UNCLASSIFIED

AD NUMBER

AD825198

NEW LIMITATION CHANGE

TO

Approved for public release, distribution unlimited

FROM

Distribution authorized to U.S. Gov't. agencies and their contractors; Critical Technology; NOV 1967. Other requests shall be referred to Air Force Aero Propulsion Laboratory, Wright-Patterson AFB, OH 45433.

AUTHORITY

AFAPL ltr, 12 Apr 1972

THIS PAGE IS UNCLASSIFIED

AFAPL-TR-67-125
PART II

AD825198

SIMULATION OF TURBOFAN ENGINE

PART II. USER'S MANUAL AND COMPUTER PROGRAM LISTING

JOHN S. McKINNEY, CAPTAIN, USAF

TECHNICAL REPORT AFAPL-TR-67-125, PART II

NOVEMBER 1967

This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of the Air Force Aero Propulsion Laboratory, Wright-Patterson AFB, Ohio.

DDC
JAN 17 1968

AIR FORCE AERO PROPULSION LABORATORY
RESEARCH AND TECHNOLOGY DIVISION
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO

90

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

2

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

100 - January 1968 - CO455 - 19-413

AFAPL-TR-67-125
PART II

SIMULATION OF TURBOFAN ENGINE

PART II. USER'S MANUAL AND COMPUTER PROGRAM LISTING

JOHN S. McKINNEY, CAPTAIN, USAF

This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of the Air Force Aero Propulsion Laboratory, Wright-Patterson AFB, Ohio.

FOREWORD

This report was prepared in the Components Branch (AFTC), Turbine Engine Division, Air Force Aero Propulsion Laboratory, Wright-Patterson Air Force Base, Ohio, under Project 3066, "Gas Turbine Technology," Task 306803, "Advanced Engine Studies," with Charles E. Bentz as Project Engineer.

This report covers work conducted within the Components Branch in the time period between July 1965 and June 1967 and was submitted by the author 31 August 1967.

This technical report has been reviewed and is approved.

ERNEST C. SIMPSON
Chief, Turbine Engine Division
Air Force Aero Propulsion Laboratory

ABSTRACT

This report describes a digital computer program titled SMOTE (Simulation of Turbofan Engine). SMOTE is a computer program for balancing-cycle turbofan engines capable of running both design and off-design points. The program is written in Fortran IV language and was designed for use on an IBM 7090 Digital Computer, although it has also been run on an IBM System 360. Performance maps (Block Data format) of the major engine components are required. Information for setting up the Block Data and input data is given in the report. Also included in the report is a complete program listing with description of each subroutine and a sample data pack.

(Distribution of the abstract is unlimited.)

TABLE OF CONTENTS

SECTION		PAGE
I	INTRODUCTION	1
II	BLOCK DATA	2
III	INPUT DATA	3
	1. Controlled Output	3
	2. Namelist Input	3
IV	SUBROUTINE DESCRIPTIONS	4
V	PROGRAM LISTING	7

SYMBOLS

STATION NUMBERS

1	ambient
2	fan entrance
21	fan exit/compressor and duct entrance
3	compressor exit/burner entrance
4	burner exit/hi pressure turbine entrance
5	hi press. turbine exit/lo press. turbine entrance
55	lo press. turbine exit
6	afterburner entrance
7	afterburner exit
8	main nozzle throat
9	main nozzle exit
23	duct burner entrance
24	duct burner exit
25	duct exit (if mixed-flow engine)
28	duct nozzle throat
29	duct nozzle exit

THERMODYNAMIC PROPERTIES

T	total temperature (degrees R)
H	total enthalpy
P	total pressure (atmospheres)
S	total entropy
TS	static temperature (degrees R)
PS	static pressure (atmospheres)
FAR	fuel-air ratio
AM	Mach number
V	velocity (feet/second)

COMPONENT SYMBOLS

F	fan
C	compressor
B	burner
COM	burner
THP	hi pressure turbine
TLP	lo pressure turbine
A	afterburner
AFT	afterburner
D	duct
DUC	duct
M	main nozzle
NOZ	nozzle
OB	overboard
T	total

ENGINE SYMBOLS

ETAR	ram recovery
CN	corrected speed ratio
PCN	percent speed
PR	pressure ratio
Z	pressure-ratio ratio
ETA	efficiency
TFF	turbine flow function
DHTC	turbine delta enthalpy (temperature corrected)
DHT	turbine delta enthalpy
WA	air flow (pounds/sec)
WF	fuel flow (pounds/sec)
WG	gas flow (pounds/sec)
PCBL	percent bleed
BL	bleed (pounds/sec)
HPEXT	horsepower extracted

SYMBOLS (CONT)

DP	pressure drop
DT	temperature increase

MISCELLANEOUS

A	area
ALTP	altitude
AM	Mach number of aircraft
BYPASS	bypass ratio
CF	correction factor
CS	ambient speed of sound
CV	nozzle velocity coefficient
DEL	delta degradation coefficient
DS	design value
DUM	dummy (not used)
FG	gross thrust
FGM	momentum thrust
FGP	pressure thrust
FN	net thrust
FRD	ram drag
GU	initial or guess values
ITRYS	number of loops thru engine before quitting
SFC	specific fuel consumption
TOLALL	tolerance
VA	velocity of aircraft
VJ	jet velocity

NOTE: Some symbols may be truncated when combined with other symbols due to six character limit imposed by Fortran Computer Language.

SECTION I

INTRODUCTION

The purpose of this report is to describe a digital computer program for balancing-cycle turbofan engines. The program, titled SMOTE (Simulation of Turbofan Engine), was developed in the Components Branch, Turbine Engine Division, Air Force Aero Propulsion Laboratory. The program, as written and described in this report, is intended for use on an IBM 7090 Digital Computer, but it has also been modified and tested on an IBM System 360. It is written in Fortran IV language.

The performance of the major engine components is based on component maps which must be converted into Block Data subroutines for use by SMOTE. Presently included in the program are maps for the fan, compressor, combustor, and both turbines.

SMOTE uses a Namelist input and a controlled output, whereby only selected variables will be printed for each run.

Part I of this report describes the method of engine calculations and the balancing technique and gives some sample results. Part II is intended as a user's manual and includes instructions for setting up and running the program, as well as a program listing. The parts may be used independently of one another.

SECTION II

BLOCK DATA

To provide the basic component performance requires the maps for the fan, compressor, combustor and both turbines to be converted to Block Data subroutines. The Block Data presented in this report is very general and does not represent any specific engine. For the following discussion on setting up Block Data, refer to the program listing, Section V. The format for all the maps is very similar, with a maximum of 15 curves and a maximum of 15 points for each curve. "N" refers to the number of curves and "NP" is an array of the number of points on each curve.

For the fan and compressor maps, the corrected speed (CN) is input as a ratio of the actual corrected speed curve to the design corrected speed. The pressure ratio (PR), corrected airflow (WAC), and efficiency (ETA) are input in groups, with a group for each corrected speed curve. The corrected speeds and pressure ratios must be loaded low to high. Note that there are two points per card.

For the combustor map, curves of pressure entering the combustor (P3) in PSI are loaded low to high, as well as delta-T (T4 - T3) points for each curve. The efficiencies (ETA) corresponding to each DELT are loaded in one array. Note that in this array, each column is obtained from one pressure curve.

For the turbine maps, turbine flow function (TFF) curves are input low to high, and the corrected speed (CN), delta-H (DH), and efficiency (ETA) are input in groups, with a group for each TFF line. The corrected speed must be loaded low to high. TFF, CN, and DH are defined as

$$TFF = \frac{WG \sqrt{T_{IN}}}{P_{IN}}$$

$$CN = \frac{PCNC}{\sqrt{T_{IN}}}$$

$$DH = \frac{H_{IN} - H_{OUT}}{T_{IN}}$$

Formats other than those presented here may be used for the component performance maps. However, some change in logic would then be required.

SECTION III

INPUT DATA

The input data is divided into two sections; data cards for the controlled output, and data cards in Namelist format for running each point. For the following discussion on setting up the input data, refer to the listing of sample data immediately following the program listing.

1. CONTROLLED OUTPUT

The variables that are to be output are selected by the first section of data cards. Any variable that is in one of the main commons (DESIGN, FRONT, SIDE, or BACK) may be selected for output by punching the name of the variable as it appears in the common (with trailing blanks, if necessary) in Columns 1 through 6. Up to 102 variables (seventeen lines of six variables) may be chosen for a particular run. During the output phase, the name of the variable is printed out, with its value printed immediately below the name.

Another feature of the controlled output is the ability to change the name of a variable to be output; for example, it may be desired to change a station designation to one more common to a particular programmer. In this case, the variable name would be punched in Columns 1 through 6 as described above, but in addition, the desired name would be punched in Columns 13 through 18. Special symbols, such as /, may be used in the new name. The last card of the controlled output must be a card with "THEEND" punched in Columns 1 through 6.

2. NAMELIST INPUT

The normal data for running the desired points follows the controlled output data and is in a Namelist format, where the name of the Namelist is DATAIN. Usually the first set of data is the design point, as shown in the sample input data. When the design point is run (IDES=1), all map scaling or correction factors are printed out, as well as being retained in common. Therefore, it is possible to run off-design points immediately following the design point by making use of the values in common, or to begin running an off-design point immediately by inputting the scaling or correction factors. The first method is usually easier, but the second method may be desired if many points are to be run using the same engine parameters with no changes except for power setting, Mach number, and altitude.

The controls which fix the type of turbofan, the mode of operation, the method of calculating ram recovery, etc., are explained in the listing of subroutine INPUT. Subroutine ZERO determines what values in common will be zeroed between points. None of the design values or correction factors are ever zeroed.

A title card must be input immediately after the first point of the data pack, and ITITLE must be set equal to 1 in the data for the first point. This is because a title is always printed for each point and must, therefore, be previously defined. The input format for the title is 12A6, and the resulting 72 spaces are centered on the page when printed out. The title may be changed by setting ITITLE = 1 and inserting a new title card after the Namelist data for the point.

When an afterburning or duct-burning point is run, the exhaust nozzle areas are allowed to float to obtain optimum expansion. This means that there can be no balancing at the point, and it is necessary to prebalance the engine cycle in a nonaugmented mode. That is, an identical point, except that it is nonaugmented, must be run before either afterburning or ductburning. When either IAFTBN or IDBURN is greater than zero, the program will automatically set INIT = 1 and use the balanced values from the preceding point. The nozzle areas are returned to their standard design values after completing an augmented point. Some examples of this type of afterburning are given in the sample data listing.

SECTION IV

SUBROUTINE DESCRIPTIONS

The following will be a brief description of what each subroutine calculates or controls. See Figure 1 for a subroutine flow chart

SMOTE	Dummy main program to initiate the calculations and cause the input of the controlled output variables. Because of the looping between subroutines, control is never transferred back to this routine.
ENGBAL	Main subroutine. Controls all engine balancing loops, checks tolerances and number of loops, and loads matrix.
MATRIX	Solves error matrix.
INPUT	Reads Namelist data and title. Prints title.
ZERO	Zeroes common and certain controls.
COINLT	Determines ram recovery and performs inlet calculations.
ATM062	1962 ARDC Atmosphere Tables.
RAM	Calculates ram recovery defined by MIL-E-5008B Specifications.
GUESS	Determines initial values of independent variables (PCNF, PCNC, and T4) at each point. It may be desired to change these equations to suit a particular engine. The closer the initial values are to the final values, the faster the program will balance.
COFAN	Uses Block Data to perform fan calculations.
COCOMP	Uses Block Data to perform compressor calculations.
COCOMB	Uses Block Data to perform combustor calculations. May use either T4 or WFB as the main parameter.
COHPTB	Uses Block Data to perform high pressure turbine calculations. Calculates ERR(1) and ERR(2).
COLPTB	Uses Block Data to perform low pressure turbine calculations. Calculates ERR(3) and ERR(4).
FRTOSD	Dummy routine to transfer values from common FRONT to common SIDE.
CODUCT	Performs duct and duct-burning calculations. May use either T24 of WFD as the main parameter for duct-burning. Controls the duct nozzle and calculates ERR(5) if in separate-flow mode.
FASTBK	Dummy routine to transfer values from commons FRONT and SIDE to common BACK.

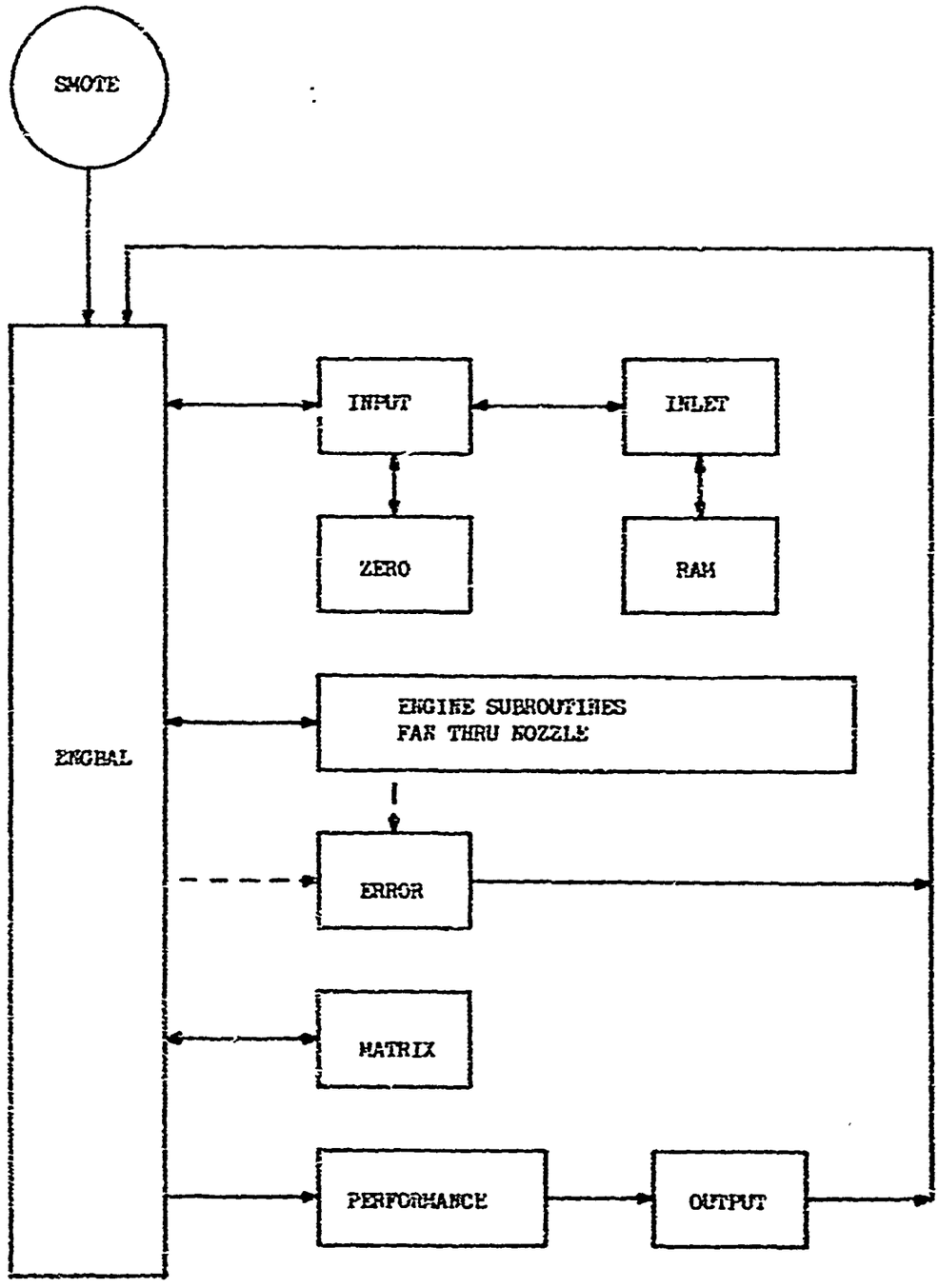

Figure 1. SMOTE Subroutine Flow Chart

COMIX	Performs gas-mixing calculations if in mixed-flow mode. At design points it calculates areas from either an input static pressure (PS55) or an input Mach number (AM55) if PS55 = 0. At off-design points it calculates static pressures and Mach numbers from the design areas. Calculates ERR(5).
COAFBN	Performs after-burning calculations. May use either T7 or WFA as the main parameter.
COMNOZ	Controls the main nozzle and calculates ERR(6).
PERF	Calculates performance after the engine is balanced.
OUTPUT	Prints output except for controlled output. Prints the main commons in a close format after each point.
CONOUT	Controls and prints the controlled output variables.
ERROR	Controls all printouts if an error occurs. Prints name of subroutine where error occurred and also prints the values of all variables in the main commons.
SYG	Controls printing from UNIT08. Throughout the program and particularly in ENGBAL, certain messages, variables, and matrix values are written on UNIT08 as an aid in determining why an error occurred or why a point did not balance. These values are printed out if subroutine ERROR is called and IDUMP is greater than zero, or after a good point if IDUMP = 2.
TAPES	Defines UNIT08, which is just a "scratch" disk and does not require a $SETUP card. Normal input and output are on UNIT05 and UNIT06, respectively.
THCOMP	Performs isentropic calculations for compressors.
THTURB	Performs isentropic calculations for turbines.
THERMO	Provides thermodynamic conditions using PROCOM.
PROCOM	Calculates thermodynamic gas properties for either air or a fuel-air mixture, based on JP-4.
SEARCH	General table look-up and interpolation routine to obtain data from the Block Data subroutines.
MAPBAC	Used when calculations result in values not on the turbine maps. Changes the map value and an independent variable (PCNF, PCNC, or T4) in an attempt to rectify the situation.
CONVRG	Performs nozzle calculations for a convergent nozzle.
CONDIV	Performs nozzle calculations for a convergent-divergent nozzle.
AFQUIR	General quadratic interpolation routine.

FANDAT	Block Data for fan.
CMPDAT	Block Data for compressor.
CMBDAT	Block Data for combustor.
HPTDAT	Block Data for high pressure turbine.
LPTDAT	Block Data for low pressure turbine.

SECTION V

PROGRAM LISTING

The following is a complete listing of all subroutines required to run SMOTE. Immediately following the program listing is a listing of a sample input data pack.

```
$IBFTC SMOTE   DECK,M94/2,XR7
      COMMON /POINT/IDATPT
      IDATPT=0
      CALL CONOUT(1)
      CALL ENGBAL
      STOP
      END
```

```
$IBFTC ENGBAL  DECK,M94/2,X87
      SUBROUTINE ENGBAL
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,NODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASMX,IDBURN,IAFTBN,IDCD  ,IMCD  ,IDSHOC,IMSHOC,NOZFLT,
     3ITRYS ,LOOPER,NOMAP ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,T4GU  ,DUMD1 ,DUMD2 ,DELFG ,DELSN ,DELSFC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCUDS,ETABDS,WA3CDS,DPCDDS,DTCDCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DNHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T25DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     APS55  ,AM55  ,CVDNOZ,CVMNOZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON / FRONT/
     1T1    ,P1    ,H1    ,S1    ,T2    ,P2    ,H2    ,S2    ,
     2T21   ,P21   ,H21   ,S21   ,T3    ,P3    ,H3    ,S3    ,
     3T4    ,P4    ,H4    ,S4    ,T5    ,P5    ,H5    ,S5    ,
     4T55   ,P55   ,H55   ,S55   ,BLF   ,BLC   ,BLDU  ,BLOB  ,
     5CNF   ,PRF   ,ETAF  ,WAFC  ,WAF   ,WA3   ,WG4   ,FAR4  ,
     6CNC   ,PRC   ,ETAC  ,WACC  ,WAC   ,ETAB  ,DPCOM ,DUMF  ,
     7CNHP  ,ETATHP,DHTCHP,DHTC  ,BLHP  ,WG5   ,FAR5  ,CS    ,
     8CNLP  ,ETATLP,DHTCLP,DHTF  ,BLLP  ,WG55  ,FAR55 ,HPEXT ,
     9AM    ,ALTP  ,ETAR  ,ZF    ,PCNF  ,ZC    ,PCNC  ,WFB   ,
     ATFFHP ,TFFLP ,PCBLF ,PCBLC ,PCBLDU,PCBLOB,PCBLHP,PCBLLP
      COMMON /  SIDE/
     XXP1   ,XWAF  ,XWAC  ,XBLF  ,XBLDU ,XH3   ,DUMS1 ,DUMS2 ,
     XXT21  ,XP21  ,XH21  ,XS21  ,T23   ,P23   ,H23   ,S23   ,
     3T24   ,P24   ,H24   ,S24   ,T25   ,P25   ,H25   ,S25   ,
     4T28   ,P28   ,H28   ,S28   ,T29   ,P29   ,H29   ,S29   ,
     5WAD   ,WFD   ,WG24  ,FAR24 ,ETAD  ,DPDUC ,BYPASS,DUMS3 ,
     6TS28  ,PS28  ,V28   ,AM28  ,TS29  ,PS29  ,V29   ,AM29
      COMMON /  BACK/
     XXT55  ,XP55  ,XH55  ,XS55  ,XT25  ,XP25  ,XH25  ,XS25  ,
     XXWFB  ,XWG55 ,XFAR55,XWFD  ,XWG24 ,XFAR24,XXP1  ,DUMB  ,
     3T6    ,P6    ,H6    ,S6    ,T7    ,P7    ,H7    ,S7    ,
     4T8    ,P8    ,H8    ,S8    ,T9    ,P9    ,H9    ,S9    ,
     5WG6   ,WFA   ,WG7   ,FAR7  ,ETAA  ,DPAFT ,V55   ,V25   ,
     6PS6   ,V6    ,AM6   ,TS7   ,PS7   ,V7    ,AM7   ,AM25  ,
     7TS8   ,PS8   ,V8    ,AM8   ,TS9   ,PS9   ,V9    ,AM9   ,
     8VA    ,FRD   ,VJD   ,FGMD  ,VJM   ,FGMM  ,FGPD  ,FGPM  ,
     9FGM   ,FGP   ,WFT   ,WGT   ,FART  ,FG    ,FN    ,SFC
      DIMENSION VAR(6),DEL(6),ERRB(6),DELVAR(6),EMAT(6,6),VMAT(6),
     1AMAT(6)
      DATA AWORD/6HENGBAL/
      CALL INPUT
      IF(INIT.EQ.1) GO TO 50
      TFFHP=TFHPDS
      TFFLP=TFLPDS
50    LOOPER=0
      NUMMAP=0
1     LOOP=0
      NOMAP=0
      ISO=2
      DO 2 I=1,6
      VMAT(I)=0.
      AMAT(I)=0.
      DELVAR(I)=0.
```

```
      DO 2 L=1,6
2     EMAT(I,L)=0.
3     LOOPER=LOOPER+1
      CALL COFAN
      WORD=ANORD
      IF(LOOPER.GT.ITRYS) GO TO 18
      IF(NOMAP.GT.0) GO TO 1
      NUMMAP=0
      VAR(1)=ZF*100.
      VAR(2)=PCNF
      VAR(3)=ZC*100.
      IF(MODE.EQ.0.OR.MODE.EQ.2) VAR(4)=PCNC
      IF(MODE.EQ.1) VAR(4)=T4/10.
      VAR(5)=TFFHP
      VAR(6)=TFFLP
      DO 4 I=1,6
4     IF(ABS(ERR(I)).GT.TOLALL) GO TO 5
      CALL PERF
      CALL ERROR
5     IF(LOOP.GT.0) GO TO 7
      MAPEDG=0
      MAPSET=0
      DO 6 I=1,6
      ERRB(I)=ERR(I)
6     DEL(I)=0.001*VAR(I)
      GO TO 9
7     IF(MAPEDG.EQ.0) GO TO 70
      MAPEDG=0
      MAPSET=1
      VAR(LOOP)=VAR(LOOP)+2.*DEL(LOOP)
      GO TO 10
70    IF(MAPSET.EQ.0) VAR(LOOP)=VAR(LOOP)+DEL(LOOP)
      IF(MAPSET.EQ.1) VAR(LOOP)=VAR(LOOP)-DEL(LOOP)
      MAPSET=0
      DO 8 I=1,6
8     EMAT(I,LOOP)=(ERRB(I)-ERR(I))/DEL(LOOP)
9     LOOP=LOOP+1
      IF(LOOP.GT.6) GO TO 11
      VAR(LOOP)=VAR(LOOP)-DEL(LOOP)
10    ZF=VAR(1)/100.
      PCNF=VAR(2)
      ZC=VAR(3)/100.
      IF(MODE.EQ.0.OR.MODE.EQ.2) PCNC=VAR(4)
      IF(MODE.EQ.1) T4=VAR(4)*10.
      TFFHP=VAR(5)
      TFFLP=VAR(6)
      IF(ZF.LT.0.) ZF=0.05
      IF(ZC.LT.0.) ZC=0.05
      GO TO (1,3),IGO
11    DO 12 I=1,6
12    AMAT(I)=-ERRB(I)
      DO 14 I=1,6
      IZERO=0
      DO 13 LOOP=1,6
13    IF(EMAT(I,LOOP).EQ.0.) IZERO=IZERO+1
      IF(IZERO.LT.6) GO TO 14
      WRITE(6,100)I
      LOOPER=ITRYS+100
      GO TO 18
14    CONTINUE
      DO 16 LOOP=1,6
```

```
      IZERO=0
      DO 15 I=1,6
15    IF(EMAT(I,LOOP).EQ.0.) IZERO=IZERO+1
      IF(IZERO.LT.6) GO TO 16
      WRITE(6,101)LOOP
      LOOPER=ITRYS+100
      GO TO 18
16    CONTINUE
      CALL MATRIX(EMAT,VMAT,AMAT)
      DO 17 LOOP=1,6
      DELVAR(LOOP)=0.6*VMAT(LOOP)
      IF(ABS(DELVAR(LOOP)).GT.0.05*VAR(LOOP))
     1DELVAR(LOOP)=0.05*VAR(LOOP)*DELVAR(LOOP)/ABS(DELVAR(LOOP))
17    VAR(LOOP)=VAR(LOOP)+DELVAR(LOOP)
18    WRITE(8,102)LOOPER
      DO 19 I=1,6
19    WRITE(8,103)AMAT(I),(EMAT(I,L),L=1,6),VMAT(I),DELVAR(I),VAR(I)
      ERRAVE=(ABS(ERRB(1))+ABS(ERRB(2))+ABS(ERRB(3))+
     1        ABS(ERRB(4))+ABS(ERRB(5))+ABS(ERRB(6)))/6.
      DELAVE=(ABS(DELVAR(1))+ABS(DELVAR(2))+ABS(DELVAR(3))+
     1        ABS(DELVAR(4))+ABS(DELVAR(5))+ABS(DELVAR(6)))/6.
      WRITE(8,104)ERRAVE,DELAVE
      IGO=1
      IF(LOOPER.LT.ITRYS) GO TO 10
      CALL ERROR
      RETURN
100   FORMAT(4HOROW,I2,16H IS ZERO IN EMAT)
101   FORMAT(7HOCOLUMN,I2,16H IS ZERO IN EMAT)
102   FORMAT(8H8   ERRB,28X23HERROR MATRIX AFTER LOOP,I4,29X4HVMAT,
     16X6HDELVAR,7X14HVARIABLE$$$$$$)
103   FORMAT(1H0,F8.4,8X6F10.4,10XF10.4,F11.4,4XF11.4,6H$$$$$$)
104   FORMAT(1H0,F8.4,32X14HAVERAGE VALUES,42XF11.4,6H$$$$$$)
      END
```

```
$IBFTC MATRIX  DECK,M94/2,XR7
      SUBROUTINE MATRIX(E,V,A)
      DIMENSION E(6,6),V(6),A(6),PIV(7),T(6,7)
      DO 1 I=1,6
      T(I,7)=A(I)
      DO 1 J=1,6
1     T(I,J)=E(I,J)
      DO 7 I=1,6
      TEMP=0.
      DO 2 J=I,6
      IF(TEMP.GT.ABS(T(J,I))) GO TO 2
      TEMP=ABS(T(J,I))
      IPIV=J
2     CONTINUE
      IP1=I+1
      DO 3 J=IP1,7
3     PIV(J)=T(IPIV,J)/T(IPIV,I)
      IFROM=6
      ITO=6
4     IF(IFROM.EQ.IPIV) GO TO 6
      RM=-T(IFROM,I)
      DO 5 J=IP1,7
5     T(ITO,J)=T(IFROM,J)+RM*PIV(J)
      ITO=ITO-1
6     IFROM=IFROM-1
      IF(IFROM.GE.I) GO TO 4
      DO 7 J=IP1,7
7     T(I,J)=PIV(J)
      DO 8 I=1,5
      J=7-I
      K=6-I
      DO 8 L=J,6
8     T(K,7)=T(K,.  T(K,L)*T(L,7)
      DO 9 I=1,6
9     V(I)=T(I,7)
      RETURN
      END
```

```
$IBFTC INPUT   DECK,M94/2,XR7
      SUBROUTINE INPUT
      COMMON /POINT/IDATPT
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,MODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASMX,IDBURN,IAFTBN,IDCD  ,IMCD  ,IDSHOC,IMSHOC,NOZFLT,
     3ITRYS ,LOOPER,NOMAP ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,T4GU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCODS,ETABDS,WA3CDS,DPCODS,DTCOCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     APS55  ,AM55  ,CVDNDZ,CVMNDZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON / FRONT/
     1T1    ,P1    ,H1    ,S1    ,T2    ,P2    ,H2    ,S2    ,
     2T21   ,P21   ,H21   ,S21   ,T3    ,P3    ,H3    ,S3    ,
     3T4    ,P4    ,H4    ,S4    ,T5    ,P5    ,H5    ,S5    ,
     4T55   ,P55   ,H55   ,S55   ,BLF   ,BLC   ,BLDU  ,BLOB  ,
     5CNF   ,PRF   ,ETAF  ,WAFC  ,WAF   ,WA3   ,WG4   ,FAR4  ,
     6CNC   ,PRC   ,ETAC  ,WACC  ,WAC   ,ETAB  ,DPCOM ,DUMF  ,
     7CNHP  ,ETATHP,DHTCHP,DHTC  ,BLHP  ,WG5   ,FAR5  ,CS    ,
     8CNLP  ,ETATLP,DHTCLP,DHTF  ,BLLP  ,WG55  ,FAR55 ,HPEXT ,
     9AM    ,ALTP  ,ETAR  ,ZF    ,PCNF  ,ZC    ,PCNC  ,WFB   ,
     ATFFHP ,TFFLP ,PCBLF ,PCBLC ,PCBLDU,PCBLOB,PCBLHP,PCBLLP
      COMMON /  SIDE/
     XXP1   ,XWAF  ,XWAC  ,XBLF  ,XBLDU ,XH3   ,DUMS1 ,DUMS2 ,
     XXT21  ,XP21  ,XH21  ,XS21  ,T23   ,P23   ,H23   ,S23   ,
     3T24   ,P24   ,H24   ,S24   ,T25   ,P25   ,H25   ,S25   ,
     4T28   ,P28   ,H28   ,S28   ,T29   ,P29   ,H29   ,S29   ,
     5WAD   ,WFD   ,WG24  ,FAR24 ,ETAD  ,DPDUC ,BYPASS,DUMS3 ,
     6TS28  ,PS28  ,V28   ,AM28  ,TS29  ,PS29  ,V29   ,AM29
      COMMON /  BACK/
     XXT55  ,XP55  ,XH55  ,XS55  ,XT25  ,XP25  ,XH25  ,XS25  ,
     XXWFB  ,XWG55 ,XFAR55,XWFD  ,XWG24 ,XFAR24,XXP1  ,DUMB  ,
     3T6    ,P6    ,H6    ,S6    ,T7    ,P7    ,H7    ,S7    ,
     4T8    ,P8    ,H8    ,S8    ,T9    ,P9    ,H9    ,S9    ,
     5WG6   ,WFA   ,WG7   ,FAR7  ,ETAA  ,DPAFT ,V55   ,V25   ,
     6PS6   ,V6    ,AM6   ,TS7   ,PS7   ,V7    ,AM7   ,AM25  ,
     7TS8   ,PS8   ,V8    ,AM8   ,TS9   ,PS9   ,V9    ,AM9   ,
     8VA    ,FRD   ,VJD   ,FGMD  ,VJM   ,FGMM  ,FGPD  ,FGPM  ,
     9FGM   ,FGP   ,WFT   ,WGT   ,FART  ,FG    ,FN    ,SFC
      NAMELIST /DATAIN/
     AITITLE,IDES  ,MODE  ,INIT  ,IDUMP ,IAMTP ,IGASMX,
     BIDBURN,IAFTBN,IDCD  ,IMCD  ,NOZFLT,ITRYS ,TOLALL,
     CZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     DZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     ET4DS  ,WFBDS ,DTCODS,ETABDS,WA3CDS,DPCODS,DTCOCF,ETABCF,
     FTFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     GTFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     HT24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     IT7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     JA55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     KT2    ,P2    ,T4    ,ZF    ,PCNF  ,ZC    ,PCNC  ,
     LWFB   ,TFFHP ,TFFLP ,AM    ,ALTP  ,ETAR  ,HPEXT ,
     MPCBLF ,BLF   ,PCBLC ,BLC   ,PCBLDU,BLDU  ,PCBLOB,BLOB  ,
     NPCBLHP,BLHP  ,PCBLLP,BLLP  ,
```

```
     DPS55  ,AM55  ,T24   ,ETAD  ,WFD   ,
     PT7    ,ETAA  ,WFA   ,CVDNOZ,CVMNOZ,DELFS ,DELFN ,DELSFC
C *** ITITLE=1  WILL READ IN TITLE
C *** IDES  =1  FOR CALCULATING DESIGN POINT
C *** MODE  =0  FOR CONSTANT T4
C *** MODE  =1  FOR CONSTANT PCNC
C *** MODE  =2  FOR CONSTANT WFB
C *** INIT  =1  WILL NOT INITIALIZE POINT
C *** IDUMP =1  WILL DUMP LOOPING WRITE-OUTS IF ERROR OCCURS
C *** IDUMP =2  WILL DUMP LOOPING WRITE-OUTS AFTER EVERY POINT
C *** IAMTP =0  WILL USE INPUT AM AND MIL SPEC ETAR
C *** IAMTP =1  WILL USE INPUT AM AND INPUT ETAR
C *** IAMTP =2  WILL USE T2 AS T1=T1+T2 AND STANDARD P1
C *** IAMTP =3  WILL USE P2 AND STANDARD T1
C *** IAMTP =4  WILL USE T2 AND P2
C *** IGASMX=-1 SEPARATE FLOW, INPUT A6
C *** IGASMX=0  SEPARATE FLOW, A6=A55
C *** IGASMX=1  WILL MIX DUCT AND MAIN STREAMS, A6=A25+A55
C *** IGASMX=2  WILL MIX DUCT AND MAIN STREAMS, INPUT A6
C *** IDBURN=1  FOR DUCT BURNING, INPUT T24
C *** IDBURN=2  FOR DUCT BURNING, INPUT WFD
C *** IAFTBN=1  FOR AFTERBURNING, INPUT T7
C *** IAFTBN=2  FOR AFTERBURNING, INPUT WFA
C *** IDCD  =1  DUCT NOZZLE WILL BE C-D
C *** IMCD  =1  MAIN NOZZLE WILL BE C-D
C *** NOZFLT=1  FOR FLOATING MAIN NOZZLE
C *** NOZFLT=2  FOR FLOATING DUCT NOZZLE
C *** NOZFLT=3  FOR FLOATING MAIN AND DUCT NOZZLES
C *** ITRYS =N  NUMBER OF PASSES THRU ENGINE BEFORE QUITTING
      DIMENSION TITLE(12)
      DATA AWORD/6H INPUT/
      IDATPT=IDATPT+1
      PRINT 100, IDATPT
      CALL ZERO
      WORD=AWORD
      READ(5,DATAIN)
      IF(IAFTBN.GT.0.OR.IDBURN.GT.0) INIT=1
      IF(ITITLE.EQ.1) READ(5,101) TITLE
      ITITLE=0
      WRITE(6,102) TITLE
      IF(MODE.EQ.0) WRITE(6,103) IDES,AM,ALTP,T4  ,T24,T7
      IF(MODE.EQ.1) WRITE(6,104) IDES,AM,ALTP,PCNC,T24,T7
      IF(MODE.EQ.2) WRITE(6,105) IDES,AM,ALTP,WFB ,T24,T7
      CALL COINLT
      RETURN
100   FORMAT(11H DATA POINT,I3)
101   FORMAT(12A6)
102   FORMAT(1H1,30X12A6)
103   FORMAT(1H0,7H  IDES=,I3,10X7H    AM=,F7.3,6X7H  ALTP=,F7.0,
     16X7H    T4=,F8.2,5X7H   T24=,F8.2,5X7H    T7=,F8.2,6H$$$$$$)
104   FORMAT(1H0,7H  IDES=,I3,10X7H    AM=,F7.3,6X7H  ALTP=,F7.0,
     16X7H  PCNC=,F8.3,5X7H   T24=,F8.2,5X7H    T7=,F8.2,6H$$$$$$)
105   FORMAT(1H0,7H  IDES=,I3,10X7H    AM=,F7.3,6X7H  ALTP=,F7.0,
     16X7H   WFB=,F8.4,5X7H   T24=,F8.2,5X7H    T7=,F8.2,6H$$$$$$)
      END
```

```
$IBFTC ZERO    DECK,M94/2,XR7
      SUBROUTINE ZERO
      COMMON /   ALL/
     1WORD   ,IDES   ,JDES   ,KDES   ,MODE   ,INIT   ,IDUMP  ,IAMTP  ,
     2IGASMX,IDBURN,IAFTBN,IDCD   ,IMCD   ,IDSHOC,IMSHOC,MUZFLT,
     3ITRYS ,LGOPER,NOMAP  ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON / FRONT/
     1T1     ,P1     ,H1     ,S1     ,T2     ,P2     ,H2     ,S2     ,
     2T21    ,P21    ,H21    ,S21    ,T3     ,P3     ,H3     ,S3     ,
     3T4     ,P4     ,H4     ,S4     ,T5     ,P5     ,H5     ,S5     ,
     4T55    ,P55    ,H55    ,S55    ,BLF    ,BLC    ,BLDU   ,BLUB   ,
     5CNF    ,PRF    ,ETAF   ,WAFC   ,WAF    ,WA3    ,WG4    ,FAR4   ,
     6CNC    ,PRC    ,ETAC   ,WACC   ,WAC    ,ETAB   ,DPCOMB ,DUMF   ,
     7CNHP   ,ETATHP,DHTCHP,DHTC   ,BLHP   ,WG5    ,FAR5   ,US     ,
     8CNLP   ,ETATLP,DHTCLP,DHTF   ,BLLP   ,WG55   ,FAR55  ,HPEXT  ,
     9AM     ,ALTP   ,ETAR   ,ZF     ,PCNF   ,ZC     ,PCNC   ,WFB    ,
     ATFFHP  ,TFFLP  ,PCBLF  ,PCBLC  ,PCBLDU,PCBLOB,PCBLHP,PCBLLP
      COMMON /  SIDE/
     XXP1    ,XWAF   ,XWAC   ,XBLF   ,XBLDU  ,XH3    ,DUMS1  ,DUMS2  ,
     XXT21   ,XP21   ,XH21   ,XS21   ,T23    ,P23    ,H23    ,S23    ,
     3T24    ,P24    ,H24    ,S24    ,T25    ,P25    ,H25    ,S25    ,
     4T26    ,P26    ,H26    ,S26    ,T29    ,P29    ,H29    ,S29    ,
     5WAD    ,WFD    ,WG24   ,FAR24  ,ETAD   ,DPDUC  ,BYPASS,DUMS3  ,
     6TS28   ,PS28   ,V28    ,AM28   ,TS29   ,PS29   ,V29    ,AM29
      COMMON /  BACK/
     XXT55   ,XP55   ,XH55   ,XS55   ,XT25   ,XP25   ,XH25   ,XS25   ,
     XXWFB   ,XWG55  ,XFAR55,XWFD    ,XWG24  ,XFAR24,XXP1    ,DUMB   ,
     3T6     ,P6     ,H6     ,S6     ,T7     ,P7     ,H7     ,S7     ,
     4T8     ,P8     ,H8     ,S8     ,T9     ,P9     ,H9     ,S9     ,
     5WG6    ,WFA    ,WG7    ,FAR7   ,ETAA   ,DPAFT  ,V55    ,V25    ,
     6TS6    ,V6     ,AM6    ,TS7    ,PS7    ,V7     ,AM7    ,AM25   ,
     7TS8    ,PS8    ,V8     ,AM8    ,TS9    ,PS9    ,V9     ,AM9    ,
     8VA     ,FRD    ,VJD    ,FGMD   ,VJM    ,FGMM   ,FGPD   ,FGPM   ,
     9FGM    ,FGP    ,WFT    ,WGT    ,FART   ,FG     ,FN     ,SFC
      DIMENSION Z1(63),Z2(48),Z3(72)
      EQUIVALENCE (Z1,T1),(Z2,XP1),(Z3,XT55)
      IDES=0
      JDES=0
      INIT=0
      IDBURN=0
      IAFTBN=0
      IDSHOC=3
      IMSHOC=3
      T20=T2
      P20=P2
      T40=T4
      DO 1 I=1,63
1     Z1(I)=0.
      DO 2 I=1,48
2     Z2(I)=0.
      DO 3 I=1,72
3     Z3(I)=0.
      T2=T20
      P2=P20
      T4=T40
      CALL SYG(1)
      RETURN
      END
```

```
$IBFTC COINLT  DECK,N94/2,XR7
      SUBROUTINE COINLT
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,MODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASMF,IDBURN,IAFTBN,IDCD  ,IMCD  ,IDSHOC,IMSHOC,NOZFLT,
     3ITRYS ,LOOPSR,NOMAP ,NUMMAP,RAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,T4GU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCUDS,ETABDS,WA3CDS,DPCODS,DTCUCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T24DS ,WFBDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     APS55  ,AM55  ,CVDNOZ,CVMNOZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON / FRONT/
     1T1    ,P1    ,H1    ,S1    ,T2    ,P2    ,H2    ,S2    ,
     2T21   ,P21   ,H21   ,S21   ,T3    ,P3    ,H3    ,S3    ,
     3T4    ,P4    ,H4    ,S4    ,T5    ,P5    ,H5    ,S5    ,
     4T55   ,P55   ,H55   ,S55   ,BLF   ,BLC   ,BLDU  ,BLDB  ,
     5CNF   ,PRF   ,ETAF  ,WAFC  ,WAF   ,WA3   ,WG4   ,FAR4  ,
     6CNC   ,PRC   ,ETAC  ,WACC  ,WAC   ,ETAB  ,DPCOM ,DUMF  ,
     7CNHP  ,ETATHP,DHTCHP,DHTC  ,BLHP  ,WG5   ,FAR5  ,CS    ,
     8CNLP  ,ETATLP,DHTCLP,DHTF  ,BLLP  ,WG55  ,FAR55 ,HPEXT ,
     9AM    ,ALTP  ,ETAR  ,ZF    ,PCNF  ,ZC    ,PCNC  ,WFB   ,
     ATFFHP ,TFFLP ,PCBLF ,PCBLC ,PCBLDU,PCBLDB,PCDLHP,PCBLLP
      DATA AWORD/6HCOINLT/
      WORD=AWORD
      AJ=778.26
      G=32.174049
      ALT=ALTP*2.0855531E+07/(2.0855531E+07-ALTP)
      CALL ATMOS(ALT,T1,XX1,XX2,XX3,P1,CS,XX4,IIER)
      IF(IAMTP.EQ.2) T1=T1+T2
1     IF(IAMTP.NE.1) CALL RAM(AM,ETAR)
      FAR=0.0
      CALL PROCOM(FAR,T1,CS,XX2,XX3,R1,PHI1,H1)
      S1=PHI1-R1*ALOG(P1)
      H2=H1+(AM*CS)**2/(2.*AJ*G)
      P2T=1.
      DO 2 I=1,10
      CALL THERMO(P2T,H2,T2T,S2T,AW,0,0.0,1)
      IF(ABS(S2T-S1).LE.0.0001*S1) GO TO 3
2     P2T=P1*EXP((AW/1.986375)*((S2T-S1)+(1.986375/AW)*ALOG(P2T/P1)).
      CALL ERROR
      RETURN
3     IF(IAMTP.EQ.3.OR.IAMTP.EQ.4) ETAR=P2/P2T
      P2=ETAR*P2T
      IF(IAMTP.NE.4) CALL THERMO(P2,H2,T2,S2,XX5,0,0.0,1)
      IF(IAMTP.EQ.4) CALL THERMO(P2,H2,T2,S2,XX5,0,0.0,0)
      IF(INIT.EQ.1) GO TO 6
      IF(IDES.EQ.1) GO TO 4
      PCNF=GUESS(MODE,T4,T4DS,PCNC,PCNCDS,WFB,WFBDS,T2,T2DS,PCNFDS)
      PCNFGU=PCNF
      GO TO 5
4     PCNF=PCNFDS
      PCNFGU=PCNF
      T2DS=T2
5     ZF=ZFDS
6     RETURN
```

END

```
$IBFTC ATM062  DECK,M94/2,XR7
      SUBROUTINE ATMOS (ZFT, TM, SIGMA, RHO, THETA, DELTA, CA, AMU, K)
C   THIS IS A SUBROUTINE TO COMPUTE CERTAIN ELEMENTS OF THE 1962
C   U.S. STANDARD ATMOSPHERE UP TO 90 KILOMETERS.
C    CALLING SEQUENCE...
C
C    CALL ATMOS (ZFT, TM, SIGMA, RHO, THETA, DELTA, CA, AMU, K)
C         ZFT   =   GEOMETRIC ALTITUDE (FEET)
C         TM    =   MOLECULAR SCALE TEMPERATURE (DEGREES RANKINE)
C         SIGMA =   RATIO OF DENSITY TO THAT AT SEA LEVEL
C         RHO   =   DENSITY (LB-SEC**2-FT**(-4)   OR   SLUGS-FT**3)
C         THETA =   RATIO OF TEMPERATURE TO THAT AT SEA LEVEL
C         DELTA =   RATIO OF PRESSURE TO THAT AT SEA LEVEL
C         CA    =   SPEED OF SOUND (FT/SEC)
C         AMU   =   VISCOSITY COEFFICIENT (LB-SEC/FT**2)
C
C         K = 1 NORMAL
C           = 2 ALTITUDE LESS THAN -5000 METERS OR GREATER THAN 90 KM
C           = 3 FLOATING POINT OVERFLOW
C
C   ALL DATA AND FUNDAMENTAL CONSTANTS ARE IN THE METRIC SYSTEM AS
C   THESE QUANTITIES ARE DEFINED AS EXACT IN THIS SYSTEM.
C
C   THE RADIUS OF THE EARTH (REFT59) IS THE VALUE ASSOCIATED WITH THE
C   1959 ARDC ATMOSPHERE SO THAT PROGRAMS CURRENTLY USING THE LIBRARY
C   ROUTINE WILL NOT REQUIRE ALTERATION TO USE THIS ROUTINE.
      DIMENSION HB(10),TMB(10),DELTAB(10),ALM(10)
      DATA(HB(I),    TMB(I),    DELTAB(I),    ALM(I),I=1,10)/
     A   -5.0,      320.65,   1.75363E 00,   -6.5,
     B    0.0,      288.15,   1.00000E 00,   -6.5,
     C   11.0,      216.65,   2.23361E-01,    0.0,
     D   20.0,      216.65,   5.40328E-02,    1.0,
     E   32.0,      228.65,   8.56663E-03,    2.8,
     F   47.0,      270.65,   1.09455E-03,    0.0,
     G   52.0,      270.65,   5.82289E-04,   -2.0,
     H   61.0,      252.65,   1.79718E-04,   -4.0,
     I   79.0,      180.65,   1.0241 E-05,    0.0,
     J   88.743,    180.65,   1.6223 E-06,    0.0/
      DATA  REFT59/2.0855531E 07/, GZ    /9.80665/,
     A      AMZ   /28.9644      /, RSTAR /8.31432/,
     B      FTTOKM/3.048E-04    /, S     /110.4  /,
     C      AMUZ  /1.2024E-05   /, CAZ   /1116.45/,
     D      RHOZ  /0.076474     /, GZENG /32.1741/
C  CONVERT GEOMETRIC ALTITUDE TO GEOPOTENTIAL ALTITUDE
      HFT = (REFT59/(REFT59+ZFT))*ZFT
C  CONVERT HFT AND ZFT TO KILOMETERS
      Z = FTTOKM*ZFT
      H = FTTOKM*HFT
      K = 1
      TMZ = TMB(2)
      IF (H.LT.-5.0.OR.Z.GT.90.0)  GO TO 16
      DO 10 M=1,10
      IF (H-HB(M)) 11,12,10
   10 CONTINUE
      GO TO 16
   11 M = M-1
   12 DELH = H-HB(M)
      IF (ALM(M).EQ.0.0)  GO TO 13
      TMX = TMB(M)+ALM(M)*DELH
Z  GRADIENT IS NON ZERO, PAGE 10, EQUATION 1.2.10-(3)
      DELTA = DELTAB(M)*((TMB(M)/TMX)**(GZ*AMZ/(RSTAR*ALM(M))))
```

```
      GO TO 14
   13 TMK = TMB(M)
C  GRADIENT IS ZERO, PAGE 10, EQUATION I.2.10-(4)
      DELTA = DELTAB(M)*EXP(-GZ*AMZ*DELH/(RSTAR*TMB(M)))
   14 THETA = TMK/TMZ
      SIGMA = DELTA/THETA
      ALPHA = SQRT(THETA**3)*((TMZ+S)/(TMK+S))
C  CONVERSION TO ENGLISH UNITS
      TM = 1.8*TMK
      RHO = RHOZ*SIGMA/GZENG
      CA = CAZ*SQRT(THETA)
      AMU = AMUZ*ALPHA/GZENG
      CALL OVERFL(J)
        GO TO (15,17), J
   15 K = K+2
      GO TO 17
   16 K = 2
   17 RETURN
      END
```

```
$IBFTC RAM      DECK,M94/2,XR7
      SUBROUTINE RAM(AM,ETAR)
      IF(AM.GT.1.) GO TO 2
      ETAR=1.
1     RETURN
2     IF(AM.GT.5.) GO TO 3
      ETAR=1.-0.075*((AM-1.)**1.35)
      GO TO 1
3     ETAR=800./((AM**4)+935.)
      GO TO 1
      END
```

```
$IBFTC GUESS    DECK,M94/2,XR7
      FUNCTION GUESS(M,T,TD,P,PD,W,WD,D,DD,VD)
      IF(M.EQ.0) GUESS=VD*((T/TD)**1.60)*((DD/D)**0.50)
      IF(M.EQ.1) GUESS=VD*((P/PD)**1.80)*((DD/D)**0.33)
      IF(M.EQ.2) GUESS=VD*((W/WD)**0.33)*((DD/D)**1.00)
      IF(M.EQ.3) GUESS=VD*((W/WD)**0.00)*((P/PD)**0.50)
      IF(M.EQ.4) GUESS=VD*((W/WD)**0.00)*((P/PD)**0.50)
      IF(M.EQ.5) GUESS=VD*((T/TD)**1.10)*((DD/D)**0.60)
      IF(M.EQ.6) GUESS=VD*((P/PD)**1.00)*((D/DD)**0.25)
      RETURN
      END
```

```
$IBFTC COFAN   DECK,M94/2,XR7
      SUBROUTINE COFAN
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,MODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASMX,IDBURN,IAFTBN,IDCD  ,IMCU  ,IDSHOC,IMSHOC,NOZFLT,
     3ITRYS ,LOOPER,NOMAP ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,T4GU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WF8DS ,DTCODS,ETABDS,WA3CDS,DPCODS,DTCOCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     APS55  ,AM55  ,CVDNOZ,CVHNOZ,ABS. V ,A9SAV ,A28SAV,A29SAV
      COMMON / FRONT/
     1T1    ,P1    ,H1    ,S1    ,T2    ,P2    ,H2    ,S2    ,
     2T21   ,P21   ,H21   ,S21   ,T3    ,P3    ,H3    ,S3    ,
     3T4    ,P4    ,H4    ,S4    ,T5    ,P5    ,H5    ,S5    ,
     4T55   ,P55   ,H55   ,S55   ,BLF   ,BLC   ,BLDU  ,BLDB  ,
     5CNF   ,PRF   ,ETAF  ,WAFC  ,WAF   ,WA3   ,WG4   ,FAR4  ,
     6CNC   ,PRC   ,ETAC  ,WACC  ,WAC   ,ETAB  ,DPCOM ,DUMF  ,
     7CNHP  ,ETATHP,DHTCHP,DHTC  ,BLHP  ,WG5   ,FAR5  ,CS    ,
     8CNLP  ,ETATLP,DHTCLP,DHTF  ,BLLP  ,WG55  ,FAR55 ,HPEXT ,
     9AM    ,ALTP  ,ETAR  ,ZF    ,PCNF  ,ZC    ,PCNC  ,WF8   ,
     ATFFHP ,TFFLP ,PCBLF ,PCBLC ,PCBLDU,PCBLDB,PCBLHP,PCBLLP
      COMMON /  FAN/CNX(15),PRX(15,15),WACX(15,15),ETAX(15,15),
     1NCN,NPT(15)
      DIMENSION WLH(2)
      DATA AWORD,WLH/6H COFAN,6H (LO) ,6H (HI) /
      WORD=AWORD
      THETA=SQRT(T2/518.668)
      CNF=PCNF/(100.*THETA)
      IF(ZF.LT.0.) ZF=0.
      IF(ZF.GT.1.) ZF=1.
      CNFS=CNF
      CALL SEARCH(ZF,CNF,PRF,WAFC,ETAF,
     1CNX(1),NCN,PRX(1,1),WACX(1,1),ETAX(1,1),NPT(1),15,15,IGO)
      IF((CNF-CNFS).GT.0.0005*CNF) MAPEDG=1
      IF(IGO.EQ.1.OR.IGO.EQ.2) WRITE(8,1000) CNFS,WLH(IGO)
1000  FORMAT(19H0* * *  CNF OFF MAP,F10.4,2XA6,11H> * *$$$$$$)
      WAF=WAFC*P2/THETA
      IF(IDES.NE.1) GO TO 1
      PRFCF=(PRFDS-1.)/(PRF-1.)
      ETAFCF=ETAFDS/ETAF
      WAFCF=WAFDS/WAF
      WRITE(6,100)PRFCF,ETAFCF,WAFCF,T2DS
100   FORMAT(11H0FAN DESIGN,13X8H  PRFCF=,E15.8,8H ETAFCF=,E15.8,
     18H  WAFCF=,E15.8,8H   T2DS=,F15.8)
1     PRF=PRFCF*(PRF-1.)+1.
      ETAF=ETAFCF*ETAF
      WAF=WAFCF*WAF
      PCNF=100.*THETA*CNF
      DUMD1=PCNF
      CALL THCOMP(PRF,ETAF,T2,H2,S2,P2,T21,H21,S21,P21)
      IF(PCBLF.GT.0.) BLF=PCBLF*WAF
      IF(JDES.EQ.1) GO TO 7
      JDES=1
      IF(INIT.EQ.1) GO TO 6
```

```
      IF(IDES.EQ.1) GO TO 4
      IF(MODE.NE.2) GO TO 2
      T4=GUESS(3,Y1,Y2,PCNF,PCNFDS,WFB,WFBDS,Y7,Y8,T4DS)
      PCNC=GUESS(4,Y1,Y2,PCNF,PCNFDS,WFB,WFBDS,Y7,Y8,PCNCDS)
      GO TO 5
2     IF(MODE.EQ.1) GO TO 3
      PCNC=GUESS(5,T4,T4DS,Y3,Y4,Y5,Y6,T21,T21DS,PCNCDS)
      GO TO 5
3     T4=GUESS(6,Y1,Y2,PCNC,PCNCDS,Y5,Y6,T21,T21DS,T4DS)
      GO TO 5
4     PCNC=PCNCDS
      T4=T4DS
      WFB=WFBDS
      T21DS=T21
5     ZC=ZCDS
      PCNCGU=PCNC
      T4GU=T4
6     INIT=0
7     CALL COCOMP
      IF(NOMAP.EQ.7) PCNF=DUMD1
      RETURN
      END
```

```
$IBFTC COCOMP  DECK,M94/2,XR7
      SUBROUTINE COCOMP
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,MODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASMX,IDBURN,IAFTBN,IDCD  ,IMCD  ,IDSHUC,INSHUC,NOZFLT,
     3ITRYS ,LOOPER,KOMAP ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,T4GU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
     2FDS   ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3CDS   ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCUDS,ETABDS,WA3CDS,DPCUDS,DTCUCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     APS55  ,AM55  ,CVDNOZ,CVMNOZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON / FRONT/
     1T1    ,P1    ,H1    ,S1    ,T2    ,P2    ,H2    ,S2    ,
     2T21   ,P21   ,H21   ,S21   ,T3    ,P3    ,H3    ,S3    ,
     3T4    ,P4    ,H4    ,S4    ,T5    ,P5    ,H5    ,S5    ,
     4T55   ,P55   ,H55   ,S55   ,BLF   ,BLC   ,BLDU  ,BLOB  ,
     5CNF   ,PRF   ,ETAF  ,WAFC  ,WAF   ,WA3   ,WG4   ,FAR4  ,
     6CNC   ,PRC   ,ETAC  ,WACC  ,WAC   ,ETAB  ,DPCOM ,DUMF  ,
     7CNHP  ,ETATHP,DHTCHP,DHTC  ,BLHP  ,WG5   ,FAR5  ,CS    ,
     8CNLP  ,ETATLP,DHTCLP,DHTF  ,BLLP  ,WG55  ,FAR55 ,HPEXT ,
     9AN    ,ALTP  ,ETAK  ,ZF    ,PCHF  ,ZC    ,PCNC  ,WFB   ,
     ATFFHP ,TFFLP ,PCBLF ,PCBLC ,PCBLDU,PCBLOB,PCBLHP,PCBLLP
      COMMON / COMP/CNX(15),PRX(15,15),WACX(15,15),ETAX(15,15),
     1NCN,NPT(15)
      DIMENSION WLH(2)
      DATA AWORD,WLH/6HCOCOMP,6H (LO) ,6H (HI) /
      WORD=AWORD
      THETA=SQRT(T21/518.668)
      CNC=PCNC/(100.*THETA)
      IF(ZC.LT.0.) ZC=0.
      IF(ZC.GT.1.) ZC=1.
      CNCS=CNC
      CALL SEARCH(ZC,CNC,PRC,WACC,ETAC,
     1CNX(1),NCN,PRX(1,1),WACX(1,1),ETAX(1,1),NPT(1),15,15,IGO)
      IF(MODE.EQ.1) GO TO 1
      IF((CNC-CNCS).GT.0.0005*CNC) MAPEDG=1
1     IF(IGO.EQ.1.OR.IGO.EQ.2) WRITE(8,1000) CNCS,WLH(IGO)
1000  FORMAT(19H0* * *  CNC OFF MAP,F10.4,2XA6,11H* * *$$$$$$)
      WAC=WACC*P21/THETA
      IF(IDES.NE.1) GO TO 2
      PRCCF=(PRCDS-1.)/(PRC-1.)
      ETACCF=ETACDS/ETAC
      WACCF=WACDS/WAC
      WRITE(6,100)PRCCF,ETACCF,WACCF,T21DS
100   FORMAT(18H0COMPRESSOR DESIGN,6X8H  PRCCF=,E15.8,8H ETACCF=,E15.8,
     18H  WACCF=,E15.8,8H  T21DS=,F15.8)
2     PRC=PRCCF*(PRC-1.)+1.
      ETAC=ETACCF*ETAC
      WAC=WACCF*WAC
      CALL THCOMP(PRC,ETAC,T21,H21,S21,P21,T3,H3,S3,P3)
      IF(PCBLC.GT.0.) BLC=PCBLC*WAC
      WA3=WAC-BLC
      BLDU=PCBLDU*BLC
      BLOB=PCBLOB*BLC
      BLHP=PCBLHP*BLC
```

```
      BLLP=PCBLLP*BLC
      IF(MODE.NE.1) GO TO 3
      IF(ABS(CNC-CNCS).LE.0.001*CNCS) GO TO 4
      WRITE(8,2000)CNCS,CNC
2000  FORMAT(10HOCNC WAS= ,E15.8,11H  AND NOW= ,E15.8,
     124H  CHECK PCNC INPUT$$$$$$)
      CALL ERROR
3     PCNC=100.*THETA*CNC
4     CALL COCOMB
      RETURN
      END
```

```
$IBFTC CUCOMB  DECK,M94/2,XR7
      SUBROUTINE CUCOMB
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,MODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASMX,IDBURN,IAFTBN,IDCD  ,IMCD  ,IDSMOC,IMSMOC,NUZFLT,
     3ITRYS ,LOOPER,NUMAP ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,T4CU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSPC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCODS,ETABDS,WA3CDS,DPCODS,DTCOCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     APS55  ,AM55  ,CVDMDZ,CVMMDZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON / FRONT/
     1T1    ,P1    ,H1    ,S1    ,T2    ,P2    ,H2    ,S2    ,
     2T21   ,P21   ,H21   ,S21   ,T3    ,P3    ,H3    ,S3    ,
     3T4    ,P4    ,H4    ,S4    ,T5    ,P5    ,H5    ,S5    ,
     4T55   ,P55   ,H55   ,S55   ,BLF   ,BLC   ,BLDU  ,BLDB  ,
     5WNF   ,PRF   ,ETAF  ,WAFC  ,WAF   ,WA3   ,WG4   ,FAR4  ,
     6CNC   ,PRC   ,ETAC  ,WACC  ,WAC   ,ETAB  ,DPCOM ,DUMF  ,
     7CNHP  ,ETATHP,ETHP  ,DHTC  ,BLHP  ,WG5   ,FAR5  ,CS    ,
     8CNLP  ,ETATLP,DHTLP,DHTF   ,BLLP  ,WG55  ,FAR55 ,HPEXT ,
     9AM    ,ALTP  ,ETAK  ,ZF    ,PCNF  ,ZC    ,PCNC  ,WFB   ,
     ATFFHP ,TFFLP ,PCBLF ,PCBLC ,PCBLDU,PCBLDB,PCBLHP,PCBLLP
      COMMON / COMB/PSI(15),DELT(15,15),ETA(15,15),NPS,NPT(15)
      DIMENSION Q(9),DUMBU(15,15)
      DATA AWORD/6HCUCOMB/
      WORD=AWORD
      Q(2)=0.
      Q(3)=0.
      P3PSI=14.696*P3
      WA3C=WA3*SQRT(T3)/P3PSI
      IF(IDES.EQ.1) WA3CDS=WA3C
      DPCOM=DPCODS*(WA3C/WA3CDS)
      IF(DPCOM.GT.1.) DPCOM=1.
      P4=P3*(1.-DPCOM)
1     IF(T4.GT.3000.) T4=3000.
      IF(T4.GE.1000.) GO TO 2
      T4=1000.
      IF(MODE.EQ.1) MAPEDG=1
2     DTCO=T4-T3
      IF(IDES.NE.1) GO TO 3
      DTCOCF=DTCODS/DTCO
3     DTCO=DTCOCF*DTCO
      P3PSIN=P3PSI
      CALL SEARCH(-1.,P3PSIN,DTCO,ETAB,DUMMY,
     1PSI(1),NPS,DELT(1,1),ETA(1,1),DUMBU(1,1),NPT(1),15,15,IGO)
      IF(IGO.EQ.7) CALL ERROR
4     IF(IDES.NE.1) GO TO 5
      ETABCF=ETABDS/ETAB
5     ETAB=ETABCF*ETAB
      HV=(((((-.4594317E-19*T4)-.2034116E-15)*T4+.2783643E-11)*T4
     1+.2051501E-07)*T4-.2453116E-03)*T4-.9433296E-01)*T4+.1845537E+05
      CALL THERMU(P4,HA,T4,XX1,XX2,0,0.0,0)
      FAR4=(HA-H3)/(HV*ETAB)
      IF(FAR4.LT.0.  FAR4=0.
      WFBX=FAR4*WA3
```

```
      IF(MODE.NE.2) GO TO 8
      ERRW=(WFB-WFBX)/WFB
      DIR=SQRT(WFB/WFBX)
      CALL AEQUIR(0(1),T4,ERRW,0.,20.,0.0001,DIR,T4T,IGO)
      GO TO (6,9,7),IGO
6     T4=T4T
      GO TO 1
7     CALL ERROR
8     WFB=WFBX
9     CALL THERMO(P4,H4,T4,S4,XX2,1,FAR4,0)
      WG4=WFB-WA3
      IF(IDES.EQ.1) WRITE(6,100) WA3CDS,ETABCF,DTCOCF
100   FORMAT(17HOCOMBUSTOR DESIGN,7X8H WA3CDS=,E15.8,8H ETABCF=,E15.8,
     18H DTCOCF=,E15.8)
      CALL COMPTB
      RETURN
      END
```

```
$IBFTC COMPTB  DECK,M94/2,XR7
      SUBROUTINE COMPTB
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,MODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASMX,IDBURN,IAFTBN,IDCD  ,IMCD  ,IDSHOC,IMSHOC,NOZFLT,
     3ITRYS ,LUGPER,NOMAP ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,T4GU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCGDS,ETABDS,WA3CDS,DPCUDS,DTCOCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T3DS  ,WF4DS ,DTAFDS,ETAADS,WC6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     A9555  ,AM55  ,CVDNOZ,CVMNOZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON / FRONT
     1T1    ,P1    ,H1    ,S1    ,T2    ,P2    ,H2    ,S2    ,
     2T21   ,P21   ,H21   ,S21   ,T3    ,P3    ,H3    ,S3    ,
     3T4    ,P4    ,H4    ,S4    ,T5    ,P5    ,H5    ,S5    ,
     4T55   ,P55   ,H55   ,S55   ,BLF   ,BLC   ,BLDU  ,BLUB  ,
     5CNF   ,PRF   ,ETAF  ,WAFC  ,WAF   ,WA3   ,WG4   ,FAR4  ,
     6CNC   ,PRC   ,ETAC  ,WACC  ,WAC   ,ETAB  ,DPCOM ,DUMF  ,
     7CNHP  ,ETATHP,DHTCHP,DHTC  ,BLHP  ,WG5   ,FAR5  ,CS    ,
     8CNLP  ,ETATLP,DHTCLP,DHTF  ,BLLP  ,WG55  ,FAR55 ,HPEXT ,
     9AM    ,ALTP  ,ETAR  ,ZF    ,PCNF  ,ZC    ,PCNC  ,WFB   ,
     ATFFHP ,TFFLP ,PCBLF ,PCBLC ,PCBLDU,PCBLUB,PCBLHP,PCBLLP
      COMMON /HTURB/TFFX(15),CNX(15,15),DHTCX(15,15),ETATX(15,15),
     1NTFFS,NPTTFF(15)
      DATA AWORD,WLO,WHI/6HCOMPTB,6H (LO) ,6H (HI) /
      WORD=AWORD
      IF(IDES.EQ.0) GO TO 1
      CNHPCF=CNHPDS*SQRT(T4)/PCNC
1     CNHP=CNHPCF*PCNC/SQRT(T4)
      CNHPS=CNHP
      TFFHPS=TFFHP
      CALL SEARCH(-1.,TFFHP,CNHP,DHTCHP,ETATHP,
     1TFFX(1),NTFFS,CNX(1,1),DHTCX(1,1),ETATX(1,1),NPTTFF(1),15,15,IGO)
      IF(IGO.EQ. 1.OR.IGO.EQ.11.OR.IGO.EQ.21) WRITE(8,1000)TFFHPS,WLO
      IF(IGO.EQ. 2.OR.IGO.EQ.12.OR.IGO.EQ.22) WRITE(8,1000)TFFHPS,WHI
      IF(IGO.EQ.10.OR.IGO.EQ.11.OR.IGO.EQ.12) WRITE(8,2000) CNHPS,WLO
      IF(IGO.EQ.20.OR.IGO.EQ.21.OR.IGO.EQ.22) WRITE(8,2000) CNHPS,WHI
1000  FORMAT(19H0*****TFFHP OFF MAP,F10.4,2XA6,11H*****$$$$$$)
2000  FORMAT(19H0***** CNHP OFF MAP,F10.4,2XA6,11H*****$$$$$$)
      IF(IGO.NE.7) GO TO 3
2     CALL ERROR
      RETURN
3     MAPGO=0
      IF(ABS(TFFHPS-TFFHP).LE.0.001*TFFHPS) GO TO 4
      MAPGO=1
      IF(ABS(CNHPS-CNHP).GT.0.001*CNHPS) MAPGO=3
      GO TO 5
4     IF(ABS(CNHPS-CNHP).GT.0.001*CNHPS) MAPGO=2
5     IF(MAPGO.GT.0) CALL MAPBAC(1,MAPGO,TFFHPS,TFFHP,CNHPS,CNHP,PCNC,
     1T4,MODE,NOMAP,NUMMAP)
      IF(NOMAP.GT.0) RETURN
      TFHCAL=WG4*SQRT(T4)/(14.696*P4)
      BTUEXT=0.706705*HPEXT
      DHTCC=(BTUEXT+WAC*(H3-H21))/(WG4*T4)
      IF(IDES.EQ.0) GO TO 6
```

```
      TFHPCF=TFHPDS/TFHCAL
      DHHPCF=DHTCC/DHTCHP
      ETHPCF=ETHPDS/ETATHP
      WRITE(6,102)CNHPCF,TFHPCF,ETHPCF,DHHPCF
102   FORMAT(20H0H.P. TURBINE DESIGN,5X7HCNHPCF=,E15.8,8H TFHPCF=,E15.8,
     18H ETHPCF=,E15.8,8H DHHPCF=,E15.8)
6     TFHCAL=TFHPCF*TFHCAL
      DHTCHP=DHHPCF*DHTCHP
      ETATHP=ETHPCF*ETATHP
      DHTC=DHTCC*T4
      ERR(1)=(TFHCAL-TFHHP)/TFHCAL
      ERR(2)=(DHTCC-DHTCHP)/DHTCC
      CALL THTURB(DHTC,ETATHP,FAR4,H4,S4,P4,T5,H5,S5,P5)
      IF(BLHP.LE.0.) GO TO 7
      FAR5=WFB/(WA3+BLHP)
      WG5=WG4+BLHP
      H5=(BLHP*H3+WG4*H5)/WG5
      CALL THERMO(P5,H5,T5,S5,XX2,1,FAR5,1)
      GO TO 8
7     FAR5=FAR4
      WG5=WG4
8     CALL COLPTB
      RETURN
      END
```

```
$IBFTC COLPTB  DECK,M94/2,XR7
      SUBROUTINE COLPTB
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,MODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASRX,IDBURN,IAFTBN,IDCD  ,IMCD  ,IDSHOC,IMSHOC,NOZFLT,
     3ITRYS ,LOOPER ,NOMAP ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,T4GU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCODS,ETABDS,WA3CDS,DPCODS,DTCOCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T7DS  ,WFADS ,DTAFDS,ETAADS,WG8CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     APS55  ,AM55  ,CVDNOZ,CVMNOZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON / FRONT/
     1T1    ,P1    ,H1    ,S1    ,T2    ,P2    ,H2    ,S2    ,
     2T21   ,P21   ,H21   ,S21   ,T3    ,P3    ,H3    ,S3    ,
     3T4    ,P4    ,H4    ,S4    ,T5    ,P5    ,H5    ,S5    ,
     4T55   ,P55   ,H55   ,S55   ,BLF   ,BLC   ,BLDU  ,BLDB  ,
     5CNF   ,PRF   ,ETAF  ,WAFC  ,WAF   ,WA3   ,WG4   ,FAR4  ,
     6CNC   ,PRC   ,ETAC  ,WACC  ,WAC   ,ETAB  ,DPCOM ,DUMF  ,
     7CNHP  ,ETATHP,DHTCHP,DHTC  ,BLHP  ,WG5   ,FAR5  ,CS    ,
     8CNLP  ,ETATLP,DHTCLP,DHTF  ,BLLP  ,WG55  ,FAR55 ,HPEXT ,
     9AM    ,ALTP  ,ETAR  ,ZF    ,PCNF  ,ZC    ,PCNC  ,WFB   ,
     ATFFHP ,TFFLP ,PCBLF ,PCBLC ,PCBLDU,PCBLDB,PCBLHP,PCBLLP
      COMMON /LTURB/TFFX(15),CNX(15,15),DHTCX(15,15),ETATX(15,15),
     1NTFFS,HPTTFF(15)
      DATA AWORD,WLO,WHI/6HCOLPTB,6H (LO) ,6H (HI) /
      WORD=AWORD
      IF(IDES.EQ.0) GO TO 1
      CNLPCF=CNLPDS*SQRT(T5)/PCNF
1     CNLP=CNLPCF*PCNF/SQRT(T5)
      CNLPS=CNLP
      TFFLPS=TFFLP
      CALL SEARCH(-1.,TFFLP,CNLP,DHTCLP,ETATLP,
     1TFFX(1),NTFFS,CNX(1,1),DHTCX(1,1),ETATX(1,1),HPTTFF(1),15,15,IGO)
      IF(IGO.EQ. 1.OR.IGO.EQ.11.OR.IGO.EQ.21) WRITE(8,1000)TFFLPS,WLO
      IF(IGO.EQ. 2.OR.IGO.EQ.12.OR.IGO.EQ.22) WRITE(8,1000)TFFLPS,WHI
      IF(IGO.EQ.10.OR.IGO.EQ.11.OR.IGO.EQ.12) WRITE(8,2000) CNLPS,WLO
      IF(IGO.EQ.20.OR.IGO.EQ.21.OR.IGO.EQ.22) WRITE(8,2000) CNLPS,WHI
1000  FORMAT(19H0*****TFFLP OFF MAP,F10.4,2XA6,11H*****$$$$$$)
2000  FORMAT(19H0***** CNLP OFF MAP,F10.4,2XA6,11H*****$$$$$$)
      IF(IGO.NE.7) GO TO 3
2     CALL ERROR
      RETURN
3     MAPGO=0
      IF(ABS(TFFLPS-TFFLP).LE.0.001*TFFLPS) GO TO 4
      MAPGO=1
      IF(ABS(CNLPS-CNLP).GT.0.001*CNLPS) MAPGO=3
      GO TO 5
4     IF(ABS(CNLPS-CNLP).GT.0.001*CNLPS) MAPGO=2
5     IF(MAPGO.GT.0) CALL MAPBAC(2,MAPGO,TFFLPS,TFFLP,CNLPS,CNLP,PCNF,
     1T4,MODE,NOMAP,NUMMAP)
      IF(NOMAP.GT.0) RETURN
      TFLCAL=WG5*SQRT(T5)/(14.696*F5)
      DHTCF=WAF*(H21-H2)/(WG5*T5)
      IF(IDES.EQ.0) GO TO 6
      TFLPCF=TFLPDS/TFLCAL
```

```
      DHLPCF=DHTCF/DHTCLP
      ETLPCF=ETLPDS/ETATLP
      WRITE(6,102)CNLPCF,TFLPCF,ETLPCF,DHLPCF
102   FORMAT(20HOL.P. TURBINE DESIGN,5X7HCNLPCF=,E15.8,8H TFLPCF=,E15.8,
     19H ETLPCF=,E15.8,8H DHLPCF=,E15.8)
6     TFLCAL=TFLPCF*TFLCAL
      DHTCLP=DHLPCF*DHTCLP
      ETATLP=ETLPCF*ETATLP
      DHTF=DHTCF*T5
      ERR(3)=(TFLCAL-TFFLP)/TFLCAL
      ERR(4)=(DHTCF-DHTCLP)/DHTCF
      CALL THTURB(DHTF,ETATLP,FAR5,H5,S5,P5,T55,H55,S55,P55)
      IF(BLLP.LE.0.) GO TO 7
      FAR55=WFB/(WA3+BLHP+BLLP)
      WG55=WG5+BLLP
      H55=(BLLP*H3+WG5*H55)/WG55
      CALL THERMO(P55,H55,T55,S55,XX2,1,FAR55,1)
      GO TO 8
7     FAR55=FAR5
      WG55=WG5
8     CALL FRTOSD
      RETURN
      END
```

```
$IBFTC FRTOSD  DECK,M94/2,XR7
      SUBROUTINE FRTOSD
      COMMON / FRONT/
     1T1     ,P1     ,H1     ,S1     ,T2     ,P2     ,H2     ,S2     ,
     2T21    ,P21    ,H21    ,S21    ,T3     ,P3     ,H3     ,S3     ,
     3T4     ,P4     ,H4     ,S4     ,T5     ,P5     ,H5     ,S5     ,
     4T55    ,P55    ,H55    ,S55    ,BLF    ,BLC    ,BLDU   ,BLUB   ,
     5CNF    ,PRF    ,ETAF   ,WAFC   ,WAF    ,WA3    ,WG4    ,FAR4   ,
     6CNC    ,PRC    ,ETAC   ,WACC   ,WAC    ,ETAB   ,DPCOM  ,DUMF   ,
     7CNHP   ,ETATHP,DHTCHP,DHTC   ,BLHP   ,WG5    ,FAR5   ,CS     ,
     8CNLP   ,ETATLP,DHTCLP,DHTF   ,BLLP   ,WG55   ,FAR55  ,HPEXT  ,
     9AM     ,ALTP   ,ETAR   ,ZF     ,PCNF   ,ZC     ,PCNC   ,WFB    ,
     ATFFHP  ,TFFLP  ,PCBLF  ,PCBLC  ,PCBLDU,PCBLUB,PCBLHP,PCBLLP
      COMMON /  SIDE/
     XXP1    ,XWAF   ,XWAC   ,XBLF   ,XBLDU  ,XH3    ,DUMS1  ,DUMS2  ,
     XXT21   ,XP21   ,XH21   ,XS21   ,T23    ,P23    ,H23    ,S23    ,
     3T24    ,P24    ,H24    ,S24    ,T25    ,P25    ,H25    ,S25    ,
     4T28    ,P28    ,H28    ,S28    ,T29    ,P29    ,H29    ,S29    ,
     5WAD    ,WFD    ,WG24   ,FAR24  ,ETAD   ,DPDUC  ,BYPASS,DUMS3  ,
     6TS28   ,PS28   ,V28    ,AM28   ,TS29   ,PS29   ,V29    ,AM29
      XP1=P1
      XWAF=WAF
      XWAC=WAC
      XBLF=BLF
      XBLDU=BLDU
      XH3=H3
      XT21=T21
      XP21=P21
      XH21=H21
      XS21=S21
      CALL CUDUCT
      RETURN
      END
```

```
$IBFTC CODUCT  DECK,M94/2,XR7
      SUBROUTINE CODUCT
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,MODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASMX,IDBURN,IAFTBN,IDCD  ,IMCD  ,IDSHOC,IMSHOC,NOZFLT,
     3ITRYS ,LOOPER,NOMAP ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,T4GU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCODS,ETABDS,WA3CDS,DPCODS,DTCOCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     APS55  ,AM55  ,CVDNOZ,CVHNOZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON /  SIDE/
     1P1    ,WAF   ,WAC   ,BLF   ,BLDU  ,H3    ,DUMS1 ,DUMS2 ,
     2T21   ,P21   ,H21   ,S21   ,T23   ,P23   ,H23   ,S23   ,
     3T24   ,P24   ,H24   ,S24   ,T25   ,P25   ,H25   ,S25   ,
     4T28   ,P28   ,H28   ,S28   ,T29   ,P29   ,H29   ,S29   ,
     5WAD   ,WFD   ,WG24  ,FAR24 ,ETAD  ,DPDUC ,BYPASS,DUMS3 ,
     6TS28  ,PS28  ,V28   ,AM28  ,TS29  ,PS29  ,V29   ,AM29
      DIMENSION Q(9)
      DATA AWORD1,AWORD2/6HCODUCT,6HDNOZZL/
      WORD=AWORD1
      Q(2)=0.
      Q(3)=0.
      WAX=WAF-WAC-BLF
      WAD=WAX+BLDU
      P23=P21
      H23=(BLDU*H3+WAX*H21)/WAD
      CALL THERMO(P23-H23,T23,S23,XX2,1,0.0,1)
      BYPASS=(WAF-WAC)/WAC
      WA23C=WAD*SQRT(T23)/P23
      IF(IDES.EQ.1) WA23DS=WA23C
      DPDUC=DPDUDS*(WA23C/WA23DS)
      IF(DPDUC.GT.1.) DPDUC=1.
      P24=P23*(1.-DPDUC)
      IF(IGASMX.GT.0) IDBURN=0
      IF(IDBURN.NE.0) GO TO 2
      T24=T23
      WFD=0.
      FAR24=0.
      GO TO 7
2     IF(IDBURN.EQ.2) T24=T23+2000.
3     IF(T24.GT.4000.) T24=4000.
      IF(T24.LT.T23) T24=T23
C *** IF DESIRED, ENTER CALCULATIONS FOR ETAD HERE
      HV=((((((-.4594317E-19*T24)-.2034116E-15)*T24+.2783643E-11)*T24+
     1.2051501E-07)*T24-.2453116E-03)*T24-.9433296E-01)*T24+.1845537E+05
      CALL THERMO(P24,HA,T24,XX1,XX2,0,0.0,0)
      FAR24=(HA-H23)/(HV*ETAD)
      IF(FAR24.LT.0.) FAR24=0.
      WFDX=FAR24*WAD
      IF(IDBURN.NE.2) GO TO 6
      ERRW=(WFD-WFDX)/WFD
      DIR=SQRT(WFD/WFDX)
      CALL AFQUIR(Q(1),T24,ERRW,0.,20.,0.0001,DIR,T24T,IGO)
      GO TO (4,7,5),IGO
```

```
4       T24=T24T
        GO TO 3
5       CALL ERROR
6       WFD=WFDX
7       CALL THERMO(P24,H24,T24,S24,XX2,1,FAR24,0)
        WG24=WFD+WAD
        IF(IDES.EQ.1) WRITE(6,101) WA23DS
101     FORMAT(12HODUCT DESIGN,12X8H WA23DS=,E15.8)
C *** IF DESIRED, ENTER OTHER LOSSES HERE
        T25=T24
        P25=P24
        H25=H24
        S25=S24
        IF(IGASMX.GT.0) GO TO 11
        WORD=AWORD2
        A28SAV=A28
        A29SAV=A29
        NOZD=0
        IDNOZ=0
        IF(NOZFLT.EQ.2.OR.NOZFLT.EQ.3) NOZD=1
        IF(IDES.EQ.1.OR.IDBURN.GT.0.OR.NOZD.EQ.1) IDNOZ=1
        IF(IDCD.EQ.1) GO TO 8
        CALL CONVRG(T25,H25,P25,S25,FAR24,WG24,P1,IDNOZ,A28,    P25R,
       1T28,H28,P28,S28,TS28,PS28,V28,AM28,ICON)
        GO TO (9,9,9,5),ICON
8       CALL CONDIV(T25,H25,P25,S25,FAR24,WG24,P1,IDNOZ,A28,A29,P25R,
       1T28,H28,P28,S28,T29,H29,P29,S29,TS28,TS29,PS28,PS29,V28,V29,AM28,
       2AM29,ICON)
        IDSHOC=ICON
        GO TO (10,10,10,5),ICON
9       T29=T28
        H29=H28
        P29=P28
        S29=S28
        TS29=TS28
        PS29=PS28
        V29=V28
        AM29=AM28
        A29=A28
        IDSHOC=ICON+3
10      ERR(5)=(P25R-P25)/P25R
        IF(IDNOZ.EQ.1) WRITE(6,100) A28,AM28,A29,AM29
100     FORMAT(19HODUCT NOZZLE DESIGN,5X8H    A28=,E15.8,8H    AM28=,E15.8,
       18H    A29=,E15.8,8H    AM29=,E15.8)
11      CALL FAST8K
        RETURN
        END
```

```
$IBFTC FASTBK  DECK,M94/2,XR7
      SUBROUTINE FASTBK
      COMMON / FRONT/
     1T1     ,P1     ,H1     ,S1     ,T2     ,P2     ,H2     ,S2     ,
     2T21    ,P21    ,H21    ,S21    ,T3     ,P3     ,H3     ,S3     ,
     3T4     ,P4     ,H4     ,S4     ,T5     ,P5     ,H5     ,S5     ,
     4T55    ,P55    ,H55    ,S55    ,BLF    ,BLC    ,BLDU   ,BLOB   ,
     5CNF    ,PRF    ,ETAF   ,WAFC   ,WAF    ,WA3    ,WG4    ,FAR4   ,
     6CNC    ,PRC    ,ETAC   ,WACC   ,WAC    ,ETAB   ,DPCON  ,DUMF   ,
     7CNHP   ,ETATHP,DHTCHP,DHTC   ,BLHP   ,WG5    ,FAR5   ,CS     ,
     8CNLP   ,ETATLP,DHTCLP,DHTF   ,BLLP   ,WG55   ,FAR55  ,HPEXT  ,
     9AM     ,ALTP   ,ETAR   ,ZF     ,PCNF   ,ZC     ,PCNC   ,WFB    ,
     ATFFHP  ,TFFLP  ,PCBLF  ,PCBLC  ,PCBLDU,PCBLOB,PCBLHP,PCBLLP
      COMMON /  SIDE/
     XXP1    ,XWAF   ,XWAC   ,XBLF   ,XBLDU  ,XH3    ,DUMS1  ,DUMS2  ,
     XXT21   ,XP21   ,XH21   ,XS21   ,T23    ,P23    ,H23    ,S23    ,
     3T24    ,P24    ,H24    ,S24    ,T25    ,P25    ,H25    ,S25    ,
     4T28    ,P28    ,H28    ,S28    ,T29    ,P29    ,H29    ,S29    ,
     5WAD    ,WFD    ,WG24   ,FAR24  ,ETAD   ,DPDUC  ,BYPASS,DUMS3  ,
     6TS28   ,PS28   ,V28    ,AM28   ,TS29   ,PS29   ,V29    ,AM29
      COMMON /  BACK/
     XXT55   ,XP55   ,XH55   ,XS55   ,XT25   ,XP25   ,XH25   ,XS25   ,
     XXWFB   ,XWG55  ,XFAR55,XWFD   ,XWG24  ,XFAR24,XXP1   ,DUMB   ,
     3T6     ,P6     ,H6     ,S6     ,T7     ,P7     ,H7     ,S7     ,
     4T8     ,P8     ,H8     ,S8     ,T9     ,P9     ,H9     ,S9     ,
     5WG6    ,WFA    ,WG7    ,FAR7   ,ETAC   ,DPAFT  ,V55    ,V25    ,
     6PS6    ,V6     ,AM6    ,TS7    ,PS7    ,V7     ,AM7    ,AM25   ,
     7TS8    ,PS8    ,V8     ,AM8    ,TS9    ,PS9    ,V9     ,AM9    ,
     8VA     ,FRD    ,VJD    ,FGMD   ,VJM    ,FGMM   ,FGPD   ,FGPM   ,
     9FGM    ,FGP    ,WFT    ,WGT    ,FART   ,FG     ,FN     ,SFC
      XT55=T55
      XP55=P55
      XH55=H55
      XS55=S55
      XT25=T25
      XP25=P25
      XH25=H25
      XS25=S25
      XWFB=WFB
      XWG55=WG55
      XFAR55=FAR55
      XWFD=WFD
      XWG24=WG24
      XFAR24=FAR24
      XXP1=P1
      CALL COMIX
      RETURN
      END
```

```
$IBFTC COMIX   DECK,H94/2,XR7
      SUBROUTINE COMIX
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,MODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASMX,IDBURN,IAFTBN,IDCD  ,IMCD  ,IDSHOC,IMSHOC,NOZFLT,
     3ITRYS ,LOOPER,NOMAP ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1FCNFGU,PCNCGU,T4GU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCODS,ETABDS,WA3CDS,DPCODS,DTCOCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DNHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DNLPCF,T21DS ,
     7T24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     APS55  ,AM55  ,CVDNOZ,CVMNOZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON /  BACK/
     1T55   ,P55   ,H55   ,S55   ,T25   ,P25   ,H25   ,S25   ,
     2WFB   ,WG55  ,FAR55 ,WFD   ,WG24  ,FAR24 ,P1    ,DUMB  ,
     3T6    ,P6    ,H6    ,S6    ,T7    ,P7    ,H7    ,S7    ,
     4T8    ,P8    ,H8    ,S8    ,T9    ,P9    ,H9    ,S9    ,
     5WG6   ,WFA   ,WG7   ,FAR7  ,ETAA  ,DPAFT ,V55   ,V25   ,
     6PS6   ,V6    ,AM6   ,TS7   ,PS7   ,V7    ,AM7   ,AM25  ,
     7TS8   ,PS8   ,V8    ,AM8   ,TS9   ,PS9   ,V9    ,AM9   ,
     8VA    ,FRD   ,VJD   ,FGRD  ,VJM   ,FGRM  ,FGPD  ,FGPM  ,
     9FGM   ,FGP   ,WFT   ,WGT   ,FART  ,FG    ,FN    ,SFC
      DATA AWORD/6H COMIX/
      DIMENSION QQ(9)
      WORD=AWORD
      AJ=778.26
      CAPSF=2116.2170
      G=32.174049
      CALL PROCON(FAR55,T55,XX1,XX2,XX3,XX4,PHI55,XX5)
      CALL PROCON(FAR24,T25,XX1,XX2,XX3,XX4,PHI25,XX5)
      IF(IDES.EQ.0) GO TO 6
C *** CALCULATE A55 AND A25 WITH PS25=PS55
      IF(PS55.EQ.0.) GO TO 50
      TS55=T55*(PS55/P55)**0.286
      DO 1 I=1,15
      CALL PROCON(FAR55,TS55,CS55,AK55,CP55,REX55,PHIS55,HS55)
      PHIS=PHI55-REX55*ALOG(P55/PS55)
      DELPHI=PHIS-PHIS55
      IF(ABS(DELPHI).LE.0.0001*PHIS) GO TO 3
1     TS55=TS55*EXP(4.0*DELPHI)
2     CALL ERROR
      RETURN
50    TS55=0.875*T55
      DO 51 I=1,15
      CALL PROCON(FAR55,TS55,CS55,AK55,CP55,REX55,PHIS55,HS55)
      V55=AM55*CS55
      HSCAL=H55-V55**2/(2.*G*AJ)
      DELHS=HSCAL-HS55
      IF(ABS(DELHS).LE.0.0005*HSCAL) GO TO 52
51    TS55=TS55+DELHS/CP55
      GO TO 2
52    PS55=P55/EXP((PHI55-PHIS55)/REX55)
3     IF(H55.GT.HS55) GO TO 53
      WRITE(6,101)P55,PS55,T55,TS55,H55,HS55
101   FORMAT(22HOSQRT OF H55-HS55 NEG ,6E15.6,6H$$$$$$)
      CALL ERROR
```

```
53    V55=SQRT(2.*G*AJ*(H55-HS55))
      RHO=CAPSF*PS55/(AJ*REX55*TS55)
      A55=WG55/(RHO*V55)
      AM55=V55/CS55
      IF(IGASMX.GT.0) GO TO 54
      WRITE(6,104)A55,AM55
104   FORMAT(20HOTURBINE AREA DESIGN,6X6H  A55=,E15.8,8H   AM55=,E15.8)
      GO TO 28
54    PS25=PS55
      TS25=T25*(PS25/P25)**0.286
      DO 4 I=1,15
      CALL PROCOR(FAR24,TS25,CS25,AK25,CP25,REX25,PHIS25,HS25)
      PHIS=PHI25-REX25*ALOG(P25/PS25)
      DELPHI=PHIS-PHIS25
      IF(ABS(DELPHI).LE.0.0001*PHIS) GO TO 5
4     TS25=TS25*EXP(4.0*DELPHI)
      GO TO 2
5     IF(H25.GT.HS25) GO TO 55
      WRITE(8,102)P25,PS25,T25,TS25,H25,HS25
102   FORMAT(22HOSQRT OF H25-HS25 NEG ,6E15.6,6H$$$$$$)
      CALL ERROR
55    V25=SQRT(2.*G*AJ*(H25-HS25))
      RHO=CAPSF*PS25/(AJ*REX25*TS25)
      A25=WG24/(RHO*V25)
      AM25=V25/CS25
      WRITE(6,100)A55,AM55,A25,AM25
100   FORMAT(25HOTURBINE/DUCT AREA DESIGN,7H   A55=,E15.8,
     18H   AM55=,E15.8,8H    A25=,E15.8,8H   AM25=,E15.8)
      GO TO 20
C *** CALCULATE PS55 AND PS25
6     WQA=WG55/A55
      C1=P55*SQRT(G/(T55*AJ))*CAPSF
      MCON=0
      QQ(2)=0.
      QQ(3)=0.
      AM55=0.50
      TS55=0.875*T55
7     DO 8 I=1,15
      CALL PROCOM(FAR55,TS55,CS55,AK55,CP55,REX55,PHIS55,HS55)
      V55=AM55*CS55
      HSCAL=H55-V55**2/(2.*G*AJ)
      DELHS=HSCAL-HS55
      IF(ABS(DELHS).LE.0.0005*HSCAL) GO TO 9
8     TS55=TS55+DELHS/CP55
      GO TO 2
9     WQAT=C1*SQRT(AK55/REX55)*AM55/(1.+(AK55-1.)*AM55**2/2.)**
     1((AK55+1.)/(2.*(AK55-1.)))
      AMX=AM55
      IGOGO=0
10    DIP=WQA/WQAT
      EN=(WQA-WQAT)/WQA
      CALL AFOUIR(QQ(1),AMX,EN,0.,30.,0.0005,DIP,AMXT,ICON)
      GO TO (11,15,2),ICON
11    IF(AMXT.LE.1.0) GO TO 13
      AMXT=0.7
      MCON=MCON+1
      IF(MCON.LE.1) GO TO 13
      PCNF=DUMD1
      WRITE(8,103)PCNF,AMX,P55,PS55,P25,PS25
103   FORMAT(32HOCOMIX PCNF=,F7.4,4H AM=,F8.6,5H P55=,F9.5,
     16H PS55=,F9.5,5H P25=,F9.5,6H PS25=,F9.5,6H$$$$$$)
```

```
      PCNF=1.01*PCNF
      DUMO1=PCNF
12    MOMAP=7
      RETURN
13    IF(IGOGO.EQ.1) GO TO 14
      AM55=AMXT
      GO TO 7
14    AM25=AMXT
      GO TO 16
15    IF(IGOGO.EQ.1) GO TO 19
      PS55=P55/EXP((PHI55-PHIS55)/REX55)
      IF(IGASMX.LE.0) GO TO 28
      WQA=WG24/A25
      C1=P25*SQRT(G/(T25*AJ))*CAPSF
      MCON=0
      QQ(2)=0.
      QQ(3)=0.
      AM25=0.25
      TS25=0.875*T25
16    DO 17 I=1,15
      CALL PROCOM(FAR24,TS25,CS25,AK25,CP25,REX25,PHIS25,HS25)
      V25=AM25*CS25
      HSCAL=H25-V25**2/(2.*G*AJ)
      DELHS=HSCAL-HS25
      IF(ABS(DELHS).LE.0.0005*HSCAL) GO TO 18
17    TS25=TS25+DELHS/CP25
      GO TO 2
18    WQAT=C1*SQRT(AK25/REX25)*AM25/(1.+(AK25-1.)*AM25**2/2.)**
     1((AK25+1.)/(2.*(AK25-1.)))
      AMX=AM25
      IGOGO=1
      GO TO 10
19    PS25=P25/EXP((PHI25-PHIS25)/REX25)
20    WG6=WG24+WG55
      ERR(5)=(PS25-PS55)/PS25
      WF6=WFD+WFB
      FAR6=WF6/(WG6-WF6)
      H6=(WG24*H25+WG55*H55)/WG6
      CALL THERMO(1.,H6,T6,PHI6,AMX,1,FAR6,1)
      C1=PS55*A55*(1.+AK55*AM55**2)+PS25*A25*(1.+AK25*AM25**2)
      TS6=0.833*T6
      DO 23 I=1,15
      CALL PROCOM(FAR6,TS6,CS6,AK6,CP6,REX6,PHIS6,HS6)
      C2=WG6*SQRT(AJ*REX6*T6/(AK6*G))
      C3=C2/(CAPSF*C1)
      C4=(AK6-1.)/2.-(C3*AK6)**2
      C5=1.-2.*AK6*C3**2
      C6=C5**2+4.*C4*C3**2
      IF(C6)21,22,23
21    CALL ERROR
      RETURN
22    AM62=-C5/(2.*C4)
      GO TO 24
23    AM62=(SQRT(C6)-C5)/(2.*C4)
24    IF(AM62.LE.0.) GO TO 21
      AM6=SQRT(AM62)
      V6=AM6*CS6
      HSCAL=H6-V6**2/(2.*G*AJ)
      DELHS=HSCAL-HS6
      IF(ABS(DELHS).LE.0.0005*HSCAL) GO TO 26
25    TS6=TS6+DELHS/CP6
```

```
      GO TO 21
26    IF(IGASMX.EQ.2) GO TO 27
      A6=A25+A55
27    C7=SQRT(1.+(AK6-1.)*AM62/2.)
      PS6=C2/(CAPSF*A6*AM6*C7)
      P6=PS6*EXP((PHI6-PHIS6)/REX6)
      CALL THERMO(P6,H6,T6,S6,XX1,1,FAR6,0)
      S6AVE=(WG24*S25+WG55*S55)/WG6
      IF(S6.GE.S6AVE) GO TO 29
      S6=S6AVE
      P6=EXP(AMX*(PHI6-S6)/1.986375)
      GO TO 29
28    T6=T55
      P6=P55
      H6=H55
      S6=S55
      WG6=WG55
      PS6=PS55
      V6=V55
      AM6=AM55
      IF(IGASMX.EQ.0) A6=A55
29    CALL COAFBN
      RETURN
      END
```

```
$IBFTC COAFBN  DECK,M94/2,XR7
      SUBROUTINE COAFBN
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,MODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASMX,IGBURN,IAFTBN,IDCD  ,IMCD  ,IDSMOC,IMCMOC,NOZFLT,
     3ITRYS ,LOOPER,NOMAP ,NUMMAP,MAPEDG,TOLALL,ERR(5)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,T4GU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCODS,ETABDS,WA3CDS,DPCODS,DTCOCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     AFS55  ,AM55  ,CVDNOZ,CVMNOZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON /  BACK/
     1T55   ,P55   ,H55   ,S55   ,T25   ,P25   ,H25   ,S25   ,
     2WFB   ,WG55  ,FAR55 ,WFD   ,WG24  ,FAR24 ,P1    ,DUMB  ,
     3T6    ,P6    ,H6    ,S6    ,T7    ,P7    ,H7    ,S7    ,
     4T8    ,P8    ,H8    ,S8    ,T9    ,P9    ,H9    ,S9    ,
     5WG6   ,WFA   ,WG7   ,FAR7  ,ETAA  ,DPAFT ,V55   ,V25   ,
     6PS6   ,V6    ,AM6   ,TS7   ,PS7   ,V7    ,AM7   ,AM25  ,
     7TS8   ,PS8   ,V8    ,AM8   ,TS9   ,PS9   ,V9    ,AM9   ,
     8VA    ,FRD   ,VJD   ,FGRD  ,VJM   ,FGMN  ,FGPD  ,FGPM  ,
     9FGM   ,FGP   ,WFT   ,WGT   ,FART  ,FG    ,FN    ,SFC
      DIMENSION Q(9)
      DATA AWORD/6HCOAFBN/
      WORD=AWORD
      Q(2)=0.
      Q(3)=0.
      AJ=778.26
      CAPSF=2116.2170
      G=32.174049
      WF6=WFB
      IF(IGASMX.GT.0) WF6=WF6+WFD
      WA6=WG6-WF6
C *** DRY LOSS
      WG6C=WG6*SQRT(T6)/P6
2     IF(IDES.EQ.1) WG6CDS=WG6C
      DPAFT=DPAFDS*(WG6C/WG6CDS)
      IF(DPAFT.GT.1.) DPAFT=1.
      P7=P6*(1.-DPAFT)
      A7=A6
      FAR6=WF6/WA6
      CALL PROCON(FAR6,T6,XX1,XX2,XX3,XX4,PHI6,XX6)
      WQA=WG6/A7
      C1=P7*SQRT(G/(T6*AJ))*CAPSF
      AM7=AM6
      TS7=0.875*T6
20    DO 22 I=1,15
      CALL PROCON(FAR6,TS7,CS7,AK7,CP7,REX7,PHIS7,MS7)
      V7=AM7*CS7
      HSCAL=H6-V7**2/(2.*G*AJ)
      DELHS=HSCAL-HS7
      IF(ABS(DELHS).LE.0.0005*HSCAL) GO TO 24
22    TS7=TS7+DELHS/CP7
      GO TO 8
24    WQAT=C1*SQRT(AK7/REX7)*AM7/(1.+(AK7-1.)*AM7**2/2.)**
     1((AK7+1.)/(2.*(AK7-1.)))
```

```
      DIR=WQA/WQAT
      EW=(WQA-WQAT)/WQA
      CALL AFQUIR(Q(1),AM7,EW,0.,30.,0.0005,DIR,AM7T,IGO)
      GO TO (26,28,8),IGO
26    AM7=AM7T
      IF(AM7.GE.1.0) AM7=0.9
      GO TO 20
28    PS7=P7/EXP((PHI6-PHIS7)/REX7)
      IF(IAFTBN.GT.0) GO TO 4
C *** NON-AFTERBURNING
3     T7=T6
      WFA=0.0
      FAR7=FAR6
      WG7=WG6
      GO TO 13
C *** AFTERBURNING
4     IF(IAFTBN.EQ.2) T7=T6+2000.
      IF(T7.LE.T6) GO TO 3
      RHO65=CAPSF*PS7/(AJ*REX7*TS7)
      PS65=PS7
      V65=V7
      Q(2)=0.
      Q(3)=0.
5     IF(T7.GT.4000.) T7=4000.
C *** IF DESIRED, ENTER CALCULATIONS FOR STAA HERE
      HV=((((((-.4594317E-19*T7)-.2034116E-15)*T7+.2782643E-11)*T7
     1+.2051501E-07)*T7-.2453116E-03)*T7-.9433296E-01)*T7+.1645537E+05
      CALL THERMO(P7,HA,T7,XX1,XX2,0,0,0,0)
      FAR7=(HA-H6)/(HV*ETAB)
      IF(FAR7.GT.0.) GO TO 6
      T7=T6
      GO TO 5
6     WFAX=FAR7*WG6
      IF(IAFTBN.EQ.1) GO TO 9
      ERRW=(WFA-WFAX)/WFA
      DIR=SQRT(WFA/WFAX)
      CALL AFQUIR(Q(1),T7,ERRW,0.,20.,0.0001,DIR,T7T,IGO)
      GO TO (7,10,8),IGO
7     T7=T7T
      GO TO 5
8     CALL ERROR
9     WFA=WFAX
10    FAR7=(WF6+WFA)/WA6
      WG7=WG6+WFA
C *** MOMENTUM LOSS
      CALL PROCON(FAR7,T7,XX1,XX2,XX3,REX7,PHI7,H7)
      RHO7=CAPSF*P7/(AJ*REX7*T7)
      V7=WG7/(RHO7*A7)
      Q(2)=0.
      Q(3)=0.
      PS7=PS65-0.01
11    RHO7=WG7/(V7*A7)
      HS7=H7-V7**2/(2.*G*AJ)
      CALL THERMO(1.0,HS7,TS7,PHIS7,XX2,1,FAR7,1)
      IF(TS7.GE.301.) GO TO 110
      CALL THERMO(1.0,HS7,400.,PHIS7,XX2,1,FAR7,0)
      V7=SQRT(2.*G*AJ*(H7-HS7))
      GO TO 11
110   PS7=RHO7*AJ*REX7*TS7/CAPSF
      PS7A=PS65+(RHO65*V65**2-RHO7*V7**2)/(G*CAPSF)
      DIR=SQRT(ABS(PS7/PS7A))
```

```
      EP=(PS7-PS7A)/PS7
      CALL AFQUIR(Q(1),V7,EP,0.,50.,0.0005,DIR,V7T,IGO)
      V7=V7T
      IF(V7.LT.100.) V7=100.
      GO TO (11,12,8),IGO
12    P7=PS7*EXP((PHI7-PHIS7)/REX7)
      CALL PROCOM(FAR7,TS7,CS7,XX2,XX3,XX4,XX5,XX6)
      AM7=V7/CS7
13    CALL THERMO(P7,H7,T7,S7,XX2,1,FAR7,0)
      IF(IDES.EQ.1) WRITE(6,100) WG6CDS
100   FORMAT(19HOAFTERBURNER DESIGN,5X8H WG6CDS=,E15.8)
      CALL COMNOZ
      RETURN
      END
```

```
$IBFTC COMNOZ  DECK,M94/2,XR7
      SUBROUTINE COMNOZ
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,MODE  ,INIT  ,IDUMP ,IANTP ,
     2IGASMX,IDBURN,IAFTBN,IDCD  ,IMCD  ,IDSHOC,IMSHOC,NOZFLT,
     3ITRYS ,LOOPER,NOMAP ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,TAGU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCODS,ETABDS,WA3CDS,DPCODS,DTCOCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     APS55  ,AM55  ,CVDNOZ,CVMNOZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON /  BACK/
     1T55   ,P55   ,H55   ,S55   ,T25   ,P25   ,H25   ,S25   ,
     2WFB   ,WG55  ,FAR55 ,WFD   ,WG24  ,FAR24 ,P1    ,DUMB  ,
     3T6    ,P6    ,H6    ,S6    ,T7    ,P7    ,H7    ,S7    ,
     4T8    ,P8    ,H8    ,S8    ,T9    ,P9    ,H9    ,S9    ,
     5WG6   ,WFA   ,WG7   ,FAR7  ,ETAA  ,DPAFT ,V55   ,V25   ,
     6PS6   ,V6    ,AM6   ,TS7   ,PS7   ,V7    ,AM7   ,AM25  ,
     7TS8   ,PS8   ,V8    ,AM8   ,TS9   ,PS9   ,V9    ,AM9   ,
     8VA    ,FRD   ,VJD   ,FGMD  ,VJM   ,FGMR  ,FGPD  ,FGPM  ,
     9FGM   ,FGP   ,WFT   ,WGT   ,FART  ,FG    ,FM    ,SFC
      DATA AWORD/6HMNOZZL/
      WORD=AWORD
      A8SAV=A8
      A9SAV=A9
      NOZM=0
      IMNOZ=0
      IF(NOZFLT.EQ.1.OR.NOZFLT.EQ.3) NOZM=1
      IF(IDES.EQ.1.OR.IAFTBN.GT.0.OR.NOZM.EQ.1) IMNOZ=1
      IF(IMCD.EQ.1) GO TO 1
      CALL CONVRG(T7,H7,P7,S7,FAR7,WG7,P1,IMNOZ,A8,   P7R,
     1T8,H8,P8,S8,TS8,PS8,V8,AM8,ICON)
      GO TO (3,3,3,2),ICON
1     CALL CONDIV(T7,H7,P7,S7,FAR7,WG7,P1,IMNOZ,A8,A9,P7R,
     1T8,H8,P8,S8,T9,H9,P9,S9,TS8,TS9,PS8,PS9,V8,V9,AM8,AM9,ICON)
      IMSHOC=ICON
      GO TO (4,4,4,2),ICON
2     CALL ERROR
3     T9=T8
      H9=H8
      P9=P8
      S9=S8
      TS9=TS8
      PS9=PS8
      V9=V8
      AM9=AM8
      A9=A8
      IMSHOC=ICON+3
4     ERR(6)=(P7R-P7)/P7R
      IF(IMNOZ.EQ.1) WRITE(6,100) A8,AM8,A9,AM9
100   FORMAT(14HONOZZLE DESIGN,10X8H     A8=,E15.8,8H    AM8=,E15.8,
     18H     A9=,E15.8,8H    AM9=,E15.8)
      RETURN
      END
```

```
$IBFTC PERF    DECK,M94/2,XR7
      SUBROUTINE PERF
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,NODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASMX,IDBURN,IAFTBN,IDCD  ,IMCD  ,IDSHOC,IMSHOC,NOZFLT,
     3ITRYS ,LOOPER,NONAP ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,T4GU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCODS,ETABDS,WA3CDS,DPCODS,DTCOCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     6YFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T24DS ,WFODS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     APS55  ,AM55  ,CVDNOZ,CVKNOZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON / FRONT/
     1T1    ,P1    ,H1    ,S1    ,T2    ,P2    ,H2    ,S2    ,
     2T21   ,P21   ,H21   ,S21   ,T3    ,P3    ,H3    ,S3    ,
     3T4    ,P4    ,H4    ,S4    ,T5    ,P5    ,H5    ,S5    ,
     4T55   ,P55   ,H55   ,S55   ,BLF   ,BLC   ,BLDU  ,BLOB  ,
     5CHF   ,PRF   ,ETAF  ,WAFC  ,WAF   ,WA3   ,WG4   ,FAR4  ,
     6CHC   ,PRC   ,ETAC  ,WACC  ,WAC   ,ETAB  ,DPCOM ,DUMF  ,
     7CHHP  ,ETATHP,DHTCHP,CHTC  ,BLHP  ,WG5   ,FAR5  ,CS    ,
     8CHLP  ,ETATLP,DHTCLP,CHTF  ,BLLP  ,WG55  ,FAR55 ,HPEXT ,
     9AM    ,ALTP  ,ETAR  ,ZF    ,PCNF  ,ZC    ,PCNC  ,WFB   ,
     ATFFHP ,TFFLP ,PCBLF ,PCBLC ,PCBLDU,PCBLOB,PCBLHP,PCBLLP
      COMMON /  SIDE/
     XXP1   ,XWAF  ,XWAC  ,XBLF  ,XBLDU ,XH3   ,DUMS1 ,DUMS2 ,
     XXT21  ,XP21  ,XH21  ,XS21  ,T23   ,P23   ,H23   ,S23   ,
     3T24   ,P24   ,H24   ,S24   ,T25   ,P25   ,H25   ,S25   ,
     4T28   ,P28   ,H28   ,S28   ,T29   ,P29   ,H29   ,S29   ,
     5WAD   ,WFD   ,WG24  ,FAR24 ,ETAD  ,DPDUM ,BYPASS,DUMS3 ,
     6TS28  ,PS28  ,V28   ,AM28  ,TS29  ,PS29  ,V29   ,AM29
      COMMON /  BACK/
     XXT55  ,XP55  ,XH55  ,XS55  ,XT25  ,XP25  ,XH25  ,XS25  ,
     XXWFB  ,XWG55 ,XFAR55,XWFD  ,XWG24 ,XFAR24,XXP1  ,DUMB  ,
     3T6    ,P6    ,H6    ,S6    ,T7    ,P7    ,H7    ,S7    ,
     4T8    ,P8    ,H8    ,S8    ,T9    ,P9    ,H9    ,S9    ,
     5WG6   ,WFA   ,WG7   ,FAR7  ,ETAA  ,DPAFT ,V55   ,V25   ,
     6PS4   ,V6    ,AM6   ,TS7   ,PS7   ,V7    ,AM7   ,AM25  ,
     7TS8   ,PS8   ,V8    ,AM8   ,TS9   ,PS9   ,V9    ,AM9   ,
     8CA    ,FRD   ,VJD   ,FGMD  ,VJM   ,FGMM  ,FGPD  ,FGPM  ,
     9FGM   ,FGP   ,WFT   ,WGT   ,FART  ,FG    ,FN    ,SFC
      DATA AWORD/8H  PERF/
      WORD=AWORD
      G=32.174049
      CAPSF=2116.2170
      WFT=WFD+WFD+WFA
      WAT=WAC-BLOB
      WGT=WAT+WFT
      FART=WFT/WAT
      VA=AM*CS
      FRD=VA*WAF/G
      VJM=CVKNOZ*V9
      FGMM=CVDNOZ*WG7/G
      FGPM=CAPSF*(PS9-P1)*A9
      IF(IGASMX.GT.0) GO TO 1
      VJD=CVDNOZ*V29
      FGMD=VJD*WG24/G
```

```
      FGPD=CAPSF*(PS29-P1)*A29
1     FGN=FGMM+FGMD
      FGP=FGPM+FGPD
      FG=FGM+FGP
      FN=FG-FRD
      SFC=3600.*WFT/FN
      FG=DELFG*FG
      FN=DELFN*FN
      SFC=DELSFC*SFC
      CALL OUTPUT
      CALL ERROR
      RETURN
      END
```

```
$IBFTC OUTPUT  DECK,M94/2,XR7
      SUBROUTINE OUTPUT
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,KODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASMX,IDBURN,IAFTBN,IDCD  ,IMCD  ,IDSHOC,IMSHOC,NOZFLT,
     3ITRYS ,LOOPER,NOMAP ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,T4GU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCODS,ETABDS,WA3CDS,DPCODS,DTCOCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8T7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     APS55  ,AM55  ,CVDNOZ,CVMNOZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON / FRONT/
     1T1    ,P1    ,H1    ,S1    ,T2    ,P2    ,H2    ,S2    ,
     2T21   ,P21   ,H21   ,S21   ,T3    ,P3    ,H3    ,S3    ,
     3T4    ,P4    ,H4    ,S4    ,T5    ,P5    ,H5    ,S5    ,
     4T55   ,P55   ,H55   ,S55   ,BLF   ,BLC   ,BLDU  ,BLDB  ,
     5CNF   ,PRF   ,ETAF  ,WAFC  ,WAF   ,WA3   ,WG4   ,FAR4  ,
     6CNC   ,PRC   ,ETAC  ,WACC  ,WAC   ,ETAB  ,DPCOM ,DUMF  ,
     7CNHP  ,ETATHP,DHTCHP,DHTC  ,BLHP  ,WG5   ,FAR5  ,CS    ,
     8CNLP  ,ETATLP,DHTCLP,DHTF  ,BLLP  ,WG55  ,FAR55 ,HPEXT ,
     9AM    ,ALTP  ,ETAR  ,ZF    ,PCNF  ,ZC    ,PCNC  ,WFB   ,
     ATFFHP ,TFFLP ,PCBLF ,PCBLC ,PCBLDU,PCBLDB,PCBLHP,PCBLLP
      COMMON /  SIDE/
     XXP1   ,XWAF  ,XWAC  ,XBLF  ,XBLDU ,XH3   ,DUMS1 ,DUMS2 ,
     XXT21  ,XP21  ,XH21  ,XS21  ,T23   ,P23   ,H23   ,S23   ,
     3T24   ,P24   ,H24   ,S24   ,T25   ,P25   ,H25   ,S25   ,
     4T28   ,P28   ,H28   ,S28   ,T29   ,P29   ,H29   ,S29   ,
     5WAD   ,WFD   ,WG24  ,FAR24 ,ETAD  ,DPDUC ,BYPASS,DUMS3 ,
     6TS28  ,PS28  ,V28   ,AM28  ,TS29  ,PS29  ,V29   ,AM29
      COMMON /  BACK/
     XXTS5  ,XP55  ,XH55  ,XS55  ,XT25  ,XP25  ,XH25  ,XS25  ,
     XXWFB  ,XWG55 ,XFAR55,XWFD  ,XWG24 ,XFAR24,XXP1  ,DUMB  ,
     3T6    ,P6    ,H6    ,S6    ,T7    ,P7    ,H7    ,S7    ,
     4T8    ,P8    ,H8    ,S8    ,T9    ,P9    ,H9    ,S9    ,
     5WG6   ,WFA   ,WG7   ,FAR7  ,ETAA  ,DPAFT ,V55   ,V25   ,
     6PS6   ,V6    ,AM6   ,TS7   ,PS7   ,V7    ,AM7   ,AM25  ,
     7TS8   ,PS8   ,V8    ,AM8   ,TS9   ,PS9   ,V9    ,AM9   ,
     8VA    ,FRD   ,VJD   ,FGMD  ,VJM   ,FGMM  ,FGPD  ,FGPM  ,
     9FGM   ,FGP   ,WFT   ,WGT   ,FART  ,FG    ,FN    ,SFC
      DIMENSION W(5,4),ANS1(80),ANS2(80),ANS3(48),ANS4(72)
      EQUIVALENCE (ANS1,PCNFGU),(ANS2,T1),(ANS3,XP1),(ANS4,XT55)
      DATA ANORD1,ANORD2/6HOUTPUT,6HCONMGN/
      DATA (W(1,I),I=1,4)/6HSUBSON,6HIC C-D,6H NOZZL,6HE     /
      DATA (W(2,I),I=1,4)/6HSHOCK ,6HINSIDE,6H C-D N,6HOZZLE /
      DATA (W(3,I),I=1,4)/6HSHOCK ,6HOUTSID,6HE C-D ,6HNOZZLE/
      DATA (W(4,I),I=1,4)/6HSUBSON,6HIC CON,6HVERG. ,6HNOZZLE/
      DATA (W(5,I),I=1,4)/6HSONIC ,6HCONVER,6HGENT N,6HOZZLE /
      WORD=ANORD1
      IF(IDES.EQ.1) GO TO 4
      IF(IDBURN.GT.0) GO TO 2
      IF(IAFTBN.GT.0) GO TO 1
      WRITE(6,100)WORD,AM,ALTP,T4,ETAR
      GO TO 3
1     WRITE(6,101)WORD,AM,ALTP,T4,T7,ETAR
      GO TO 3
```

```
2       WRITE(6,102)WORD,AM,ALTP,T4,T24,ETAR
3       CALL CONOUT(2)
4       WRITE(6,104)(W(IMSHOC,I),I=1,4),FG,FN,SFC
        IF(IGASMX.GT.0) GO TO 5
        WRITE(6,105)(W(IDSHOC,I),I=1,4)
5       WRITE(6,106)LOOPER
        WORD=AWORD2
        WRITE(6,107)WORD,ZF,PCNF,ZC,PCNC,T4,MODE
        WRITE(6,108)
        WRITE(6,109)(ANS1(I),I=1,80)
        WRITE(6,108)
        WRITE(6,109)(ANS2(I),I=1,80)
        WRITE(6,108)
        WRITE(6,109)(ANS3(I),I=1,48)
        WRITE(6,108)
        WRITE(6,109)(ANS4(I),I=1,72)
        IF(IDES.EQ.1) GO TO 6
        A8=A8SAV
        A9=A9SAV
        A28=A28SAV
        A29=A29SAV
        IF(IDUMP.NE.2) GO TO 6
        WRITE(6,110)
        CALL SYG(2)
6       CALL ENGBAL
        RETURN
100     FORMAT(1H8,A6,14X7H    AM=,F7.3,6X7H  ALTP=,F7.0,
       16X7H    T4=,F8.2,25X7H  ETAR=,F7.4)
101     FORMAT(1H6,A6,14X7H    AM=,F7.3,6X7H  ALTP=,F7.0,
       16X7H    T4=,F8.2,5X7H    T7=,F8.2,5X7H  ETAR=,F7.4)
102     FORMAT(1H8,A6,14X7H    AM=,F7.3,6X7H  ALTP=,F7.0,
       16X7H    T4=,F8.2,5X7H   T24=,F8.2,5X7H  ETAR=,F7.4)
104     FORMAT(6HOMAIN ,4A6,9X3HFG=,F9.2,18X3HFN=,F9.2,18X4HSFC=,F8.5)
105     FORMAT(6H DUCT ,4A6)
106     FORMAT(16H1CONVERGED AFTER,I4,6H LOOPS,/,1H8)
107     FORMAT(1H ,A6,9X,5E15.6,I4)
108     FORMAT(1H )
109     FORMAT(1H ,8E15.6)
110     FORMAT(1H1)
        END
```

```
$IBFTC CONOUT  DECK,M94/2,XR7
      SUBROUTINE CONOUT(ICON)
      COMMON /   ALL/
     1WORD  ,IDES  ,JDES  ,KDES  ,MODE  ,INIT  ,IDUMP ,IAMTP ,
     2IGASMX,IDBURM,IAFTBN,IDCD  ,IMCD  ,IDSHOC,IMSHOC,NOZFLT,
     3ITRYS ,LGOPER,NOMAP ,NUMMAP,MAPEDG,TOLALL,ERR(6)
      COMMON /DESIGN/
     1PCNFGU,PCNCGU,T4GU  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
     2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
     3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
     4T4DS  ,WFBDS ,DTCODS,ETABDS,WA3CDS,DPCODS,DTCOCF,ETABCF,
     5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
     6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
     7T24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
     8TTDS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
     9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
     APS55  ,AM55  ,CVDNOZ,CVMNOZ,A8SAV ,A9SAV ,A28SAV,A29SAV
      COMMON / FRONT/
     1T1    ,P1    ,H1    ,S1    ,T2    ,P2    ,H2    ,S2    ,
     2T21   ,P21   ,H21   ,S21   ,T3    ,P3    ,H3    ,S3    ,
     3T4    ,P4    ,H4    ,S4    ,T5    ,P5    ,H5    ,S5    ,
     4T55   ,P55   ,H55   ,S55   ,BLF   ,BLC   ,BLDU  ,BLDB  ,
     5CHF   ,PRF   ,ETAF  ,WAFC  ,WAF   ,WA3   ,WG4   ,FAR4  ,
     6CHC   ,PRC   ,ETAC  ,WACC  ,WAC   ,ETAB  ,DPCOM ,DUMF  ,
     7CHHP  ,ETATHP,DHTCHP,DHTC  ,BLHP  ,WG5   ,FAR5  ,CS    ,
     8CHLP  ,ETATLP,DHTCLP,DHTF  ,BLLP  ,WG55  ,FAR55 ,HPEXT ,
     9AM    ,ALTP  ,ETAR  ,ZF    ,PCHF  ,ZC    ,PCHC  ,WFB   ,
     ATFFHP ,TFFLP ,PCBLF ,PCBLC ,PCBLDU,PCBLDB,PCBLHP,PCBLLP
      COMMON /  SIDE/
     XXP1   ,XWAF  ,XWAC  ,XBLF  ,XBLDU ,XH3   ,DUMS1 ,DUMS2 ,
     XXT21  ,XP21  ,XH21  ,XS21  ,T23   ,P23   ,H23   ,S23   ,
     3T24   ,P24   ,H24   ,S24   ,T25   ,P25   ,H25   ,S25   ,
     4T28   ,P28   ,H28   ,S28   ,T29   ,P29   ,H29   ,S29   ,
     5WAD   ,WFD   ,WG24  ,FAR24 ,ETAD  ,DPDUC ,BYPASS,DUMS3 ,
     6TS28  ,PS28  ,V28   ,AM28  ,TS29  ,PS29  ,V29   ,AM29
      COMMON /  BACK/
     XXT55  ,XP55  ,XH55  ,XS55  ,XT25  ,XP25  ,XH25  ,XS25  ,
     XXWFB  ,XWG55 ,XFAR55,XWFD  ,XWG24 ,XFAR24,XXP1  ,DUM8  ,
     3T6    ,P6    ,H6    ,S6    ,T7    ,P7    ,H7    ,S7    ,
     4T8    ,P8    ,H8    ,S8    ,T9    ,P9    ,H9    ,S9    ,
     5WG6   ,WFA   ,WG7   ,FAR7  ,ETAA  ,DPAFT ,V55   ,V25   ,
     6PS6   ,V6    ,AM6   ,TS7   ,PS7   ,V7    ,AM7   ,AM25  ,
     7TS8   ,PS8   ,V8    ,AM8   ,TS9   ,PS9   ,V9    ,AM9   ,
     8VA    ,FRD   ,VJD   ,FGMD  ,VJM   ,FGMM  ,FGPD  ,FGPM  ,
     9FGR   ,FGP   ,WFT   ,WGT   ,FART  ,FG    ,FN    ,SFC
      DIMENSION PARAM(280),WORDY(260),IOUT(103),AOUT(6),WOUT(6)
      EQUIVALENCE (PARAM,PCNFGU)
      DATA (WORDY(I),I=1,98)/
     16HPCNFGU,6HPCNCGU,6HT4GU  ,6HDUMD1 ,6HDUMD2 ,6HDELFG ,6HDELFN ,
     26HDELSFC,6HZFDS  ,6HPCNFDS,6HPRFDS ,6HETAFDS,6HWAFDS ,6HPRFCF ,
     36HETAFCF,6HWAFCF ,6HZCDS  ,6HPCNCDS,6HPRCDS ,6HETACDS,6HWACDS ,
     46HPRCCF ,6HETACCF,6HWACCF ,6HT4DS  ,6HWFBDS ,6HDTCODS,6HETABDS,
     56HWA3CDS,6HDPCODS,6HDTCOCF,6HETABCF,6HTFHPDS,6HCNHPDS,6HETHPDS,
     66HTFHPCF,6HCNHPCF,6HETHPCF,6HDHHPCF,6HT2DS  ,6HTFLPDS,6HCNLPDS,
     76HETLPDS,6HTFLPCF,6HCNLPCF,6HETLPCF,6HDHLPCF,6HT21DS ,6HT24DS ,
     86HWFDDS ,6HDTDUDS,6HETADDS,6HWA23DS,6HDPDUDS,6HDTDUCF,6HETADCF,
     96HT7DS  ,6HWFADS ,6HDTAFDS,6HETAADS,6HWG6CDS,6HDPAFDS,6HDTAFCF,
     16HETAACF,6HAS5   ,6HA25   ,6HA6    ,6HA7    ,6HA8    ,6HA9    ,
     26HA28   ,6HA29   ,6HPS55  ,6HAM55  ,6HCVDNOZ,6HCVMNOZ,6HA8SAV ,
     36HA9SAV ,6HA28SAV,6HA29SAV,6HT1    ,6HP1    ,6HH1    ,6HS1    ,
     46HT2    ,6HP2    ,6HH2    ,6HS2    ,6HT21   ,6HP21   ,6HH21   ,
```

```
     56HS21    ,6HT3     ,6HP3     ,6HH3     ,6HS3     ,6HT4     ,6HP4     /
      DATA (WORDY(I),I=99,189)/
     66HH4     ,6HS4     ,6HT5     ,6HP5     ,6HH5     ,6HS5     ,6HT55    ,
     76HP55    ,6HH55    ,6HS55    ,6HBLF    ,6HBLC    ,6HBLDU   ,6HBLOB   ,
     86HCNF    ,6HPAF    ,6HETAF   ,6HWAFC   ,6HWAF    ,6HWA3    ,6HWGA    ,
     96HFAR4   ,6HCNC    ,6HPRC    ,6HETAC   ,6HWACC   ,6HWAC    ,6HETAB   ,
     16HDHTCOM ,6HDUMF   ,6HCNHP   ,6HETATHP,6HDHTCHP,6HDHTC   ,6HBLHP   ,
     26HWG5    ,6HFAR5   ,6HCS     ,6HCNLP   ,6HETATLP,6HDHTCLP,6HDHTF   ,
     36HBLLP   ,6HWG55   ,6HFAR55  ,6HHPEXT  ,6HAM    ,6HALTP   ,6HETAR   ,
     46HZF     ,6HPCNF   ,6HZC     ,6HPCNC   ,6HWFB    ,6HTFFHP  ,6HTFFLP  ,
     56HPCBLF  ,6HPCBLC  ,6HFCBLDU,6HPCBLOB,6HPCBLHP,6HPCBLLP,6HXP1    ,
     66HXWAF   ,6HXWAC   ,6HXBLF   ,6HXBLDU  ,6HXH3    ,6HDUMS1  ,6HDUMS2  ,
     76HXT21   ,6HXP21   ,6HXH21   ,6HXS21   ,6HT23    ,6HP23    ,6HH23    ,
     86HS23    ,6HT24    ,6HP24    ,6HH24    ,6HS24    ,6HT25    ,6HP25    ,
     96HH25    ,6HS25    ,6HT28    ,6HP28    ,6HH28    ,6HS28    ,6HT29    /
      DATA (WORDY(I),I=190,280)/
     16HP29    ,6HH29    ,6HS29    ,6HWAD    ,6HWFD    ,6HWG24   ,6HFAR24  ,
     26HETAD   ,6HDPDUM  ,6HBYPASS,6HDUMS3  ,6HTS28   ,6HPS28   ,6HV28    ,
     36HAM28   ,6HTS29   ,6HPS29   ,6HV29    ,6HAM29   ,6HXT55   ,6HXP55   ,
     46HXH55   ,6HXS55   ,6HXT25   ,6HXP25   ,6HXH25   ,6HXS25   ,6HXWFB   ,
     56HXWG55  ,6HXFAR55,6HXWFD   ,6HXWG24  ,6HXFAR24,6HXXP1   ,6HDUMB   ,
     66HT6     ,6HP6     ,6HH6     ,6HS6     ,6HT7     ,6HP7     ,6HH7     ,
     76HS7     ,6HT8     ,6HP8     ,6HH8     ,6HS8     ,6HT9     ,6HP9     ,
     86HH9     ,6HS9     ,6HWG6    ,6HWFA    ,6HWG7    ,6HFAR7   ,6HETAA   ,
     96HDPAFT  ,6HV55    ,6HV25    ,6HPS6    ,6HV6     ,6HAM6    ,6HTS7    ,
     16HPS7    ,6HV7     ,6HAM7    ,6HAM25   ,6HTS8    ,6HPS8    ,6HV8     ,
     26HAM8    ,6HTS9    ,6HPS9    ,6HV9     ,6HAM9    ,6HVA     ,6HFRD    ,
     36HVJD    ,6HFGMD   ,6HVJM    ,6HFGMM   ,6HFGPD   ,6HFGPM   ,6HFGM    ,
     46HFGP    ,6HWFT    ,6HWGT    ,6HFART   ,6HFG     ,6HFN     ,6HSFC    /
      DATA THEEND,BLANK,LIMIT/6HTHEEND,6H      ,280/
      GO TO (1,12),ICON
C *** INPUT SECTION
1     DO 4 N=1,102
      NUM=N
      READ(5,100)AIN,CHANGE
      IF(AIN.EQ.THEEND) GO TO 5
      DO 2 J=1,LIMIT
      JJ=J
      IF(AIN.EQ.WORDY(J)) GO TO 3
2     CONTINUE
      WRITE(6,101)AIN
      GO TO 4
3     IOUT(NUM)=JJ
      IF(CHANGE.NE.BLANK) WORDY(JJ)=CHANGE
4     CONTINUE
      WRITE(6,102)
5     NUM=NUM-1
      RETURN
C *** OUTPUT SECTION
12    IF(NUM.EQ.1) GO TO 16
      N=NUM
      J=6
      DO 15 I=1,NUM,6
      IF(N.GT.6) GO TO 13
      J=N
13    N=N-6
      DO 14 K=1,J
      L=I+K-1
      M=IOUT(L)
      WOUT(K)=WORDY(M)
14    AOUT(K)=PARAM(M)
```

```
      WRITE(6,103)(NOUT(K),K=1,J)
      WRITE(6,104)(ROUT(K),K=1,J)
      IF(N.LE.0) GO TO 16
15    CONTINUE
16    RETURN
100   FORMAT(A6,6X,A6)
101   FORMAT(1040THE WORD ,A6,26H NOT FOUND IN COMMON ARRAY)
102   FORMAT(/30HOERROR IN CONOUT INPUT)
103   FORMAT(1H0,25XA6,5(9XA6))
104   FORMAT(1H ,20X6E15.6)
      END
```

```
[illegible]
[illegible]
[illegible]
[illegible] ,IDES  ,IDES  ,NODE  ,IRT  ,IDUMP ,IAMTP ,
[illegible],IBEURN,IFTBN,IUCD  ,IRCD  ,IDSHOC,IFSHOC,NOZFLT,
[illegible] ,LOOPER,NOMAP ,NUMMAP,PAPEDG,TOLALL,ERR(6)
 COMMON /DESIGN/
1PCNFGU,PCNCGU,T4GM  ,DUMD1 ,DUMD2 ,DELFG ,DELFN ,DELSFC,
2ZFDS  ,PCNFDS,PRFDS ,ETAFDS,WAFDS ,PRFCF ,ETAFCF,WAFCF ,
3ZCDS  ,PCNCDS,PRCDS ,ETACDS,WACDS ,PRCCF ,ETACCF,WACCF ,
4T4DS  ,WFBDS ,DTCODS,ETABDS,WA3CDS,DPCODS,DTCOCF,ETABCF,
5TFHPDS,CNHPDS,ETHPDS,TFHPCF,CNHPCF,ETHPCF,DHHPCF,T2DS  ,
6TFLPDS,CNLPDS,ETLPDS,TFLPCF,CNLPCF,ETLPCF,DHLPCF,T21DS ,
7T24DS ,WFDDS ,DTDUDS,ETADDS,WA23DS,DPDUDS,DTDUCF,ETADCF,
8T7DS  ,WFADS ,DTAFDS,ETAADS,WG6CDS,DPAFDS,DTAFCF,ETAACF,
9A55   ,A25   ,A6    ,A7    ,A8    ,A9    ,A28   ,A29   ,
APS55  ,AH55  ,CVDNOZ,CVHNOZ,A8SAV ,A9SAV ,A28SAV,A29SAV
 COMMON / FRONT/
1T1    ,P1    ,H1    ,S1    ,T2    ,P2    ,H2    ,S2    ,
2T21   ,P21   ,H21   ,S21   ,T3    ,P3    ,H3    ,S3    ,
3T4    ,P4    ,H4    ,S4    ,T5    ,P5    ,H5    ,S5    ,
4T55   ,P55   ,H55   ,S55   ,BLF   ,BLC   ,BLDU  ,BLOB  ,
5CNF   ,PRF   ,ETAF  ,WAFC  ,WAF   ,WA3   ,WG4   ,FAR4  ,
6CNC   ,PRC   ,ETAC  ,WACC  ,WAC   ,ETAB  ,DPCOM ,DUMF  ,
7CNHP  ,ETATHP,DHTCHP,DHTC  ,BLHP  ,WG5   ,FAR5  ,CS    ,
8CNLP  ,ETATLP,DHTCLP,DHTF  ,BLLP  ,WG55  ,FAR55 ,HPEXT ,
9AM    ,ALTP  ,ETAR  ,ZF    ,PCNF  ,ZC    ,PCNC  ,WFB   ,
ATFFHP ,TFFLP ,PCBLF ,PCBLC ,PCBLDU,PCBLOB,PCBLHP,PCBLLP
 COMMON /  SIDE/
XXP1   ,XWAF  ,XWAC  ,XBLF  ,XBLDU ,XH3   ,DUMS1 ,DUMS2 ,
XXT21  ,XP21  ,XH21  ,XS21  ,T23   ,P23   ,H23   ,S23   ,
3T24   ,P24   ,H24   ,S24   ,T25   ,P25   ,H25   ,S25   ,
4T28   ,P28   ,H28   ,S28   ,T29   ,P29   ,H29   ,S29   ,
5WAD   ,WFD   ,WG24  ,FAR24 ,ETAD  ,DPDUC ,BYPASS,DUMS3 ,
6TS28  ,PS28  ,V28   ,AM28  ,TS29  ,PS29  ,V29   ,AM29
 COMMON /  BACK/
XXT55  ,XP55  ,XH55  ,XS55  ,XT25  ,XP25  ,XH25  ,XS25  ,
XXWFB  ,XWG55 ,XFAR55,XWFD  ,XWG24 ,XFAR24,XXP1  ,DUM8  ,
3T6    ,P6    ,H6    ,S6    ,T7    ,P7    ,H7    ,S7    ,
4T8    ,P8    ,H8    ,S8    ,T9    ,P9    ,H9    ,S9    ,
5WG6   ,WFA   ,WG7   ,FAR7  ,ETAA  ,DPAFT ,V55   ,V25   ,
6PS6   ,V6    ,AM6   ,TS7   ,PS7   ,V7    ,AM7   ,AM25  ,
7TS8   ,PS8   ,V8    ,AM8   ,TS9   ,PS9   ,V9    ,AM9   ,
8VA    ,FRD   ,VJD   ,FGMD  ,VJM   ,FGMM  ,FGPD  ,FGPM  ,
9FGM   ,FGP   ,WFT   ,WGT   ,FART  ,FG    ,FN    ,SFC
 DIMENSION TRASH1(80),TRASH2(80),TRASH3(48),TRASH4(72)
 EQUIVALENCE (TRASH1,PCNFGU),(TRASH2,T1),(TRASH3,XP1),(TRASH4,XT55)
 DATA AWORD/6HCOMMON/
 WRITE(6,100)WORD
 WORD=AWORD
 WRITE(6,102)WORD,ZF,PCNF,ZC,PCNC,T4,MODE
 WRITE(6,103)
 WRITE(6,104)(TRASH1(I),I=1,80)
 WRITE(6,105)
 WRITE(6,104)(TRASH2(I),I=1,80)
 WRITE(6,103)
 WRITE(6,104)(TRASH3(I),I=1,48)
 WRITE(6,103)
 WRITE(6,104)(TRASH4(I),I=1,72)
 WRITE(6,103)
 WRITE(6,106)LOOPER
```

```
      IF(IDUMP.EQ.0) GO TO 2
      WRITE(6,105)
      CALL SYG(2)
2     CALL ENGBAL
      RETURN
100   FORMAT(28H0AN ERROR HAS BEEN FOUND IN ,A6)
102   FORMAT(1H0,A6,9X,5E15.6,I4)
103   FORMAT(2H0 )
104   FORMAT(1H0,8E15.6)
105   FORMAT(1H1)
106   FORMAT(25H0FAILED TO CONVERGE AFTER,I4,6H LOOPS)
      END
```

```
$IBFTC SYG      DECK,K94/2,XR7
      SUBROUTINE SYG(ICON)
      DIMENSION WORD(132)
      DATA ONEDOL/6H$     /
      GO TO (1,2),ICON
1     END FILE 8
      REWIND 8
      RETURN

C        TERMINATE THE FILE

2     WRITE(8,500)
500   FORMAT(12H$$$$$$$$$$$$)
      END FILE 8
      REWIND 8

C        READ RECORD

5     READ(8,501)(WORD(I),I=1,132)
501   FORMAT(132A1)

C        CHECK FOR 12 LEADING DOLLAR SIGNS

      DO 10 I=1,12
      IF(WORD(I)-ONEDOL)11,10,11
10    CONTINUE
      RETURN

C        CHECK FOR 6 TRAILING DOLLAR SIGNS
11    DO 15 I=1,132
      I=I

      IF(WORD(I)-ONEDOL)15,12,15
12    K=I+5
      DO 13 J=I,K
      IF(WORD(J)-ONEDOL)15,13,15
13    CONTINUE
      GO TO 20
15    CONTINUE
      WRITE(6,502)
502   FORMAT(1H0,12HERROR IN SYG)
      RETURN

C        PRINT LINE

20    I=I-1
      WRITE(6,501)(WORD(M),M=1,I)
      GO TO 5
      END
```

```
$IBMAP TAPES    DECK
       ENTRY    .UNOB.
.UNOB. PZE      UNITOB
UNITOB FILE     ,C(1),BCD,BLK=22,READY,INOUT
       END
```

```
$IBFTC THCOMP  DECK,H94/2,XR7
      SUBROUTINE THCOMP(PR,ETA,T,H,S,P,TO,HO,SO,PO)
      PO=P*PR
      TP=T*PR**0.28572
      DO 1 I=1,25
      CALL THERMO(PO,HP,TP,SP,X1,0,X2,0)
      DELS=SP-S
      IF(ABS(DELS).LE.0.00005*S) GO TO 2
1     TP=TP/EXP(4.*DELS)
      CALL ERROR
2     HO=H+((HP-H)/ETA)
      CALL THERMO(PO,HO,TO,SO,X1,0,X2,1)
      RETURN
      END
```

```
$IBFTC THTURB  DECK,M94/2,XR7
      SUBROUTINE THTURB(DH,ETA,FAR,H,S,P,TO,HO,SO,PO)
      HO=H-DH
      HOP=H-DH/ETA
      PT=P/2.
      DO 1 I=1,25
      CALL THERMJ(PT,HOP,TT,ST,AMWT,1,FAR,1)
      DELS=ST-S
      IF(ABS(DELS).LE.0.00005*S) GO TO 2
1     PT=P*EXP(DELS*AMWT/1.986375+ALOG(PT/P))
      CALL ERROR
2     PO=PT
      CALL THERMO(PO,HO,TO,SO,X1,1,FAR,1)
      RETURN
      END
```

```
$IBFTC THERMO  DECK,M94/2,XR7
      SUBROUTINE THERMO(PX,HX,TX,SX,AMX,L,FAR,K)
      FX=0.
      IF(L.EQ.1) FX=FAR
      IF(K.EQ.1) GO TO 1
      CALL PROCOM(FX,TX,CS,AK,CP,R,PHI,HX)
      GO TO 3
1     TX=4.*HX
      DO 2 I=1,15
      CALL PROCOM(FX,TX,CS,AK,CP,R,PHI,H)
      DELH=HX-H
      IF(ABS(DELH).LE.0.00001*HX) GO TO 3
2     TX=TX+4.*DELH
      WRITE(8,100)
100   FORMAT(31HONO CONVERGENCE IN THERMO$$$$$$)
3     SX=PHI-R*ALOG(PX)
      AMX=1.986375/R
      RETURN
      END
```

```
$IBFTC PROCOM  DECK,M94/2,XR7
      SUBROUTINE PROCOM(FARX,TEX,CSEX,AKEX,CPEX,REX,PHI,HEX)
      IF(FARX.LE.0.067623) GO TO 1
      FARX=0.067623
      WRITE(8,101)
1     IF(TEX.GE.300.) GO TO 2
      TEX=300.
      WRITE(8,102)
2     IF(TEX.LE.4000.) GO TO 3
      TEX=4000.
      WRITE(8,103)
3     IF(FARX.GE.0.0) GO TO 4
      FARX=0.0
      WRITE(8,104)
C     AIR PATH
4     CPA =(((((1.0115540E-25*TEX-1.4526770E-21)*TEX
     1+7.6215767E-18)*TEX-1.5128259E-14)*TEX-6.7178376E-12)
     2*TEX+6.5519486E-08)*TEX-5.1536879E-05)*TEX+2.5020051E-01
      HEA=(((((((1.2644425E-26*TEX-2.0752522E-22)*TEX
     1+1.2702630E-18)*TEX-3.0256518E-15)*TEX-3.6794594E-12)*TEX
     2+2.1839826E-08)*TEX-2.5768440E-05)*TEX+2.5020051E-01)*TEX
     3-1.7558886E+00
      SEA=+2.5020051E-01*ALOG(TEX)+((((((1.4450767E-26*TEX
     1-2.4211288E-22)*TEX+1.5243153E-18)*TEX-3.7850648E-15)*TEX
     2-2.2392790E-12)*TEX+3.2759743E-08)*TEX-5.1576879E-05)*TEX
     3+4.5432390E-02
      IF(FARX.LE.0.0) GO TO 5
C     FUEL/AIR PATH
      CPF =(((((7.2678710E-25*TEX-1.3335668E-20)*TEX
     1+1.0212913E-16)*TEX-4.2051104E-13)*TEX+9.9686793E-10)*TEX
     2-1.3771901E-06)*TEX+1.2258630E-03)*TEX+7.3816638E-02
      HEF=(((((((9.0848388E-26*TEX-1.9050949E-21)*TEX
     1+1.7021525E-17)*TEX-8.4102208E-14)*TEX+2.4921698E-10)*TEX
     2-4.5906332E-07)*TEX+6.1293150E-04)*TEX+7.3816638E-02)
     3*TEX+3.0581530E+01
      SEF=+7.3816638E-02*ALOG(TEX)+(((((1.0382670E-25*TEX
     1-2.2226118E-21)*TEX+2.0425826E-17)*TEX-1.0512776E-13)*TEX
     2+3.3228928E-10)*TEX-6.3859505E-07)*TEX+1.2258630E-03)*TEX
     3+6.485398E-01
5     CPEX=(CPA+FARX*CPF)/(1.+FARX)
      HEX=(HEA+FARX*HEF)/(1.+FARX)
      PHI=(SEA+FARX*SEF)/(1.+FARX)
      AMW=28.97-.946186*FARX
      REX=1.986375/AMW
      AKEX=CPEX/(CPEX-REX)
      CSEX=SQRT(AKEX*REX*TEX*25031.37)
      RETURN
101   FORMAT(1H0,63HINPUT FUEL-AIR RATIO ABOVE LIMITS, IT HAS BEEN RESET
     2TO 0.067623,6H$$$$$$)
102   FORMAT(1H0,35HPROCOM INPUT TEMPERATURE BELOW 300.,6H$$$$$$)
103   FORMAT(1H0,36HPROCOM INPUT TEMPERATURE ABOVE 4000.,6H$$$$$$)
104   FORMAT(1H0,38HPROCOM INPUT FUEL-AIR RATIO BELOW ZERO,6H$$$$$$)
      END
```

```
$IBFTC SEARCH  DECK,MAP/2,XR7
      SUBROUTINE SEARCH(P,A,B,C,D,AX,NA,BX,CX,DX,NO,NAM,NOM,NCODE)
      DIMENSION AX(NAM),BX(NAM,NOM),CX(NAM,NOM),DX(NAM,NOM),NO(NAM),Q(3)
C *** NEEDS SUBROUTINE ARQUIR
C *** AX AND BX MUST BE STORED LO TO HI
C *** P=INPUT PROPORTION BETWEEN 0.0 AND 1.0
C     IF NOT INPUT, P MUST EQUAL -1.
C *** NCODE=00   OK
C     NCODE=01   A LO
C     NCODE=02   A HI
C     NCODE=07   ERROR
C     NCODE=10   B LO
C     NCODE=20   B HI
      NCODE=0
      C=0.
      D=0.
C *** FIND A
      DO 1 I=1,NA
      IH=I
      IF(A.LT.AX(1)) GO TO 2
1     CONTINUE
      IF(A.GT.AX(IH)) NCODE=2
      A=AX(IH)
      GO TO 3
2     IF(IH.GT.1) GO TO 3
      NCODE=1
      IH=2
      A=AX(1)
3     IL=IH-1
      LIMH=NO(IH)
      LIML=NO(IL)
C *** FIND B
      PRM=(A-AX(IL))/(AX(IH)-AX(IL))
      PP=P
      IF(P.GE.0.) GO TO 6
      BL=BX(IL,1)+PRM*(BX(IH,1)-BX(IL,1))
      BH=BX(IL,LIML)+PRM*(BX(IH,LIMH)-BX(IL,LIML))
      IF(B.GE.BL) GO TO 4
      NCODE=NCODE+10
      B=BL
      GO TO 5
4     IF(B.LE.BH) GO TO 5
      NCODE=NCODE+20
      B=BH
5     PP=0.5
      Q(2)=0.
      Q(3)=0.
6     BH=PP*(BX(IH,LIMH)-BX(IH,1))+BX(IH,1)
      BL=PP*(BX(IL,LIML)-BX(IL,1))+BX(IL,1)
      DO 7 J=2,LIMH
      JH=J
      IF(BH.LT.BX(IH,J)) GO TO 8
7     CONTINUE
8     JL=JH-1
      DO 9 K=2,LIML
      KH=K
      IF(BL.LT.BX(IL,K)) GO TO 10
9     CONTINUE
10    KL=KH-1

      PR=(BX(IH,JL)-BH)/(BX(IH,JH)-BX(IH,JL))
```

```
      CH= CX(IH,JL)-PR *(CX(IH,JH)-CX(IH,JL))
      DH= DX(IH,JL)-PR *(DX(IH,JH)-DX(IH,JL))

      PR=(BX(IL,KL)-BL)/(BX(IL,KH)-BX(IL,KL))
      CL= CX(IL,KL)-PR *(CX(IL,KH)-CX(IL,KL))
      DL= DX(IL,KL)-PR *(DX(IL,KH)-DX(IL,KL))

      BT=BL+PRM*(BH-BL)
      CT=CL+PRM*(CH-CL)
      DT=DL+PRM*(DH-DL)

      IF(P.GE.0.) GO TO 13
      DIR=SQRT(B/BT)
      ERR=(B-BT)/B
      CALL AFQUER(Q(1),PP,ERR,0.,25.,0.001,DIR,PT,ICON)
      GO TO (11,13,12),ICON
11    PP=PT
      IF(PP.LT.0.) PP=0.
      IF(PP.GT.1.) PP=1.
      GO TO 6
12    NCODE=7
13    B=BT
      C=CT
      D=DT
      RETURN
      END
```

```
$IBFTC MAPBAC  DECK,M94/2,XR7
      SUBROUTINE MAPBAC(MAP,MAPGO,TFFS,TFF,CNS,CN,PCN,T,MODE,IGO,NUM)
      DATA WH,WL,WT,WS/6H H.P. ,6H L.P. ,6H TFF  ,6HSPEED /
1     FORMAT(1H0,A6,12HTURBINE MAP ,A6,4HWAS=,E13.6,10H  AND NOW=,E13.6,
     16H$$$$$$)
2     FORMAT(1H0,A6,A6,22HWAS ALSO CHANGED FROM ,E13.6,5H  TO ,E13.6,
     16H$$$$$$)
      IF(NUM.GT.0) GO TO 3
      NUMH=0
      NUML=0
3     IGO=MAPGO+3*(MAP-1)
      GO TO (100,200,300,400,500,600),IGO
C *** HIGH PRESSURE TURBINE
100   TFF=TFF+0.1*(TFF-TFFS)
      WRITE(8,1)WH,WT,TFFS,TFF
      RETURN
200   CN=CN+0.05*(CN-CNS)
      IF(MODE.NE.1) PCN=PCN*(CN/CNS)
      IF(MODE.EQ.1) T  =T  *(CNS/CN)**2
      WRITE(8,1)WH,WS,CNS,CN
      IF(NUMH.GT.2) GO TO 210
      NUM=1
      NUMH=NUMH+1
      RETURN
210   DELCN=CN-CNS
      IF(DELCN.GE.0.) RETURN
      TFF=TFF*(1.+DELCN/CN)
      WRITE(8,2)WH,WT,TFFS,TFF
      RETURN
300   TFF=TFF+0.1*(TFF-TFFS)
      WRITE(8,1)WH,WT,TFFS,TFF
      GO TO 200
C *** LOW PRESSURE TURBINE
400   TFF=TFF+0.1*(TFF-TFFS)
      WRITE(8,1)WL,WT,TFFS,TFF
      RETURN
500   CN=CN+0.05*(CN-CNS)
      PCN=PCN*(CN/CNS)
      WRITE(8,1)WL,WS,CNS,CN
      IF(NUML.GT.2) GO TO 510
      NUM=1
      NUML=NUML+1
      RETURN
510   DELCN=CN-CNS
      IF(DELCN.GE.0.) RETURN
      TFF=TFF*(1.+DELCN/CN)
      WRITE(8,2)WL,WT,TFFS,TFF
      RETURN
600   TFF=TFF+0.1*(TFF-TFFS)
      WRITE(8,1)WL,WT,TFFS,TFF
      GO TO 400
      END
```

```
$IBFTC CONVRG  DECK,M94/2,XR7
      SUBROUTINE CONVRG(TI,HI,PI,SI,FAR,WG,PA,IDES,AO,PR,
     1TO,HO,PO,SO,TSO,PSO,VO,AMO,ICON)
C     ICON=1      SUBSONIC, COMPARE PI WITH PR
C     ICON=2      SONIC, COMPARE PI WITH PR
C     ICON=4      ERROR
      AJ=778.26
      CAPSF=2116.217
      G=32.174049
      CALL PROCOM(FAR,TI,XX1,XX2,XX3,XX4,PHII,XX6)

C *** SONIC CALCULATIONS

      J=0
      TSS=0.833*TI
1     J=J+1
      CALL PROCOM(FAR,TSS,CSS,AKS,CP,REXS,PHISS,HSS)
      HSCAL=HI-CSS**2/(2.*G*AJ)
      DELHS=HSCAL-HSS
      IF(ABS(DELHS)-0.0005*HSCAL)4,4,2
2     TSS=TSS+DELHS/CP
      IF(J-15)1,1,3
3     ICON=4
      RETURN
4     IF(IDES)12,12,5

C *** ISENTROPIC EXPANSION CALCULATIONS

5     J=0
      TSI=TI*(PA/PI)**0.286
6     J=J+1
      CALL THERMO(PA,HSI,TSI,SSI,XX1,1,FAR,0)
      IF(ABS(SSI-SI)-0.0001*SI)8,8,7
7     TSI=TSI/EXP(4.*(SSI-SI))
      IF(J-30)6,6,3
8     VIS=SQRT(2.*G*AJ*(HI-HSI))
      IF(VIS-CSS)9,11,11

C *** SUBSONIC DESIGN, CALCULATE AO

9     VO=VIS
      TSO=TSI
      PSO=PA
      CALL PROCOM(FAR,TSO,CSO,XX2,XX3,REX,PHISO,HSO)
      RHO=CAPSF*PSO/(AJ*REX*TSO)
      AO=WG/(RHO*VO)
      AMO=VO/CSO
      PR=PI
      ICON=1
10    TO=TI
      HO=HI
      PO=PI
      SO=SI
      RETURN

C *** SONIC DESIGN, CALCULATE AO

11    VO=CSS
      TSO=TSS
      PSO=PI*(TSO/TI)**(AKS/(AKS-1.))
      RHO=CAPSF*PSO/(AJ*REXS*TSO)
```

```
      AO=WG/(RHO*VO)
      AMO=1.0
      PR=PI
      ICON=2
      GO TO 10

C *** NON-DESIGN, CALCULATE CRITICAL CONDITIONS

12    VO=CSS
      TSO=TSS
      PSO=PA
      RHO=CAPSF*PSO/(AJ*REXS*TSO)
      AOCRIT=WG/(RHO*VO)
      AMO=1.0
      PR=PSO*(TI/TSO)**(AKS/(AKS-1.))
      IF(AO-AOCRIT)13,13,14

C *** NON-DESIGN, CRITICAL AND SUPERCRITICAL CONDITIONS

13    PSO=PSO*AOCRIT/AO
      PR=PR*AOCRIT/AO
      ICON=2
      GO TO 10

C *** NON-DESIGN, SUBSONIC CALCULATIONS

14    PSO=PA
      J=0
      TSO=0.833*TSO
15    J=J+1
      CALL PROCOM(FAR,TSO,CSO,AKO,CP,REX,PHISO,HSO)
      RHO=CAPSF*PSO/(AJ*REX*TSO)
      VO=WG/(RHO*AO)
      HSCAL=HI-VO**2/(2.*G*AJ)
      DELHS=HSCAL-HSO
      IF(ABS(DELHS)-0.0005*HSCAL)17,17,16
16    TSO=TSO+DELHS/CP
      IF(J-15)15,15,3
17    AMO=VO/CSO
      PR=PSO*(TI/TSO)**(AKO/(AKO-1.))
      ICON=1
      GO TO 10
      END
```

```
$IBFTC CONDIV  DECK,M94/2,XR7
      SUBROUTINE CONDIV(TI,HI,PI,SI,FAR,WG,PA,IDES,AT,AO,PIR,
     1TT,HT,PT,ST,TO,HO,PO,SO,TST,TSO,PST,PSO,VT,VO,AMT,AMO,ICON)
C     ICON=1  SUBSONIC, COMPARE PIR WITH PI
C     ICON=2  SONIC, SHOCK INSIDE NOZZLE, COMPARE PIR WITH PI
C     ICON=3  SONIC, SHOCK OUTSIDE NOZZLE, COMPARE PIR WITH PI
C     ICON=4  ERROR
      DIMENSION Q(9)
      Q(2)=0.
      Q(3)=0.
      AJ=778.26
      CAPSF=2116.2170
      G=32.174049
      CALL PROCON(FAR,TI,XX1,XX2,XX3,XX4,PHII,XX6)

C *** SONIC CALCULATIONS

      J=0
      TSS=0.833*TI
1     J=J+1
      CALL PROCON(FAR,TSS,CSS,AK,CP,REXS,PHISS,HSS)
      HSCAL=HI-CSS**2/(2.*G*AJ)
      DELHS=HSCAL-HSS
      IF(ABS(DELHS)-0.0005*HSCAL)4,4,2
2     TSS=TSS+DELHS/CP
      IF(J-15)1,1,3
3     ICON=4
      RETURN
4     IF(IDES)11,11,5

C *** SONIC DESIGN, CALCULATE AT

5     VT=CSS
      TST=TSS
      PST=PI*(TST/TI)**(AK/(AK-1.))
      RHO=CAPSF*PST/(AJ*REXS*TST)
      AT=WG/(RHO*VT)
      AMT=1.0

C *** IDEAL EXPANSION DESIGN, CALCULATE AO

      PSO=PA
      J=0
      TSO=TI*(PSO/PI)**.286
6     J=J+1
      CALL PROCON(FAR,TSO,CSO,AK,CP,REX,PHISO,HSO)
      PHICAL=PHII-REX*ALOG(PI/PSO)
      DELPHI=PHICAL-PHISO
      IF(ABS(DELPHI)-0.0001*PHICAL)8,8,7
7     TSO=TSO*EXP(4.*DELPHI)
      IF(J-15)6,6,3
8     VO=SQRT(2.*G*AJ*(HI-HSO))
      AMO=VO/CSO
      AO=(AT/AMO)*(2.*(1.+(AK-1.)*AMO**2/2.)/(AK+1.))**((AK+1.)/(2.*
     1(AK-1.)))
      PIR=PI
      ICON=3
9     TO=TI
      HO=HI
      PO=PI
      SO=SI
```

```
10      TT=TI
        HT=HI
        PT=PI
        ST=SI
        RETURN

C *** ASSUME SONIC THROAT AND ISENTROPIC EXPANSION TO AO

11      VT=CSS
        AMT=1.0
        TST=TSS
        RHO=WG/(AT*VT)
        PST=RHO*AJ*REXS*TST/CAPSF
        PIR=PST*(TI/TST)**(AK/(AK-1.))
        IF(PST-PA)12,24,24
12      TSO=0.95*TI
        MAM=0
13      CALL PROCOM(FAR,TSO,CSO,AK,CP,REX,PHISO,HSO)
        AMO=SQRT(2.*((TI/TSO)-1.)/(AK-1.))
        AOCAL=(AT/AMO)*(2.*(1.+(AK-1.)*AMO**2/2.)/(AK+1.))**((AK+1.)/
       1(2.*(AK-1.)))
        EA=(AO-AOCAL)/AO
        DIR=SQRT(AO/AOCAL)
        CALL AFQUIR(Q(1),TSO,EA,0.,100.,0.0001,DIR,TSOT,JCON)
        GO TO (14,15,3),JCON
14      TSO=TSOT
        IF(TSO-TI)140,13,141
140     TSC=2.*TI/(AK+1.)
        IF(TSO.GT.TSC) GO TO 142
141     TSO=0.95*TI
        GO TO 13
142     IF(Q(2).LT.30.0.OR.AMO.LT.0.95.OR.MAM.EQ.1) GO TO 13
        TSO=2.*TI/(2.+0.98*(AK-1.))
        MAM=1
        GO TO 13
15      PSO=PIR*(TSO/TI)**(AK/(AK-1.))
        IF(PSO-PA)17,16,24
C *** CRITICAL FLOW, ISENTROPIC EXPANSION TO PA

16      VO=AMO*CSO
        ICON=1
        GO TO 9

C *** SUBSONIC FLOW

17      PSO=PA
        Q(2)=0.
        Q(3)=0.
        J=0
        TSO=0.833*TI
18      J=J+1
        CALL PROCOM(FAR,TSO,CSO,AK,CP,REX,PHISO,HSO)
        RHO=CAPSF*PSO/(AJ*REX*TSO)
        VO=WG/(RHO*AO)
        HSCAL=HI-VO**2/(2.*G*AJ)
        DELHS=HSCAL-HSO
        IF(ABS(DELHS)-0.0005*HSCAL)20,20,19
19      TSO=TSO+DELHS/CP
        IF(J-15)18,18,3
20      AMO=VO/CSO
        PIR=PSO*(TI/TSO)**(AK/(AK-1.))
```

```
      TST=TSO
21    CALL PROCOM(FAR,TST,CST,AK,CP,REX,PHI[illegible]T,HST)
      PST=PIR*(TST/TI)**(AK/(AK-1.))
      RHO=PST*CAPSF/(AJ*REX*TST)
      VT=WG/(RHO*AT)
      HSCAL=HI-VT**2/(2.*G*AJ)
      EH=(HSCAL-HST)/HSCAL
      DIR=1.+(HSCAL-HST)/(CP*TST)
      CALL AFGUIR(Q(1),TST,EH,0.,20.,0.0005,DIR,TSTT,JCON)
      GO TO (22,23,3),JCON
22    TST=TSTT
      GO TO 21
23    AMT=VT/CST
      ICON=1
      GO TO 9

C *** SUPERCRITICAL FLOW, ISENTROPIC EXPANSION TO PA

24    PSO=PA
      J=0
      TSO=TI*(PSO/PIR)**.286
25    J=J+1
      CALL PROCOM(FAR,TSO,CSO,AK,CP,REX,PHISO,HSO)
      PHICAL=PHII-REX*ALOG(PIR/PSO)
      DELPHI=PHICAL-PHISO
      IF(ABS(DELPHI)-0.0001*PHICAL)27,27,26
26    TSO=TSO*EXP(4.0*DELPHI)
      IF(J-15)25,25,3
27    VO=SQRT(2.*G*AJ*(HI-HSO))
      AMO=VO/CSO
      AOID=(AT/AMO)*(2.*(1.+(AK-1.)*AMO**2/2.)/(AK+1.))**((AK+1.)/
     1(2.*(AK-1.)))
      ICON=3
      N=0
      IF(AO-AOID)28,9,29

C *** SUPERCRITICAL FLOW, ISENTROPIC EXPANSION TO AO

28    N=1
29    TSO=0.833*TI
      J=0
30    J=J+1
      CALL PROCOM(FAR,TSO,CSO,AK,CP,REX,PHISO,HSO)
      AMO=SQRT(2.*((TI/TSO)-1.)/(AK-1.))
      AOCAL=(AT/AMO)*(2.*(1.+(AK-1.)*AMO**2/2.)/(AK+1.))**((AK+1.)/
     1(2.*(AK-1.)))
      DELA=AO-AOCAL
      IF(ABS(DELA)-0.0001*AO)32,32,31
31    TSO=TSO*SQRT(AOCAL/AO)
      IF(J-50)30,30,3
32    IF(N)34,34,33

C *** UNDEREXPANDED, SHOCK OUTSIDE NOZZLE

33    PSO=PIR*(TSO/TI)**(AK/(AK-1.))
      VO=AMO*CSO
      GO TO 9

C *** OVEREXPANDED, FIND SHOCK POSITION

34    PSX=PIR*(TSO/TI)**(AK/(AK-1.))
```

```
      PSY=PSX*(2.*AK*AMO**2/(AK+1.)-(AK-1.)/(AK+1.))
      IF(PA-PSY)35,36,36

C *** OVEREXPANDED, SHOCK OUTSIDE NOZZLE

35    PSO=PSX
      VO=AMO*CSO
      GO TO 9

C *** OVEREXPANDED, SHOCK INSIDE NOZZLE

36    PSO=PA
      J=0
      TSO=0.833*TI
37    J=J+1
      CALL PROCOM(FAR,TSO,CSO,AK,CP,REX,PHISO,HSO)
      RHO=CAPSF*PSO/(AJ*REX*TSO)
      VO=WG/(RHO*AO)
      HSCAL=HI-VO**2/(2.*G*AJ)
      DELHS=HSCAL-HSO
      IF(ABS(DELHS)-0.0005*HSCAL)39,39,38
38    TSO=TSO+DELHS/CP
      IF(J-15)37,37,3
39    AMO=VO/CSO
      TO=TI
      HO=HI
      PO=PSO*(TO/TSO)**(AK/(AK-1.))
      SO=PHII-REX*ALOG(PO)
      ICON=2
      GO TO 10
      END
```

```
$IBFTC AFQUIR  DECK,M9472,XR7
      SUBROUTINE AFQUIR(X,AIND,DEPEND,ANS,AJ,TOL,DIR,ANEW,ICON)
      DIMENSION X(9)
C X(1)=NAME OF ARRAY TO USE
C AIND=INDEPENDANT VARIABLE
C DEPEND= DEPENDANT VARIABLE
C ANS=ANSWER UPON WHICH TO CONVERGE
C AJ=MAX NUMBER OF TRYS
C TOL=PERCENT TOLERANCE FOR CONVERGENCE
C DIR=DIRECTION AND PERCENTAGE FOR FIRST GUESS
C ANEW=CALCULATED VALUE OF NEXT TRY AT INDEPENDANT VARIABLE
C ICON=CONTROL  =1 GO THRU LOOP AGAIN
C               =2 YOU HAVE REACHED THE ANSWER
C               =3 COUNTER HAS HIT LIMITS
C X(2)=COUNTER STORAGE
C X(3)=CHOOSES METHOD OF CONVERGENCE
C X(4)=THIRD DEPEND VAR
C X(5)=THIRD IND VAR
C X(6)=SECOND DEPEND VAR
C X(7)=SECOND IND VAR
C X(8)=FIRST DEPEND VAR
C X(9)=FIRST IND VAR
C X(3) MUST BE ZERO UPON FIRST ENTRY TO ROUTINE

      Y=0.
      IF(ANS)1,2,1
1     DEP=DEPEND-ANS
      TOLANS=TOL*ANS
      GO TO 3
2     DEP=DEPEND
      TOLANS=TOL
3     IF(ABS(DEP)-TOLANS)5,5,4
4     IF(X(2)-AJ)8,8,7
5     ANEW=AIND
      X(2)=0.
      ICON=2
      RETURN
6     ANEW=Y
      X(2)=X(2)+1.
      ICON=1
      RETURN
7     ANEW=Y
      X(2)=0.
      ICON=3
      RETURN
8     IF(X(3))9,9,12
C *** FIRST GUESS USING DIR
9     X(3)=1.
      X(8)=DEP
      X(9)=AIND
      IF(AIND)10,11,10
10    Y=DIR*AIND
      GO TO 6
11    Y=DIR
      GO TO 6
12    IF(X(3)-1.)13,13,16
C *** LINEAR GUESS
13    X(3)=2.
      X(6)=DEP
      X(7)=AIND
      IF(X(8)-X(6))14,9,14
```

```
14      IF(X(9)-X(7))15,9,15
15      A=(X(9)-X(7))/(X(8)-X(6))
        Y=X(9)-A*X(8)
        IF(ABS(10.*X(9))-ABS(Y))9,9,6
C *** QUADRATIC GUESS
16      X(4)=DEP
        X(5)=AIND
        IF(X(7)-X(5))18,17,18
17      IF(X(6)-X(4))13,9,13
18      IF(X(6)-X(4))19,13,19
19      IF(X(9)-X(5))23,20,23
20      IF(X(8)-X(4))21,22,21
21      X(9)=X(7)
        X(8)=X(6)
        GO TO 13
22      X(9)=X(7)
        X(8)=X(6)
        X(3)=1.
        IF(X(9))10,11,10
23      IF(X(8)-X(4))24,21,24
24      F=(X(6)-X(4))/(X(7)-X(5))
        A=(X(8)-X(4)-F*(X(9)-X(5)))/((X(9)-X(7))*(X(9)-X(5)))
        B=F-A*(X(5)+X(7))
        C=X(4)+X(5)*(A*X(7)-F)
        IF(A)242,240,242
240     IF(B)241,7,241
241     Y=-C/B
        GO TO 37
242     IF(B)247,243,247
243     IF(C)245,244,245
244     Y=0.
        GO TO 37
245     G=-C/A
        IF(G)7,7,246
246     Y=SQRT(G)
        YY=-SQRT(G)
        GO TO 270
247     IF(C)249,248,249
248     Y=-B/A
        YY=0.
        GO TO 270
249     D=4.*A*C/B**2
        IF(1.-D)13,25,26
25      Y=-B/(2.*A)
        GO TO 37
26      E=SQRT(1.-D)
27      Y=(-B/(2.*A))*(1.+E)
        YY=(-B/(2.*A))*(1.-E)
270     J=4
        DEPMIN=ABS(X(4))
        DO 29 I=6,8,2
        IF(DEPMIN-ABS(X(I)))29,29,28
28      J=I
        DEPMIN=ABS(X(I))
29      CONTINUE
        K=J+1
        IF((X(K)-Y)*(X(K)-YY))32,32,30
30      IF(ABS(X(K)-Y)-ABS(X(K)-YY))37,37,31
31      Y=YY
        GO TO 37
32      IF(J-6)33,34,34
```

```
33      JJ=J+2
        KK=K+2
        GO TO 35
34      JJ=J-2
        KK=K-2
35      SLOPE=(X(KK)-X(K))/(X(JJ)-X(J))
        IF(SLOPE*X(J)*(X(K)-Y))36,36,37
36      Y=YY
37      X(9)=X(7)
        X(8)=X(6)
        X(7)=X(5)
        X(6)=X(4)
        GO TO 6
        END
```

```
$IBFTC FANDAT  DECK,M94/2,XR7
      BLOCK DATA
      COMMON /  FAN/CN(15),PR(15,15),WAC(15,15),ETA(15,15),N,NP(15)
      DATA N,NP/10,6,3*7,5*10,8,5*0/
      DATA CN/0.3,0.4,0.5,0.6,0.7,0.8,0.9,1.0,1.1,1.2,5*0./
      DATA (PR( 1,J),WAC( 1,J),ETA( 1,J),J=1, 6)/
     A 1.000   , 73.0    ,0.7300   , 1.030   , 69.0    ,0.7350   ,
     B 1.070   , 60.0    ,0.7400   , 1.096   , 50.0    ,0.7300   ,
     C 1.112   , 40.0    ,0.7000   , 1.120   , 26.0    ,0.6200   /
      DATA (PR( 2,J),WAC( 2,J),ETA( 2,J),J=1, 7)/
     D 1.000   , 86.0    ,0.7300   , 1.050   , 81.0    ,0.7500   ,
     E 1.100   , 76.0    ,0.7650   , 1.146   , 70.0    ,0.7700   ,
     F 1.188   , 62.9    ,0.7750   , 1.230   , 55.0    ,0.7500   ,
     G 1.250   , 47.0    ,0.7150   /
      DATA (PR( 3,J),WAC( 3,J),ETA( 3,J),J=1, 7)/
     H 1.000   ,100.0    ,0.7250   , 1.064   , 96.8    ,0.7500   ,
     I 1.128   , 93.0    ,0.7750   , 1.200   , 87.5    ,0.8000   ,
     J 1.290   , 78.0    ,0.8150   , 1.330   , 72.0    ,0.8000   ,
     K 1.370   , 64.0    ,0.7500   /
      DATA (PR( 4,J),WAC( 4,J),ETA( 4,J),J=1, 7)/
     L 1.000   ,115.0    ,0.7200   , 1.092   ,112.8    ,0.7500   ,
     M 1.220   ,107.5    ,0.8000   , 1.310   ,102.0    ,0.8250   ,
     N 1.400   , 94.0    ,0.8500   , 1.474   , 83.0    ,0.8000   ,
     O 1.488   , 80.0    ,0.7800   /
      DATA (PR( 5,J),WAC( 5,J),ETA( 5,J),J=1,10)/
     P 1.000   ,132.0    ,0.7000   , 1.160   ,131.0    ,0.7500   ,
     Q 1.296   ,128.5    ,0.8000   , 1.370   ,126.2    ,0.8250   ,
     R 1.460   ,122.0    ,0.8500   , 1.524   ,118.0    ,0.8700   ,
     S 1.544   ,116.5    ,0.8730   , 1.560   ,115.0    ,0.8700   ,
     T 1.610   ,110.5    ,0.8500   , 1.668   ,102.8    ,0.8000   /
      DATA (PR( 6,J),WAC( 6,J),ETA( 6,J),J=1,10)/
     U 1.000   ,150.0    ,0.6650   , 1.250   ,149.9    ,0.7500   ,
     V 1.400   ,148.0    ,0.8000   , 1.500   ,145.6    ,0.8250   ,
     W 1.570   ,143.0    ,0.8500   , 1.638   ,140.0    ,0.8700   ,
     X 1.680   ,137.0    ,0.8800   , 1.716   ,134.5    ,0.8700   ,
     Y 1.756   ,130.0    ,0.8500   , 1.830   ,122.0    ,0.7990   /
      DATA (PR( 7,J),WAC( 7,J),ETA( 7,J),J=1,10)/
     Z 1.000   ,170.0    ,0.6200   , 1.190   ,170.0    ,0.7000   ,
     A 1.380   ,170.0    ,0.7500   , 1.548   ,168.0    ,0.8000   ,
     B 1.650   ,166.0    ,0.8300   , 1.724   ,163.5    ,0.8500   ,
     C 1.828   ,158.6    ,0.8700   , 1.904   ,152.8    ,0.8500   ,
     D 1.978   ,145.0    ,0.8000   , 2.000   ,142.2    ,0.7900   /
      DATA (PR( 8,J),WAC( 8,J),ETA( 8,J),J=1,10)/
     E 1.000   ,190.0    ,0.5800   , 1.110   ,190.0    ,0.6200   ,
     F 1.338   ,190.0    ,0.7000   , 1.582   ,189.9    ,0.7500   ,
     G 1.736   ,187.8    ,0.8000   , 1.850   ,185.0    ,0.8200   ,
     H 2.000   ,180.0    ,0.8500   , 2.070   ,176.0    ,0.8300   ,
     I 2.120   ,173.0    ,0.8000   , 2.200   ,166.0    ,0.7600   /
      DATA (PR( 9,J),WAC( 9,J),ETA( 9,J),J=1,10)/
     J 1.000   ,210.0    ,0.5500   , 1.260   ,210.0    ,0.6200   ,
     K 1.550   ,210.0    ,0.7000   , 1.818   ,210.0    ,0.7500   ,
     L 2.000   ,210.0    ,0.7750   , 2.120   ,209.5    ,0.7800   ,
     M 2.250   ,208.0    ,0.7750   , 2.334   ,205.0    ,0.7500   ,
     N 2.420   ,200.0    ,0.7200   , 2.460   ,197.0    ,0.7000   /
      DATA (PR(10,J),WAC(10,J),ETA(10,J),J=1, 6)/
     O 1.000   ,225.0    ,0.5800   , 1.406   ,225.0    ,0.6200   ,
     P 1.700   ,225.0    ,0.7000   , 2.000   ,225.0    ,0.7500   ,
     Q 2.300   ,225.0    ,0.7350   , 2.350   ,225.0    ,0.7250   ,
     R 2.450   ,224.5    ,0.7000   , 2.650   ,221.0    ,0.6200   /
      END
```

```
$IBFTC CMPDAT  DECK,M94/2,XR7
      BLOCK DATA
      COMMON / COMP/CN(15),PR(15,15),WAC(15,15),ETA(15,15),N,NP(15)
      DATA N,NP/10,2*6,2*8,4*10,2*8,5*0/
      DATA CN/0.5,0.6,0.7,0.8,0.89,0.92,0.95,1.0,1.1,1.15,5*0./
      DATA (PR( 1,J),WAC( 1,J),ETA( 1,J),J=1, 6)/
     A 1.000    , 21.4     ,0.5700    , 1.599    , 21.1     ,0.6000    ,
     B 2.018    , 20.8     ,0.6200    , 2.335    , 20.5     ,0.6300    ,
     C 3.024    , 19.6     ,0.6500    , 3.533    , 18.7     ,0.6200    /
      DATA (PR( 2,J),WAC( 2,J),ETA( 2,J),J=1, 6)/
     D 1.000    , 24.9     ,0.5700    , 1.689    , 24.9     ,0.6200    ,
     E 2.497    , 24.7     ,0.6700    , 3.096    , 24.3     ,0.7000    ,
     F 3.635    , 23.8     ,0.7200    , 4.234    , 23.1     ,0.7000    /
      DATA (PR( 3,J),WAC( 3,J),ETA( 3,J),J=1, 8)/
     G 1.000    , 29.4     ,0.5650    , 1.599    , 29.4     ,0.6200    ,
     H 2.198    , 29.4     ,0.6600    , 2.719    , 29.2     ,0.7000    ,
     I 3.515    , 28.9     ,0.7500    , 4.174    , 28.5     ,0.7650    ,
     J 4.796    , 27.9     ,0.7500    , 4.946    , 27.6     ,0.7430    /
      DATA (PR( 4,J),WAC( 4,J),ETA( 4,J),J=1, 8)/
     K 1.000    , 35.6     ,0.5600    , 1.719    , 35.6     ,0.6200    ,
     L 2.737    , 35.6     ,0.7000    , 3.575    , 35.6     ,0.7500    ,
     M 4.353    , 35.3     ,0.7800    , 5.012    , 35.0     ,0.8000    ,
     N 5.671    , 34.3     ,0.7800    , 5.970    , 33.8     ,0.7650    /
      DATA (PR( 5,J),WAC( 5,J),ETA( 5,J),J=1,10)/
     O 1.000    , 42.7     ,0.5520    , 2.084    , 42.7     ,0.6200    ,
     P 3.132    , 42.7     ,0.7000    , 4.054    , 42.7     ,0.7500    ,
     Q 4.922    , 42.5     ,0.7800    , 5.431    , 42.4     ,0.8100    ,
     R 6.000    , 42.0     ,0.8300    , 6.407    , 41.8     ,0.8100    ,
     S 6.988    , 41.2     ,0.7800    , 7.138    , 40.9     ,0.7780    /
      DATA (PR( 6,J),WAC( 6,J),ETA( 6,J),J=1,10)/
     T 1.000    , 45.4     ,0.5500    , 2.323    , 45.4     ,0.6200    ,
     U 3.353    , 45.4     ,0.7000    , 4.293    , 45.4     ,0.7500    ,
     V 5.240    , 45.2     ,0.7800    , 5.731    , 45.0     ,0.8100    ,
     W 6.389    , 44.8     ,0.8300    , 6.808    , 44.5     ,0.8100    ,
     X 7.377    , 43.9     ,0.7800    , 7.587    , 43.7     ,0.7760    /
      DATA (PR( 7,J),WAC( 7,J),ETA( 7,J),J=1,10)/
     Z 1.000    , 48.1     ,0.5400    , 2.617    , 48.1     ,0.6200    ,
     A 3.683    , 48.1     ,0.7000    , 4.653    , 48.1     ,0.7500    ,
     B 5.611    , 48.1     ,0.7800    , 6.222    , 48.0     ,0.8100    ,
     C 6.868    , 47.7     ,0.8200    , 7.168    , 47.6     ,0.8100    ,
     D 7.796    , 47.3     ,0.7800    , 8.138    , 47.2     ,0.7700    /
      DATA (PR( 8,J),WAC( 8,J),ETA( 8,J),J=1,10)/
     E 1.000    , 51.6     ,0.5200    , 1.491    , 51.6     ,0.5500    ,
     F 3.036    , 51.6     ,0.6200    , 4.192    , 51.6     ,0.7000    ,
     G 5.192    , 51.6     ,0.7500    , 6.263    , 51.6     ,0.7800    ,
     H 6.689    , 51.5     ,0.7950    , 7.347    , 51.3     ,0.8100    ,
     I 8.078    , 51.1     ,0.7800    , 8.725    , 50.7     ,0.7500    /
      DATA (PR( 9,J),WAC( 9,J),ETA( 9,J),J=1, 8)/
     J 1.000    , 58.7     ,0.4600    , 3.395    , 58.7     ,0.5800    ,
     K 5.731    , 58.7     ,0.7000    , 7.587    , 58.7     ,0.7500    ,
     L 8.186    , 58.7     ,0.7550    , 8.784    , 58.6     ,0.7500    ,
     M 9.683    , 58.5     ,0.7000    ,10.042    , 58.4     ,0.6750    /
      DATA (PR(10,J),WAC(10,J),ETA(10,J),J=1, 8)/
     N 1.000    , 61.4     ,0.4500    , 2.976    , 61.4     ,0.5500    ,
     O 4.916    , 61.4     ,0.6200    , 6.838    , 61.4     ,0.7000    ,
     P 7.587    , 61.4     ,0.7250    , 8.485    , 61.4     ,0.7250    ,
     Q 9.365    , 61.4     ,0.7000    ,10.581    , 61.4     ,0.6200    /
      END
```

```
$IBFTC CMBDAT  DECK,M94/2,XR7
      BLOCK DATA
      COMMON / COMB/PSI(15),DELT(15,15),ETA(15,15),N,NP(15)
      DATA N,NP/8,8*11,7*0/
      DATA PSI/10.,20.,30.,40.,50.,60.,[illegible]0.,7*0./
      DATA DELT/15*800.,15*900.,15*975.,[illegible]065.,15*1175.,15*1260.,
     A15*1325.,15*1450.,15*1550.,15*1685.,[illegible]5*1800.,60*0./
      DATA ETA/
     B0.8600,0.8820,0.9020,0.9200,0.9360,0.9500,0.9600,0.9680,7*0.,
     C0.8783,0.8996,0.9183,0.9347,0.9489,0.9615,0.9710,0.9787,7*0.,
     D0.8895,0.9100,0.9282,0.9435,0.9573,0.9694,0.9782,0.9850,7*0.,
     E0.9000,0.9200,0.9372,0.9520,0.9653,0.9769,0.9850,0.9850,7*0.,
     F0.9080,0.9272,0.9444,0.9587,0.9718,0.9830,0.9850,0.9850,7*0.,
     G0.9100,0.9290,0.9460,0.9610,0.9740,0.9850,0.9850,0.9850,7*0.,
     H0.9087,0.9275,0.9448,0.9600,0.9733,0.9840,0.9850,0.9850,7*0.,
     I0.9000,0.9195,0.9376,0.9543,0.9690,0.9800,0.9850,0.9850,7*0.,
     J0.8886,0.9090,0.9284,0.9470,0.9630,0.9750,0.9850,0.9850,7*0.,
     K0.8660,0.8890,0.9110,0.9328,0.9515,0.9672,0.9792,0.9850,7*0.,
     L0.8400,0.8670,0.8930,0.9180,0.9400,0.9600,0.9730,0.9800,7*0.,
     M4*0.  ,4*0.  ,4*0.  ,4*0.  ,4*0.  ,4*0.  ,4*0.  ,4*0.  ,28*0./
      END
```

```
$IBFTC HPTDAT  DECK,M94/2,XR7
      BLOCK DATA
      COMMON / HTURB/TFF(15),CN(15,15),DH(15,15),ETA(15,15),N,NP(15)
      DATA N,NP/10,9*15,12,5*0/
      DATA TFF/ 16.86     ,18.27     ,20.17     ,20.66     ,20.90     ,
     121.08     ,21.25     ,21.43     ,21.64     ,21.92     ,5*0.0     /
      DATA (CN( 1,J),DH( 1,J),ETA( 1,J),J=1,15)/
     A0.191     ,0.0029    ,0.6150    ,0.344     ,0.0051    ,0.7000    ,
     B0.526     ,0.0076    ,0.7780    ,0.727     ,0.0097    ,0.8000    ,
     C0.957     ,0.0120    ,0.8000    ,1.167     ,0.0137    ,0.7875    ,
     D1.340     ,0.0148    ,0.7692    ,1.569     ,0.0157    ,0.7340    ,
     E1.761     ,0.0161    ,0.7000    ,1.971     ,0.0158    ,0.7550    ,
     F2.193     ,0.0150    ,0.6000    ,2.454     ,0.0130    ,0.5250    ,
     G2.641     ,0.0108    ,0.4720    ,2.842     ,0.0074    ,0.4000    ,
     H3.005     ,0.0031    ,0.3000    /
      DATA (CN( 2,J),DH( 2,J),ETA( 2,J),J=1,15)/
     I0.191     ,0.0034    ,0.6000    ,0.402     ,0.0072    ,0.7000    ,
     J0.593     ,0.0102    ,0.8000    ,0.727     ,0.0122    ,0.8200    ,
     K0.861     ,0.0140    ,0.8270    ,1.000     ,0.0158    ,0.8300    ,
     L1.129     ,0.0173    ,0.8275    ,1.301     ,0.0191    ,0.8210    ,
     M1.474     ,0.0205    ,0.8162    ,1.741     ,0.0223    ,0.8000    ,
     N2.009     ,0.0234    ,0.7610    ,2.316     ,0.0235    ,0.7000    ,
     O2.751     ,0.0217    ,0.6000    ,3.158     ,0.0169    ,0.5000    ,
     P3.445     ,0.0115    ,0.4150    /
      DATA (CN( 3,J),DH( 3,J),ETA( 3,J),J=1,15)/
     Q0.191     ,0.0041    ,0.5700    ,0.445     ,0.0090    ,0.7000    ,
     R0.670     ,0.0130    ,0.8000    ,0.890     ,0.0166    ,0.8400    ,
     S1.091     ,0.0194    ,0.8448    ,1.263     ,0.0216    ,0.8420    ,
     T1.493     ,0.0241    ,0.8400    ,1.722     ,0.0263    ,0.8315    ,
     U2.009     ,0.0284    ,0.8170    ,2.258     ,0.0298    ,0.8000    ,
     V2.603     ,0.0310    ,0.7495    ,2.861     ,0.0311    ,0.7000    ,
     W3.100     ,0.0306    ,0.6578    ,3.330     ,0.0292    ,0.6000    ,
     X3.445     ,0.0281    ,0.5800    /
      DATA (CN( 4,J),DH( 4,J),ETA( 4,J),J=1,15)/
     Y0.191     ,0.0047    ,0.5580    ,0.260     ,0.0061    ,0.6000    ,
     Z0.488     ,0.0108    ,0.7000    ,0.708     ,0.0148    ,0.8000    ,
     A0.933     ,0.0184    ,0.8400    ,1.167     ,0.0220    ,0.8500    ,
     B1.416     ,0.0252    ,0.8500    ,1.593     ,0.0274    ,0.8480    ,
     C1.837     ,0.0302    ,0.8440    ,2.019     ,0.0320    ,0.8400    ,
     D2.325     ,0.0349    ,0.8270    ,2.564     ,0.0371    ,0.8170    ,
     E2.890     ,0.0397    ,0.8000    ,3.205     ,0.0425    ,0.7710    ,
     F3.445     ,0.0445    ,0.7500    /
      DATA (CN( 5,J),DH( 5,J),ETA( 5,J),J=1,15)/
     G0.191     ,0.0050    ,0.5500    ,0.306     ,0.0079    ,0.6000    ,
     H0.536     ,0.0130    ,0.7000    ,0.765     ,0.0173    ,0.8000    ,
     I0.993     ,0.0212    ,0.8400    ,1.301     ,0.0259    ,0.8600    ,
     J1.512     ,0.0289    ,0.8599    ,1.799     ,0.0324    ,0.8566    ,
     K2.086     ,0.0360    ,0.8519    ,2.383     ,0.0400    ,0.8460    ,
     L2.698     ,0.0446    ,0.8426    ,2.928     ,0.0486    ,0.8400    ,
     M3.138     ,0.0536    ,0.8400    ,3.215     ,0.0576    ,0.8437    ,
     N3.225     ,0.0595    ,0.8475    /
      DATA (CN( 6,J),DH( 6,J),ETA( 6,J),J=1,15)/
     O0.191     ,0.0061    ,0.5250    ,0.364     ,0.0108    ,0.6000    ,
     P0.632     ,0.0173    ,0.7000    ,0.880     ,0.0227    ,0.8000    ,
     Q1.115     ,0.0270    ,0.8400    ,1.311     ,0.0306    ,0.8600    ,
     R1.531     ,0.0346    ,0.8721    ,1.722     ,0.0379    ,0.8800    ,
     S1.952     ,0.0425    ,0.8840    ,2.167     ,0.0472    ,0.8870    ,
     T2.316     ,0.0508    ,0.8875    ,2.471     ,0.0551    ,0.8876    ,
     U2.545     ,0.0576    ,0.8868    ,2.588     ,0.0601    ,0.8838    ,
     V2.607     ,0.0628    ,0.8798    /
      DATA (CN( 7,J),DH( 7,J),ETA( 7,J),J=1,15)/
```

```
W0.191     ,0.0072    ,0.5005    ,0.440     ,0.0148    ,0.6000    ,
X0.698     ,0.0212    ,0.7000    ,0.976     ,0.0277    ,0.8000    ,
Y1.225     ,0.0335    ,0.8400    ,1.411     ,0.0374    ,0.8600    ,
Z1.541     ,0.0403    ,0.8700    ,1.651     ,0.0428    ,0.8800    ,
A1.780     ,0.0459    ,0.8855    ,1.899     ,0.0490    ,0.8900    ,
B1.995     ,0.0518    ,0.8910    ,2.040     ,0.0540    ,0.8900    ,
C2.086     ,0.0562    ,0.8880    ,2.124     ,0.0594    ,0.8826    ,
D2.143     ,0.0630    ,0.8695    /
 DATA (CN( 8,J),DH( 8,J),ETA( 8,J),J=1,15)/
E0.191     ,0.0079    ,0.4995    ,0.493     ,0.0176    ,0.6000    ,
F0.746     ,0.0245    ,0.7000    ,0.899     ,0.0284    ,0.7580    ,
G1.043     ,0.0320    ,0.8000    ,1.167     ,0.0353    ,0.8200    ,
H1.306     ,0.0389    ,0.8400    ,1.397     ,0.0414    ,0.8500    ,
I1.493     ,0.0439    ,0.8600    ,1.627     ,0.0475    ,0.8710    ,
J1.708     ,0.0504    ,0.8750    ,1.780     ,0.0536    ,0.8750    ,
K1.837     ,0.0576    ,0.8690    ,1.852     ,0.0598    ,0.8600    ,
L1.856     ,0.0624    ,0.8495    /
 DATA (CN( 9,J),DH( 9,J),ETA( 9,J),J=1,15)/
M0.191     ,0.0084    ,0.4855    ,0.344     ,0.0143    ,0.5320    ,
N0.545     ,0.0209    ,0.6000    ,0.689     ,0.0256    ,0.6500    ,
O0.823     ,0.0297    ,0.7000    ,0.938     ,0.0331    ,0.7380    ,
P1.033     ,0.0360    ,0.7690    ,1.148     ,0.0398    ,0.8000    ,
Q1.244     ,0.0432    ,0.8100    ,1.340     ,0.0472    ,0.8210    ,
R1.397     ,0.0500    ,0.8255    ,1.435     ,0.0522    ,0.8270    ,
S1.474     ,0.0551    ,0.8230    ,1.493     ,0.0576    ,0.8150    ,
T1.497     ,0.0601    ,0.8000    /
 DATA (CN(10,J),DH(10,J),ETA(10,J),J=1,12)/
U0.191     ,0.0119    ,0.4210    ,0.287     ,0.0162    ,0.4695    ,
V0.388     ,0.0205    ,0.5000    ,0.478     ,0.0241    ,0.5300    ,
W0.574     ,0.0283    ,0.5620    ,0.651     ,0.0317    ,0.5875    ,
X0.703     ,0.0342    ,0.6000    ,0.751     ,0.0371    ,0.6110    ,
Y0.785     ,0.0396    ,0.6170    ,0.823     ,0.0428    ,0.6240    ,
Z0.842     ,0.0454    ,0.6195    ,0.847     ,0.0477    ,0.6050    /
 END
```

```
$IBFTC LPTDAT  DECK,M94/2,XR7
      BLOCK DATA
      COMMON / LTURB/TFF(15),CN(15,15),DH(15,15),ETA(15,15),N,NP(15)
      DATA N,NP/11,9*15,12,9,4*0/
      DATA TFF/
     131.50    ,36.60    ,41.60    ,46.05    ,50.22    ,51.89    ,
     253.41    ,54.60    ,55.69    ,56.89    ,58.10    ,4*0.0    /
      DATA (CN( 1,J),DH( 1,J),ETA( 1,J),J=1,15)/
     A0.365    ,0.0029   ,0.7120   ,0.529    ,0.0042   ,0.7300   ,
     B0.730    ,0.0056   ,0.7472   ,0.967    ,0.0070   ,0.7300   ,
     C1.204    ,0.0081   ,0.7140   ,1.405    ,0.0089   ,0.7000   ,
     D1.606    ,0.0095   ,0.6850   ,1.752    ,0.0098   ,0.6730   ,
     E2.007    ,0.0100   ,0.6452   ,2.203    ,0.0098   ,0.6200   ,
     F2.445    ,0.0092   ,0.6000   ,2.591    ,0.0085   ,0.5750   ,
     G2.792    ,0.0070   ,0.5310   ,2.920    ,0.0056   ,0.5000   ,
     H3.285    ,0.0001   ,0.3850   /
      DATA (CN( 2,J),DH( 2,J),ETA( 2,J),J=1,15)/
     I0.365    ,0.0042   ,0.8000   ,0.547    ,0.0063   ,0.8100   ,
     J0.785    ,0.0086   ,0.8200   ,1.058    ,0.0111   ,0.8300   ,
     K1.277    ,0.0128   ,0.8300   ,1.432    ,0.0139   ,0.8290   ,
     L1.679    ,0.0153   ,0.8100   ,1.879    ,0.0162   ,0.8000   ,
     M2.044    ,0.0167   ,0.7850   ,2.208    ,0.0171   ,0.7600   ,
     N2.354    ,0.0173   ,0.7450   ,2.600    ,0.0170   ,0.7000   ,
     O2.847    ,0.0162   ,0.6800   ,3.029    ,0.0151   ,0.6450   ,
     P3.285    ,0.0123   ,0.5900   /
      DATA (CN( 3,J),DH( 3,J),ETA( 3,J),J=1,15)/
     Q0.365    ,0.0050   ,0.8000   ,0.586    ,0.0081   ,0.8300   ,
     R0.858    ,0.0114   ,0.8600   ,1.067    ,0.0139   ,0.8630   ,
     S1.241    ,0.0159   ,0.8670   ,1.423    ,0.0178   ,0.8700   ,
     T1.606    ,0.0195   ,0.8720   ,1.825    ,0.0215   ,0.8720   ,
     U2.007    ,0.0229   ,0.8700   ,2.226    ,0.0243   ,0.8670   ,
     V2.409    ,0.0252   ,0.8600   ,2.573    ,0.0259   ,0.8500   ,
     W2.755    ,0.0265   ,0.8300   ,3.029    ,0.0268   ,0.8000   ,
     X3.285    ,0.0263   ,0.7600   /
      DATA (CN( 4,J),DH( 4,J),ETA( 4,J),J=1,15)/
     Y0.365    ,0.0053   ,0.7995   ,0.420    ,0.0061   ,0.8000   ,
     Z0.675    ,0.0098   ,0.8400   ,0.876    ,0.0125   ,0.8600   ,
     A1.095    ,0.0153   ,0.8680   ,1.277    ,0.0176   ,0.8730   ,
     B1.496    ,0.0201   ,0.8800   ,1.733    ,0.0226   ,0.8830   ,
     C2.025    ,0.0254   ,0.8835   ,2.190    ,0.0266   ,0.8830   ,
     D2.354    ,0.0279   ,0.8800   ,2.582    ,0.0293   ,0.8740   ,
     E2.847    ,0.0305   ,0.8300   ,3.102    ,0.0312   ,0.8350   ,
     F3.285    ,0.0315   ,0.8200   /
      DATA (CN( 5,J),DH( 5,J),ETA( 5,J),J=1,15)/
     G0.365    ,0.0057   ,0.7750   ,0.502    ,0.0078   ,0.8000   ,
     H0.730    ,0.0113   ,0.8480   ,0.967    ,0.0148   ,0.8600   ,
     I1.277    ,0.0191   ,0.8750   ,1.551    ,0.0226   ,0.8900   ,
     J1.715    ,0.0248   ,0.8912   ,1.952    ,0.0276   ,0.8940   ,
     K2.080    ,0.0290   ,0.8955   ,2.244    ,0.0307   ,0.8970   ,
     L2.412    ,0.0323   ,0.8961   ,2.646    ,0.0343   ,0.8900   ,
     M2.920    ,0.0362   ,0.8790   ,3.157    ,0.0376   ,0.8671   ,
     N3.285    ,0.0382   ,0.8600   /
      DATA (CN( 6,J),DH( 6,J),ETA( 6,J),J=1,15)/
     O0.365    ,0.0061   ,0.7600   ,0.611    ,0.0103   ,0.8000   ,
     P0.830    ,0.0139   ,0.8450   ,1.095    ,0.0181   ,0.8600   ,
     Q1.277    ,0.0208   ,0.8730   ,1.515    ,0.0243   ,0.8900   ,
     R1.733    ,0.0273   ,0.8950   ,1.934    ,0.0300   ,0.9000   ,
     S2.194    ,0.0334   ,0.9005   ,2.409    ,0.0362   ,0.9010   ,
     T2.628    ,0.0390   ,0.9004   ,2.837    ,0.0415   ,0.9000   ,
     U3.111    ,0.0457   ,0.8900   ,3.230    ,0.0485   ,0.8800   ,
     V3.285    ,0.0510   ,0.8735   /
```

```
 DATA (CN( 7,J),DH( 7,J),ETA( 7,J),J=1,15)/
W0.365     ,0.0070    ,0.7310    ,0.766     ,0.0142    ,0.8000    ,
X1.004     ,0.0184    ,0.8300    ,1.255     ,0.0226    ,0.8600    ,
Y1.460     ,0.0259    ,0.8750    ,1.664     ,0.0290    ,0.8900    ,
Z1.825     ,0.0315    ,0.8930    ,2.007     ,0.0348    ,0.8975    ,
A2.171     ,0.0376    ,0.8999    ,2.280     ,0.0400    ,0.9000    ,
B2.372     ,0.0424    ,0.8980    ,2.482     ,0.0454    ,0.8937    ,
C2.536     ,0.0474    ,0.8900    ,2.591     ,0.0502    ,0.8799    ,
D2.609     ,0.0527    ,0.8710    /
 DATA (CN( 8,J),DH( 8,J),ETA( 8,J),J=1,15)/
E0.365     ,0.0077    ,0.7100    ,0.602     ,0.0125    ,0.7450    ,
F0.785     ,0.0163    ,0.7680    ,0.967     ,0.0198    ,0.8000    ,
G1.223     ,0.0245    ,0.8380    ,1.442     ,0.0284    ,0.8600    ,
H1.624     ,0.0321    ,0.8712    ,1.825     ,0.0362    ,0.8780    ,
I1.934     ,0.0387    ,0.8800    ,2.062     ,0.0418    ,0.8775    ,
J2.135     ,0.0438    ,0.8760    ,2.190     ,0.0456    ,0.8722    ,
K2.244     ,0.0479    ,0.8660    ,2.281     ,0.0502    ,0.8600    ,
L2.285     ,0.0514    ,0.8480    /
 DATA (CN( 9,J),DH( 9,J),ETA( 9,J),J=1,15)/
M0.365     ,0.0066    ,0.6780    ,0.547     ,0.0128    ,0.7000    ,
N0.657     ,0.0153    ,0.7125    ,0.821     ,0.0190    ,0.7350    ,
O1.004     ,0.0226    ,0.7690    ,1.159     ,0.0256    ,0.8000    ,
P1.223     ,0.0270    ,0.8060    ,1.369     ,0.0301    ,0.8225    ,
Q1.515     ,0.0334    ,0.8395    ,1.606     ,0.0357    ,0.8450    ,
R1.733     ,0.0390    ,0.8470    ,1.825     ,0.0421    ,0.8445    ,
S1.879     ,0.0446    ,0.8330    ,1.898     ,0.0463    ,0.8235    ,
T1.907     ,0.0485    ,0.8080    /
 DATA (CN(10,J),DH(10,J),ETA(10,J),J=1,12)/
U0.365     ,0.0098    ,0.6380    ,0.474     ,0.0125    ,0.6550    ,
V0.639     ,0.0167    ,0.6700    ,0.748     ,0.0195    ,0.6850    ,
W0.856     ,0.0223    ,0.7000    ,0.967     ,0.0251    ,0.7110    ,
X1.095     ,0.0290    ,0.7180    ,1.191     ,0.0321    ,0.7180    ,
Y1.259     ,0.0348    ,0.7170    ,1.296     ,0.0368    ,0.7140    ,
Z1.325     ,0.0390    ,0.7000    ,1.329     ,0.0401    ,0.6890    /
 DATA (CN(11,J),DH(11,J),ETA(11,J),J=1, 9)/
A0.365     ,0.0111    ,0.6000    ,0.438     ,0.0137    ,0.6000    ,
B0.547     ,0.0169    ,0.6120    ,0.639     ,0.0197    ,0.6170    ,
C0.730     ,0.0226    ,0.6210    ,0.821     ,0.0254    ,0.6258    ,
D0.876     ,0.0275    ,0.6250    ,0.931     ,0.0298    ,0.6230    ,
E0.963     ,0.0322    ,0.6009    /
 END
```

SOATA
PCNF
CNF
ZF
PRF
WAFC
WAF
PCNC
CNC
ZC
PRC
WACC
WAC
T2
P2
T21
P21
T3
P3
PCBLF
BLF
PCBLC
BLC
PCBLOB
BLOB
PCBLHP
BLHP
PCBLLP
BLLP
T4
P4
WA3
WFB
WG4
FAR4
ETAB
DPCOM
TFFHP
CNHP
DHTCHP
DHTC
T5
P5
TFFLP
CNLP
DHTCLP
DHTF
T55
P55
PCBLDU
BLDU
T24
P24
T25
P25
WAD
WFD
WG24
FAR24
ETAD
DPDUC

```
ETAF
ETAC
ETATHP
ETATLP
AM55
AM25
T6
P6
PS6
AM6
V6
WG6
T7
WFA
WG7
FAR7
ETAA
DPAFT
PS8
AM8
V8
PS9
AM9
V9
PS28
AM28
V28
PS29
AM29
V29
BYPASS
HPEXT
WFT
WGT
VA
FRD
CVMNOZ
VJM
CVDNOZ
VJD
FGM
FGP
THEEND
 $DATAIN ITITLE=1,IDES=1,MODE=0,IDUMP=1,IAMTP=0,
 IGASMX=1,IDCD=0,IMCD=0,NOZFLT=0,ITRYS=210,TOLALL=0.005,
 ZFDS=0.833333,PCMFDS=100.,PRFDS=2.00,ETAFDS=0.850,WAFDS=180.,
 ZCDS=0.814598,PCMCDS=100.,PRCDS=6.00,ETACDS=0.830,WACDS=75.0,
 T4DS=2400.,DTCODS=1260.,ETABDS=0.985,DPCODS=0.05,
 TFHPDS=21.25,CMHPDS=2.040,ETHPDS=0.890,
 TFLPDS=53.41,CMLPDS=2.280,ETLPDS=0.900,
 DPDUDS=0.04,DPAFDS=0.04,
 AK=0.0,ALTP=0.0,HPEXT=0.0,PCBLF=0.0,PCBLC=0.05,
 PCBLDU=0.20,PCBLDB=0.0,PCBLHP=0.80,PCBLLP=0.0,
 PS55=1.88,CVDNOZ=0.985,CVMNOZ=0.985,DELFG=1.0,DELFN=1.0,DELSFC=1.0$
                                  SMOTE DESIGN POINT
 $DATAIN T4=1750.,AM=0.0,ALTP=00000.,ITITLE=1$
                                        SLS IDLE
 $DATAIN T4=2400.,AM=0.0,ALTP=00000.,ITITLE=1$
                                      SLS MILITARY
 $DATAIN T4=2400.,AM=0.0,ALTP=00000.,ITITLE=1,
 IAFTEN=1,T7=3200.,ETAA=0.91$
```

```
                                        SLS TAKE-OFF
$DATAIN T4=2400.,AM=1.2,ALTP=00500.,ITITLE=1$
                                   SET-UP LOW ALTITUDE DASH
$DATAIN T4=2400.,AM=1.2,ALTP=00500.,ITITLE=1,
IAFTBN=1,T7=3200.,ETAA=0.91$
                                      LOW ALTITUDE DASH
$DATAIN T4=2100.,AM=0.8,ALTP=36100.,ITITLE=1$
                                       SUBSONIC CRUISE
$DATAIN T4=2400.,AM=1.2,ALTP=50000.,ITITLE=1$
                                 SUPERSONIC AT MILITARY POWER
$DATAIN T4=2400.,AM=1.6,ALTP=50000.,ITITLE=1$
                            SET-UP SUPERSONIC WITH AFTERBURNER
$DATAIN T4=2400.,AM=1.6,ALTP=50000.,ITITLE=1,
IAFTBN=1,T7=1700.,ETAA=0.94$
                                     SUPERSONIC PARTIAL A/B
$DATAIN T4=2400.,AM=1.6,ALTP=50000.,ITITLE=1,
IAFTBN=1,T7=3200.,ETAA=0.91$
                                      SUPERSONIC FULL A/B
$EOF
```

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* Air Force Aero Propulsion Laboratory Wright-Patterson Air Force Base, Ohio 45433	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

SIMULATION OF TURBOFAN ENGINE

Part II. User's Manual and Computer Program Listing

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5. AUTHOR(S) *(Last name, first name, initial)*

McKinney, John S.
Captain, USAF

6. REPORT DATE November 1967	7a. TOTAL NO. OF PAGES 94	7b. NO. OF REFS 0
8a. CONTRACT OR GRANT NO. b. PROJECT NO. 3066 c. Task No. 306603 d.	9a. ORIGINATOR'S REPORT NUMBER(S) AFAFL-TR-67-125, Part II 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*	

10. AVAILABILITY/LIMITATION NOTICES This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of the Air Force Aero Propulsion Laboratory, Wright-Patterson AFB, Ohio.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Air Force Aero Propulsion Laboratory Wright-Patterson Air Force Base, Ohio 45433

13. ABSTRACT

This report describes a digital computer program titled SMOTE (Simulation of Turbofan Engine). SMOTE is a computer program for balancing-cycle turbofan engines capable of running both design and off-design points. The program is written in Fortran IV language and was designed for use on an IBM 7090 Digital Computer, although it has also been run on an IBM System 360. Performance maps (Block Data format) of the major engine components are required. Information for setting up the Block Data and input data is given in the report. Also included in the report is a complete program listing with a description of each subroutine and a sample data pack.

(Distribution of the abstract is unlimited.)

DD FORM 1 JAN 64 1473

UNCLASSIFIED

Security Classification

UNCLASSIFIED

Security Classification

14 KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
Turbine Engine Simulation						
Turbine Engine Computer Programs						
Turbine Engine Cycle Analysis						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through _____."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through _____."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through _____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.

UNCLASSIFIED

Security Classification