVAB=DC(89)*T6
      CALL PROCOM(TAVAB,FAR6M,CPAB,CVAB,GMAB,HABM)
      HAB=HABM*DC(52)
      GO TO 130
  125 DC(89)=0.
      DC(90)=DC(51)
      WRITE(IW,505)
      TAVAB=DC(90)*T7
      CALL PROCOM(TAVAB,FAR6M,CPAB,CVAB,GMAB,HABM)
      HAB=HABM*DC(52)
  130 CC(18) = (H7*WG7/DC(49)-HAB*WG6)/(WF7*DC(53)*ETAAB/DC(49))
      UDC(89)=DC(89)
```

```
      UDC(90)=DC(90)
      DC(53)= DC(53)*CC(18)
  150 RETURN
  500 FORMAT(3X,45HHP COMPRESSOR EFFICIENCY REQUIRES TAV LT TIN./)
  501 FORMAT(3X,46HHP COMPRESSOR EFFICIENCY REQUIRES TAV GT TOUT./)
  502 FORMAT(3X,38HBURNER EFFICIENCY REQUIRES TAV LT TIN./)
  503 FORMAT(3X,39HBURNER EFFICIENCY REQUIRES TAV GT TOUT./)
  504 FORMAT(3X,41HAUGMENTOR EFFICIENCY REQUIRES TAV LT TIN./)
  505 FORMAT(3X,42HAUGMENTOR EFFICIENCY REQUIRES TAV GT TOUT./)
      END
C
C
C**********DUCT*********
C
C
      SUBROUTINE DUCT(AQL,V,SFPIN,SFPOUT,SFTIN,SFWG,TSC,PIN,POUT,TIN,
     1 WG,POT,IGAIN,SFW)
C     DUCT ROUTINE
      DIMENSION POT(1)
      PIN=PIN*SFPIN
      POUT=POUT*SFPOUT
      WG=WG*SFWG
      TIN=TIN*SFTIN
      GC=386.26
      RA=640.1
      POT(1)=WG/SFWG
      POT2I=AQL*GC*SFPIN/(SFWG*TSC)
      POT3I=SFPOUT*POT2I/SFPIN
      POT4I=AQL*GC*PIN*(PIN-POUT)*SFWG*V/(WG*WG*RA*TIN*SFW*TSC)
      IF(POT2I.GT.1.)  GO TO 2
      IGAIN=0
      POT(2)=POT2I
      POT(3)=POT3I
      POT(4)=POT4I
      GO TO 7
    2 IF(POT2I.GT.10.)  GO TO 3
      IGAIN=1
      POT(2)=POT2I/10.
      POT(3)=POT3I/10.
      POT(4)=POT4I/10.
      GO TO 7
    3 IF(POT2I.GT.100.)  GO TO 4
      IGAIN=2
      POT(2)=POT2I/100.
      POT(3)=POT3I/100.
      POT(4)=POT4I/100.
      GO TO 7
    4 IF(POT2I.GT.1000.)  GO TO 5
      IGAIN=3
      POT(2)=POT2I/1000.
      POT(3)=POT3I/1000.
      POT(4)=POT4I/1000.
      GO TO 7
    5 IF(POT2I.GT.10000.)   GO TO 6
      IGAIN=4
      POT(2)=POT2I/10000.
      POT(3)=POT3I/10000.
      POT(4)=POT4I/10000.
      GO TO 7
    6 IGAIN=5
      POT(2)=POT2I/100000.
      POT(3)=POT3I/100000.
      POT(4)=POT4I/100000.
    7 CONTINUE
      PIN=PIN/SFPIN
      POUT=POUT/SFPOUT
      WG=WG/SFWG
      TIN=TIN/SFTIN
      RETURN
      END
C
C
```

```
C**********ENGINE**********
C
C
      SUBROUTINE ENGINE
C     THIS ROUTINE PERFORMS ALL THE ENGINE CALCULATIONS
C                  AND IS SIMILAR IN STRUCTURE TO THE
C                  LOOP ROUTINE OF THE HYBRID SIMULATION
      REAL MAP,MAPL,KBLWHT,KBLWLT
      INTEGER PLA
      DIMENSION N7(3)
      DIMENSION XY(45),X7(15),Y71(15),Y72(15)
      EQUIVALENCE (X7(1),XY(1)),(Y71(1),XY(16)),(Y72(1),XY(31))
      DATA N7/1,1,15/
      DATA X7/                                                          -
     * .05000,.34732,.40345,.43196,.45522,.48029,.51127,.54541,         -
     * .57747,.61735,.66017,.72709,.84377,.91296,.95000/
      DATA Y71/                                                         -
     * .96610,.96610,.95366,.95208,.95019,.95172,.95369,.95709,         -
     * .95519,.94861,.93186,.88707,.88481,.90661,.92250/
      DATA Y72/                                                         -
     *.97034,.97034,.97184,.97431,.97727,.98022,.98406,.98933,          -
     *.99500,1.0000,1.0000,.95010,.95372,.96547,.97600/
      COMMON /IVARS/ IP,JP,JPA,JPD,KBH,KBL,KBV,NAUG,NDRY,NTOTAL,PLA
      COMMON /NMAPS/F1(322),F2(322),F3(854),F4(518),F5(224),F6(224),    -
     1              N1(5),N2(5),N3(5),N4(5),N5(5),N6(5)
      COMMON /AVARS/ CC(50),DACI(24),DC(125),XC(6),YC(6),ZC(12),UDC(125)
      COMMON /RVARS/ CBLHR(50),CBLLR(50),CBLVR(50),DH4QR(50),DH4TR(50),-
     A               DH41QR(50),DH41TR(50),DP13R(50),DP6R(50),ETAABR(50),-
     B               ETABR(50),FGR(50),P0R(50),P2R(50),P22R(50),-
     C               P5R(50),T13PR(50),T2R(50),T22R(50),T3PR(50),-
     D               WA2R(50),WA22R(50),WA3R(50),WG4R(50),WG41R(50),-
     E               WG7R(50)
      COMMON /XVARS/ A1(50),A2(50),A3(50),A4(50),A5(50),A6(50),A7(50),  -
     1               A81(50),A9(50),A10(50),A11(50),A12(50),A13(50),    -
     2               A14(50),A15(50),A16(50),A17(50),A18(50),A19(50),   -
     3               A20(50),A21(50),A22(50),A23(50),A24(50),A25(50),   -
     4               A26(50)
      EQUIVALENCE (A8    , A1(  1)),(AE    , A1(  2)),(ALT   , A1(  3)),  -
     1            (ALTM  , A1(  4))
      EQUIVALENCE (CD7   , A3(  1)),(CDN   , A3(  2)),(CIVV  , A3(  3)),  -
     2            (CP13  , A3(  4)),(CP13P , A3(  5)),(CP2   , A3(  6)),  -
     2            (CP22  , A3(  7)),(CP3   , A3(  8)),(CP3P  , A3(  9)),  -
     3            (CP4   , A3( 10)),(CP41  , A3( 11)),(CP6   , A3( 12)),  -
     4            (CP7   , A3( 13)),(CPAB  , A3( 14)),(CPB   , A3( 15)),  -
     5            (CPHC  , A3( 16)),(CSHIFT, A3( 17)),(CV13  , A3( 18))
      EQUIVALENCE (CV13P , A3( 19)),(CV2   , A3( 20)),(CV22  , A3( 21)), -
     7            (CV3   , A3( 22)),(CV3P  , A3( 23)),(CV4   , A3( 24)), -
     8            (CV41  , A3( 25)),(CV6   , A3( 26)),(CV7   , A3( 27)), -
     9            (CV8   , A3( 28)),(CVAB  , A3( 29)),(CVB   , A3( 30)), -
     A            (CVHC  , A3( 31)),(CVN   , A3( 32))
      EQUIVALENCE (DH4   , A4(  1)),(DH41  , A4(  2)),(DTQW13, A4(  3)), -
     1            (DTQW3 , A4(  4)),(DTQW4 , A4(  5)),(DTQW41, A4(  6)), -
     2            (DTQW6 , A4(  7)),(DTQW7 , A4(  8)),(DW13  , A4(  9)), -
     3            (DW3   , A4( 10)),(DW4   , A4( 11)),(DW41  , A4( 12)), -
     4            (DW6   , A4( 13)),(DW7   , A4( 14)),(DXNH  , A4( 15)), -
     5            (DXNL  , A4( 16))
      EQUIVALENCE (ETAAB , A5(  1)),(ETAB  , A5(  2)),(ETAHCM, A5(  3)), -
     1            (ETAIFM, A5(  4)),(ETAOFM, A5(  5))
      EQUIVALENCE (FAR4M , A6(  1)),(FAR41M, A6(  2)),(FAR6M , A6(  3)), -
     1            (FAR7M , A6(  4)),(FG    , A6(  5)),(FGM3  , A6(  6)), -
     2            (FGPT3 , A6(  7)),(FN    , A6(  8)),(FNET  , A6(  9)), -
     3            (FNM   , A6( 10)),(FSHIFT, A6( 11))
      EQUIVALENCE (GM13  , A7(  1)),(GM13M , A7(  2)),(GM13P , A7(  3)), -
     1            (GM2   , A7(  4)),(GM22  , A7(  5)),(GM3   , A7(  6)), -
     2            (GM3M  , A7(  7)),(GM3P  , A7(  8)),(GM4   , A7(  9)), -
     3            (GM41  , A7( 10)),(GM4M  , A7( 11)),(GM41M , A7( 12)), -
     4            (GM6   , A7( 13)),(GM6M  , A7( 14)),(GM7   , A7( 15)), -
     5            (GM7M  , A7( 16)),(GMAB  , A7( 17)),(GMB   , A7( 18)), -
     6            (GMHC  , A7( 19))
      EQUIVALENCE (H13   ,A81(  1)),(H13M  ,A81(  2)),(H13P  ,A81(  3)), -
     1            (H13PM ,A81(  4)),(H2    ,A81(  5)),(H22   ,A81(  6)), -
     2            (H2M   ,A81(  7)),(H22M  ,A81(  8)),(H3    ,A81(  9)), -
     3            (H3M   ,A81( 10)),(H3P   ,A81( 11)),(H3PM  ,A81( 12)), -
     4            (H4    ,A81( 13)),(H41   ,A81( 14)),(H4M   ,A81( 15)), -
```

```
     5              (H41M   ,A81( 16)),(H6     ,A81( 17)),(H6M    ,A81( 18)), -
     6              (H7     ,A81( 19)),(H7M    ,A81( 20)),(HAB    ,A81( 21)), -
     7              (HABM   ,A81( 22)),(HB     ,A81( 23)),(HBM    ,A81( 24)), -
     8              (HHCM   ,A81( 25)),(HP4    ,A81( 26)),(HP41   ,A81( 27))
      EQUIVALENCE (KBLWHT,A11(  1)),(KBLWLT,A11(  2))
      EQUIVALENCE (PE     ,A16(  1)),(P0     ,A16(  2)),(P0A    ,A16(  3)), -
     1              (P0QT7 ,A16(  4)),(P13    ,A16(  5)),(P2     ,A16(  6)), -
     2              (P2A    ,A16(  7)),(P22    ,A16(  8)),(P22Q2M,A16(  9)), -
     3              (P3     ,A16( 10)),(P4     ,A16( 11)),(P41    ,A16( 12)), -
     4              (P5     ,A16( 13)),(P6     ,A16( 14)),(P7     ,A16( 15)), -
     5              (PRHC   ,A16( 16)),(PRIF   ,A16( 17)),(PROF   ,A16( 18))
      EQUIVALENCE (RCVV   ,A18(  1)),(RTT2   ,A18(  2)),(RTT22  ,A18(  3)), -
     1              (RTT4   ,A18(  4)),(RTT41  ,A18(  5))
      EQUIVALENCE (T0A    ,A20(  1)),(T13    ,A20(  2)),(T13M   ,A20(  3)), -
     1              (T13P   ,A20(  4)),(T13PM  ,A20(  5)),(T2     ,A20(  6)), -
     2              (T2A    ,A20(  7)),(T2M    ,A20(  8)),(T22    ,A20(  9)),-
     3              (T22M   ,A20( 10)),(T3     ,A20( 11)),(T3M    ,A20( 12)),-
     4              (T3P    ,A20( 13)),(T3PM   ,A20( 14)),(T4     ,A20( 15)), -
     5              (T4M    ,A20( 16)),(T41    ,A20( 17)),(T41M   ,A20( 18)), -
     6              (T6     ,A20( 19)),(T6M    ,A20( 20)),(T7     ,A20( 21)), -
     7              (T7M    ,A20( 22)),(TAM    ,A20( 23)),(TAVAB  ,A20( 24)), -
     8              (TAVB   ,A20( 25)),(TAVHC  ,A20( 26)),(TRHCM1,A20( 27)), -
     9              (TRIFM1,A20( 28)),(TROFM1,A20( 29))
      EQUIVALENCE (WA13   ,A23(  1)),(WA2    ,A23(  2)),(WA22   ,A23(  3)), -
     1              (WA3    ,A23(  4)),(WAR2   ,A23(  5)),(WAR2M  ,A23(  6)), -
     2              (WAR22  ,A23(  7)),(WAR22M,A23(  8)),(WBLHT  ,A23(  9)), -
     3              (WBLLT  ,A23( 10)),(WBLOV  ,A23( 11)),(WF4    ,A23( 12)),-
     4              (WF7    ,A23( 13)),(WG4    ,A23( 14)),(WG41   ,A23( 15)), -
     5              (WG6    ,A23( 16)),(WG7    ,A23( 17)),(WG7M   ,A23( 18)), -
     6              (WP4    ,A23( 19)),(WP41   ,A23( 20))
      EQUIVALENCE (X3     ,A24(  1)),(X4     ,A24(  2)),(X5     ,A24(  3)), -
     1              (X6     ,A24(  4)),(XF     ,A24(  5)),(XMN    ,A24(  6)), -
     2              (XMNM   ,A24(  7)),(XNH    ,A24(  8)),(XNL    ,A24(  9))
      EQUIVALENCE (Y3     ,A25(  1)),(Y4     ,A25(  2)),(Y5     ,A25(  3)),-
     1              (Y6     ,A25(  4))
C.....COMPUTE FAN INLET CONDITION AND FAN PERFORMANCE PARAMETERS
  100 ALTM = DC(1)*ALT
      XMNM = DC(2)*XMN
      CALL FLCOND(ALTM,XMNM,TAM,P2A,T2A,P0A,T0A)
      P2C= (.5*P2A)/DC(3)
      P2R(IP)= P2/P2C
      P2=P2C
      T2C= (.5*T2A)/DC(4)
      T2R(IP)= T2/T2C
      T2=T2C
      IF(IP.EQ.1)  DACI(1)=P2
      IF(IP.EQ.1)  DACI(2)=T2
  105 X3 = (DC(58)*P13)/P2
      Y3 = (DC(59)*XNL)/SQRT(T2)
      WAR2M= MAP(N3,F3,X3,Y3)
      ETAOFM = MAPL(F3)
      P22Q2M = MAPL(F3)
      ETAIFM = MAPL(F3)
      IF(CIVV .LT. .0) CIVV=.0
  110 FSHIFT = MAP(N1,F1,CIVV,Y3)
      WAR2 = (WAR2M*(.5+.5*FSHIFT))/.5
      IF(IP.EQ.1)  DACI(3)=X3
      IF(IP.EQ.1)  DACI(4) = WAR2
      WA2C=(WAR2*P2)/(SQRT(T2)*DC(57))
      WA2R(IP)= WA2/WA2C
      WA2=WA2C
      P22C=(P22Q2M*P2)/DC(60)
      P22R(IP)=P22/P22C
      P22=P22C
      PRIF=(DC(10)*P22)/P2
      T2M = DC(6)*T2
  115 CALL PROCOM(T2M,0.,CP2,CV2,GM2,H2M)
      CALL TRAT(1,PRIF,GM2,TRIFM1)
      T22C=(DC(11)*T2*(.2+DC(91)*TRIFM1/ETAIFM))/.2
      T22R(IP)=T22/T22C
      T22=T22C
C.....COMPUTE COMPRESSOR PERFORMANCE PARAMETERS
   120 X4=(DC(62)*P3)/P22
      Y4=(DC(63)*XNH)/SQRT(T22)
```

```
      WAR22M=MAP(N4,F4,X4,Y4)
      ETAHCM=MAPL(F4)
      IF(RCVV .LT. .0) RCVV=.0
      CSHIFT = MAP(N2,F2,RCVV,Y4)
      WAR22 = (WAR22M*(.5+.5*CSHIFT))/.5
      IF(IP.EQ.1)  DACI(5)=X4
      IF(IP.EQ.1)  DACI(6)=WAR22
      WA22C=(WAR22 *P22)/SQRT(T22)/DC(61)
      WA22I=WA22
      WA22R(IP)=WA22/WA22C
      WA22=WA22C
C.....COMPUTE DERIVATIVES AT STATION 13
  125 DW13=WA2 - DC(18)*WA22 - DC(56)*WA13
      PROF=(DC(8)*P13)/P2
      CALL TRAT(2,PROF,GM2,TROFM1)
      T13PC= (DC(9)*T2*(.2+DC(96)*TROFM1/ETAOFM))/.2
      T13PR(IP)=T13/T13PC
      T13P=T13PC
      T13PM=DC(34)*T13P
      CALL PROCOM(T13PM,0.,CP13P,CV13P,GM13P,H13PM)
      H13P=H13PM/DC(15)
      T13M=DC(34)*T13
  130 CALL PROCOM(T13M,0.,CP13,CV13,GM13,H13M)
      H13=H13M/DC(15)
      GM13M=(GM13-.5)/.5
      DTQW13=((WA2-DC(18)*WA22)*(H13P-H13))/CV13+T13*DW13*GM13M
C.....COMPUTE BLEEDS AND COMBUSTOR AIRFLOW
  135 T3M=DC(13)*T3
      CALL PROCOM(T3M,0.,CP3,CV3,GM3,H3M)
      FGM3=(.66572-.23396*GM3)*GM3+.33337
      FGPT3=.5*FGM3*P3/SQRT(T3)
      IF(KBH.GT.0) GO TO 140
      WBLHT=0.
      CBLHR(IP)=1.
      GO TO 145
  140 WBLHT=FGPT3/DC(98)
      CBLHR(IP)=(WG41-DC(30)*WG4)/(DC(72)*WBLHT)
      WBLHTN=(WG41-DC(30)*WG4)/(DC(72)*CBLHR(IP))
  145 IF(KBL.GT.0) GO TO 150
      WBLLT=0.
      CBLLR(IP)=1.
      GO TO 155
  150 WBLLT=FGPT3/DC(99)
      CBLLR(IP)=(WG6-DC(36)*WG41-DC(35)*WA13)/(DC(80)*WBLLT)
      WBLLTN=(WG6-DC(36)*WG41-DC(35)*WA13)/(DC(80)*CBLLR(IP))
  155 IF(KBV.GT.0) GO TO 160
      WBLOV=0.
      CBLVR(IP)=1.
      GO TO 165
  160 WBLOV=FGPT3/DC(100)
      CBLVR(IP)=(WA22I-DC(43)*WA3-.2*WBLHTN-.02*WBLLTN)/(.002*WBLOV)
C.....COMPUTE DERIVATIVES AT STATION 3
  165 WA3C=SQRT(P3*(P3-P4)/T3)/DC(64)
      WA3R(IP)=WA3/WA3C
      WA3=WA3C
      DW3=WA22-.2*WBLHT-.02*WBLLT-.002*WBLOV-DC(43)*WA3
      H3=H3M/DC(17)
      PRHC=(DC(14)*P3)/P22
      TAVHC=DC(83)*T22+DC(84)*T3
  170 CALL PROCOM(TAVHC,0.,CPHC,CVHC,GMHC,HHCM)
      CALL TRAT(3,PRHC,GMHC,TRHCM1)
      T3PC=(DC(12)*T22*(.2+DC(92)*TRHCM1/ETAHCM))/.2
      T3PR(IP)=T3/T3PC
      T3P=T3PC
      T3PM=DC(13)*T3P
      CALL PROCOM(T3PM,0.,CP3P,CV3P,GM3P,H3PM)
      H3P=H3PM/DC(17)
      GM3M=(GM3-.5)/.5
      DTQW3=(WA22*(H3P-H3))/CV3-T3*DW3*GM3M
C.....COMPUTE HP TURBINE PERFORMANCE PARAMETERS
  175 X5=(DC(66)*P41)/P4
      Y5=(DC(67)*XNH)/SQRT(T4)
      WP4=MAP(N5,F5,X5,Y5)
      HP4=MAPL(F5)
```

```
      WG4C=(.5*WP4*P4*XNH)/(T4*DC(65))
      WG4R(IP)=WG4/WG4C
      WG4=WG4C
C.....COMPUTE DERIVATIVES AT STATION 4
      DW4=DC(23)*WA3-WG4+DC(44)*WF4
      TAVB=DC(85)*T3+DC(86)*T4
  180 CALL PROCOM(TAVB,0.,CPB,CVB,GMB,HBM)
      HB=HBM*DC(22)
      FAR4M=(DC(20)*WF4)/WA3
      T4M=DC(21)*T4
  185 CALL PROCOM(T4M,FAR4M,CP4,CV4,GM4,H4M)
      H4=H4M*DC(22)
      GM4M=(GM4-.5)/.5
      ETABC=(-DC(23)*WA3*HB+H4*(DC(23)*WA3+DC(44)*WF4))/(DC(24)*WF4)
      ETABR(IP)=ETABC/ETAB
      DTQW4=(DC(23)*WA3*HB-H4*(DC(23)*WA3+DC(44)*WF4)+DC(24)*ETAB*WF4)   -
     1 /CV4+T4*DW4*GM4M
C.....COMPUTE LP TURBINE PERFORMANCE PARAMTERS
  190 P5C=P6/DC(76)
      P5R(IP)=P5/P5C
      P5=P5C
      X6=(DC(74)*P5)/P41
      Y6=(DC(75)*XNL)/SQRT(T41)
      WP41=MAP(N6,F6,X6,Y6)
      HP41=MAPL(F6)
      WG41C=(WP41*P41*XNL)/(T41*DC(73))
      WG41R(IP)=WG41/WG41C
      WG41=WG41C
C.....COMPUTE DERIVATIVES AT STATION 4.1
      DW41=DC(30)*WG4+DC(72)*WBLHT-WG41
      DH4=(HP4*SQRT(T4)*XNH)/DC(87)
      FAR41M=(FAR4M*.5)/(.5+(.5+.04*FAR4M)*DC(71)*WBLHT/WG4)
      T41M=DC(28)*T41
  195 CALL PROCOM(T41M,FAR41M,CP41,CV41,GM41,H41M)
      H41=H41M*DC(29)
      GM41M=(GM41-.5)/.5
      DTQW41=(WG4*H4/DC(26)-(DC(30)*WG4+DC(72)*WBLHT)*H41+DC(68)*
     1 WBLHT*H3-DC(27)*DH4*(WG4+DC(70)*WBLHT))/CV41+T41*DW41*GM41M
C.....COMPUTE DERIVATIVES AT STATION 6
  200 DW6=DC(35)*WA13-WG6+DC(80)*WBLLT+DC(36)*WG41
      DH41=(HP41*SQRT(T41)*XNL)/DC(88)
      T6M=DC(38)*T6
      FAR6M =(FAR41M*.25)/(.25+(.5+.04*FAR41M)*DC(37)*(WA13+DC(81)* -
     1 WBLLT)/WG41)
  205 CALL PROCOM(T6M,FAR6M,CP6,CV6,GM6,H6M)
      H6=H6M/DC(39)
      GM6M=(GM6-.5)/.5
      DTQW6=(DC(31)*WA13*H13-(DC(35)*WA13+DC(80)*WBLLT+DC(36)*WG41)*     -
     1 H6+DC(32)*WG41*H41+DC(77)*WBLLT*H3-DC(33)*DH41*(WG41+            -
     2 DC(79)*WBLLT))/CV6+T6*DW6*GM6M
C.....COMPUTE NOZZLE PERFORMANCE PARAMETERS
  210 P0C=(.5*P0A)/DC(5)
      P0R(IP)=P0/P0C
      P0=P0C
      P0QT7=DC(111)*P0/P7
      CDN=FUN1(N7,X7,P0QT7)
      CVN=FUN1L(X7)
  213 CALL NOZZL (P0,P7,P0QT7,T7,A8,AE,CDN,CVN,WG7M,FNM,XF)
      FGR(IP)=(FG-XF)/(FNM-XF)
      FNET=FNM-XMNM*SQRT(T0A)*WA2/DC(45)
      IF(IP.EQ.1)  DACI(21)=FNET
      WG7C=(WG7M*DC(95))/.5
      WG7R(IP)=WG7/WG7C
      WG7=WG7C
C.....COMPUTE DERIVATIVES AT STATION 7
      DW7=DC(49)*WG6-WG7+DC(55)*WF7
      TAVAB=DC(89)*T6+DC(90)*T7
  215 CALL PROCOM(TAVAB,FAR6M,CPAB,CVAB,GMAB,HABM)
      HAB=HABM*DC(52)
      T7M=DC(51)*T7
      FAR7M=FAR6M+(DC(50)*(.04*FAR6M+.5)*WF7)/(.5*WG6)
  220 CALL PROCOM(T7M,FAR7M,CP7,CV7,GM7,H7M)
      H7=H7M*DC(52)
      GM7M=(GM7-.5)/.5
```

```
      IF(IP.GT.NDRY) GO TO 224
      ETAABR(IP)=1
      GO TO 230
  224 IF(FAR7M.GT..39762) GO TO 225
      ETAAB=((.81226-.56042*FAR7M)*FAR7M+.25)/.5
      GO TO 226
  225 ETAAB=((.75331-.72512*FAR7M)*FAR7M+.29948)/.5
  226 ETAABC=(-DC(49)*WG6*HAB+H7*(DC(49)*WG6+DC(55)*WF7))/(DC(53)*WF7)
      ETAABR(IP)=ETAABC/ETAAB
  230 DTQW7=(DC(49)*WG6*HAB-H7*(DC(49)*WG6+DC(55)*WF7)+DC(53)*WF7*       -
     1 ETAAB)/CV7+T7*DW7*GM7M
C.....COMPUTE SPEED DERIVATIVES
      H2=H2M/DC(7)
      T22M=DC(19)*T22
  235 CALL PROCOM(T22M,0.,CP22,CV22,GM22,H22M)
      H22=H22M/DC(16)
      DH41QC=((WA2-DC(18)*WA22)*(H13P-DC(9)*H2)/DC(41)+                 -
     1  WA22*(H22-DC(11)*H2)/DC(42))/(.5*(WG41+DC(79)*WBLLT)/DC(93))
      DH41QR(IP)=DH41QC/DH41
      DH41TC=(DC(31)*WA13*H13-(DC(35)*WA13+DC(80)*WBLLT+DC(36)*          -
     1 WG41)*H6+DC(32)*WG41*H41+DC(77)*WBLLT*H3)/(DC(33)*(WG41+          -
     2 DC(79)*WBLLT))
      DH41TR(IP)=DH41TC/DH41
      DXNL=(.5*DH41*(WG41+DC(79)*WBLLT)/DC(93)-(WA2-DC(18)*              -
     1 WA22)*(H13P-DC(9)*H2)/DC(41)-WA22*(H22-DC(11)*H2)/DC(42))/XNL
      DH4QC=(WA22*(H3P-DC(12)*H22)/DC(40))/(.5*(WG4+DC(70)*WBLHT)        -
     1 /DC(94))
      DH4QR(IP)=DH4QC/DH4
      DH4TC=(WG4*H4/DC(26)-(DC(30)*WG4+DC(72)*WBLHT)*H41+                -
     1 DC(68)*WBLHT*H3)/(DC(27)*(WG4+DC(70)*WBLHT))
      DH4TR(IP)=DH4TC/DH4
      DXNH=(.5*DH4*(WG4+DC(70)*WBLHT)/DC(94)-WA22*(H3P-DC(12)*           -
     1 H22)/DC(40))/XNH
      RETURN
      END
C
C
C***********FLCOND**********
C
C
      SUBROUTINE FLCOND(HT,XM0,TAS,PT,TT,PS,TS)
C     THIS ROUTINE CALCULATES FLIGHT CONDITION INFORMATION
      DIMENSION N1(3),N2(3),N3(3)
      DIMENSION XY(56),X1(11),Y1(11),X2(10),Y2(10),X3(7),Y3(7)
      EQUIVALENCE (X1(1),XY(1)),(Y1(1),XY(12)),(X2(1),XY(23)),         -
     *            (Y2(1),XY(33)),(X3(1),XY(43)),(Y3(1),XY(50))
      COMMON/INL/INLET
      DATA N1/2,1,11/,N2/3,1,10/,N3/4,1,7/
      DATA X1          /.0,.05,.1,.175,.275,.35,.425,.525,.625,          -
     1 .75,.99999/
      DATA Y1          /.7348,.6346,.54585,.43180,.31062,.23922,         -
     1 .18176,.12404,.08458,.05244,.02018/
      DATA X2/  .00000,.11111,.22222,.33333,.44444,.55555,              -
     1 .66666,.77777,.88888,.99999/
      DATA Y2          /.0,.00046,.00517,.02138,.05853,.12780,          -
     1 .24192,.41493,.66214,.99999/
      DATA  X3/ .00000,.11111,.22222,.33333,.44444,.55555,              -
     1  .66666/
      DATA Y3         /.0,.05149,.13126,.27692,.33461,.45224,.57846/
      TS=TAS-.2848*HT
      IF(TS.LT..38999) TS=.38999
      IF(HT.LT.0.0) HT=0.
      PS=FUN1(N1,X1,HT)
  100 TTQS= .33333 + .60000*(XM0*XM0)
      TR35=FUN1(N2,X2,TTQS)
      PTQS=TR35/.85533
      ETAI=.99999
      IF(INLET.NE.1) GO TO 110
      IF(XM0.LE..33333) GO TO 110
      XMP=XM0-.33333
```

```
      XPP=FUN1(N3,X3,XMP)
      ETAI=.99999-.33050*XPP
  110 TT= (TTQS*TS)/.33333
      PT= (ETAI*PTQS*PS)/.05
      RETURN
      END
C
C
C***********FOOR**********
C
C
      SUBROUTINE FOOR(N,XIN)
C     FUNCTION-OUT-OF-RANGE ROUTINE
      COMMON /IO/ IR,IW,IPNCH
  100 WRITE(IW,400) N,XIN
  400 FORMAT(/12HFUNCTION NO.,I3,19H INPUT OUT OF RANGE/6HXIN = ,F7.5/)
      RETURN
      END
C
C
C***********FUN1L**********
C
C
      FUNCTION FUN1L(F)
C     FUNCTION SECOND EVALUATION ROUTINE
C******XIN IS DUMMY ARGUMENT AND DOES NOT AFFECT RESULTS
      N=0
      FUN1L=FUN1(N,F,XIN)
      RETURN
      END
C
C
C************FUN1**********
C
C
      FUNCTION FUN1(N,F,XIN)
C     FUNCTION EVALUATION ROUTINE
      DIMENSION N(1),F(1)
      IF(N(1).EQ.0) GO TO 200
      I=N(2)
      NXP=N(3)
  100 X1=XIN-F(I)
      IF(X1.GT..0) GO TO 110
      IF(X1.EQ..0) GO TO 120
      IF(I.LE.1) GO TO 140
      I=I-1
      GO TO 100
  110 X2=XIN-F(I+1)
      IF(X2.LT..0) GO TO 180
      IF(X2.EQ..0) GO TO 130
      I=I+1
      IF(I.GE.NXP) GO TO 150
      X1=X2
      GO TO 110
  120 XFRAC=.0
      GO TO 190
  130 XFRAC=1.0
      GO TO 190
  140 XFRAC=.0
      GO TO 160
  150 XFRAC=1.0
      I=I-1
  160 CALL FOOR(N,XIN)
      GO TO 190
  180 XFRAC=X1/(X1-X2)
  190 N(2)=I
  200 I=I+NXP
      FUN1=F(I)+XFRAC*(F(I+1)-F(I))
      RETURN
      END
```

```
C
C
C***********MAP**********
C
C
      FUNCTION MAP(N,F,XIN,YIN)
C     MAP EVALUATION ROUTINE
      DIMENSION N(1),F(1)
      REAL MAP
      IF(N(1).EQ.0) GO TO 400
      I=N(2)
      J=N(3)
      NXP=N(4)
      NYC=N(5)
      NPROD=NYC*NXP
  100 Y1=YIN-F(J)
      IF(Y1.GT..0) GO TO 110
      IF(Y1.EQ..0) GO TO 120
      IF(J.LE.1) GO TO 140
      J=J-1
      GO TO 100
  110 Y2=YIN-F(J+1)
      IF(Y2.LT..0) GO TO 180
      IF(Y2.EQ..0) GO TO 130
      J=J+1
      IF(J.GE.NYC) GO TO 150
      Y1=Y2
      GO TO 110
  120 YINCR=.0
      GO TO 190
  130 YINCR=.99999
      GO TO 190
  140 YINCR=.0
      GO TO 160
  150 YINCR=.99999
      J=J-1
  160 CALL MOOR(N,XIN,YIN)
      GO TO 190
  180 YINCR=Y1/(Y1-Y2)
  190 KX=J*NXP+NYC+I
      LX=KX-NXP
  200 XLO=F(LX)+YINCR*(F(KX)-F(LX))
      IF(XIN.GT.XLO) GO TO 210
      IF(XIN.EQ.XLO) GO TO 220
      IF(I.LE.1) GO TO 240
      I=I-1
      LX=LX-1
      KX=KX-1
      GO TO 200
  210 XHI=F(LX+1)+YINCR*(F(KX+1)-F(LX+1))
      IF(XIN.LT.XHI) GO TO 280
      IF(XIN.EQ.XHI) GO TO 230
      I=I+1
      IF(I.GE.NXP) GO TO 250
      LX=LX+1
      KX=KX+1
      XLO=XHI
      GO TO 210
  220 XFRAC=.0
      GO TO 300
  230 XFRAC=.99999
      GO TO 300
  240 XFRAC=.0
      GO TO 260
  250 XFRAC=.99999
      I=I-1
  260 CALL MOOR(N,XIN,YIN)
      GO TO 300
  280 XFRAC=(XIN-XLO)/(XHI-XLO)
  300 N(3)=J
      N(2)=I
      LZ=LX+NPROD
      KZ=LZ+NXP
  350 ZL=F(LZ)+YINCR*(F(KZ)-F(LZ))
```

```
C      ZH=F(LZ+1)+YINCR*(F(KZ+1)-F(LZ+1))
C      MAP=ZL+XFRAC*(ZH-ZL)
C      RETURN
C  400 LZ=LZ+NPROD
C      KZ=LZ+NXP
C      GO TO 350
C      END
C
C
C***********MAPIN**********
C
 C
       SUBROUTINE MAPIN
C      THE ROUTINE DOES THE INITIALIZATION AND READ IN OF MAP DATA
C                      ALSO CALCULATES SOME DIGITAL COEFFICIENTS
       COMMON /IO/IR,IW,IPNCH
       COMMON /NMAPS/F1(322),F2(322),F3(854),F4(518),F5(224),F6(224),   -
      1                  N1(5),N2(5),N3(5),N4(5),N5(5),N6(5)
       COMMON /AVARS/ CC(50),DACI(24),DC(125),XC(6),YC(6),ZC(12),UDC(125)
       COMMON /SVARS/ BSCIVV,BSRCVV,SFA8,SFAE,SFALT,SFCIVV,SFDH4,SFDH41,-
      A               SFFG,SFFN,SFP0,SFP13,SFP2,SFP22,SFP3,SFP4,SFP41,-
      B               SFP5,SFP6,SFP7,SFRCVV,SFT0,SFT13,SFT2,SFT22,SFT3,-
      C               SFT4,SFT41,SFT6,SFT7,SFVEL,SFWA13,SFWA2,SFWA22,-
      D               SFWA3,SFWF4,SFWF7,SFWG4,SFWG41,SFWG6,SFWG7,SFXMN, -
      E               SFXNH,SFXNL
       WRITE(IW,500)
C******READ AND SCALING OF PERFORMANCE DATA
       CALL DATAIN(N1,F1)
       DO 10 N=1,2
    10 READ(IR,501) XC(N),YC(N),ZC(N)
       READ(IR,501) XC(3),YC(3),(ZC(N),N=3,6)
       DO 15 N=4,6
       K1=2*N-1
       K2=2*N
    15 READ(IR,501) XC(N),YC(N),ZC(K1),ZC(K2)
C******READ SCALE FACTORS
    20 READ(IR,502) SFP0,SFP2,SFP13,SFP22,SFP3,SFP4,SFP41,SFP5,SFP6,SFP7,-
      1 SFT2,SFT13,SFT22,SFT3,SFT4,SFT41,SFT6,SFT7,SFWA2,SFWA13,SFWA22,  -
      2 SFWA3,SFWG4,SFWG41,SFWG6,SFWG7,SFDH4,SFDH41,SFFN,SFXNL,SFXNH,    -
      3 SFWF4,SFWF7,SFA8,SFAE,SFALT,SFXMN,BSCIVV,SFCIVV,BSRCVV,SFRCVV    -
      4 ,SFVEL,SFT0,SFFG
       WRITE(IW,503)
       WRITE(IW,504)
       WRITE(IW,505)
       WRITE(IW,506)
       WRITE(IW,507)
       WRITE(IW,508)
       WRITE(IW,509)
       WRITE(IW,510)
       WRITE(IW,511)
       WRITE(IW,512)
       WRITE(IW,513)
       WRITE(IW,514)
       WRITE(IW,502) SFP0,SFP2,SFP13,SFP22,SFP3,SFP4,SFP41,SFP5,SFP6,    -
      1 SFP7,SFT2,SFT13,SFT22,SFT3,SFT4,SFT41,SFT6,SFT7,SFWA2,SFWA13,    -
      2 SFWA22,SFWA3,SFWG4,SFWG41,SFWG6,SFWG7,SFDH4,SFDH41,SFFN,SFXNL,   -
      3 SFXNH,SFWF4,SFWF7,SFA8,SFAE,SFALT,SFXMN,BSCIVV,SFCIVV,BSRCVV,    -
      4 SFRCVV,SFVEL,SFT0,SFFG
C******COMPUTE DIGITAL COEFFICIENTS FROM SCALE FACTORS
       DO 25 I=1,125
       DC(I)=.0
    25 CONTINUE
       DC(1)= SFALT/80000.
       DC(2)= SFXMN/3.0
       DC(3)=SFP2/80.
       DC(4)=SFT2/2000.
       DC(5)=SFP0/40.
       DC(6)=.4*DC(4)
       DC(7)=SFT2/3000.
       DC(8)=SFP13/(15.*SFP2)
       DC(9)=SFT2/SFT13
       DC(10)=SFP22/(15.*SFP2)
       DC(11)=SFT2/SFT22
       DC(12)=SFT22/SFT3
```

```
      DC(13)=SFT3/5000.
      DC(14)=SFP3/(15.*SFP22)
      DC(15)=SFT13/3000.
      DC(16)=SFT22/3000.
      DC(17)=SFT3/3000.
      DC(18)=SFWA22/SFWA2
      DC(19)=.6*DC(16)
      DC(20)=SFWF4/(.08*SFWA3)
      DC(21)=SFT4/5000.
      DC(22)=3000./SFT4
      DC(23)=SFWA3/SFWG4
      DC(24)=(37000.*SFWF4)/(SFT4*SFWG4)
      DC(25)=SFT3/SFT4
      DC(26)=(SFT41*SFWG41)/(SFT4*SFWG4)
      DC(27)=(SFDH4*SFWG4)/(.5*SFT41*SFWG41)
      DC(28)=SFT41/5000.
      DC(29)=3000./SFT41
      DC(30)=SFWG4/SFWG41
      DC(31)=(SFT13*SFWA13)/(SFT6*SFWG6)
      DC(32)=(SFT41*SFWG41)/(SFT6*SFWG6)
      DC(33)=(SFDH41*SFWG41)/(.5*SFT6*SFWG6)
      DC(34)=.6*DC(15)
      DC(35)=SFWA13/SFWG6
      DC(36)=SFWG41/SFWG6
      DC(37)=.5*SFWA13/SFWG41
      DC(38)=SFT6/5000.
      DC(39)=SFT6/3000.
      DC(40)=(SFDH4*SFWG4)/(.5*SFT3*SFWA22)
      DC(41)=(SFDH41*SFWG41)/(.5*SFT13*SFWA2)
      DC(42)=(SFDH41*SFWG41)/(.5*SFT22*SFWA22)
      DC(43)=SFWA3/SFWA22
      DC(44)=SFWF4/SFWG4
      DC(45)=SFFN/(SFXMN*SQRT(SFT0)*SFWA2*1.52317)
      DC(49)=SFWG6/SFWG7
      DC(50)=SFWF7/(.08*SFWG6)
      DC(51)=SFT7/5000.
      DC(52)=3000./SFT7
      DC(53)=(37000.*SFWF7)/(SFT7*SFWG7)
      DC(54)=SFT6/SFT7
      DC(55)=SFWF7/SFWG7
      DC(56)=SFWA13/SFWA2
  500 FORMAT(3X,36HCOMPONENT AND VARIABLE GEOMETRY DATA///)
  501 FORMAT(6F8.2)
  502 FORMAT(5F12.5)
  503 FORMAT(1H1)
  504 FORMAT(3X,23HSCALE FACTOR INPUT DATA/)
  505 FORMAT(6X,2HP0,10X,2HP2,10X,3HP13,9X,3HP22,9X,2HP3)
  506 FORMAT(6X,2HP4,10X,3HP41,9X,2HP5,10X,2HP6,10X,2HP7)
  507 FORMAT(6X,2HT2,10X,3HT13,9X,3HT22,9X,2HT3,10X,2HT4)
  508 FORMAT(6X,3HT41,9X,2HT6,10X,2HT7,10X,3HWA2,9X,4HWA13)
  509 FORMAT(6X,4HWA22,8X,3HWA3,9X,3HWG4,9X,4HWG41,8X,3HWG6)
  510 FORMAT(6X,3HWG7,9X,3HDH4,9X,4HDH41,8X,2HFN,10X,3HXNL)
  511 FORMAT(6X,3HXNH,9X,3HWF4,9X,3HWF7,9X,2HA8,10X,2HAE)
  512 FORMAT(6X,3HALT,9X,3HXMN,9X,5HBCIVV,7X,4HCIVV,8X,5HBRCVV)
  513 FORMAT(6X,4HRCVV,8X,3HVEL,9X,2HT0,10X,2HFG)
  514 FORMAT(/)
      RETURN
      END
C
C
C
C**********MAPL**********
C
C
C     FUNCTION MAPL(F)
C     MAP SECOND EVALUATION ROUTINE
C     REAL MAPL,MAP
CC    XIN AND YIN ARE DUMMY ARGUMENTS AND DO NOT AFFECT RESULTS
C     N=0
C     MAPL=MAP(N,F,XIN,YIN)
C     RETURN
C     END
C
C
```

```
C***********MOOR$**********
C
C
C       SUBROUTINE MOOR(N,XIN,YIN)
C       MAP-OUT-OF-RANGE ROUTINE
C       COMMON/IO/IR,IW,IPNCH
C       WRITE(IW,600) N
C   600 FORMAT(3X,7HMAP NO.,I3,20H INPUTS OUT OF RANGE)
C       WRITE(IW,601) XIN,YIN
C   601 FORMAT(3X,6HXIN = ,F7.5,8H  YIN = ,F7.5/)
C       RETURN
C       END
C
C
C***********NOZZL**********
C
C
        SUBROUTINE NOZZL (P0,P7,P0QT7,T7,A7,A8,CD7,CV8,W7,F8,XF)
C       THIS ROUTINE CALCULATES NOZZLE INFORMATION FOR
C                        CONVERGENT-DIVERGENT NOZZLE COFIGURATIONS
        COMMON /AVARS/ CC(50),DACI(24),DC(125),XC(6),YC(6),ZC(12),UDC(125)
        DIMENSION N1(3),N2(3),N3(3)
        DIMENSION XY(135),X1(15),Y1(15),X2(15),Y21(15),Y22(15),
       *             X3(15),Y31(15),Y32(15),Y33(15)
        EQUIVALENCE (X1(1),XY(1)),(Y1(1),XY(16)),(X2(1),XY(31)),
       *                (Y21(1),XY(46)),(Y22(1),XY(61)),(X3(1),XY(76)),
       *                (Y31(1),XY(91)),(Y32(1),XY(106)),(Y33(1),XY(121))
        DATA N1/6,1,15/,N2/7,1,15/,N3/8,1,15/
        DATA X1/
       * .50000,.50035,.50155,.50645,.51185,.52350,.53405,.54715,
       * .56780,.59410,.64320,.70090,.75035,.91145,1.0000/
        DATA Y1/
       * .52828,.54691,.56578,.60412,.63000,.66905,.69507,.72092,
       * .75283,.78400,.82589,.85958,.88065,.92312,.93611/
        DATA X2/
       * .52828,.54691,.56578,.60412,.63000,.66905,.69507,.72092,
       * .75283,.78400,.82589,.85958,.88065,.92312,.93611/
        DATA Y21/
       * .50000,.48740,.47462,.44851,.43070,.40338,.38479,.36589,
       * .34187,.31740,.28246,.25192,.23128,.18410,.16698/
        DATA Y22/
       * .50000,.50035,.50155,.50645,.51185,.52350,.53405,.54715,
       * .56780,.59410,.64320,.70090,.75035,.91145,1.0000/
        DATA X3/
       * .50000,.50395,.50870,.51520,.52335,.53315,.54450,.55745,
       * .58810,.60575,.64610,.71950,.77760,.88000,1.0000/
        DATA Y31/
       * .20000,.22000,.23000,.24000,.25000,.26000,.27000,.28000,
       * .30000,.31000,.33000,.36000,.38000,.41000,.43942/
        DATA Y32/
       * .52828,.46835,.43983,.41238,.38606,.36092,.33697,.31424,
       * .27240,.25320,.21839,.17404,.14924,.11823,.09396/
        DATA Y33/
       * .50000,.54062,.56015,.57915,.59760,.61555,.63300,.64995,
       * .68230,.69775,.72720,.76800,.79305,.82765,.85835/
        XF=0.0
C     **CALC AREA RATIO
        A8Q7 = DC(110)*A8/A7
        PEQ7=FUN1(N1,X1,A8Q7)
C    ***SUBSONIC FLOW
        ATXX =A7
        IF (P0QT7 .LT. PEQ7) GO TO 10
        XMX=FUN1(N2,X2,P0QT7)
        PTOL = ABS(P0QT7 - PEQ7)
        IF (PTOL .LE. .0005) GO TO 100
        AEQTXX=FUN1L(X2)
        ATXX = A8/AEQTXX
        GO TO 100
C    ***SUPERSONIC FLOW AND SONIC FLOW FOR CONV-ONLY NOZZLE
     10 IF (A8Q7 .LE. 0.5) GO TO 30
        XMN=FUN1(N3,X3,A8Q7)
        PEQ7=FUN1L(X3)
        P0YQX=(XMN*(XMN/.72217-.07958)/.4)+.08956
        PYQX=(XMN*(XMN/.85714)/.4)-.01667
```

```
      POYQOX=(XMN*(-XMN/.87600+.40163)/.2)+.83626
      PYQT = (PYQX * POYQOX)/POYQX
      IF (P0QT7 .GT. PYQT) GO TO 40
      XMX=FUN1L(X3)
C    *REGIME V
   20 PE=P7*PEQ7/DC(111)
      XF=(A8*(PE-P0))*DC(112)
      GO TO 100
   30 XMX=0.5
      GO TO 20
C    *REGIME III
   40 IF(ABS(A8Q7-.56345) .LE. .0005) GO TO 50
      XMX= .85585 + (P0QT7*((-.41254) + .17418 * P0QT7))/ .4
      XSHFT = 0.0
      IF (A8Q7 .GT. .6) XSHFT = .61755 + A8Q7 *(A8Q7*(A8Q7*(-.12575)+.38380)+
      (-.44243))/.25
      IF(A8Q7 .LT. .6) XSHFT = (-.53378*A8Q7)+.31910
      XMX=XMX+XSHFT
      GO TO 100
   50 XMX= .88080+(P0QT7*((-.41488)+.17255*P0QT7))/.4
C  ***VELOCITY,FLOW,AND THRUST CALCULATIONS
  100 VE = XMX*SQRT(T7)*CV8/DC(113)
  105 W7 = (ATXX*P7)/SQRT(T7) * CD7 / DC(115)
      F8 = W7*VE/DC(116) + XF
  500 RETURN
      END
C
C
C***********PRINT**********
C
C
      SUBROUTINE PRINT(IPRINT)
C     THIS ROUTINE PRINTS THE OPERATING POINT DATA,
C               ADC AND DAC VALUES, SCALED AND UNSCALED
C               ENGINE VARIABLES, AND ERROR RATIO INFORMATION
      DIMENSION DDC(125),GLD(125),ZD(125),FD(125)
      INTEGER PLA,ADDR,GAIN(10),GLD,ZD,FD,KBLWHT,KBLWLT
      COMMON /INL/ INLET
      COMMON /IO/ IR,IW,IPNCH
      COMMON /IVARS/ IP,JP,JPA,JPD,KBH,KBL,KBV,NAUG,NDRY,NTOTAL,PLA
      COMMON /AVARS/ CC(50),DACI(24),DC(125),XC(6),YC(6),ZC(12),UDC(125)
      COMMON /RVARS/ CBLHR(50),CBLLR(50),CBLVR(50),DH4QR(50),DH4TR(50),-
     A               DH41QR(50),DH41TR(50),DP13R(50),DP6R(50),ETAABR(50),-
     B               ETABR(50),FGR(50),P0R(50),P2R(50),P22R(50),-
     C               P5R(50),T13PR(50),T2R(50),T22R(50),T3PR(50),-
     D               WA2R(50),WA22R(50),WA3R(50),WG4R(50),WG41R(50),-
     E               WG7R(50)
      COMMON /ANVARS/ AQL13,AQL6,SFT,SFW13,SFW3,SFW4,SFW41,SFW6,SFW7,     -
     1                W13,W3,W4,W41,W6,W7,SW13,SW3,SW4,SW41,SW6,SW7,     -
     A                V13,V3,V4,V41,V6,V7,XIH,XIL,PADR(53),PVAL(53),     -
     B                IG(10),AADR(16)
      COMMON /SVARS/ BSCIVV,BSRCVV,SFA8,SFAE,SFALT,SFCIVV,SFDH4,SFDH41,-
     A               SFFG,SFFN,SFP0,SFP13,SFP2,SFP22,SFP3,SFP4,SFP41,-
     B               SFP5,SFP6,SFP7,SFRCVV,SFT0,SFT13,SFT2,SFT22,SFT3,-
     C               SFT4,SFT41,SFT6,SFT7,SFVEL,SFWA13,SFWA2,SFWA22,-
     D               SFWA3,SFWF4,SFWF7,SFWG4,SFWG41,SFWG6,SFWG7,SFXMN, -
     E               SFXNH,SFXNL
      COMMON /XVARS/ A1(50),A2(50),A3(50),A4(50),A5(50),A6(50),A7(50),  -
     1               A81(50),A9(50),A10(50),A11(50),A12(50),A13(50),     -
     2               A14(50),A15(50),A16(50),A17(50),A18(50),A19(50),   -
     3               A20(50),A21(50),A22(50),A23(50),A24(50),A25(50),  -
     4               A26(50)
      EQUIVALENCE (A8     , A1(  1)),(AE     , A1(  2)),(ALT    , A1(  3)),  -
     1            (ALTM   , A1(  4))
      EQUIVALENCE (CD7    , A3(  1)),(CDN    , A3(  2)),(CIVV   , A3(  3)),  -
     2            (CP13   , A3(  4)),(CP13P  , A3(  5)),(CP2    , A3(  6)),  -
     2            (CP22   , A3(  7)),(CP3    , A3(  8)),(CP3P   , A3(  9)),  -
     3            (CP4    , A3( 10)),(CP41   , A3( 11)),(CP6    , A3( 12)),  -
     4            (CP7    , A3( 13)),(CPAB   , A3( 14)),(CPB    , A3( 15)),  -
     5            (CPHC   , A3( 16)),(CSHIFT, A3( 17)),(CV13   , A3( 18))
      EQUIVALENCE (CV13P  , A3( 19)),(CV2    , A3( 20)),(CV22   , A3( 21)), -
     7            (CV3    , A3( 22)),(CV3P   , A3( 23)),(CV4    , A3( 24)), -
     8            (CV41   , A3( 25)),(CV6    , A3( 26)),(CV7    , A3( 27)), -
     9            (CV8    , A3( 28)),(CVAB   , A3( 29)),(CVB    , A3( 30)), -
```

```
     A            (CVHC  , A3( 31)),(CVN   , A3( 32))
      EQUIVALENCE (DH4   , A4(  1)),(DH41  , A4(  2)),(DTQW13, A4(  3)), -
     1            (DTQW3 , A4(  4)),(DTQW4 , A4(  5)),(DTQW41, A4(  6)), -
     2            (DTQW6 , A4(  7)),(DTQW7 , A4(  8)),(DW13  , A4(  9)), -
     3            (DW3   , A4( 10)),(DW4   , A4( 11)),(DW41  , A4( 12)), -
     4            (DW6   , A4( 13)),(DW7   , A4( 14)),(DXNH  , A4( 15)), -
     5            (DXNL  , A4( 16))
      EQUIVALENCE (ETAAB , A5(  1)),(ETAB  , A5(  2)),(ETAHCM, A5(  3)), -
     1            (ETAIFM, A5(  4)),(ETAOFM, A5(  5))
      EQUIVALENCE (FAR4M , A6(  1)),(FAR41M, A6(  2)),(FAR6M , A6(  3)), -
     1            (FAR7M , A6(  4)),(FG    , A6(  5)),(FGM3  , A6(  6)), -
     2            (FGPT3 , A6(  7)),(FN    , A6(  8)),(FNET  , A6(  9)), -
     3            (FNM   , A6( 10)),(FSHIFT, A6( 11))
      EQUIVALENCE (GM13  , A7(  1)),(GM13M , A7(  2)),(GM13P , A7(  3)), -
     1            (GM2   , A7(  4)),(GM22  , A7(  5)),(GM3   , A7(  6)), -
     2            (GM3M  , A7(  7)),(GM3P  , A7(  8)),(GM4   , A7(  9)), -
     3            (GM41  , A7( 10)),(GM4M  , A7( 11)),(GM41M , A7( 12)), -
     4            (GM6   , A7( 13)),(GM6M  , A7( 14)),(GM7   , A7( 15)), -
     5            (GM7M  , A7( 16)),(GMAB  , A7( 17)),(GMB   , A7( 18)), -
     6            (GMHC  , A7( 19))
      EQUIVALENCE (H13   ,A81(  1)),(H13M  ,A81(  2)),(H13P  ,A81(  3)), -
     1            (H13PM ,A81(  4)),(H2    ,A81(  5)),(H22   ,A81(  6)), -
     2            (H2M   ,A81(  7)),(H22M  ,A81(  8)),(H3    ,A81(  9)), -
     3            (H3M   ,A81( 10)),(H3P   ,A81( 11)),(H3PM  ,A81( 12)), -
     4            (H4    ,A81( 13)),(H41   ,A81( 14)),(H4M   ,A81( 15)), -
     5            (H41M  ,A81( 16)),(H6    ,A81( 17)),(H6M   ,A81( 18)), -
     6            (H7    ,A81( 19)),(H7M   ,A81( 20)),(HAB   ,A81( 21)), -
     7            (HABM  ,A81( 22)),(HB    ,A81( 23)),(HBM   ,A81( 24)), -
     8            (HHCM  ,A81( 25)),(HP4   ,A81( 26)),(HP41  ,A81( 27))
      EQUIVALENCE (KBLWHT,A11(  1)),(KBLWLT,A11(  2))
      EQUIVALENCE (PE    ,A16(  1)),(P0    ,A16(  2)),(P0A   ,A16(  3)), -
     1            (P0QT7 ,A16(  4)),(P13   ,A16(  5)),(P2    ,A16(  6)), -
     2            (P2A   ,A16(  7)),(P22   ,A16(  8)),(P22Q2M,A16(  9)), -
     3            (P3    ,A16( 10)),(P4    ,A16( 11)),(P41   ,A16( 12)), -
     4            (P5    ,A16( 13)),(P6    ,A16( 14)),(P7    ,A16( 15)), -
     5            (PRHC  ,A16( 16)),(PRIF  ,A16( 17)),(PROF  ,A16( 18))
      EQUIVALENCE (RCVV  ,A18(  1)),(RTT2  ,A18(  2)),(RTT22 ,A18(  3)), -
     1            (RTT4  ,A18(  4)),(RTT41 ,A18(  5))
      EQUIVALENCE (T0A   ,A20(  1)),(T13   ,A20(  2)),(T13M  ,A20(  3)), -
     1            (T13P  ,A20(  4)),(T13PM ,A20(  5)),(T2    ,A20(  6)), -
     2            (T2A   ,A20(  7)),(T2M   ,A20(  8)),(T22   ,A20(  9)),-
     3            (T22M  ,A20( 10)),(T3    ,A20( 11)),(T3M   ,A20( 12)),-
     4            (T3P   ,A20( 13)),(T3PM  ,A20( 14)),(T4    ,A20( 15)), -
     5            (T4M   ,A20( 16)),(T41   ,A20( 17)),(T41M  ,A20( 18)), -
     6            (T6    ,A20( 19)),(T6M   ,A20( 20)),(T7    ,A20( 21)), -
     7            (T7M   ,A20( 22)),(TAM   ,A20( 23)),(TAVAB ,A20( 24)), -
     8            (TAVB  ,A20( 25)),(TAVHC ,A20( 26)),(TRHCM1,A20( 27)), -
     9            (TRIFM1,A20( 28)),(TROFM1,A20( 29))
      EQUIVALENCE (WA13  ,A23(  1)),(WA2   ,A23(  2)),(WA22  ,A23(  3)), -
     1            (WA3   ,A23(  4)),(WAR2  ,A23(  5)),(WAR2M ,A23(  6)), -
     2            (WAR22 ,A23(  7)),(WAR22M,A23(  8)),(WBLHT ,A23(  9)), -
     3            (WBLLT ,A23( 10)),(WBLOV ,A23( 11)),(WF4   ,A23( 12)),-
     4            (WF7   ,A23( 13)),(WG4   ,A23( 14)),(WG41  ,A23( 15)), -
     5            (WG6   ,A23( 16)),(WG7   ,A23( 17)),(WG7M  ,A23( 18)), -
     6            (WP4   ,A23( 19)),(WP41  ,A23( 20))
      EQUIVALENCE (X3    ,A24(  1)),(X4    ,A24(  2)),(X5    ,A24(  3)), -
     1            (X6    ,A24(  4)),(XF    ,A24(  5)),(XMN   ,A24(  6)), -
     2            (XMNM  ,A24(  7)),(XNH   ,A24(  8)),(XNL   ,A24(  9))
      EQUIVALENCE (Y3    ,A25(  1)),(Y4    ,A25(  2)),(Y5    ,A25(  3)),-
     1            (Y6    ,A25(  4))
      IPRINT=IPRINT+1
      GO TO (10,20,30),IPRINT
C******PRINT OUT OF INPUT DATA
   10 CONTINUE
      WRITE(IW,502)
      WRITE(IW,510)IP,PLA
      WRITE(IW,511)
      WRITE(IW,512)
      WRITE(IW,513)
      WRITE(IW,514)
      WRITE(IW,515)
      WRITE(IW,516)
      WRITE(IW,517)
      WRITE(IW,518)
```

```
      WRITE(IW,519)
      WRITE(IW,520) P0,P2,P13,P22,P3,P4,P41,P5,P6,P7,TAM,T2,T13,T22,T3, -
     1 T4,T41,T6,T7,WA2,WA13,WA22,WA3,WG4,WG41,WG6,WG7,DH4,DH41,ETAB,    -
     2 ETAAB,FN,XNL,XNH,WF4,WF7,A8,AE,ALT,XMN,CDN,CVN,CIVV,RCVV,FG
      WRITE(IW,501)
      RETURN
C******PRINT OUT OF ENGINE DATA
   20 WRITE(IW,521)
      WRITE(6,2270)
      WRITE(6,2500)  P13,            P3,             P4,             P41,   -
     1           P6
      WRITE(6,2280)
      WRITE(6,2500)  P7,             T13,            T3,             T4,    -
     2           T41
      WRITE(6,2290)
      WRITE(6,2500)  T6,             T7,             WA13,           WG6,   -
     3           XNL
      WRITE(6,2300)
      WRITE(6,2500)  XNH,            WF4,            WF7,            A8,    -
     4           AE
      WRITE(6,2310)
      WRITE(6,2500)  ALT,            XMN,            CIVV,           RCVV,  -
    5            TAM
      WRITE(IW,501)
      WRITE(IW,522)
      WRITE(6,2320)
      WRITE(6,2500)   P2,            T2,             X3,             WAR2
      WRITE(6,2330)
      WRITE(6,2544)  X4,             WAR22,          DW13,           DTQW13-
     7,          DW3
      WRITE(6,2340)
      WRITE(6,2544)  DTQW3,          DW4,            DTQW4,          DW41,  -
     8           DTQW41
      WRITE(6,2350)
      WRITE(6,2544)  DW6,            DTQW6,          DW7,            DTQW7,-
     9           DXNL
      WRITE(6,2360)
      WRITE(6,2544)  DXNH
      WRITE(IW,502)
      WRITE(IW,523)
      WRITE(6,2370)
      WRITE(6,2500)  ALTM,           XMNM,           P2A,            T2A,   -
     1           P0A
      WRITE(6,2380)
      WRITE(6,2500)  Y3,             ETAOFM,         P22Q2M,         ETAIFM-
     2,          WA2
      WRITE(6,2390)
      WRITE(6,2500)  RTT2,           T2M,            CP2,            CV2,   -
     3           GM2
      WRITE(6,2400)
      WRITE(6,2500)  H2M,            PROF,           TROFM1,         T13P,  -
     4           P22
      WRITE(6,2410)
      WRITE(6,2500)  PRIF,           TRIFM1,         T22,            Y4,    -
     5           RTT22
      WRITE(6,2420)
      WRITE(6,2500)  ETAHCM,           WA22,         PRHC,          TAVHC
      WRITE(6,2430)
      WRITE(6,2500)  CPHC,           CVHC,           GMHC,           HHCM,  -
     7           TRHCM1
      WRITE(6,2440)
      WRITE(6,2500)  T3P,            WA3,            T3M,            CP3,   -
     8           CV3
      WRITE(6,2450)
      WRITE(6,2500)  GM3,            H3M,            H3,             T13PM,-
     9           CP13P
      WRITE(6,2460)
      WRITE(6,2500)  CV13P,          GM13P,          H13PM,          H13P,  -
     *           T13M
      WRITE(6,2470)
      WRITE(6,2500)  CP13,           CV13,           GM13,           H13M,  -
     1           H13
      WRITE(6,2480)
      WRITE(6,2500)  T22M,           CP22,           CV22,           GM22,  -
```

```
     2           H22M
      WRITE(6,2490)
      WRITE(6,2500)  H22,            T3PM,           CP3P,           CV3P, -
     3           GM3P
      WRITE(6,2140)
      WRITE(6,2500)  H3PM,           H3P,            T4M,            FAR4M,-
     4           CP4
      WRITE(6,2150)
      WRITE(6,2500)  CV4,            GM4,            H4M,            H4,   -
     5           TAVB
      WRITE(6,2160)
      WRITE(6,2500)  CPB,            CVB,            GMB,            HBM,  -
     6           HB
      WRITE(6,2170)
      WRITE(6,2500)  X5,             Y5,             RTT4,           WP4,  -
     7           HP4
      WRITE(6,2180)
      WRITE(6,2500)  WG4,            FAR41M,         T41M,           CP41, -
     8           CV41
      WRITE(6,2190)
      WRITE(6,2500)  GM41,           H41M,           H41,            DH4,  -
     9           FAR6M
      WRITE(6,2200)
      WRITE(6,2500)  T6M,            CP6,            CV6,            GM6,  -
     *           H6M
      WRITE(6,2210)
      WRITE(6,2500)  H6,             ETAB,           T7M,            FAR7M,-
     1           CP7
      WRITE(6,2220)
      WRITE(6,2500)  CV7,            GM7,            H7M,            H7,   -
     2           TAVAB
      WRITE(6,2230)
      WRITE(6,2500)  CPAB,           CVAB,           GMAB,           HABM, -
     3           HAB
      WRITE(6,2240)
      WRITE(6,2500)  CVN,            P0,             CDN,            WG7M, -
     4           WG7
      WRITE(6,2250)
      WRITE(6,2500)  ETAAB,          H2,             RTT41,          X6,   -
     5           Y6
      WRITE(6,2260)
      WRITE(6,2500)  WP41,           HP41,           WG41,           DH41, -
     6           TAM
      WRITE(6,2265)
      WRITE(6,2500)  WBLHT,          WBLLT,          WBLOV,          FNET, -
     7           FNM
      WRITE(IW,501)
      WRITE(IW,501)
      WRITE(IW,524)
      P0=P0*SFP0
      P2=P2*SFP2
      IF(IP.EQ.1)  P2D=P2
      P13=P13*SFP13
      P22=P22*SFP22
      IF(IP.EQ.1)  P22D=P22
      P3=P3*SFP3
      P4=P4*SFP4
      P41=P41*SFP41
      P6=P6*SFP6
      P7=P7*SFP7
      T2=T2*SFT2
      IF(IP.EQ.1)  T2D=T2
      T13P=T13P*SFT13
      T13=T13*SFT13
      T22=T22*SFT22
      IF(IP.EQ.1)  T22D=T22
      TAVHC=TAVHC*5000.
      T3P=T3P*SFT3
      T3=T3*SFT3
      TAVB=TAVB*5000.
      T4=T4*SFT4
      T41=T41*SFT41
      T6=T6*SFT6
      TAVAB=TAVAB*5000.
```

```
      T7=T7*SFT7
      WA2=WA2*SFWA2
      IF(IP.EQ.1)  WA2D=WA2
      WAR2=WAR2*ZC(3)*WA2D*SQRT(T2D)*CC(4)/P2D/1.5497
      WA22=WA22*SFWA22
      IF(IP.EQ.1)  WA22D=WA22
      WAR22=WAR22*ZC(4)*CC(8)*WA22D*SQRT(T22D)/P22D/1.5497
      WA3=WA3*SFWA3
      WA13=WA13*SFWA13
      WF4=WF4*SFWF4
      WG4=WG4*SFWG4
      WG41=WG41*SFWG41
      WG6=WG6*SFWG6
      WF7=WF7*SFWF7
      WG7=WG7*SFWG7
      H2=H2*.5*SFT2
      H13P=H13P*.5*SFT13
      H13=H13*.5*SFT13
      H22=H22*.5*SFT22
      H3P=H3P*.5*SFT3
      H3=H3*.5*SFT3
      HB=HB*.5*SFT4
      H4=H4*.5*SFT4
      DH4=DH4*SFDH4
      H41=H41*.5*SFT41
      DH41=DH41*SFDH41
      H6=H6*.5*SFT6
      HAB=HAB*.5*SFT7
      H7=H7*.5*SFT7
      PROF=PROF*15.
      PRIF=PRIF*15.
      PRHC=PRHC*15.
      FAR4M=FAR4M*.08
      FAR41M=FAR41M*.08
      FAR6M=FAR6M*.08
      FAR7M=FAR7M*.08
      XNH=XNH*SFXNH
      XNL=XNL*SFXNL
      ETAOFM=ETAOFM*CC(5)
      ETAIFM=ETAIFM*CC(7)
      ETAHCM=ETAHCM*CC(9)
      WBLHT=WBLHT*.2*SFWA22
      WBLLT=WBLLT*.02*SFWA22
      WBLOV=WBLOV*.002*SFWA22
      FNET=FNET*SFFN
      FNM=FNM*SFFG
      WRITE(6,2010)
      WRITE(6,2500)  P0,            P2,            P13,           P22,   -
     7             P3
      WRITE(6,2020)
      WRITE(6,2500)  P4,            P41,           P6,            P7,    -
     8             T2
      WRITE(6,2030)
      WRITE(6,2500)  T13P,          T13,           T22,           TAVHC,-
     9             T3P
      WRITE(6,2040)
      WRITE(6,2500)  T3,            TAVB,          T4,            T41,   -
     *             T6
      WRITE(6,2050)
      WRITE(6,2500)  TAVAB,         T7,            WA2,           WAR2,  -
     1             WA22
      WRITE(6,2060)
      WRITE(6,2500)  WAR22,         WA3,           WA13,          WF4,   -
     2             WG4
      WRITE(6,2070)
      WRITE(6,2500)  WG41,          WG6,           WF7,           WG7,   -
     3             H2
      WRITE(6,2080)
      WRITE(6,2500)  H13P,          H13,           H22,           H3P,   -
     4             H3
      WRITE(6,2090)
      WRITE(6,2500)  HB,            H4,            DH4,           H41,   -
     5             DH41
      WRITE(6,2100)
```

```
      WRITE(6,2500)  H6,              HAB,             H7,              CDN,  -
     6             CVN
      WRITE(6,2110)
      WRITE(6,2500)  XNH,             XNL,             PROF,            ETAOFM-
     7,            PRIF
      WRITE(6,2120)
      WRITE(6,2500)  ETAIFM,          PRHC,            ETAHCM,          ETAB, -
     8             ETAAB
      WRITE(6,2130)
      WRITE(6,2500)  FAR4M,           FAR41M,          FAR6M,           FAR7M
      WRITE(6,2265)
      WRITE(6,2500)  WBLHT,           WBLLT,           WBLOV,           FNET, -
     9             FNM
      RETURN
C******CHECK OF DIGITAL COEFFICIENTS (UNMODIFIED AND MODIFIED)
   30 CONTINUE
      K1=0
      DO 35 I=1,125
      IF(ABS(UDC(I)).LT.1.0) GO TO 35
      K1=K1+1
      GLD(K1)=I
   35 CONTINUE
      K3=0
      DO 37 I=1,125
      IF(UDC(I).NE.0.0) GO TO 37
      K3=K3+1
      ZD(K3)=I
   37 CONTINUE
      K4=0
      DO 45 I=1,125
      IF(ABS(DC(I)).LT.1.0) GO TO 45
      K4=K4+1
      FD(K4)=I
   45 CONTINUE
      DO 50 I=1,125
      DDC(I)=DC(I)
      IF(DC(I).GE.1.0)  DDC(I)=.99999
      IF(DC(I).LE.-1.0)  DDC(I)=-.99999
   50 CONTINUE
C******PRINT OUT OF DIGITAL COEFFICIENTS (UNMODIFIED AND MODIFIED), CORRECTION
C      FACTORS, AND ERROR RATIOS
      WRITE(IW,502)
      WRITE(IW,525)
      WRITE(IW,526) (UDC(I),I=1,125)
      WRITE(IW,501)
      IF(K1.NE.0) WRITE(IW,527)
      IF(K1.NE.0) WRITE(IW,532) (GLD(I),I=1,K1)
      WRITE(IW,501)
      IF(K3.NE.0) WRITE(IW,528)
      IF(K3.NE.0) WRITE(IW,532) (ZD(I),I=1,K3)
      WRITE(IW,501)
      WRITE(IW,529)
      WRITE(IW,526)(CC(I),I=1,50)
      WRITE(IW,504)
      WRITE(IW,501)
      WRITE(IW,530)
      WRITE(IW,533)(DDC(I),I=1,125)
      WRITE(IW,501)
      IF(K4.NE.0) WRITE (IW,531)
      IF(K4.NE.0) WRITE (IW,532) (FD(I),I=1,K4)
      WRITE(IW,504)
C******OUTPUT OF DATA FOR USE IN THE HYBRID PROGRAM
C******PUNCH OUT DIGITAL COEFFICIENTS AND INLET & BLEED FLAGS
      IF((JP.EQ.1).OR.(JPD.EQ.1)) GO TO 51
      GO TO 52
   51 CONTINUE
      WRITE(IW,501)
      WRITE(IW,534)
      WRITE(IPNCH,500)
      WRITE(IW,535)
      WRITE(IPNCH,533) (DDC(I),I=1,125)
      WRITE(IW,536)
      WRITE(IPNCH,540) INLET,KBH,KBL,KBV
   52 CONTINUE
```

```
C******PUNCH OUT INTEGRATOR,DAC IC,POT DATA
      IF((JP.EQ.1).OR.(JPA.EQ.1)) GO TO 53
      GO TO 85
   53 CONTINUE
      WRITE(IW,537)
      WRITE(IW,538)
      WRITE(IW,539)
      DO 70 K=1,10
      N=IG(K)
      GAIN(K)=10**N
      IF(K.GT.6) GO TO 60
      J=2*K
      JJ=J-1
      WRITE(IPNCH,541) AADR(JJ),AADR(J),IG(K)
      GO TO 65
   60 J=K+6
      WRITE(IPNCH,541) AADR(J),IG(K)
   65 CONTINUE
   70 CONTINUE
      WRITE(IPNCH,533) (DACI(I),I=1,24)
      DO 80  I=1,53
      WRITE(IPNCH,542) PADR(I),PVAL(I)
   80 CONTINUE
   85 CONTINUE
C******ERROR RATIOS OUTPUT
   90 WRITE(IW,501)
      WRITE(IW,504)
      WRITE(IW,543)
      WRITE(IW,545)
      WRITE(IW,551)(P2R(I),T2R(I),WA2R(I),T13PR(I),P22R(I),I,I=1,NTOTAL)
      WRITE(IW,501)
      WRITE(IW,546)
      WRITE(IW,551)(T22R(I),WA22R(I),T3PR(I),WA3R(I),CBLHR(I),I,I=1,NTOT-
     1AL)
      WRITE(IW,501)
      WRITE(IW,547)
      WRITE(IW,551)(WG4R(I),CBLLR(I),WG41R(I), CBLVR(I),ETABR(I),I,I=1,N-
     1TOTAL)
      WRITE(IW,501)
      WRITE(IW,548)
      WRITE(IW,551)(DH4TR(I),DH41TR(I),P0R(I),WG7R(I),ETAABR(I),I,I=1,NT-
     1OTAL)
      WRITE(IW,502)
      WRITE(IW,501)
      WRITE(IW,549)
      WRITE(IW,551)(DH4QR(I),DH41QR(I),DP13R(I),DP6R(I),P5R(I),I,I=1,NTO-
     1TAL)
      WRITE(IW,501)
      WRITE(IW,550)
      WRITE(IW,552) (FGR(I),I,I=1,NTOTAL)
  500 FORMAT(1X)
  501 FORMAT(/)
  502 FORMAT(1H1)
  503 FORMAT(1H1/)
  504 FORMAT(1H1//)
  510 FORMAT(3X,26HOPERATING POINT INPUT DATA,5X,8HPT. NO. ,I2,5X,6HPLA -
     1= ,I3/)
  511 FORMAT(6X,2HP0,10X,2HP2,10X,3HP13,9X,3HP22,9X,2HP3)
  512 FORMAT(6X,2HP4,10X,3HP41,9X,2HP5,10X,2HP6,10X,2HP7)
  513 FORMAT(6X,3HTAM,9X,2HT2,10X,3HT13,9X,3HT22,9X,2HT3)
  514 FORMAT(6X,2HT4,10X,3HT41,9X,2HT6,10X,2HT7,10X,3HWA2)
  515 FORMAT(6X,4HWA13,8X,4HWA22,8X,3HWA3,9X,3HWG4,9X,4HWG41)
  516 FORMAT(6X,3HWG6,9X,3HWG7,9X,3HDH4,9X,4HDH41,8X,4HETAB)
  517 FORMAT(6X,5HETAAB,7X,2HFN,10X,3HXNL,9X,3HXNH,9X,3HWF4)
  518 FORMAT(6X,3HWF7,9X,2HA8,10X,2HAE,10X,3HALT,9X,3HXMN)
  519 FORMAT(6X,3HCDN,9X,3HCVN,9X,4HCIVV,8X,4HRCVV,8X,2HFG/)
  520 FORMAT(5F12.5)
  521 FORMAT(3X,26HOPERATING POINT ADC INPUTS)
  522 FORMAT(3X,27HOPERATING POINT DAC OUTPUTS)
  523 FORMAT(3X,31HOTHER OPERATING POINT VARIABLES)
  524 FORMAT(3X,29HUNSCALED OPERATING POINT DATA)
  525 FORMAT(3X,31HUNMODIFIED DIGITAL COEFFICIENTS/)
  526 FORMAT(5(F8.5,2X))
  527 FORMAT(3X,33HDIGITAL COEFFICIENTS > OR = |1.0|/)
```

```
  528 FORMAT(3X,26HDIGITAL COEFFICIENTS = 0.0/)
  529 FORMAT(3X,18HCORRECTION FACTORS/)
  530 FORMAT(3X,29HMODIFIED DIGITAL COEFFICIENTS/)
  531 FORMAT(3X,43HMODIFIED DIGITAL COEFFICIENTS > OR = 11.01 /)
  532 FORMAT(5(I5,2X))
  533 FORMAT((3X,5(1X,F6.5,2X)))
  534 FORMAT(3X,16HMAP DATA PUNCHED/)
  535 FORMAT(3X,31HPUNCH MODIFIED COEFFICIENT DATA/)
  536 FORMAT(3X,21HINLET AND BLEED SPECS/)
  537 FORMAT(3X,21HPUNCH INTEGRATOR DATA/)
  538 FORMAT(3X,17HPUNCH DAC IC DATA/)
  539 FORMAT(3X,14HPUNCH POT DATA/)
  540 FORMAT((1X,4(I7,2X)))
  541 FORMAT(3I4)
  542 FORMAT(1X,A4,1X,F5.4)
  543 FORMAT(3X,38HOFF-DESIGN MODELING ERROR COEFFICIENTS/)
  545 FORMAT(6X,3HP2R,9X,3HT2R,9X,4HWA2R,8X,5HT13PR,7X,4HP22R)
  546 FORMAT(6X,4HT22R,8X,5HWA22R,7X,4HT3PR,8X,4HWA3R,8X,5HCBLHR)
  547 FORMAT(6X,4HWG4R,8X,5HCBLLR,7X,5HWG41R,7X, 5HCBLVR,6X,5HETABR)
  548 FORMAT(6X,5HDH4TR,7X,6HDH41TR,6X,3HP0R,9X,4HWG7R,8X,6HETAABR)
  549 FORMAT(6X,5HDH4QR,7X,6HDH41QR,6X,5HDP13R,7X,4HDP6R,8X,3HP5R)
  550 FORMAT(6X,3HFGR)
  551 FORMAT(5F12.5,5X,I2)
  552 FORMAT(F12.5,53X,I2)
 2010 FORMAT(/,9X,81HP0,                  P2,             P13,            P22,   -
     7              P3                    )
 2020 FORMAT(/,9X,81HP4,                  P41,            P6,             P7,    -
     8              T2                    )
 2030 FORMAT(/,9X,81HT13P,                T13,            T22,            TAVHC,-
     9              T3P                   )
 2040 FORMAT(/,9X,81HT3,                  TAVB,           T4,             T41,   -
     *              T6                    )
 2050 FORMAT(/,9X,81HTAVAB,               T7,             WA2,            WAR2M,-
     1              WA22                  )
 2060 FORMAT(/,9X,81HWAR22M,              WA3,            WA13,           WF4,   -
     2              WG4                   )
 2070 FORMAT(/,9X,81HWG41,                WG6,            WF7,            WG7,   -
     3              H2                    )
 2080 FORMAT(/,9X,81HH13P,                H13,            H22,            H3P,   -
     4              H3                    )
 2090 FORMAT(/,9X,81HHB,                  H4,             DH4,            H41,   -
     5              DH41                  )
 2100 FORMAT(/,9X,81HH6,                  HAB,            H7,             CD7,   -
     6              CV8                   )
 2110 FORMAT(/,9X,81HXNH,                 XNL,            PROF,           ETAOFM-
     7,             PRIF                  )
 2120 FORMAT(/,9X,81HETAIFM,              PRHC,           ETAHCM,         ETAB,  -
     8              ETAAB                 )
 2130 FORMAT(/,9X,81HFAR4M,               FAR41M,         FAR6M,          FAR7M  -
     9                                    )
 2270 FORMAT(/,9X,81HP13,                 P3,             P4,             P41,   -
     1              P6                    )
 2280 FORMAT(/,9X,81HP7                   T13,            T3,             T4,    -
     2              T41                   )
 2290 FORMAT(/,9X,81HT6,                  T7,             WA13,           WG6,   -
     3              XNL                   )
 2300 FORMAT(/,9X,81HXNH,                 WF4,            WF7,            A8,    -
     4              AE                    )
 2310 FORMAT(/,9X,81HALT,                 XMN,            CIVV,           RCVV   -
     5              TAM                   )
 2320 FORMAT(/,9X,81HP2,                  T2,             X3,             WAR2   -
     6                                    )
 2330 FORMAT(/,9X,81HX4,                  WAR22,          DW13,           DTQW13-
     7,             DW3                   )
 2340 FORMAT(/,9X,81HDTQW3,               DW4,            DTQW4,          DW41,  -
     8              DTQW41                )
 2350 FORMAT(/,9X,81HDW6,                 DTQW6,          DW7,            DTQW7,-
     9              DXNL                  )
 2360 FORMAT(/,9X,81HDXNH                                                        -
     *                                    )
 2370 FORMAT(/,9X,81HALTM,                XMNM,           P2A,            T2A,   -
     1              P0A                   )
 2380 FORMAT(/,9X,81HY3,                  ETAOFM,         P22Q2M,         ETAIFM-
     2,             WA2                   )
```

```
 2390 FORMAT(/,9X,81HRTT2,                T2M,        CP2,        CV2,   -
     3             GM2                    )
 2400 FORMAT(/,9X,81HH2M,                 PROF,       TROFM1,     T13P,  -
     4             P22                    )
 2410 FORMAT(/,9X,81HPRIF,                TRIFM1,     T22,        Y4,    -
     5             RTT22                  )
 2420 FORMAT(/,9X,81HETAHCM,                WA22,     PRHC,      TAVHC,  -
     6                                    )
 2430 FORMAT(/,9X,81HCPHC,                CVHC,       GMHC,       HHCM,  -
     7             TRHCM1                 )
 2440 FORMAT(/,9X,81HT3P,                 WA3,        T3M,        CP3,   -
     8             CV3                    )
 2450 FORMAT(/,9X,81HGM3,                 H3M,        H3,         T13PM,-
     9             CP13P                  )
 2460 FORMAT(/,9X,81HCV13P,               GM13P,      H13PM,      H13P,  -
     *             T13M                   )
 2470 FORMAT(/,9X,81HCP13,                CV13,       GM13,       H13M,  -
     1             H13                    )
 2480 FORMAT(/,9X,81HT22M,                CP22,       CV22,       GM22,  -
     2             H22M                   )
 2490 FORMAT(/,9X,81HH22,                 T3PM,       CP3P,       CV3P,  -
     3             GM3P                   )
 2140 FORMAT(/,9X,81HH3PM,                H3P,        T4M,        FAR4M,-
     4             CP4                    )
 2150 FORMAT(/,9X,81HCV4,                 GM4,        H4M,        H4,    -
     5             TAVB                   )
 2160 FORMAT(/,9X,81HCPB,                 CVB,        GMB,        HBM,   -
     6             HB                     )
 2170 FORMAT(/,9X,81HX5,                  Y5,         RTT4,       WP4,   -
     7             HP4                    )
 2180 FORMAT(/,9X,81HWG4,                 FAR41M,     T41M,       CP41,  -
     8             CV41                   )
 2190 FORMAT(/,9X,81HGM41,                H41M,       H41,        DH4,   -
     9             FAR6M                  )
 2200 FORMAT(/,9X,81HT6M,                 CP6,        CV6,        GM6,   -
     *             H6M                    )
 2210 FORMAT(/,9X,81HH6,                  ETAB,       T7M,        FAR7M,-
     1             CP7                    )
 2220 FORMAT(/,9X,81HCV7,                 GM7,        H7M,        H7,    -
     2             TAVAB                  )
 2230 FORMAT(/,9X,81HCPAB,                CVAB,       GMAB,       HABM,  -
     3             HAB                    )
 2240 FORMAT(/,9X,81HCVN,                 P0,         CDN,        WG7M,  -
     4             WG7                    )
 2250 FORMAT(/,9X,81HETAAB,               H2,         RTT41,      X6,    -
     5             Y6                     )
 2260 FORMAT(/,9X,81HWP41,                HP41,       WG41,       DH41,  -
     6             TAM                    )
 2265 FORMAT(/,9X,81HWBLHT,               WBLLT,      WBLOV,      FNET,  -
     7             FG                     )
 2500 FORMAT(1X,5F15.7)
 2544 FORMAT(1X,5E15.7)
      RETURN
      END
C
C
C***********PROCOM**********
C
C
      SUBROUTINE PROCOM(T,FA,CP,CV,GAM,H)
C     PRODUCTS OF COMBUSTION ROUTINE
      IF(T.GE..46000) GO TO 50
      IF(T.GE..24000) GO TO 40
      CPA= .48068-(.12464-T/.98246)*T
      HA= .00176+(.56558+.14075*T)*T
      GO TO 60
   40 CPA= .40528+(.52545-.38182*T)*T
      HA= .00119+(.56298+.16150*T)*T
      GO TO 60
   50 CPA= .46063+(.30024-.15378*T)*T
      HA= -.01870+(.64968+.06700*T)*T
   60 AMW= .57940-.00144*FA
      R= .07945/AMW
      TD= .70000 -T
```

```
      CPF= .93330 - (.29350+.81750*TD)*TD
      HF= -.03305 +(.63624+.38625*T)*T
      H= (.66667*HA + .10667*HF*FA)/(.50000 + .04000 * FA)
      CP= (.80000*CPA+.12800*CPF*FA)/(.80000+.06400*FA)
      CV= CP - R
      GAM= .50000*CP/CV
      RETURN
      END
C
C
C**********SPOOL**********
C
C
      SUBROUTINE SPOOL (XI,XN,SFWG,SFDH,SFXN,TSC,IGAIN,POT)
C     SPOOL ROUTINE
      DIMENSION POT(1)
      XN=XN*SFXN
      XJ=9339.6
      PI= 3.1416
      POT1I =900.*XJ*SFDH*SFWG/(PI*PI*XI*SFXN*SFXN*TSC)
      IF(POT1I.GT.1.) GO TO 2
      IGAIN=0
      POT(1)= POT1I
      GO TO 7
    2 IF(POT1I.GT.10.) GO TO 3
      IGAIN=1
      POT(1)= POT1I/10.
      GO TO 7
    3 IF(POT1I.GT.100.) GO TO 4
      IGAIN=2
      POT(1)= POT1I/100.
      GO TO 7
    4 IF(POT1I.GT.1000.) GO TO 5
      IGAIN=3
      POT(1)=POT1I/1000.
      GO TO 7
    5 IF(POT1I.GT.10000.) GO TO 6
      IGAIN=4
      POT(1)=POT1I/10000.
      GO TO 7
    6 IGAIN=5
      POT(1)=POT1I/100000.
    7 POT(2)= XN/SFXN
      XN=XN/SFXN
      RETURN
      END
C
C
C**********TRAT**********
C
C
      SUBROUTINE TRAT(N,PRC,GAM,TR)
C     TEMPERATURE RATIO ROUTINE
      DIMENSION N1(3),N2(3),N3(3)
      DIMENSION XY(50),X1(25),Y1(25)
      EQUIVALENCE (X1(1),XY(1)),(Y1(1),XY(26))
      DATA N1/5,1,25/,N2/5,1,25/,N3/5,1,25/
      DATA X1/.06667,.07333,.08000,.09333,.10667,.12000,.13333,      -
     1  .14667,.16667,.20000,.23333,.26667,.30000,.33333,.36667,      -
     2  .40000,.46667,.53333,.60000,.66667,.73333,.80000,.86667,      -
     3  .93333,.99999/
      DATA Y1/.00000,.02001,.03872,.07292,.10367,.13169,.15749,      -
     1  .18145,.21452,.26362,.30699,.34599,.38152,.41424,.44462,      -
     2  .47301,.52492,.57159,.61412,.65328,.68964,.72363,.75557,      -
     3  .78575,.81437/
  150 IF(GAM.GE..67500) GO TO 300
      IF(PRC.GE..33333) GO TO 200
      S= .50526 + .77229*PRC -.95164*PRC*PRC
      GO TO 700
  200 S= .58745 + .23850*PRC - .09000*PRC*PRC
      GO TO 700
  300 IF(PRC.GE..33333) GO TO 400
      S= .45911 + .89475*PRC - (PRC*PRC)/.92895
      GO TO 700
```

```
  400 S= .55215 + .29319 * PRC - .10919* PRC*PRC
  700 C= ((.67500 - GAM)*S)/.02500
      IF(N.NE.1) GO TO 750
      TRC=FUN1(N1,X1,PRC)
      GO TO 900
  750 IF(N.NE.2) GO TO 800
      TRC=FUN1(N2,X1,PRC)
      GO TO 900
  800 TRC=FUN1(N3,X1,PRC)
  900 TR= (TRC*(.80000 -.16000*C))/.80000
 1000 RETURN
      END
C
C
C***********VOLUME**********
C
C
      SUBROUTINE VOLUME(P,TT,V,W,WS,TSC,SFWG,SFP,SFTT,IGAIN,POT,SFW)
C     VOLUME ROUTINE
      DIMENSION POT(1)
      P=P*SFP
      TT=TT*SFTT
      RA= 640.1
      W= P*V / (RA*TT)
      DO 10 J=1,3
      IF(W .GT. (1.0*10.**(J-2))) GO TO 10
      SFW=1.0*10.**(J-3)
    1 WS=W/SFW
      IF(WS .LE. .8) GO TO 2
      SFW=SFW+0.5*10.**(J-3)
      GO TO 1
   10 CONTINUE
    2 POT(1)= WS
      POT2I= SFWG/(SFW * TSC)
      IF(POT2I.GT. 1.)  GO TO 3
      IGAIN= 0
      POT(2)= POT2I
      GO TO 8
    3 IF(POT2I.GT.10. ) GO TO 4
      IGAIN= 1
      POT(2)=POT2I/10.
      GO TO 8
    4 IF(POT2I.GT.100.)  GO TO 5
      IGAIN= 2
      POT(2)= POT2I/100.
      GO TO 8
    5 IF(POT2I.GT.1000.) GO TO 6
      IGAIN= 3
      POT(2)= POT2I/1000.
      GO TO 8
    6 IF(POT2I.GT.10000.) GO TO 7
      IGAIN= 4
      POT(2)= POT2I/10000.
      GO TO 8
    7 IGAIN= 5
      POT(2)= POT2I/100000.
    8 POT(3)= POT(2)
      POT(4)=TT/SFTT
      POT(5)= (RA*SFW*SFTT)/(10.*V*SFP)
      P=P/SFP
      TT=TT/SFTT
      RETURN
      END
C
C
```

Fortran Symbols

MAIN. – Same symbol used for unscaled and scaled variables.

AE	exhaust nozzle exit area, cm^2 (in^2)
ALT	altitude, m (ft)
APRINT	index for subroutine ANALOG (integer)

A8 exhaust nozzle throat area, cm^2 (in^2)
BSCIVV bias on fan variable-geometry parameter, deg
BSRCVV bias on compressor variable-geometry parameter, deg
CDN exhaust nozzle flow coefficient
CIVV fan variable-geometry parameter, deg
CVN exhaust nozzle velocity coefficient
DH4 high-pressure-turbine enthalpy drop, J/kg (Btu/lbm)
DH41 low-pressure-turbine enthalpy drop, J/kg (Btu/lbm)
ETAAB augmentor efficiency
ETAB combustor efficiency
FN net thrust, N (lbf)
FG gross thrust, N (lbf)
INLET inlet configuration option (integer)
IP index on operating points (integer)
IPNCH card punch device number (integer)
IPRINT index for subroutine PRINT (integer)
IR card input device number (integer)
IW line printer device number (integer)
JP punch option (integer) (1 for output of all data)
JPA punch option (integer) (1 for output of analog data)
JPD punch option (integer) (1 for output of digital data)
KBH high-pressure-turbine bleed flow indicator (integer)
KBL low-pressure-turbine bleed flow indicator (integer)
KBLWHT fraction of high-pressure-turbine cooling bleed doing work
KBLWLT fraction of low-pressure-turbine cooling bleed doing work
KBV overboard bleed flow indicator (integer)
NAUG number of augmented operating points input to host program
NDRY number of dry operating points input to host program
NTOTAL total number of operating points input to host program
PLA power lever angle (operating point label), deg
PI total pressure at station I, N/cm^2 (psia)
RCVV compressor variable-geometry parameter, deg
RTT2 square root of scaled fan inlet total temperature
RTT22 square root of scaled compressor inlet total temperature
RTT4 square root of scaled high-pressure-turbine inlet total temperature
RTT41 square root of scaled low-pressure-turbine inlet total temperature
SF$\underline{X}$ scale factor on variable X, appropriate units
TAM sea-level ambient temperature, K (°R)
TI total temperature at station I, K (°R)
WAI airflow rate leaving station I, kg/sec (lbm/sec)
WBLHT high-pressure-turbine cooling bleed flow rate, kg/sec (lbm/sec)
WBLLT low-pressure-turbine cooling bleed flow rate, kg/sec (lbm/sec)
WBLOV overboard bleed flow rate, kg/sec (lbm/sec)
WF4 combustor fuel flow rate, kg/sec (lbm/sec)
WF7 augmentor fuel flow rate, kg/sec (lbm/sec)
WGI gas flow rate leaving station I, kg/sec (lbm/sec)
XMN Mach number
XNH high-spool rotor speed, rpm
XNL low-spool rotor speed, rpm

ANALOG. – All variables are scaled unless otherwise specified.

AADR(k) integrator address array (integer), $k=1$ to 16
AE exhaust nozzle exit area
ALT altitude
AQL13 ratio of bypass duct area to length, cm (in.)
AQL6 ratio of augmentor duct area to length, cm (in.)
CIVV fan variable-geometry parameter
DACI(k) DAC initial condition array, $k=1$ to 24
DP13R(k) bypass duct model evaluation ratio array, $k=1$ to NTOTAL
DP6R(k) augmentor duct model evaluation ratio array, $k=1$ to NTOTAL
GAIN(k) integrator gain array (integer), $k=1$ to 10
I integer index

IG(k)	integrator gain integer array, $k=1$ to 10
IK	integrator gain integer
IP	index on operating points (integer)
IR	card reader device number (integer)
IW	line printer device number (integer)
J	integer index
JJ	integer index
K	integer index
N	integer index
PADR(k)	potentiometer address array (alphanumeric), $k=1$ to 53
POT(k)	potentiometer setting array, $k=1$ to 5
PVAL(k)	potentiometer setting array, $k=1$ to 53
PI	total pressure at station I
RCVV	compressor variable-geometry parameter
SFX	scale factor on variable X, appropriate units
SWI	stored mass at station I
TAM	sea-level ambient temperature
TI	total temperature at station I
VI	volume at station I, cm^3 (in^3)
WA13	bypass duct flow rate
WF4	combustor fuel flow rate
WF7	augmentor fuel flow rate
WG6	augmentor flow rate
WI	stored mass at station I, kg (lbm)
XC(k)	map scale factor array for X input variables, $k=1$ to 6
XIH	high-rotor moment of inertia, N cm sec^2 (lbf in sec^2)
XIL	low-rotor moment of inertia, N cm sec^2 (lbf in sec^2)
XNH	high-spool rotor speed
XNL	low-spool rotor speed

DATAIN

F(k)	map data array (scaled fraction), k determined by calling program
I	integer index
INC	number of points defining map (integer)
IPNCH	card punch device number (integer)
IR	card input device number (integer)
ISC(k)	map scale factor format array (alphanumeric), $k=1$ to 4
IW	line printer device number (integer)
IX(k)	map X input data format array (alphanumeric), $k=1$ to 4
IY(k)	map Y input data format array (alphanumeric), $k=1$ to 4
IZ1(k)	map Z1 output data format array (alphanumeric), $k=1$ to 4
IZ2(k)	map Z2 output data format array (alphanumeric), $k=1$ to 4
IZ3(k)	map Z3 output data format array (alphanumeric), $k=1$ to 4
IZ4(k)	map Z4 output data format array (alphanumeric), $k=1$ to 4
J	integer index
JF	integer index
JMNPT	integer index
JP	punch option (integer) (1 for output of all data)
JPD	punch option (integer) (1 for output of digital data)
JS	integer index
K	integer index
KPNCH	map data punch integer (1 if JP or JPD equal to 1)
L	integer index
M	map number (integer)
N(k)	map integer array, k determined by calling program
NCOM	integer indicating rectilinear map data (nonzero for rectilinear map data)
NCV	number of curves defining map
NPCT	number of functions in map
NF	integer index
NPT	number of points per curve
TEST	map data increment used for detecting overflows (scaled fraction)
VALS(k)	array used to store unscaled Y and Z_i map data (floating point), $k=1$ to 25
XSC	map scale factor for X input variable (floating point)
XVALS(k)	array used to store unscaled X map data (floating point), $k=1$ to 25
YSC	map scale factor for Y input variable (floating point)
ZSC(k)	map scale factor array for Z_i output variables (floating point), $k=1$ to 4

DCOEF. – All variables are scaled unless otherwise specified.

AE	exhaust nozzle exit area
ALT	altitude
ALTM	FLCOND-rescaled altitude
BETAAB	augmentor temperature interpolation constant

BETAB	combustor temperature interpolation constant
BETAHC	compressor temperature interpolation constant
CC(k)	correction factor array (floating point), $k = 1$ to 50
CDN	exhaust nozzle flow coefficient
CIVV	fan variable-geometry parameter
CPAB	average specific heat at constant pressure in augmentor
CPB	average specific heat at constant pressure in combustor
CPHC	average specific heat at constant pressure in compressor
CPI	specific heat at constant pressure at station I
CSHIFT	compressor variable-geometry effect on corrected flow
CVAB	average specific heat at constant volume in augmentor
CVB	average specific heat at constant volume in combustor
CVHC	average specific heat at constant volume in compressor
CVN	exhaust nozzle velocity coefficient
CVI	specific heat at constant volume at station I
DC(k)	corrected digital coefficient array, $k = 1$ to 125
DH4	high-pressure-turbine enthalpy drop
DH41	low-pressure-turbine enthalpy drop
ERRHAB	difference between computed and desired augmentor specific enthalpy
ERRHB	difference between computed and desired compressor specific enthalpy
ERRTRHC	difference between computed and desired compressor temperature rise parameter
ETAAB	augmentor efficiency
ETAB	combustor efficiency
ETAHCM	compressor efficiency
ETAIFM	fan inside-diameter efficiency
ETAOFM	fan outside-diameter efficiency
FARIM	fuel-air ratio at station I
FG	gross thrust
FGM3	function of specific heat ratio at station 3
FNET	net thrust
FNM	nozzle gross thrust
FSHIFT	fan variable-geometry effect on corrected flow
F1(k)	fan variable-geometry-effects ap data array, $k = 1$ to 322
F2(k)	compressor variable-geometry-effects map data array, $k = 1$ to 322
F3(k)	fan map data array, $k = 1$ to 854
F4(k)	compressor map data array, $k = 1$ to 518
F5(k)	high-pressure-turbine map data array, $k = 1$ to 224
F6(k)	low-pressure-turbine map data array, $k = 1$ to 224
GMAB	average specific heat ratio in augmentor
GMB	average specific heat ratio in combustor
GMHC	average specific heat ratio in compressor
GMI	specific heat ratio at station I
HAB	average specific enthalpy in augmentor
HABD	desired augmentor specific enthalpy
HB	combustor specific enthalpy
HBD	desired combustor specific enthalpy
HHCM	PROCOM-rescaled compressor specific enthalpy
HP4	high-pressure-turbine enthalpy drop parameter
HP41	low-pressure-turbine enthalpy drop parameter
HI	specific enthalpy at station I
HIM	PROCOM-rescaled specific enthalpy at station I
I	integer index
IP	index on operating points (integer)
IW	line printer device number (integer)
KBH	high-pressure-turbine bleed flow indicator (integer)
KBL	low-pressure-turbine bleed flow indicator (integer)
KBLWHT	fraction of high-pressure-turbine cooling bleed doing work
KBLWLT	fraction of low-pressure-turbine cooling bleed doing work
KBV	overboard bleed flow indicator (integer)
NDRY	number of dry operating points input to host program
N1(k)	fan variable-geometry-effects map integer array, $k = 1$ to 5
N2(k)	compressor variable-geometry-effects map integer array, $k = 1$ to 5
N3(k)	fan map integer array, $k = 1$ to 5

N4(k)	compressor map integer array, $k=1$ to 5
N5(k)	high-pressure-turbine map integer array, $k=1$ to 5
N6(k)	low-pressure-turbine map integer array, $k=1$ to 5
N7(k)	nozzle functions integer array, $k=1$ to 3
PRHC	TRAT-rescaled compressor pressure ratio
PRIF	TRAT-rescaled fan inside-diameter pressure ratio
PROF	TRAT-rescaled fan outside-diameter pressure ratio
PI	total pressure at station I
POA	FLCOND-rescaled ambient pressure
POQT7	exhaust nozzle pressure ratio
P2A	FLCOND-rescaled fan inlet total pressure
P22Q2M	fan inside-diameter pressure ratio
RCVV	compressor variable-geometry parameter
SFX	scale factor on variable X, appropriate units
TAM	sea-level ambient temperature
TAVAB	average total temperature in augmentor
TAVB	average total temperature in combustor
TAVHC	average total temperature in compressor
TRHCM1	compressor temperature rise parameter
TRHC1D	desired compressor temperature rise parameter
TRIFM1	fan inside-diameter temperature rise parameter
TROFM1	fan outside-diameter temperature rise parameter
TOA	FLCOND-rescaled ambient temperature
TI	total temperature at station I
TIM	PROCOM-rescaled total temperature at station I
T2A	FLCOND-rescaled fan inlet total temperature
UDC(k)	uncorrected digital coefficient array, $k=1$ to 125
WAR2	fan corrected flow rate
WAR2M	nominal fan corrected flow rate
WAR22	compressor corrected flow rate
WAI	airflow rate leaving station I
WBLHT	high-pressure-turbine cooling bleed flow rate
WBLLT	low-pressure-turbine cooling bleed flow rate
WBLOV	overboard bleed flow rate
WF4	combustor fuel flow rate
WF7	augmentor fuel flow rate
WGI	gas flow rate leaving station I
WG7M	NOZZL-rescaled exhaust nozzle flow rate
WP4	high-pressure-turbine flow parameter
WP41	low-pressure-turbine flow parameter
XC(k)	map scale factor array for X input variables (floating point), $k=1$ to 6
XF	pressure area term in gross thrust calculation
XMN	Mach number
XMNM	FLCOND-rescaled Mach number
XNH	high-spool rotor speed
XNL	low-spool rotor speed
XI	X input for map I
YC(k)	map scale factor array for Y input variables (floating point), $k=1$ to 6
YI	Y input for map number I
Y71(k)	exhaust nozzle flow coefficient function data array, $k=1$ to 15
Y72(k)	exhaust nozzle velocity coefficient function data array, $k=1$ to 15
ZC(k)	map scale factor array for Z output variables (floating point), $k=1$ to 12

DUCT. – Same symbol used for unscaled and scaled pressures, temperature, and flow rate.

AQL	ratio of duct area to length, cm (in.)
GC	gravitational conversion factor, 100 cm kg/N sec^2 (386.26 lbm in/lbf sec^2)
IGAIN	integrator gain integer
PIN	duct inlet total pressure, N/cm^2 (psia)
POT(k)	potentiometer setting array, $k=1$ to 4
POT2I	required gain of PIN potentiometer-integrator combination
POT3I	required gain of POUT potentiometer-integrator combination
POT4I	required gain of feedback potentiometer-integrator combination
POUT	duct exit total pressure, N/cm^2 (psia)
RA	gas constant of air, 2.8699×10^4 N cm/kg k (640.1 in lbf/lbm °R)

SFPIN	scale factor on duct inlet total pressure, appropriate units
SFPOUT	scale factor on duct exit total pressure, appropriate units
SFTIN	scale factor on duct inlet total temperature, appropriate units
SFW	scale factor on duct inlet stored mass, appropriate units
SFWG	scale factor on flow rate through duct, appropriate units
TIN	duct inlet total temperature, K (°R)
TSC	time scale factor
V	duct inlet volume, cm^3 (in^3)
WG	flow rate through duct, kg/sec (lbm/sec)

ENGINE. – All variables are scaled unless otherwise specified.

AE	exhaust nozzle exit area
ALT	altitude
ALTM	FLCOND-rescaled altitude
CBLHR(k)	high-pressure-turbine bleed model evaluation ratio array, $k=1$ to NTOTAL
CBLLR(k)	low-pressure-turbine bleed model evaluation ratio array, $k=1$ to NTOTAL
CBLVR(k)	overboard bleed model evaluation ratio array, $k=1$ to NTOTAL
CDN	exhaust nozzle flow coefficient
CIVV	fan variable-geometry parameter
CPAB	average specific heat at constant pressure in augmentor
CPB	average specific heat at constant pressure in combustor
CPHC	average specific heat at constant pressure in compressor
CPI	specific heat at constant pressure at station I
CSHIFT	compressor variable-geometry effect on corrected flow
CVAB	average specific heat at constant volume in augmentor
CVB	average specific heat at constant volume in combustor
CVHC	average specific heat at constant volume in compressor
CVN	exhaust nozzle velocity coefficient
CVI	specific heat at constant volume at station I
DACI(k)	DAC initial condition array, $k=1$ to 24
DC(k)	corrected digital coefficient array, $k=1$ to 125
DH4	high-pressure-turbine enthalpy drop
DH4QC	high-pressure-turbine enthalpy required for torque balance
DH4QR(k)	high-pressure-turbine torque model evaluation ratio array, $k=1$ to NTOTAL
DH4TC	high-pressure-turbine enthalpy drop required for energy balance
DH4TR(k)	high-pressure-turbine temperature model evaluation ratio array, $k=1$ to NTOTAL
DH41	low-pressure-turbine enthalpy drop
DH41QC	low-pressure-turbine enthalpy drop required for torque balance
DH41QR(k)	low-pressure-turbine torque model evaluation ratio array, $k=1$ to NTOTAL
DH41TC	low-pressure-turbine enthalpy drop required for energy balance
DH41TR(k)	low-pressure-turbine temperature model evaluation ratio array, $k=1$ to NTOTAL
DTQWI	specific temperature derivative at station I
DXNH	high-rotor-speed derivative
DXNL	low-rotor-speed derivative
DWI	stored mass derivative at station I
ETAAB	augmentor efficiency
ETAABC	augmentor efficiency required for energy balance
ETAABR(k)	combustor efficiency model evaluation ratio array, $k=1$ to NTOTAL
ETAHCM	compressor efficiency
ETAIFM	fan inside-diameter efficiency
ETAOFM	fan outside-diameter efficiency
FARIM	fuel-air ratio at station I
FG	gross thrust
FGM3	function of specific heat ratio at station 3
FGPT3	compressor discharge bleed flow parameter
FGR(k)	nozzle velocity model evaluation ratio array, $k=1$ to NTOTAL
FNET	net thrust
FNM	nozzle gross thrust
FSHIFT	fan variable-geometry effect on corrected flow
F1(k)	fan variable-geometry-effects map data array, $k=1$ to 322

F2(k) compressor variable-geometry-effects map data array, $k=1$ to 322

F3(k) fan map data array, $k=1$ to 854

F4(k) compressor map data array, $k=1$ to 518

F5(k) high-pressure-turbine map data array, $k=1$ to 224

F6(k) low-pressure-turbine map data array, $k=1$ to 224

GMAB average specific heat ratio in augmentor

GMB average specific heat ratio in combustor

GMHC average specific heat ratio in compressor

GMI specific heat ratio at station I

GMIM specific heat ratio minus 1 at station I

HAB average specific enthalpy in augmentor

HABM PROCOM-rescaled augmentor specific enthalpy

HB average specific enthalpy in combustor

HBM PROCOM-rescaled combustor specific enthalpy

HHCM PROCOM-rescaled compressor specific enthalpy

HP4 high-pressure-turbine enthalpy drop parameter

HP41 low-pressure-turbine enthalpy drop parameter

HI specific enthalpy at station I

HIM PROCOM-rescaled specific enthalpy at station I

IP index on operating points (integer)

KBH high-pressure-turbine bleed flow indicator (integer)

KBL low-pressure-turbine bleed flow indicator (integer)

KBV overboard bleed flow indicator (integer)

NDRY number of dry operating points input to host program

N1(k) fan variable-geometry-effects map integer array, $k=1$ to 5

N2(k) compressor variable-geometry-effects map integer array, $k=1$ to 5

N3(k) fan map integer array, $k=1$ to 5

LTSHFT low-pressure-turbine enthalpy map shift

N4(k) compressor map integer array, $k=1$ to 5

N5(k) high-pressure-turbine map integer array, $k=1$ to 5

N6(k) low-pressure-turbine map integer array, $k=1$ to 5

N7(k) nozzle functions integer array, $k=1$ to 3

PRHC TRAT-rescaled compressor pressure ratio

PRIF TRAT-rescaled fan inside-diameter pressure ratio

PROF TRAT-rescaled fan outside-diameter pressure ratio

PI total pressure at station I

PIC calculated total pressure at station I

PIR(k) model evaluation ratio array for total pressure at station I, $k=1$ to NTOTAL

POA FLCOND-rescaled ambient pressure

POQT7 exhaust nozzle pressure ratio

P2A FLCOND-rescaled fan inlet total pressure

P22Q2M fan inside-diameter pressure ratio

RCVV compressor variable-geometry parameter

TAM sea-level ambient temperature

TAVAB average total temperature in augmentor

TAVB average total temperature in combustor

TAVHC average total temperature in compressor

TRHCM1 compressor temperature rise parameter

TRIFM1 fan inside-diameter temperature rise parameter

TROFM1 fan outside-diameter temperature rise parameter

TOA FLCOND-rescaled ambient temperature

TI total temperature at station I

TIC calculated total temperature at station I

TIM PROCOM-rescaled total temperature at station I

TIR(k) model evaluation ratio array for total temperature at station I, $k=1$ to NTOTAL

T2A FLCOND-rescaled fan inlet total temperature

WAR2 fan corrected flow rate

WAR2M nominal fan corrected flow rate

WAR22 compressor corrected flow rate

WAR22M nominal compressor corrected flow rate

WAI airflow rate leaving station I

WAIC calculated airflow rate leaving station I

WAIR(k) model evaluation ratio array for airflow leaving station I, $k=1$ to NTOTAL

WBLHT high-pressure-turbine cooling bleed flow rate

WBLLT low-pressure-turbine cooling bleed flow rate

WBLOV overboard bleed flow rate

WF4 combustor fuel flow rate

WF7 augmentor fuel flow rate

WGI gas flow rate leaving station I

WGIC calculated ratio array for gas flow rate leaving station I, $k=1$ to NTOTAL

WG7M NOZZL-rescaled exhaust nozzle flow rate

WP4 high-pressure-turbine flow parameter

WP41 low-pressure-turbine flow parameter

XF pressure-area term in gross thrust calculation

XMN Mach number

XMNM FLCOND-rescaled Mach number

XNH high-spool rotor speed

XNL low-spool rotor speed

XI *X* input for map I

X7(k) exhaust nozzle pressure ratio data array, $k=1$ to 15

X8(k) low-pressure-turbine corrected speed data array, $k=1$ to 13

YI *Y* input for map number I

Y71(k) exhaust nozzle flow coefficient function data array, $k=1$ to 15

Y72(k) exhaust nozzle velocity coefficient function data array, $k=1$ to 15

Y8(k) low-pressure-turbine enthalpy map shift function data array, k = 1 to 13

FLCOND. – All variables are scaled unless otherwise specified.

ETAI inlet efficiency

HT altitude

INLET inlet configuration option (integer)

N1(k) ambient pressure function integer array, k = 1 to 3

N2(k) $(T_2/T_0)^{\gamma_I/(\gamma_I-1)}$ function integer array, $k=1$ to 3

N3(k) $(M_0-1)^{1.35}$ function integer array, $k=1$ to 3

PS ambient pressure

PT fan inlet total pressure

PTQS isentropic inlet pressure ratio

TAS sea-level ambient pressure

TR35 $(T_2/T_0)^{\gamma_I/(\gamma_I-1)}$

TS ambient pressure

TTQS inlet temperature ratio

TT fan inlet total temperature

XMO Mach number

XMP supersonic Mach number minus 1

XPP $(M_0-1)^{1.35}$

X1(k) altitude data array, $k=1$ to 11

X2(k) inlet temperature ratio data array, $k=1$ to 10

X3(k) Mach-number-minus-1 data array, $k=1$ to 7

Y1(k) ambient pressure data array, $k=1$ to 11

Y2(k) $(T_2/T_0)^{\gamma_I/(\gamma_I-1)}$ data array, $k=1$ to 11

Y3(k) $(M_0-1)^{1.35}$ data array, $k=1$ to 7

FUN1/FUN1L/FOOR.-All variables are scaled unless otherwise specified.

F(k) function data array, *k* determined by calling program

FUN1/FUN1L function output

I *X* variable search index (integer)

MN saved function number for out-of-range message (integer)

N(k) function integer array, *k* determined by calling program

NXP number of points defining function (integer)

X saved value of XIN for out-of-range message

XFRAC fraction of X1NX2 interval covered by XIN

XIN *X* input

X1 difference between XIN and I^{th} value of *X*

X2 difference between XIN and $(I+1)^{\text{th}}$ value of *X*

MAP/MAPL/MOOR. – All variables are scaled unless otherwise specified.

F(k) map data array, *k* determined by calling program

I *X* variable search index (integer)

J	Y variable search index (integer)
KX	map data array index corresponding to X(I, J+1) (integer)
KZ	map data array index corresponding to Z(I, J+1) (integer)
LX	map data array index corresponding to X(I, J+1) (integer)
LZ	map data array index corresponding to Z(I, J+1) (integer)
MAP/MAPL	map output
MN	saved map number for out-of-range message (integer)
N(k)	map integer array, k determined by calling program
NPROD	number of points defining map
NXP	number of points per curve (integer)
MUC	number of curves in map (integer)
X	saved value of XIN for out-of-range message
XFRAC	fraction of XHI–XLO interval covered by XIN
XHI	$(I+1)^{\text{th}}$ breakpoint on interpolated YIN curve
XIN	X input
XLO	I^{th} breakpoint on interpolated YIN curve
Y	saved value of YIN for out-of-range message
Y1	difference between YIN and J^{th} value of Y in map
Y2	difference between YIN and $(J+1)^{\text{th}}$ value of Y in map
YIN	Y input
YINCR	fraction on Y(J+1)–Y(J) interval covered by YIN
ZH	map output at XHI,YIN
ZL	map output at XLO,YIN

MAPIN

BSCIVV	bias on fan variable-geometry parameter, deg
BSRCVV	bias on compressor variable-geometry parameter, deg
DC(k)	corrected digital coefficient array (scaled fraction), $k=1$ to 125
F1(k)	fan variable-geometry-effects map data array (scaled fraction), $k=1$ to 322
F2(k)	compressor variable-geometry-effects map data array (scaled fraction), $k=1$ to 322
F3(k)	fan map data array (scaled fraction), $k=1$ to 854
F4(k)	compressor map data array (scaled fraction), $k=1$ to 518
F5(k)	high-pressure-turbine map data array (scaled fraction), $k=1$ to 224
F6(k)	low-pressure-turbine map data array (scaled fraction), $k=1$ to 224
I	integer index
IR	card input device number (integer)
IW	line printer device number (integer)
K1	integer index
K2	integer index
N	integer index
N1(k)	fan variable-geometry-effects map integer array, $k=1$ to 5
N2(k)	compressor variable-geometry-effects map integer array, $k=1$ to 5
N3(k)	fan map integer array, $k=1$ to 5
N4(k)	compressor map integer array, $k=1$ to 5
N5(k)	high-pressure-turbine map integer array, $k=1$ to 5
N6(k)	low-pressure-turbine map integer array, $k=1$ to 5
SFX	scale factor on variable X, appropriate units
XC(k)	map scale factor array for X input variables, $k=1$ to 6
YC(k)	map scale factor array for Y input variables, $k=1$ to 6
ZC(k)	map scale factor array for Z output variables, $k=1$ to 12

NOZZL. – All variables are scaled unless otherwise specified.

AEQTXX	ratio of exit area to area at which sonic flow is reached
ATXX	reduced throat area that results in sonic flow
A7	throat area
A8	exit area
A8Q7	expansion ratio
CD7	flow coefficient
CV8	velocity coefficient
DC(k)	corrected digital coefficient, $k=1$ to 125
F8	gross thrust
N1(k)	critical pressure ratio function integer array, k=1 to 3

N2(k) subsonic functions integer array, $k=1$ to 3

N3(k) supersonic functions integer array, $k=1$ to 3

PE exit plane pressure

PEQ7 critical pressure ratio

PTOL difference between computed exit plane pressure and ambient pressure for subsonic exit flow

PYQT pressure ratio at which shock is in exit plane

PYQX static pressure ratio across shock

PO ambient pressure

POQT7 ambient to total pressure ratio

POYQX total to static pressure ratio across shock

P7 inlet total pressure

POYQOX total pressure ratio across shock

T7 inlet total temperature

VE exit velocity

W7 mass flow rate

XF pressure-area term in gross thrust equation

XMN Mach number upstream of shock

XMX dimensionless velocity

XSHFT area ratio shift on dimensionless velocity-pressure ratio fit

X1(k) subsonic flow area ratio data array, $k=1$ to 15

X2(k) subsonic flow pressure ratio data array, $k=1$ to 15

X3(k) supersonic flow area ratio data array, $k=1$ to 15

Y1(k) critical pressure ratio data array, $k=1$ to 15

Y21(k) subsonic dimensionless velocity data array, $k=1$ to 15

Y22(k) AEQTXX data array, $k=1$ to 15

Y31(k) XMN data array, $k=1$ to 15

Y32(k) supersonic pressure ratio data array, $k=1$ to 15

Y33(k) supersonic dimensionless velocity data array, $k=1$ to 15

PRINT. – Same symbols used for unscaled and scaled variables.

AADR(k) integrator address array (integer), $k=1$ to 16

AE exhaust nozzle exit area, cm^2 (in^2)

ALT altitude, m (ft)

ALTM FLCOND-rescaled altitude

CBLHR(k) high-pressure-turbine bleed model evaluation ratio array, $k=1$ to NTOTAL

CBLLR(k) low-pressure-turbine bleed model evaluation ration, $k=1$ to NTOTAL

CBLVR(k) overboard bleed model evaluation ratio array, $k=1$ to NTOTAL

CC(k) correction factor array (floating point), $k=1$ to 50

CDN exhaust nozzle flow coefficient

CIVV fan variable-geometry parameter, deg

CPAB average specific heat at constant pressure in augmentor, J/kg K (Btu/lbm °R)

CPB average specific heat at constant pressure in combustor, J/kg K (Btu/lbm °R)

CPHC average specific heat at constant pressure in compressor, J/kg K (Btu/lbm °R)

CPI specific heat at constant pressure at station I, J/kg K (Btu/lbm °R)

CVAB average specific heat at constant volume in augmentor, J/kg K (Btu/lbm °R)

CVB average specific heat at constant volume in combustor, J/kg K (Btu/lbm °R)

CVHC average specific heat at constant volume in compressor, J/kg K (Btu/lbm °R)

CVN exhaust nozzle velocity coefficient

CVI specific heat at constant volume at station I, J/kg K (Btu/lbm °R)

DACI(k) DAC initial condition array, $k=1$ to 24

DC(k) corrected digital coefficient array, $k=1$ to 125

DDC(k) array used to save unlimited digital coefficients, $k=1$ to 125

DH4 high-pressure-turbine enthalpy drop, J/kg (Btu/lbm)

DH4QR(k) high-pressure-turbine torque model evaluation ratio array, $k=1$ to NTOTAL

DH4TR(k) high-pressure-turbine temperature model evaluation ratio array, $k=1$ to NTOTAL

DH41 low-pressure-turbine enthalpy drop, J/kg (Btu/lbm)

DH41QR(k) low-pressure-turbine torque model evaluation ratio array, $k=1$ to NTOTAL

DH41TR(k)	low-pressure-turbine temperature model evaluation ratio array, $k=1$ to NTOTAL
DP13R(k)	bypass duct model evaluation ratio array, $k=1$ to NTOTAL
DP6R(k)	augmentor duct model evaluation ratio array, $k=1$ to NTOTAL
DTQWI	specific temperature derivative at station I (scaled)
DXNH	high-rotor-speed derivative (scaled)
DXNL	low-rotor-speed derivative (scaled)
DWI	stored mass derivative at station I (scaled)
ETAAB	augmentor efficiency
ETAABR(k)	augmentor efficiency model evaluation ratio array, $k=1$ to NTOTAL
ETAB	combustor efficiency
ETABR(k)	combustor efficiency model evaluation ratio array, $k=1$ to NTOTAL
ETAHCM	compressor efficiency
ETAIFM	fan inside-diameter efficiency
ETAOFM	fan outside-diameter efficiency
FARIM	fuel-air ratio at station I
FD(k)	array used to save indices of out-of-range corrected digital coefficients, $k=1$ to 125
FN	net thrust, N (lbf)
GLD	array used to save indices of out-of-range uncorrected digital coefficients, $k=1$ to 125
GMAB	average specific heat ratio in augmentor
GMB	average specific heat ratio in combustor
GMHC	average specific heat ratio in compressor
GMI	specific heat ratio at station I
HAB	average specific enthalpy in augmentor, J/kg (Btu/lbm)
HABM	PROCOM-rescaled augmentor specific enthalpy, J/kg (Btu/lbm)
HB	average specific enthalpy in combustor, J/kg (Btu/lbm)
HBM	PROCOM-rescaled combustor specific enthalpy, J/kg (Btu/lbm)
HHCM	PROCOM-rescaled compressor specific enthalpy
HP4	high-pressure-turbine enthalpy drop parameter (scaled)
HP.41	low-pressure-turbine enthalpy drop parameter (scaled)
HI	specific enthalpy at station I, J/kg (Btu/lbm)
HIM	PROCOM-rescaled specific enthalpy at station I
I	integer index
IG(k)	integrator gain integer array, $k=1$ to 10
INLET	inlet configuration option (integer)
IP	index on operating points (integer)
IPNCH	card punch device number (integer)
IPRINT	index for subroutine PRINT (integer)
IW	line printer device number (integer)
J	integer index
JJ	integer index
JP	punch option (integer) (1 for output of all data)
JPA	punch option (integer) 1 for output of analog data)
JPD	punch option (integer) (1 for output of digital data)
K	integer index
KBH	high-pressure-turbine bleed flow indicator (integer)
KBL	low-pressure-turbine bleed flow indicator (integer)
KBV	overboard bleed flow indicator (integer)
K1	number of out-of-range uncorrected digital coefficients (integer)
K3	number of zero uncorrected digital coefficients (integer)
K4	number of out-of-range corrected digital coefficients (integer)
N	integer index
NTOTAL	total number of operating points input to host program (integer)
PADR(k)	potentiometer address array (alphanumeric), $k=1$ to 53
PLA	power lever angle (operating point label), deg
PRHC	TRAT-rescaled compressor pressure ratio
PRIF	TRAT-rescaled fan inside-diameter pressure ratio
PROF	TRAT-rescaled fan outside-diameter pressure ratio
PVAL(k)	potentiometer setting array, $k=1$ to 53
PI	total pressure at station I, N/cm^2 (psia)
PIR(k)	model evaluation ratio array for total pressure at station I, $k=1$ to NTOTAL

POA	FLCOND-rescaled ambient pressure
P2A	FLCOND-rescaled fan inlet total pressure
P2D	fan inlet total pressure at design point, N/cm^2 (psia)
P22D	compressor inlet total pressure at design point, N/cm^2 (psia)
P22Q2M	fan inside-diameter pressure ratio
RCVV	compressor variable geometry parameter, deg
RTT2	square root of scaled fan inlet total temperature
RTT22	square root of scaled compressor inlet total temperature
RTT4	square root of scaled high-pressure-turbine inlet total temperature
RTT41	square root of scaled low-pressure-turbine inlet total temperature
SFX	scale factor on variable X, appropriate units
TAM	sea-level ambient temperature, K (°R)
TAVAB	average total temperature in augmentor, K (°R)
TAVB	average total temperature in combustor, K (°R)
TAVHC	average total temperature in compressor, K (°R)
TRHCM1	compressor temperature rise parameter
TRIFM1	fan inside-diameter temperature rise parameter
TROFM1	fan outside-diameter temperature rise parameter
TI	total temperature at station I, K (°R)
TIM	PROCOM-rescaled total temperature at station I
TIR(k)	model evaluation ratio array for total temperature at station I, $k=1$ to NTOTAL
T2A	FLCOND-rescaled fan inlet total temperature
UDC(k)	uncorrected digital coefficient array, $k=1$ to 125
WAR2	fan corrected flow rate, kg/sec (lbm/sec)
WAR22	compressor corrected flow rate, kg/sec (lbm/sec)
WAI	airflow rate leaving station I, kg/sec (lbm/sec)
T2D	fan inlet total temperature at design point, K (°R)
T22D	compressor inlet total temperature at design point, K (°R)
WAIR(k)	model evaluation ratio array for airflow leaving station I, $k=1$ to NTOTAL
WA2D	fan flow rate at design point, kg/sec (lbm/sec)
WA22D	compressor flow rate at design point, kg/sec (lbm/sec)
WBLHT	high-pressure-turbine cooling bleed flow rate, kg/sec (lbm/sec)
WBLLT	low-pressure-turbine cooling bleed flow rate, kg/sec (lbm/sec)
WBLOV	overboard bleed flow rate, kg/sec (lbm/sec)
WF4	combustor fuel flow rate, kg/sec (lbm/sec)
WF7	augmentor fuel flow rate, kg/sec (lbm/sec)
WGI	gas flow rate leaving station I, kg/sec (lbm/sec)
WGIR(k)	model evaluation ratio array for gas flow rate leaving station I, $k=1$ to NTOTAL
WP4	high-pressure-turbine flow parameter (scaled)
WP41	low-pressure-turbine flow parameter (scaled)
XMN	Mach number
XMNM	FLCOND-rescaled Mach number
XNH	high-spool rotor speed, rpm
XNL	low-spool rotor speed, rpm
XI	X input for map I
YI	Y input for map I
ZC(k)	map scale factor array for Z output variables, $k=1$ to 12
DZ(k)	array used to save indices of zero uncorrected digital coefficients, $k=1$ to 125

PROCOM. – All variables are scaled.

AMW	molecular weight
CP	specific heat at constant pressure
CPA	specific heat at constant pressure of air
CPF	specific heat at constant pressure of fuel
CV	specific heat at constant volume
FA	fuel-air ratio
GAM	specific heat ratio
H	specific enthalpy
HA	specific enthalpy of air
HF	specific enthalpy of fuel
R	gas constant

T	total temperature
TD	3500° F minus T

SPOOL. – Same symbol used for unscaled and scaled rotor speed.

IGAIN	integrator gain integer
PI	3.1416
POT(k)	potentiometer setting array, $k=1$ to 2
POT1I	required gain of potentiometer-integrator combination
SFDH	scale factor on turbine enthalpy drop, appropriate units
SFWG	scale factor on turbine flow rate, appropriate units
SFXN	scale factor on rotor speed, rpm/computer unit
TSC	time scale factor
XI	rotor moment of inertia, N cm sec^2 (lbf in sec^2)
XJ	mechanical equivalent of heat, 100 N cm/J (9339.6 lbf/in)
XN	rotor speed, rpm

TRAT. – All variables are scaled fractions unless otherwise specified.

C	variable specific heat ratio effect on $(\Delta T/T)_{id}$
GAM	average component specific heat ratio
N	component identifying integer, $N=1$ to 3
N1(k)	fan outside-diameter function integer array, $k=1$ to 3
N2(k)	fan inside-diameter function integer array, $k=1$ to 3
N3(k)	compressor function integer array, $k=1$ to 3
PRC	component pressure ratio
S	pressure ratio effect on specific heat ratio shift
TR	$(\Delta T/T)_{id}$
TRC	$(\Delta T/T)_{id}$ for specific heat ratio of 1.35
X1(k)	component pressure ratio data array, $k=1$ to 25
Y1(k)	function data array, $k=1$ to 25

VOLUME. – Same symbol used for unscaled and scaled pressure and temperature.

IGAIN	integrator gain integer
P	total pressure in volume, N/cm^2 (psia)
POT(k)	potentiometer setting array, $k=1$ to 5
POT2I	required gain of potentiometer-integrator combination
RA	gas constant of air, 2.8699×10^4 N cm/kg K (640.1 in lbf/lbm °R)
SFP	scale factor on total pressure, appropriate units
SFTT	scale factor on total temperature, appropriate units
SFW	scale factor on stored mass, appropriate units
SFWG	scale factor on flow rate through volume, appropriate units
TSC	time scale factor
TT	total temperature in volume, K (°R)
V	volume, cm^3 (in^3)
W	stored mass in volume, kg (lbm)
WS	stored mass in volume (scaled)

References

1. McKinney, John S.: Simulation of Turbofan Engine. Part I: Description of Method of Balancing Technique. AFAPL-TR-67-125-Pt.-1, Air Force Aeropropulsion Lab., Air Force Systems Command, 1967. (AD-825197)
2. McKinney, John S.: Simulation of Turbofan Engine. Part II: User's Manual and Computer Program Listing. AFAPL-TR-67-125-Pt.-2, Air Force Aeropropulsion Lab, Air Force Systems Command, 1967. (AD-825198)
3. Koenig, Robert W.; and Fishbach, Laurence H.: GENENG—A Program for Calculating Design and Off-Design Performance for Turbojet and Turbofan Engines. NASA TN D-6552, 1972.
4. Fishbach, Laurence H.; and Koenig, Robert W.: GENENG II—A Program for Calculating Design and Off-Design Performance of Two-and Three-Spool Turbofans with as Many as Three Nozzles. NASA TN D-6553, 1972.
5. Sellers, James F.; and Daniele, Carl J.: DYNGEN—A Program for Calculating Steady-State and Transient Performance of Turbojet and Turbofan Engines. NASA TN D-7901, 1975.
6. Fishbach, Laurence H.; and Caddy, Michael J.: NNEP—The Navy NASA Engine Program. NASA TM X-71857, 1975.
7. Szuch, John R.; and Bruton, William M.: Real-Time Simulation of the TF30-P-3 Turbofan Engine Using a Hybrid Computer. NASA TM X-3106, 1974.
8. Szuch, John R.; and Seldner, Kurt: Real-Time Simulation of the F100-PW-100 Turbofan Engine Using the Hybrid Computer. NASA TM X-3261, 1975.
9. Szuch, John R.; Seldner, Kurt; and Cwynar, David S.: Development and Verification of Real-Time, Hybrid Computer Simulation of the F100-PW-100(3) Turbofan Engine. NASA TP-1034, 1977.
10. Cwynar, David S.; and Batterton, Peter G.: Digital Implementation of the TF30-P-3 Turbofan Engine Control. NASA TM X-3105, 1975.
11. Hrach, Frank J.; Arpasi, Dale J.; and Bruton, William M.: Design and Evaluation of a Sensor Fail-Operational Control System for a Digitally Controlled Turbofan Engine. NASA TM X-3260, 1975.
12. Arpasi, Dale J.; and Bruton, William M.: Method of Discrete Modeling and Its Application to Estimation of TF30 Engine Variables. NASA TM X-3443, 1976.
13. Szuch, John R.; et al.: F100 Multivariable Control Synthesis Program—Evaluation of a Multivariable Control Using a Real-Time Engine Simulation. NASA TP-1056, 1977.
14. Szuch, John R.: HYDES, A Generalized Hybrid Computer Program for Studying Turbojet or Turbofan Engine Dynamics. NASA TM X-3014, 1974.
15. Szuch, John R.; Krosel, Susan M.; and Bruton, William M.: An Automated Procedure for Developing Hybrid Computer Simulations of Turbofan Engines—Part II: Computer Printouts. NASA TM-81730, 1981.
16. Shames, Irving H.: Mechanics of Fluids. McGraw-Hill, Inc., 1962.
17. Keenan, Joseph H.; and Kaye, Joseph: Gas Tables. John Wiley & Sons, Inc., 1948.
18. Shapiro, Ascher H.: The Dynamics and Thermodynamics of Compressible Fluid Flow. Vol. I. Ronald Press Co., 1953.
19. Cota, R. A.; Kunkle, J. S.; and Wilson, S. D.; eds.: Compressed Gas Handbook. NASA SP-3045, 1969.
20. Bruton,. William M.; and Cwynar, David S.: Lewis Hybrid Computing System—Users Manual. NASA TM-79111, 1979.
21. Cwynar, David S.: INFORM—An Interactive Data Collection and Display Program with Debugging Capability. NASA TP-1424, 1980.
22. Szuch, John R.: Application of Real-Time Engine Simulations to the Development of Propulsion System Controls. NASA TM X-71764, 1975.

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TP-1851		

4. Title and Subtitle AUTOMATED PROCEDURE FOR DEVELOPING HYBRID COMPUTER SIMULATIONS OF TURBOFAN ENGINES I - GENERAL DESCRIPTION	5. Report Date August 1982 6. Performing Organization Code 505-32-6B
7. Author(s) John R. Szuch, Susan M. Krosel, and William M. Bruton	8. Performing Organization Report No. E-779
9. Performing Organization Name and Address National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135	10. Work Unit No. 11. Contract or Grant No.
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546	13. Type of Report and Period Covered Technical Paper 14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

This report offers a systematic, computer-aided, self-documenting methodology for developing hybrid computer simulations of turbofan engines. The methodology that is presented makes use of a host program that can run on a large digital computer and a machine-dependent target (hybrid) program. The host program performs all the calculations and data manipulations that are needed to transform user-supplied engine design information to a form suitable for the hybrid computer. The host program also trims the self-contained engine model to match specified design-point information. Part I contains a general discussion of the methodology, describes a test case, and presents comparisons between hybrid simulation and specified engine performance data. Part II, a companion document, contains documentation, in the form of computer printouts, for the test case.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
	Unclassified - unlimited STAR Category 61

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	120	A06

* For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1982

National Aeronautics and
Space Administration

Washington, D.C.
20546

Official Business
Penalty for Private Use, $300

Postage and Fees Paid
National Aeronautics and
Space Administration
NASA-451

U.S. MAIL

NASA

Undeliverable (Section 158
stal Manual) Do Not Return