GENERAL ELECTRIC
COMPANY

VALLEY FORGE SPACE TECHNOLOGY CENTER (MAIL: P. O. BOX 8555, PHILA., PA. 19101) . . TEL. 969-2000

MISSILE AND
SPACE DIVISION

SPACE SCIENCES
LABORATORY

August 9, 1967

Headquarters
National Aeronautics & Space Administration
Washington, D. C. 20546

Attention: Code BCN-1/Mr. J. F. Meikrantz, Jr.

Subject: Contract NASW - 1586
Quarterly Report No. 2

Gentlemen:

In accordance with the requirements of Article IV-B of subject contract, forwarded herewith are twenty-two (22) copies of Quarterly Report No. 2 covering the period April 25, 1967 to July 24, 1967.

Very truly yours,

E. A. Blum

E. A. Blum
Contract Administrator

/lrr

Enclosures: (22)

cc: Code UT/New Technology Representative (1 copy)
NASA Representative, Scientific and Technical Information Facility (2 copies)
DCASR-Phila. /M. A. Hyland

N67-34921
(ACCESSION NUMBER)
46
(PAGES)
CR-87508
(NASA CR OR TMX OR AD NUMBER)
(THRU)
1
(CODE)
25
(CATEGORY)
FACILITY FORM 602

Second Quarterly Report

MHD BOUNDARY LAYERS INVOLVING NON-EQUILIBRIUM IONIZATION

(25 April 1967 - 24 July 1967)

by

Arthur Sherman
Eli Reshotko (Consultant)
Giles Crimi

Prepared under Contract No. NASw-1586 by

Space Sciences Laboratory
GENERAL ELECTRIC COMPANY
King of Prussia, Pa.

for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

TABLE OF CONTENTS

Section		Page
I.	Introduction	1
II.	Analysis	2
III.	Computer Program	4
IV.	Channel Flow for Initial Studies	5
V.	Validity of Collision Free Sheath Assumption	9
VI.	Further Work	10
	Appendix A	11
	Appendix B	12

I. INTRODUCTION

The present report is the second quarterly report under Contract NASw-1586 and covers the period 25 April 1967 to 24 July 1967. The problem being considered is the boundary layer that will develop over the segmented electrode wall in an MHD channel when a non-equilibrium plasma is flowing. In our analysis the characteristics of the plasma sheath at the base of the continuum boundary layer as well as the variation of the electron temperature throughout the boundary layer are considered. Due to our desire to study finite electrode segments in the wall over which the boundary layer develops, we shall use a finite difference technique in solving the boundary layer equations. This will permit the study of non-similar behavior in a reasonably accurate way.

II. ANALYSIS

In order to complete the formulation of our boundary layer problem one item, in addition to our development in the first quarterly report, remained. We needed some convenient way of specifying conditions at some initial ξ station before proceeding with the full finite difference calculation. This problem has been resolved by assuming "local similarity" at this initial location ($j_y = E_x = 0$). In other words, our original equations were simplified by neglecting all $\frac{\partial}{\partial \xi}$ terms and treating $\xi_{initial}$ as a parameter. The resulting equations are shown below:

Momentum:

$$(\ell f'')' + f f'' = 0 \tag{1}$$

Overall Energy:

$$\left(\frac{\ell}{P_R} g'\right)' + f g' + \frac{u_\infty^2}{h_\infty}\left[\ell (f'')^2\right] + \frac{T_{e_\infty}}{T_\infty}(\lambda \theta')'$$

$$+ I\left(\frac{S_1 T_{e_\infty}}{\rho_\infty C_p T_\infty} f g \theta' - \frac{S_2 f g g'}{\rho_\infty} + \frac{n_e}{\rho_\infty C_p T_\infty} f g'\right) = 0 \tag{2}$$

Electron Energy:

$$(\lambda \theta')' + \left[\frac{\frac{3}{2} k n_e + (\frac{3}{2} k T_{e_\infty} \theta + I) S_1}{\rho_\infty C_p}\right] f g \theta'$$

$$- \left[\frac{(\frac{3}{2} k T_{e_\infty} \theta + I) T_\infty S_2}{\rho_\infty T_{e_\infty}}\right] f g g' + \left[\frac{n_e (\frac{5}{2} k T_{e_\infty} \theta + I)}{\rho_\infty C_p T_{e_\infty}}\right] f g'$$

$$- \left[\left(\frac{2 \xi_I}{(\rho \mu)_R u_\infty^2}\right)\left(\frac{3 n_e k}{\rho_\infty C_p}\right)\left(m_e \sum_s \frac{\nu_s}{m_s}\right) g \left(\theta - \frac{T_\infty}{T_{e_\infty}} g\right)\right] = 0 \tag{3}$$

where $(\ )' = \frac{d}{d\eta}$

Boundary Conditions:

$$\eta = \infty \qquad f' = g = \theta = 1$$

$$\eta = 0 \qquad f = f' = 0, \quad g = g_w,$$

$$\theta_w' = \frac{2}{5} \frac{\sqrt{2\xi}\, m_c\, n_{e_w}}{\lambda (\rho\mu)_R\, u_\infty} \sqrt{\frac{k\, T_{e_\infty}}{m_s}} \left(2 + \ln \sqrt{\frac{m_s}{2\pi\, m_e}}\right) \theta_w^{3/2}$$

The above set represents three coupled non-linear ordinary differential equations that have to be solved simultaneously while satisfying two point boundary conditions. Their solution is only needed once, however, for any known wall temperature and chosen value of ξ_I.

III. COMPUTER PROGRAM

The program described in the first quarterly report for the main finite difference calculation has been written and operated. A listing is included as Appendix A. To date the program has run properly and produced velocity, gas temperature, and electron temperature profiles that are plausible. Further more detailed checks on the self-consistency of the results are now being carried out before any extensive calculations will be attempted. The initial results suggest that as the current flow through an electrode is increased the electron temperature near the wall can build up, possibly to a value larger than in the free stream.

Another program has been written to carry out the calculation of the initial profile. The three equations are reduced to seven first-order non-linear equations which are then solved as a two-point boundary value problem. The approach is to guess f_w'', g_w', θ_w and check the profiles to see if f_∞', g_∞, θ_∞ all go to unity. If not other guesses are made and the calculation repeated. A flow diagram for this program and a listing are shown in Appendix B. So far the program has run and yielded plausible profiles. Again further checks are being made to verify the accuracy of the calculation. Initial results, however, show that the electron temperature profile will be several times thicker than the velocity or gas temperature profiles. One would expect this since the electron thermal conductivity is so much larger than the gas thermal conductivity.

IV. CHANNEL FLOW FOR INITIAL STUDIES

The channel flow chosen for the initial boundary layer calculation has been developed by Les Nichols and is his case #001351 dated November 16, 1966. He has chosen the following conditions for the channel flow.

$K = 0.700$ Seed = 0.01 $P^o = 2 \times 10^5 \frac{\text{newtons}}{m^2}$

$T^o = 2000^oK$ $M = 0.500$ $B = 10,000$ gauss

Argon + Cesium

The ξ variation of velocity, gas temperature, pressure, and density can be taken directly from his calculation. They are shown in Table I where $\xi_I = .01$ and all edge quantities are to be specified at the half station. We have chosen a $\Delta\xi = .00002$ for our calculation.

ξ	u_∞	$\frac{du_\infty}{d\xi}$	T_∞	$\frac{dT_\infty}{d\xi}$	p	ρ_∞	$\frac{d\rho_\infty}{d\xi}$
.01001	395.61	222	1919.98	1459	164,000	0.5	0
.01003	395.61	222	1919.95	1459	163,999	0.5	0
.01005	395.62	222	1919.92	1459	163,998	0.5	0
.01007	395.62	222	1919.89	1459	163,997	0.5	0
.01009	395.63	222	1919.86	1459	163,996	0.5	0
.01011	395.63	222	1919.83	1459	163,995	0.5	0
.01013	395.64	222	1919.81	1459	163,994	0.5	0
.01015	395.64	222	1919.79	1459	163,994	0.5	0
.01017	395.65	222	1919.76	1459	163,993	0.5	0
.01019	395.65	222	1919.74	1459	163,992	0.5	0

Table I

In addition, we have taken $\ell = g^{-1/4}$ and $P_R = 2/3$. For the electrical quantities and T_e we cannot use the channel flow values directly as they

assume an infinitely fine segmentation whereas we are calculating a boundary layer with finite segments. To resolve this difficulty we have calculated j_{y_∞} and E_{x_∞} at the channel centerline to be

$$j_{y_\infty} = \pm 2250 \text{ amps/m}^2 \qquad + \text{ anode, } - \text{ cathode}$$

$$E_{x_\infty} = -226 \text{ volts/meter}$$

These uniform values were then redistributed over the electrode-insulator wall making use of previously known theoretical calculations. As yet there is no calculation which corresponds accurately to our conditions so the above redistribution is for the time being approximate. The j_{y_∞} and E_{x_∞} distributions are shown in Figure 1 and Table II.

ξ	j_{y_∞}	E_{x_∞}	T_{e_∞}	$\frac{dT_{e_\infty}}{d\xi}$
.01001	0	0	1921	0
.01003	0	0	1922	50,000
.01005	0	0	1925	150,000
.01007	0	0	1927	100,000
.01009	-100	0	1928	50,000
.01011	-200	0	1928	0
.01013	-350	0	1928	0
.01015	-500	0	1928	0
.01017	-900	0	1928	0
.01019	-1300	0	1928	0

Table II

It should be noted that the nozzle throat has been assumed to be approximately 34 inches from the start of the MHD generator electrode wall.

The first numerical results will be obtained for the above case since edge quantities are varying moderately and should not introduce any unexpected numerical difficulties. Later calculations will look at supersonic

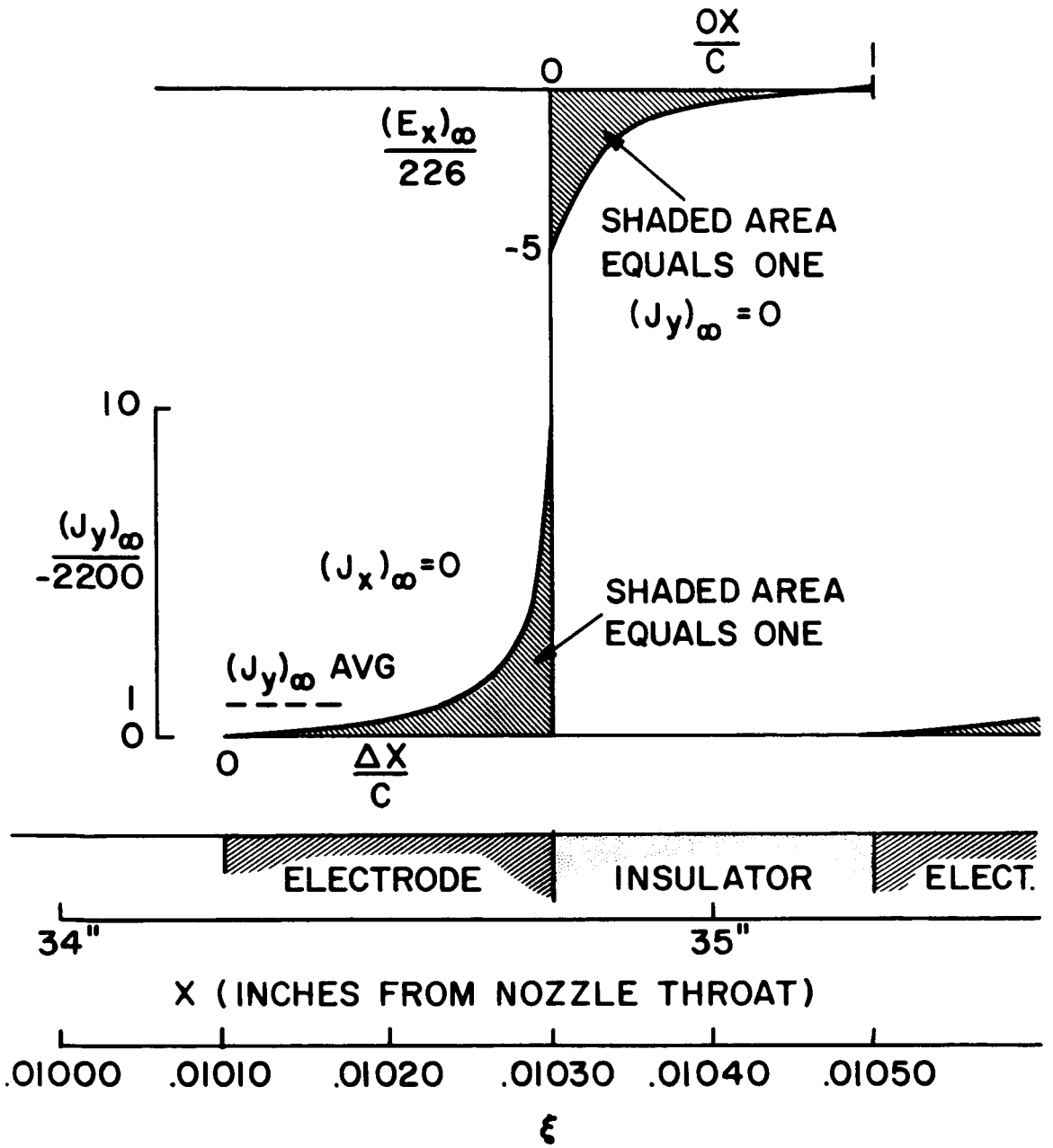


Figure 1. Sketch of J_{y_∞} and E_{x_∞} Distributions

free stream flows. Also observe that the first calculation only carries to $\xi = .01019$ (half-way along the first electrode). This is a practical limitation due to storage problems in the computer. The final profiles from this calculation will serve as the initial profiles for the next ten ξ stations and so on.

V. VALIDITY OF COLLISION FREE SHEATH ASSUMPTION

Until recent years all plasma-surface interaction analysis worked from the assumption of a sheath of some specified thickness separating the wall and collision dominated plasma (assuming Debye length << mean free path). A matching procedure is then used to correlate the solutions in the two regions.

This is the procedure that we are following in the present analysis. However, it is not clear what error is involved when the Debye length is the same magnitude as the mean free path, a case which may be of practical significance. Also, the potential due to the charge distribution near the wall does not in fact vanish at some well-defined boundary, as has been assumed.

It is in order to clarify such questions as the above that we have begun a review of current analysis of plasma-surface interactions. The most relevant would seem to be the "moment" methods since they do not require the assumption of a sheath. Of particular interest is the work by Bienkowski (Princeton), Probstien (MIT), and Terhune (GE Research Laboratory). They should all be valid, in general, whether $d < \lambda$ or $d > \lambda$.

VI. FURTHER WORK

In the next reporting period, we expect to have correct calculations completed for the first channel flow case. This includes the locally similar solution in the entrance region and the finite difference solution over the electrode wall.

Efforts will continue to clarify the range of validity of our sheath assumption. We shall also attempt to formulate a more accurate sheath analysis for possible future use where conditions warrant it.

A review of the literature will also be begun to try to determine more accurate formulas for transport property evaluations.

APPENDIX A

The listing included here is for the main finite difference computation. The flow charts for the main program and its subroutines are recorded in Section III of the first quarterly progress report (25 January - 24 April 1967).

```
      CMAIN
            COMMON/COM1/BK,EM,EC,C1,PI,AKO,R,Y(50),TE,DUM(10),TW,EKSIS,EKSIM,D
           1UM1(7),MEDGE,EKSI
            COMMON/COM3/SMLHH(3),DUM2(350),TEEKSI(10)
            COMMON/COM5/N,NPL1,PCC,NWRIT,KK,NK,CONST,NRTST,NTIMES,DUM3(9),NVMN
      C     RETURN HERE FOR START OF NEXT CASE
         10 MEDGE=1
            NRTST = 1
            NTIMES = 0
            NK = 1
            BK = 1.38E-23
            EM = 9.107E-31
            EC = 1.602E-19
            C1 = 2.42E21
            PI = 3.1416
            AKO = 8.854E-12
            Y(1) = 0.
            R = 8.317E3
            CALL READIN
         40 CALL EDGE1
            SMLHH(2) = TW/TEEKSI(MEDGE)
         50 N = 0
            CALL NXTLST
         60 N = 1
            CALL NXTLST
         70 N = N+1
            CALL ABCD
            CALL EKPK
            IF (NRTST-N) 80,80,90
         80 CALL TEST
            NRTST = N
            IF (CONST) 90,200,200
         90 IF (N-49) 70,110,110
      C     CONVERGENCE NOT ATTAINED, PRINT AND GO TO NEXT CASE
        110 WRITE (6,903)
            NWRIT = NPL1
            CALL WOSUB
            GO TO 10
      C     CONVERGENCE ATTAINED, GO TO NEXT PROFILE
        200 CALL WNSUB
            IF (PCC) 201,202,202
        201 NK = KK
        202 IF (KK-NK) 220,220,210
      C     ITERATE UP TO KK TIMES
        210 NVMN = 2
            NK = NK+1
            CALL VMNSUB($50)
      C     INITIALIZE FOR NEXT PROFILE
        220 NK = 1
            NVMN = 1
            NWRIT = NPL1
            CALL VMNSUB($50)
```

```
      IF (EKSIM-EKSI) 260,260,40                                        44
  260 WRITE (6,902)                                                     45
      GO TO 10                                                          47
  902 FORMAT (16H THATS ALL FOLKS)                                      48
  903 FORMAT (36H CONVERGENCE NOT ATTAINED, TRY AGAIN)                  48
      END                                                               48
```

```
      CREADIN
            SUBROUTINE READIN
            DIMENSION ARAKSI(10)
            COMMON/COM1/DUM1(68),TW,EKSIS,EKSIM,DKSI,DUM2(3),DETA,TSQSI,DELTA,
           1MEDGE,EKSI
            COMMON/COM2/EDGE(10,11),TEMPRA(50),QEAT(50),QECST(50)
            COMMON/COM3/SMLHH(3),V(50),WLST(50,3),W(50,3),TEEKSI(10)
            COMMON/COM5/N,NPL1,PCC,NWRIT,KK,NK,CONST,NRTST,NTIMES,EP1,EP2,EP3,
           1ERR,EPP1,EPP2,EPP3,NPRINT,NTAB,NVMN
            COMMON/COM6/DUM3(25),C2,PER,QIN,CP,ROMR,CM,TIW,PIZ,SM,BZ
            DIMENSION FF(12)
            NAMELIST/NAME1/DELTA,EKSIS,EKSIM,DKSI,DETA,QIN,CM,SM,TIW,CP,PER,PI
           1Z,C2,BZ,ERR,EPP1,EPP2,EPP3,EP1,EP2,EP3
            NAMELIST/NAME2/TEMPRA,QEAT,QECST,EDGE,SMLHH,TEEKSI,V,W,NWRIT
            NAMELIST/NAME3/KK,NTAB
            READ (5,911)(FF(I),I = 1,12)
            CALL SPGHDR(FF)
            READ (5,NAME1)
            READ (5,NAME2)
            IF (ERR) 7,8,8
          8 CALL RUNKUT
          7 READ (5,NAME3)
            ROMR=3.1E-7*EDGE(1,7)*EDGE(1,2)**.75
            WRITE (6,900)EKSIS,EKSIM,DKSI,DETA,QIN
        900 FORMAT (7H0EKSIS=E16.8,3X,6HEKSIM=E16.8,3X,5HDKSI=E16.8,3X,5HDETA=
           1E16.8,3X,4HQIN=E16.8//)
            WRITE (6,901)DELTA,ROMR,CM,SM,CP
        901 FORMAT (7H0DELTA=E16.8,3X,6HROMR =E16.8,3X,5HCM  =E16.8,3X,5HSM  =
           1E16.8,3X,4HCP =E16.8//)
            WRITE (6,902)TIW,PER,PIZ,C2,BZ
        902 FORMAT (7H0TIW  =E16.8,3X,6HPER  =E16.8,3X,5HPIZ =E16.8,3X,5HC2  =
           1E16.8,3X,4HBZ =E16.8//)
            CALL INITIL
            CALL WOSUB
            WRITE (6,905)
        905 FORMAT (1H1/14X6HTEMPRA20X4HQEAT21X5HQECST/)
            WRITE (6,906)(TEMPRA(I),QEAT(I),QECST(I),I = 1,NTAB)
        906 FORMAT (1H 3E25.8)
            NN = (EKSIM-EKSIS)/DKSI+1.5
            WRITE (6,907)
        907 FORMAT (1H1/10X4HEKSI17X2HUE18X2HTE18X3HETE17X3HDUE17X3HDTE/)
            ARAKSI(1) = EKSIS
            DO 10 I = 2,NN
         10 ARAKSI(I) = ARAKSI(I-1)+DKSI
            WRITE (6,908)(ARAKSI(I),(EDGE(I,J),J = 1,5),I = 1,NN)
        908 FORMAT (1H 6E20.8)
            WRITE (6,909)
        909 FORMAT (1H1/7X4HEKSI13X4HDETE13X4HRHOE13X5HDRHOE12X4HAJYE14X3HEXE1
           13X4HPRES/)
            WRITE (6,910)(ARAKSI(I),(EDGE(I,J),J = 6,11),I = 1,NN)
        910 FORMAT (1H 7E17.8)
        911 FORMAT (12A6)
```

```
 53        9999 RETURN
 54             END
```

```
CINITIL
      SUBROUTINE INITIL
      COMMON/COM1/DUM1(57),TE,DUM2(10),TW,DUM3(3),TDA,TDDA,DETA2,DETA,DU
     1M4(4)
      COMMON/COM3/SMLHH(3),V(50),WLST(50,3),W(50,3),TEEKSI(10)
      COMMON/COM5/N,NPL1,PCC,NWRIT,DUM5(15)
      DO 10 I = 1,NWRIT
      DO 11 J = 1,3
   11 WLST(I,J) = W(I,J)
   10 CONTINUE
C     EXTEND PROFILES
      NPL1 = NWRIT+1
      DO 12 I = NPL1,50
      V(I) = V(I-1)-DETA
      DO 13 J = 1,3
      WLST(I,J) = 1.
   13 W(I,J) = 1.
   12 CONTINUE
      TW = SMLHH(2)*TEEKSI(1)
      TDA = 2.*DETA
      TDDA = 2./DETA
      DETA2 = DETA**2
 9999 RETURN
      END
```

```
CEDGE1
      SUBROUTINE EDGE1
      COMMON/COM1/DUM1(57),TE,ETE,PRES,AJYE,UE,DUE,DTE,DETE,RHOE,DRHOE,E
     1XE,TW,EKSIS,EKSIM,DKSI,DUM2(4),TSQSI,DELTA,MEDGE,EKSI
      COMMON/COM2/EDGE(10,11),DUM3(150)
      COMMON/COM6/DUM4(29),ROMR,DUM5(5)
      UE = EDGE(MEDGE,1)
      TE = EDGE(MEDGE,2)
      ETE = EDGE(MEDGE,3)
      DUE = EDGE(MEDGE,4)
      DTE = EDGE(MEDGE,5)
      DETE = EDGE(MEDGE,6)
      RHOE = EDGE(MEDGE,7)
      DRHOE = EDGE(MEDGE,8)
      AJYE = EDGE(MEDGE,9)
      EXE = EDGE(MEDGE,10)
      PRES = EDGE(MEDGE,11)
      ROMR=RHOE*3.1E-7*TE**.75
      AA = MEDGE-1
      EKSI = EKSIS*AA*DKSI
      MEDGE = MEDGE+1
      TSQSI = SQRT(2.*EKSI)
 9999 RETURN
      END
```

```
CNXTLST
      SUBROUTINE NXTLST
      COMMON/COM1/BK,EM,EC,C1,PI,AKO,R,Y(50),TE,ETE,PRES,AJYE,UE,DUM(5),
     1EXE,DUM1(7),DETA,TSQSI,DUM2(3)
      COMMON/COM2/EDGE(10,11),TEMPRA(50),QEAT(50),QECST(50)
      COMMON/COM3/DUM3(53),WLST(50,3),W(50,3),TEEKSI(10)
      COMMON/COM4/DUM4(900),H(3,3),EF(3,3),DUM5(60)
      COMMON/COM5/N,NPL1,DUM6(15),NTAB,NVMN
      COMMON/COM6/ELLS,EL,ELNX,YAMLS,YAM,YAMNX,PRNLS,PRN,PRNNX,ALF1LS,AL
     1F1,ALF1NX,ALF2LS,ALF2,ALF2NX,S,GNX,THENX,ENX,CONX,SXM,BETAX,QEI,QE
     2A,QECS,C2,PER,QIN,CP,ROMR,CM,TIW,PIZ,SM,BZ
      QAA = 1.E-16
      NPL1 = N+1
      ELLS = EL
      EL = ELNX
      YAMLS = YAM
      YAM = YAMNX
      PRNLS = PRN
      PRN = PRNNX
      ALF1LS = ALF1
      ALF1 = ALF1NX
      ALF2LS = ALF2
      ALF2 = ALF2NX
      GNX = .5*(W(NPL1,2)+WLST(NPL1,2))
      THENX = .5*(W(NPL1,3)+WLST(NPL1,3))
      ET = ETE*THENX
      PRNNX = 2./3.
      ELNX = GNX**(-.25)
      S=C1*(ET)**1.5*EXP(-C2/ET)
      GRP = 4.*PER/(BK*S)*PRES/TE/GNX
      IF (.01-GRP) 2,1,1
    1 ENX = GRP/4.*(1.-GRP/4.)*S
      GO TO 3
    2 ENX = S/2.*(SQRT(1.+GRP)-1.)
    3 ENUIN = (QIN/BK)*(PRES/GNX/TE)*SQRT((8.*BK*TE*GNX)/(PI*EM))
      CALL TABLE(TEMPRA,QEAT,QECST,ET,NTAB,N,QEA,QECS)
      QEI = PI*((EC/AKO)*(EC/(16.*BK*ETE*THENX)))**2*THENX**1.5
      QEI = QEI*ALOG(32.*(BK*ETE)**1.5/EC/EC*AKO**1.5/(EC*ENX**.5))
      ENUE = PRES*QEA/(BK*GNX*TE)+(PER*PRES/BK/TE/GNX-ENX)*QECS+ENX*QEI
      SQT = SQRT(8.*BK/EM*ETE*THENX/PI)
      ENUE = ENUE*SQT
      CONX = EC/EM*EC*ENX/ENUE
      BETAX = BZ*EC/ENUE/EM
      BETAI = BZ*EC/ENUIN/CM
      SXM = 1.+BETAX*BETAI
      A1 = 1.24E7*ET**1.5/ENX**.5
      A2 = 1.8E5*ET/ENX**(1./3.)
      GAM = ENX*(BK*TE/PRES)*GNX
      BKSKA = 3.E-6*ET**2.5/(TE*GNX)**.75/ALOG(55.+A1**4+A2**4)
      BKEKA = BKSKA/(1.+BETAX**2)/(1.+1.414*(1.-GAM)/GAM*BKSKA*
     1(QEA/QAA)*SQRT(EM/CM*GNX*TE/ET))
      YAMNX = ELNX/PRNNX*BKEKA
```

```
      ALF1NX = 1./SXM
      ALF2NX = CONX*BETAI/SXM
      IF (NPL1-1) 5,5,9999
    5 GRP = YAMNX*CP*ROMR*UE/BK/TSQSI/DETA
      AJEYW = ALF1NX*AJYE+ALF2NX*EXE
      SCRB = (AJYE+TIW)/EC/ENX*SQRT(2.*PI*EM/(BK*ETE*THENX))+SQRT((2.*PI
     1*EM)/CM)
      B2 = 2.*GRP/((2.-ALOG(SCRB))*(AJYE+TIW+EC*ENX*SQRT(BK*ETE*THENX/CM
     1))+1.5*GRP-2.5*AJEYW)
      B3 = -.25*B2
      DO 10 J = 1,3
      DO 11 I = 1,3
      H(I,J) = 0.
   11 FF(I,J) = 0.
   10 CONTINUE
      H(3,3) = B2
      FF(3,3) = B3
 9999 RETURN
      END
```

SM

SM

02A35 2 06-06-67

```
CTABLE
      SUBROUTINE TABLE(TEMP,QEA,QECS,ARG5,KTAB,N,XQEA,XQECS)
      DIMENSION TEMP(50),QEA(50),QECS(50)
      XTEMP = ARG5
    9 IF (N) 10,101,10                                                  2
   10 IF (XTEMP-XTEMPL) 101,11,11                                       3
  101 J = 1                                                             4
      NTAB = KTAB                                                       5
   11 NTAB1 = NTAB+1                                                    6
      K = J-1                                                           7
      CALL TLU1(XTEMP,NTAB,TEMP(J),J,IERR)                              8
      IF (IERR) 13,14,13                                                9
   13 WRITE (6,901)XTEMP                                               10
      GO TO 9999                                                       13
   14 NTAB = NTAB1-J                                                   14
      J = J+K                                                          15
      IF (NTAB) 9999,9999,15                                           16
   15 XQEA = TNT1(XTEMP,NTAB,TEMP(J),QEA(J),2,IERR)                    17
      XQECS = TNT1(XTEMP,NTAB,TEMP(J),QECS(J),2,IERR)                  18
   16 XTEMPL = XTEMP                                                   19
  901 FORMAT (34H THIS TEMPERATURE IS NOT IN TABLE-,E16.8)             20
 9999 RETURN                                                           20
      END                                                              21
```

```
CABCD
      SUBROUTINE ABCD
      COMMON/COM1/BK,EM,EC,C1,PI,AKO,R,Y(50),TE,ETE,PRES,AJYE,UE,DUE,DTE
     1,DETE,RHOE,DRHOE,EXE,TW,EKSIS,EKSIM,DKSI,TDA,TDDA,DETA2,DETA,TSQSI
     2,DELTA,MEDGE,EKSI
      COMMON/COM3/SMLHH(3),V(50),WLST(50,3),W(50,3),TEEKSI(10)
      COMMON/COM4/E(150,3),PHI(150,3),H(3,3),EF(3,3),TEMP(3,3),AA(3,3),B
     1(3,3),C(3,3),D(3),A(3,3),CKMAT(3,3),TEMPP(3)
      COMMON/COM5/N,NPL1,DUM(15),NTAB,NVMN
      COMMON/COM6/ELLS,EL,ELNX,YAMLS,YAM,YAMNX,PRNLS,PRN,PRNNX,ALF1LS,AL
     1F1,ALF1NX,ALF2LS,ALF2,ALF2NX,S,GNX,THENX,ENX,CONX,SXM,BETAX,QEI,QE
     2A,QECS,C2,PER,QIN,CP,ROMR,CM,TIW,PIZ,SM,BZ
      FP =  .5*(W(N,1)+WLST(N,1))
      G = GNX
      THETA = THENX
      BETAE = BETAX
      COND = CONX
      ENE = ENX
      SUM = SXM
      DO 10 I=1,3
   10 D(I) = 0.
      DO 11 I=1,3
      DO 11 J=1,3
      AA(I,J)=0.
      B(I,J)=0.
   11 C(I,J)=0.
      SNUS = (PRES/BK*QEA)/TE/G/CM+(QECS/SM)*(PER*PRES/TE/BK/G-ENE)+(EN
     1E*QEI)/CM
      SNUS = SNUS*DELTA*SQRT(8.*BK*ETE*THETA/PI/EM)
      GRP = PRES*PER/BK/TE/G
      S1 = S*(1.5/ETE/THETA+C2/(ETE*THETA)**2)*(-.5+(S/2.+GRP)/S/SQRT(1.
     1+4.*GRP/S))
      S2 = .8*(CM/BK)*GRP/TE/G/SQRT(1.+4.*GRP/S)
      S3 = 2.*GRP/PRES/SQRT(1.+4.*GRP/S)
      FPA = (WLST(N+1,1)-WLST(N-1,1))/TDA
      FPAA = (WLST(N+1,1)-2.*WLST(N,1)+WLST(N-1,1))/DETA2
      GA = (WLST(N+1,2)-WLST(N-1,2))/TDA
      GAA = (WLST(N+1,2)-2.*WLST(N,2)+WLST(N-1,2))/DETA2
      THA = (WLST(N+1,3)-WLST(N-1,3))/TDA
      THAA = (WLST(N+1,3)-2.*WLST(N,3)+WLST(N-1,3))/DETA2
      RHOS = R/PRES/40.*TE*(WLST(N,2)+WLST(N-1,2))
      Y(N) = Y(N-1)+TSQSI*RHOS*.5*DETA/UE
      CALL NXTLST
      ELA = (ELNX-ELLS)/TDA
      YAMA = (YAMNX-YAMLS)/TDA
      PRA = (PRNNX-PRNLS)/TDA
      ALF1A = (ALF1NX-ALF1LS)/TDA
      ALF2A = (ALF2NX-ALF2LS)/TDA
      V1 = V(N)
      Q1 = DKSI/(4.*EKSI*FP*TDA)
      Q2 = 2.5*BK*ETE/EC*TSQSI/(ROMR*UE*CP*TE)
      Q3 = DKSI*DTE
```

```
      Q4 = PIZ/(RHOE*CP*TE)
      Q5 = 1.5*BK*ETE*THETA+PIZ
      Q6 = 2.5*BK*ETE*THETA+PIZ
      Q7 = 5.*BK/EC*RHOE*ETE*UE/TSOSI/G
      Q9 = ROMR*UE**2
      Q0 = RHOE*CP*ETE*Q9
      Q8 = RHOE*CP*TE*Q9
      Q10 = 1.5*BK*ENE
      F1 = FP
      Q = G
      FP = WLST(N,1)
      G = WLST(N,2)
      THETA = WLST(N,3)
      AA(1,1) = Q1*(V1-2.*EL/DETA-ELA)
      AA(1,2) = 0.
      AA(1,3) = 0.
      B(1,1) = 1.+DKSI*DUE/2./UE+4.*Q1*EL/DETA
      B(1,2) = -DKSI*DUE/(2.*UE*F1)
      B(1,3) = 0.
      C(1,1) = Q1*(-V1-2.*EL/DETA+ELA)
      C(1,2) = 0.
      C(1,3) = 0.
      D(1) = FP*(1.-DKSI*DUE/2./UE)+(DKSI/4./EKSI/F1)*(-FPA*V1+2.*EKSI*G
     1*DUE/UE+ELA*FPA+EL*FPAA)
      AA(2,1) = -2.*Q1*UE**2/CP/TE*EL*FPA
      AA(2,2) = Q1*(V1-2.*EL/PRN/DETA-ELA/PRN*Q4*ENE*V1-PIZ*S2*V1*G/RHOE
     1)
      AA(2,3) = Q1*(-2.*ETE*YAM/DETA/TE-ETE*YAMA/TE-Q2*(ALF1*AJYE+ALF2*E
     1XE)+Q4*ETE*V1*S1*G)
      B(2,1) = 0
      B(2,2) = 1.+DKSI*UE*DUE/(2.*CP*TE)*(1.-PIZ*S3)+4.*Q1*EL/PRN/DETA+Q
     13/2./TE+(Q4*S1)*THETA*Q3-PIZ*S2*G/RHOE-PIZ*CP*Q3*S2*G/RHOE/UE
      B(2,2) = B(2,2)+DKSI/2.*UE*DUE*(ENE*Q4)+(PIZ*ENE)/CP/RHOE/TE+(Q4*
     1ENE)/2./TE*Q3-DKSI*COND*EXE**2/Q8/2./F1/SUM-DKSI*AJYE**2*(SUM**2+
     2BETAE**2)/(2.*COND*F1*Q8*SUM)+DKSI*AJYE*BZ*(1.-PIZ*S3)/2.*UE/Q8-
     3DKSI*(PIZ*ENE)*BZ/2./PRES*UE/Q8*AJYE
      B(2,3) = 4.*Q1*ETE*YAM/DETA/TE-Q2*DKSI/8./F1/EKSI*(AJYE*ALF1A+EXE*
     1ALF2A)+Q4*ETE*S1*G+Q4*S1*G*Q3/2.
      C(2,1) = -AA(2,1)
      C(2,2) = Q1*(-V1-2.*EL/DETA/PRN+ELA/PRN-Q4*ENE*V1+PIZ*S2*V1*G/RHOE
     1)
      C(2,3) = Q1*(ETE/TE*(-2.*YAM/DETA+YAMA)+Q2*(AJYE*ALF1+EXE*ALF2)-Q4
     1*S1*ETE*V1*G)
      D(2) = G-Q3*G/2./TE-(1.-PIZ*S3)*UE*DKSI*G/(2.*CP*TE)*DUE+2.*DETA*Q
     11*(EL/PRN*GAA+GA*(ELA/PRN-V1)+ETE/TE*(YAM*THAA+THA*YAMA)+Q2*(THA*(
     2AJYE*ALF1+EXE*ALF2)+THETA*(AJYE*ALF1A+EXE*ALF2A))-Q4*ETE*S1*G*THA*
     3V1-Q4*ENE*GA*V1+PIZ*S2*V1*G*GA/RHOE+2.*G*EKSI/SUM*(COND*EXE**2+AJY
     4E**2/COND*(SUM**2+BETAE**2))/Q8)+Q4*S1*ETE*G*THETA-PIZ*S2*G**2/RHO
     5E-DKSI*UE*RHOE*(Q4*ENE)*G/2./PRES*DUE+Q4*ENE*G+DKSI*AJYE*BZ/2.*UE
     6/Q8*(G*(1.-PIZ*S3)+PIZ*ENE/PRES)-Q4*ENE*Q3*G/2./TE
      AA(3,1) = 0.
      AA(3,2) = (ENE/Q*Q6-S2*Q5*CP*TE)*Q9*V1/8./EKSI/DETA
```

```
      AA(3,3) = Q9*ETE/4./EKSI/DETA*(V1/2.*(Q10+S1*Q5)-CP*RHOE/Q*(YAM/DE
     1TA+YAMA/2.))-Q7/8./DETA*(ALF1*AJYE+ALF2*EXE)
      B(3,1) = THETA*(Q9*DETE/2.*(Q10+S1*Q5)-.75*ETE*BK*S3*UE*(-AJYE*BZ+
     1Q9*RHOE*DUE))-(ENE*Q6)*Q9*DRHOE/2./RHOE-S2*Q5*Q9/2.*CP*DTE *G-PIZ
     2*S3/2.*(Q9*UE*RHOE*DUE-UE*AJYE*BZ)
      B(3,2) = Q9*FP*((ENE*Q6)/Q/DKSI-(Q5*S2)*CP*(TE/DKSI+DTE/2.))-Q10*
     1TE*(EM*SNUS)
      B(3,3) = (Q10+S1*Q5)*Q9*(TE*F1/DKSI+.5*DETE*FP)-.75*BK*ETE*S3*FP*Q
     19*RHOE*UE*DUE+Q0*YAM/2./(EKSI*DETA**2*Q)-.25*Q7*(ALF1A*AJYE+ALF2A*
     2EXE)+Q10*ETE*(EM*SNUS)+.75*ETE*UE*S3*AJYE*BZ*FP*BK
      C(3,1) = 0.
      C(3,2) = -(ENE/Q*Q6-S2*Q5*CP*TE)*Q9*V1/8./EKSI/DETA
      C(3,3) = -(Q10+Q5*S1)*Q9*ETE*V1/8./EKSI/DETA-Q0*(YAM/DETA-YAMA/2.)
     1/4./EKSI/DETA/Q+Q7/DETA*(ALF1*AJYE+ALF2*EXE)/8.
      D(3) = Q9*ETE*(Q10+Q5*S1)*(+F1/DKSI*THETA-V1*THA/4./EKSI)+Q9*(ENE/
     1G*Q6-S2*Q5*CP*TE)*(FP*G/DKSI-V1*GA/4./EKSI)+ENE*Q6*Q9*.5/RHOE*DRHO
     2E*FP
      D(3) = D(3)+PIZ*S3/2.*FP*(Q9*RHOE*UE*DUE-UE*AJYE*BZ)+Q0*(YAM*THAA+
     1YAMA*THA)/4./EKSI/Q+.25*Q7*(THA*(ALF1*AJYE+ALF2*EXE)+THETA*(ALF1A*
     2AJYE+ALF2A*EXE))+COND*EXE**2/SUM**2+AJYE**2/COND*(SUM**2+BETAE**2)
     3/SUM**2+Q10*TE*G*(EM*SNUS)-Q10*ETE*THETA*(EM*SNUS)
 9999 RETURN
      END
```

```
CEKPK
      SUBROUTINE EKPK
      COMMON/COM3/SMLHH(3),DUM1(360)
      COMMON/COM4/E(150,3),PHI(150,3),H(3,3),EF(3,3),TEMP(3,3),AA(3,3),B
     1(3,3),C(3,3),D(3),A(3,3),CKMAT(3,3),TEMPP(3)
      COMMON/COM5/N,NPL1,DUM(17)
      DIMENSION SCRACH(3,3)
      IF (N-2) 10,10,140
   10 DO 40 I = 1,3
      DO 30 J = 1,3
      A(I,J) = B(I,J)
      TEMP(I,J) = AA(I,J)
      DO 20 K = 1,3
      A(I,J) = C(I,K)*H(K,J)+A(I,J)
   20 TEMP(I,J) = C(I,K)*EF(K,J)+TEMP(I,J)
   30 CKMAT(I,J) = A(I,J)
   40 CONTINUE
   50 CALL MTINV(A,3,3,3,SCRACH)
      DO 80 I = 1,3
      DO 70 J = 1,3
      AA(I,J) = 0.
      DO 60 K = 1,3
   60 AA(I,J) = A(I,K)*TEMP(K,J)+AA(I,J)
   70 CONTINUE
   80 CONTINUE
      DO 110 I = 1,3
      E(6,I) = -AA(3,I)
      E(5,I) = -AA(2,I)
      E(4,I) = -AA(1,I)
      TEMPP(I) = 0.
      DO 100 J = 1,3
  100 TEMPP(I) = C(I,J)*SMLHH(J)+TEMPP(I)
  110 D(I) = D(I)-TEMPP(I)
      DO 130 I = 1,3
      PHI(2,I) = 0.
      DO 120 J = 1,3
  120 PHI(2,I) = A(I,J)*D(J)+PHI(2,I)
  130 CONTINUE
      GO TO 9999
  140 MN = (N-1)*3-2
      DO 160 J = 1,3
      TEMP(3,J) = E(MN+2,J)
      A(3,J) = B(3,J)
      TEMP(2,J) = E(MN+1,J)
      A(2,J) = B(2,J)
      TEMP(1,J) = E(MN,J)
  160 A(1,J) = B(1,J)
      DO 190 I = 1,3
      DO 180 J = 1,3
      DO 170 K = 1,3
  170 A(I,J) = C(I,K)*TEMP(K,J)+A(I,J)
  180 CONTINUE
```

```
  190 CONTINUE
  200 CALL MTINV(A,3,3,3,SCRACH)
      DO 230 I = 1,3
      DO 220 J = 1,3
      TEMP(I,J) = 0.
      DO 210 K = 1,3
  210 TEMP(I,J) = A(I,K)*AA(K,J)+TEMP(I,J)
  220 CONTINUE
  230 CONTINUE
      M = 3*N-2
      DO 250 I = 1,3
      E(M+2,I) = -TEMP(3,I)
      E(M+1,I) = -TEMP(2,I)
  250 E(M,I) = -TEMP(1,I)
      DO 270 I = 1,3
      TEMPP(I) = 0.
      PHI(N,I) = 0.
      DO 260 J = 1,3
  260 TEMPP(I) = C(I,J)*PHI(N-1,J)+TEMPP(I)
  270 D(I) = D(I)-TEMPP(I)
      DO 290 I = 1,3
      DO 280 J = 1,3
  280 PHI(N,I) = A(I,J)*D(J)+PHI(N,I)
  290 CONTINUE
      P1=PHI(N,1)
      P2=PHI(N,2)
      P3=PHI(N,3)
 9999 RETURN
      END
```

```
CTEST
      SUBROUTINE TEST
      COMMON/COM3/SMLHH(3),DUM1(360)
      COMMON/COM4/E(150,3),PHI(150,3),DUM2(78)
      COMMON/COM5/N,DUM3(5),CONST,DUM4(2),EP1,EP2,EP3,DUM5(7)
      M = 3*N-2
      TERM1 = ABS(1.-E(M,1)-E(M,2)-E(M,3)-PHI(N,1))
      TERM2 = ABS(1.-E(M+1,1)-E(M+1,2)-E(M+1,3)-PHI(N,2))
      TERM3 = ABS(1.-E(M+2,1)-E(M+2,2)-E(M+2,3)-PHI(N,3))
      IF (TERM1-EP1) 1,100,100
    1 IF (TERM2-EP2) 2,100,100
    2 IF (TERM3-EP3) 3,100,100
  100 CONST = -1.
      GO TO 9999
    3 CONST = 0.
 9999 RETURN
      END
```

```
CWNSUB
      SUBROUTINE WNSUB
      DIMENSION AAA(3),BBB(3)
      COMMON/COM3/SMLHH(3),V(50),WLST(50,3),W(50,3),TEFKSI(10)
      COMMON/COM4/E(150,3),PHI(150,3),H(3,3),EF(3,3),TEMP(3,3),DUM1(51)
      COMMON/COM5/N,NPL1,PCC,DUM2(10),EPP1,EPP2,EPP3,DUM3(3)
      NPL1=N
      DO 10 I = 1,3
   10 W(N,I) = 1.
      PCC = -1.
   35 N = N-1
      TERM1 = W(N,1)
      TERM2 = W(N,2)
      TERM3 = W(N,3)
      MN = 3*N-2
      DO 50 I = 1,3
      TEMP(3,I) = E(MN+2,I)
      TEMP(2,I) = E(MN+1,I)
   50 TEMP(1,I) = E(MN,I)
      DO 70 I = 1,3
      W(N,I) = 0.
      DO 60 J = 1,3
   60 W(N,I) = TEMP(I,J)*W(N+1,J)+W(N,I)
   70 W(N,I) = W(N,I)+PHI(N,I)
      IF (N-2) 75,75,20
   20 IF (PCC) 30,30,3
   30 IF (ABS(TERM1-W(N,1))-EPP1) 1,1,100
    1 IF (ABS(TERM2-W(N,2))-EPP2) 2,2,100
    2 IF (ABS(TERM3-W(N,3))-EPP3) 3,3,100
  100 PCC = +1.
    3 IF (N-2) 9999,9999,35
   75 DO 90 I = 1,3
      AAA(I) = 0.
      BBB(I) = 0.
      DO 80 J = 1,3
      AAA(I) = H(I,J)*W(2,J)+AAA(I)
   80 BBB(I) = EF(I,J)*W(3,J)+BBB(I)
   90 W(1,I) = BBB(I)+AAA(I)+SMLHH(I)
      GO TO 20
 9999 RETURN
      END
```

```
CVMNSUB
      SUBROUTINE VMNSUB(*)
      COMMON/COM1/DUM1(69),EKSIS,EKSIM,DKSI,TDA,TDDA,DETA2,DETA,DUM2(3),
     1EKSI
      COMMON/COM3/SMLHH(3),V(50),WLST(50,3),W(50,3),TEEKSI(10)
      COMMON/COM5/DUM3(18),NVMN
      TERM1 = EKSI*DETA/DKSI
      TERM2 = DETA/4.
      DO 10 I = 2,50
   10 V(I) = V(I-1)-(TERM1+TERM2)*(W(I,1)+W(I-1,1))+(TERM1-TERM2)*(WLST(
     1I,1)+WLST(I-1,1))
      GO TO (29,99),NVMN
   29 DO 11 I = 1,50
      DO 12 J = 1,3
   12 WLST(I,J) = W(I,J)
   11 CONTINUE
      CALL WOSUB
      GO TO 9999
   99 RETURN 1
 9999 RETURN
      END
```

01A48 C 05-22-67

```
CWOSUB
      SUBROUTINE WOSUB
      COMMON/COM1/DUM(7),Y(50),DUM1(12),EKSIS,EKSIM,DKSI,DUM2(6),MEDGE,E
     1KSI
      COMMON/COM3/SMLHH(3),V(50),WLST(50,3),W(50,3),TEEKSI(10)
      COMMON/COM5/N,NPL1,PCC,NWRIT,DUM3(15)
      IF (1-MEDGE) 1,2,1
    2 EKSI = EKSIS-.5*DKSI                                                  2
      DO 10 I = 2,NWRIT                                                     3
   10 Y(I) = Y(I-1)+1.                                                      4
      GO TO 3                                                               6
    1 EKSI = EKSI+.5*DKSI                                                   7
    3 WRITE (6,902)(EKSI)                                                   8
  902 FORMAT (1H1/5HEKSI=E16.8/)                                           11
      WRITE (6,903)                                                        11
  903 FORMAT (1H011X2HFP19X1HG16X5HTHETA18X1HV19X1HY/)                     13
      WRITE (6,904)((W(I,J),J = 1,3),V(I),Y(I),I = 1,NWRIT)                13
  904 FORMAT (1H 5E20.8)                                                   21
 9999 RETURN                                                               21
      END                                                                  22
```

APPENDIX B

The flow diagram and listing included here are for the computer program written to carry out the calculation of the initial profile. A fourth order Runge-Kutta method subroutine is employed to solve the equations described in Sections II and III.

The following equivalence between major variable names employed in the equations and in the program should be noted:

$U(1) = f = -V$

$U(2) = f'$

$U(3) = f''$

$U(4) = g$

$U(5) = g'$

$U(6) = \theta$

$U(7) = \theta'$

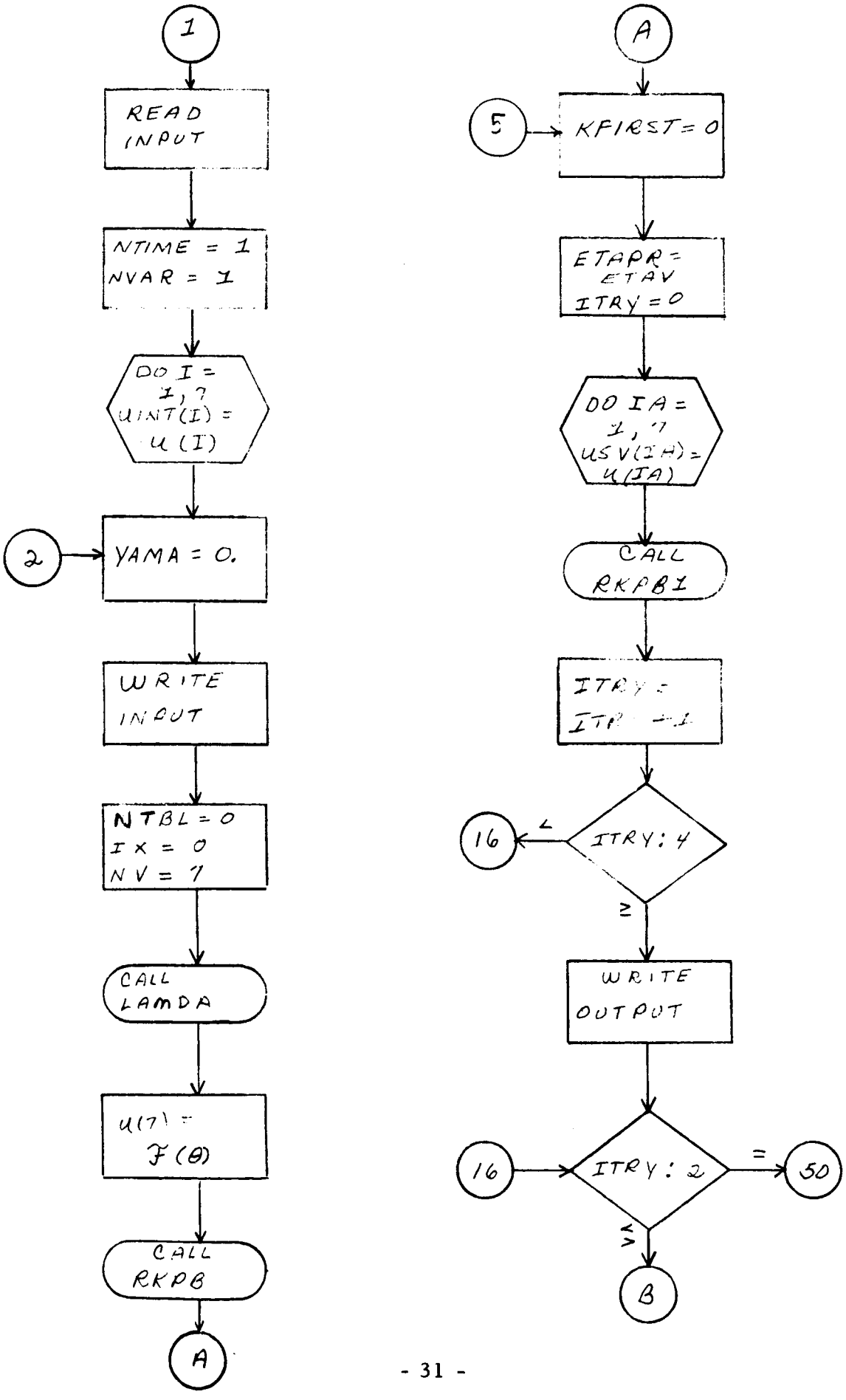
$\mathcal{F}(\ldots)$ = function of $(\ldots)$

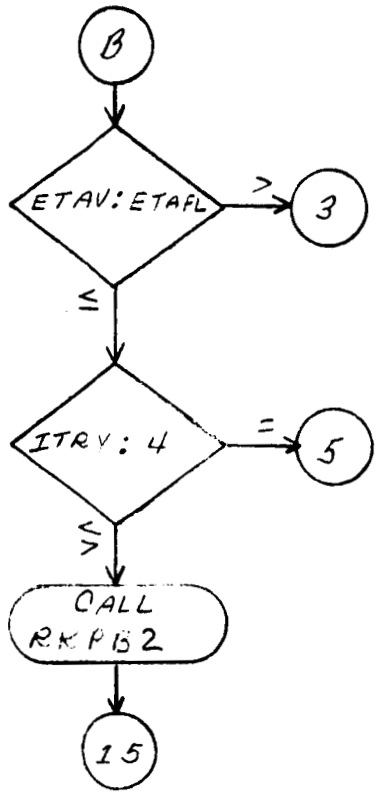
$F(3) = f''' = \mathcal{F}(f, f'', g, g')$

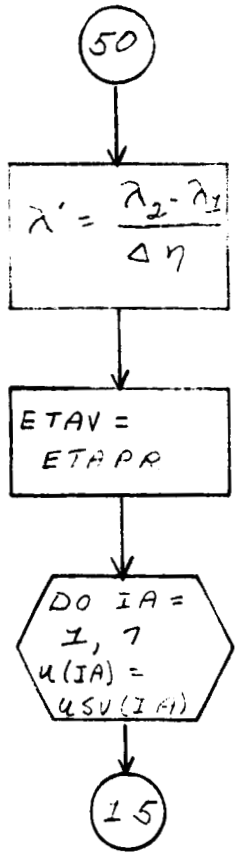
$F(5) = g'' = \mathcal{F}(f, f'', g, g', \theta, \theta')$

$F(7) = \theta'' = \mathcal{F}(f, g, g', \theta, \theta')$

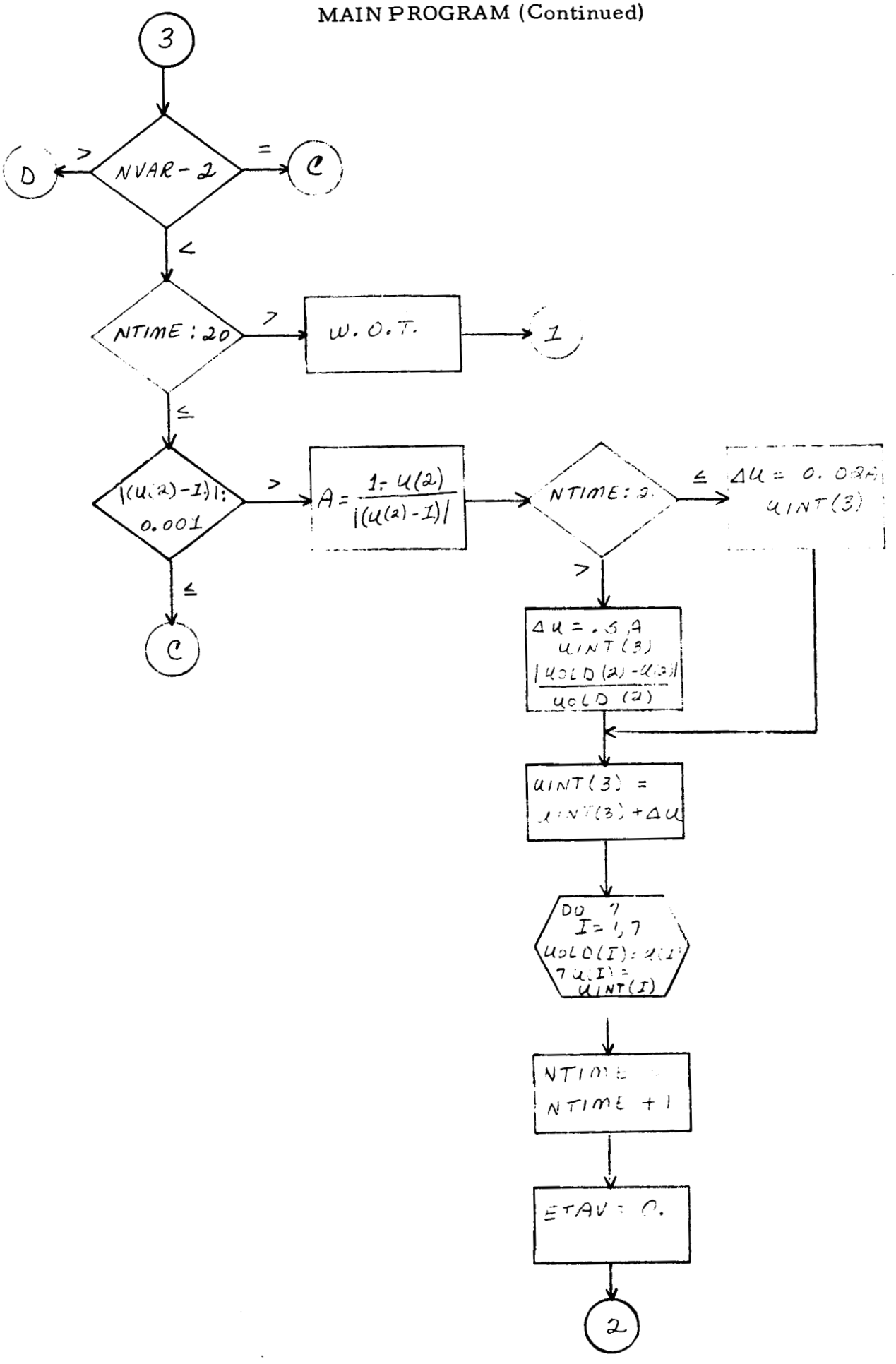
MAIN PROGRAM

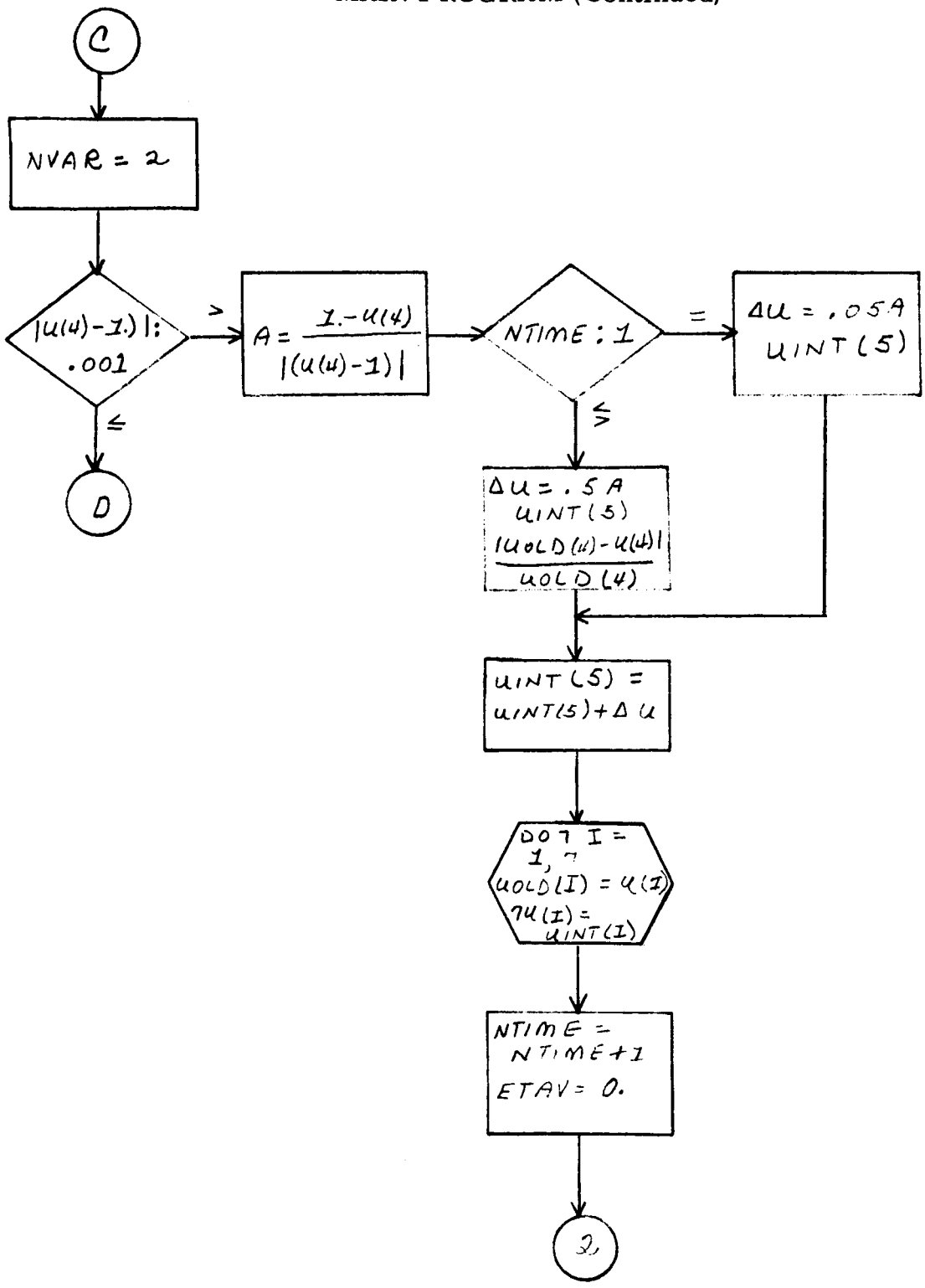


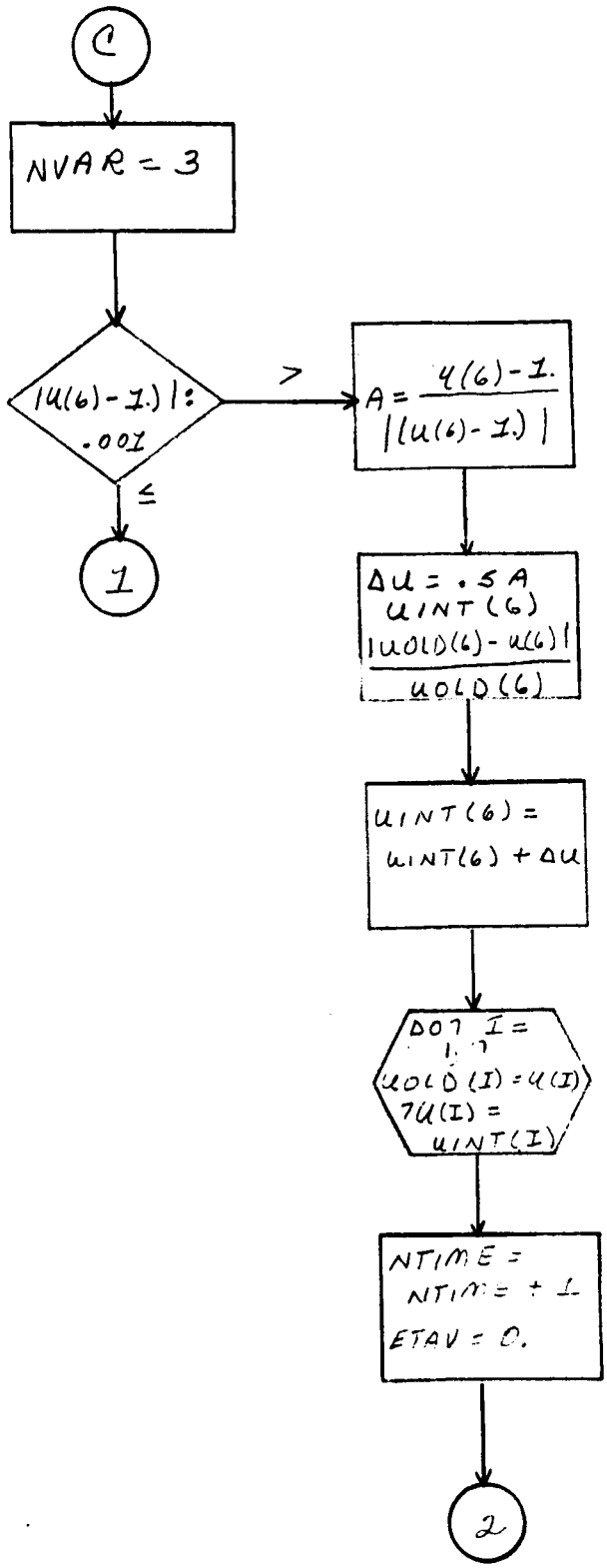

B
ETAV: ETAFL
>
3
≤
ITRV : 4
=
5
≤
>
CALL
RKPB2
15


50
$\lambda' = \frac{\lambda_2 - \lambda_1}{\Delta \eta}$
ETAV =
ETAPR
DO IA =
1, 7
u(IA) =
USV(IA)
15

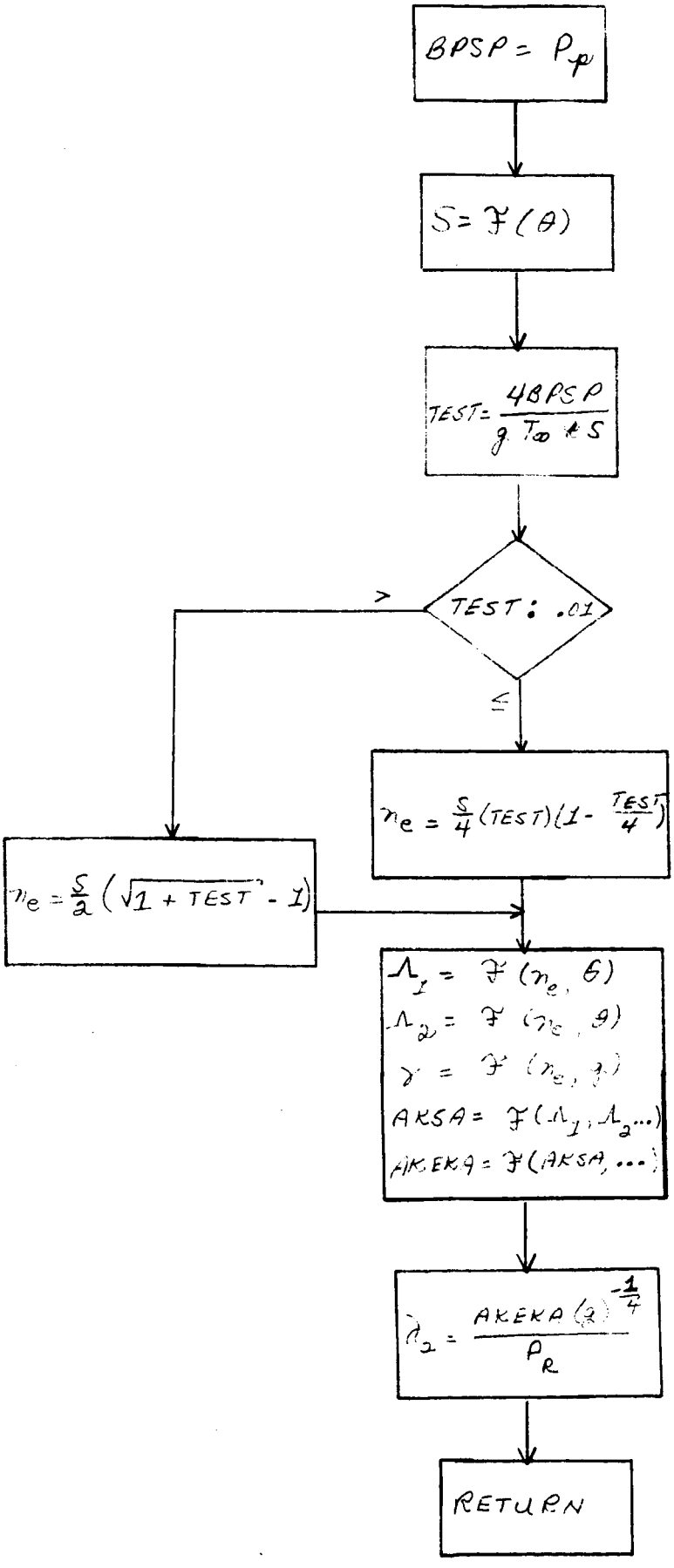
MAIN PROGRAM (Continued)



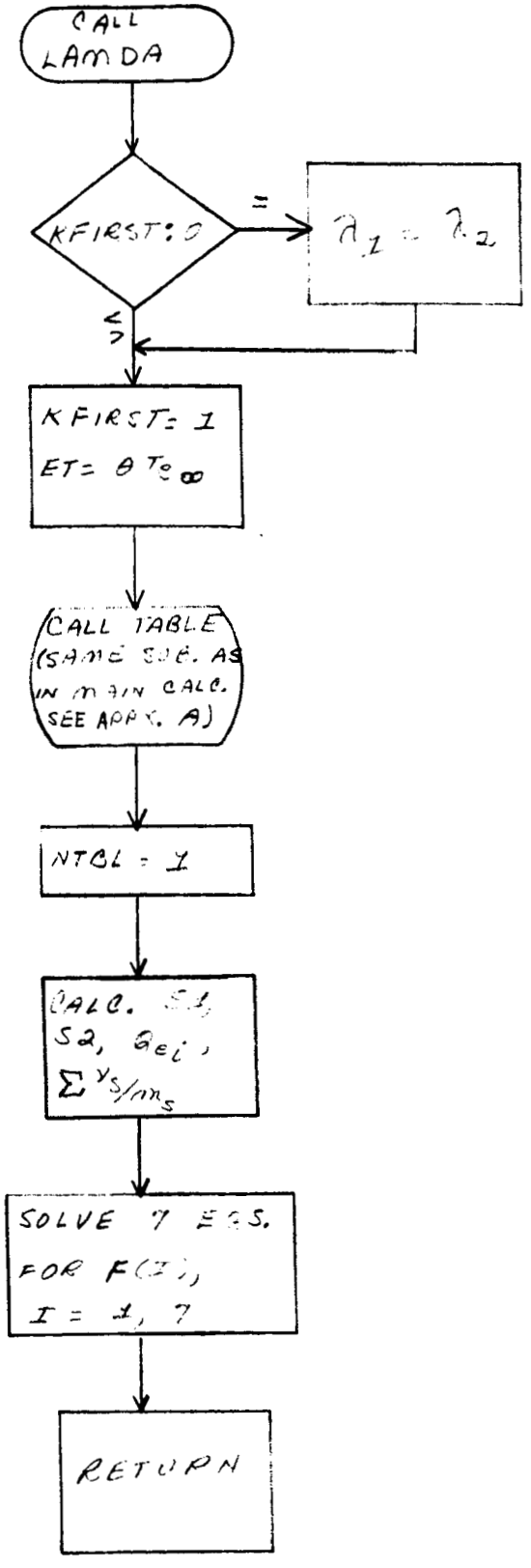

C
NVAR = 2
|U(4)-1.)| : .001
>
≤
D
A = (1.-U(4)) / |(U(4)-1)|
NTIME : 1
=
≤
>
ΔU = .05A UINT(5)
ΔU = .5A UINT(5) |UOLD(4)-U(4)| / UOLD(4)
UINT(5) = UINT(5)+ΔU
DO 7 I = 1, 7
UOLD(I) = U(I)
7U(I) = UINT(I)
NTIME = NTIME+1
ETAV = 0.
2


C
NVAR = 3
|U(6) - 1.| : .001
>
≤
1
A = U(6) - 1. / |(U(6) - 1.)|
ΔU = .5 A UINT(6) |UOLD(6) - U(6)| / UOLD(6)
UINT(6) = UINT(6) + ΔU
DO 7 I = 1, 7
UOLD(I) = U(I)
7 U(I) = UINT(I)
NTIME = NTIME + 1
ETAV = 0.
2

LAMDA


BPSP = P_p
S = $\mathcal{F}(\theta)$
TEST = $\frac{4BPSP}{g\,T_\infty\,t\,S}$
TEST : .01
>
≤
$n_e = \frac{S}{4}(TEST)(1 - \frac{TEST}{4})$
$n_e = \frac{S}{2}(\sqrt{1 + TEST} - 1)$
$\Lambda_1 = \mathcal{F}(n_e, \theta)$
$\Lambda_2 = \mathcal{F}(n_e, \theta)$
$\gamma = \mathcal{F}(n_e, \theta)$
AKSA = $\mathcal{F}(\Lambda_1, \Lambda_2 \ldots)$
AKEKA = $\mathcal{F}$(AKSA, ...)
$\lambda_2 = \frac{AKEKA\,(2)^{-\frac{1}{4}}}{P_R}$
RETURN

DERIV*



*The Runge-Kutta integration subroutine CALLS DERIV, which contains the seven first-order non-linear differential equations for simultaneous solution.

```
C MAIN          MAIN PROGRAM
      COMMON /CCOM1/ PIZ,CP,ETE,UE,CM,EM,TE,C1,C2,PER,PRN,EKS1,AKO,EC,
     1QEAQAA,ROMR,BK,PRES,PI,SM,RHOE
      COMMON/CCOM2/ U(7),F(7),YAM1,YAM2,YAMA,KFIRST
      COMMON /CCOM3/ BPSP,ET,TEST,S,ENE,NTBL
      COMMON /CCOM4/ TEMPRA,QEAT,QECST,NTBA
      DIMENSION TEMPRA(50),QEAT(50),QECST(50)
      DIMENSION USV(7),ATEMP(32)
      DIMENSION UINT(7),UOLD(7)
      EXTERNAL DERIV
      NAMELIST /INPUT/ U,DETA,ETAV,ETAFL,TEMPRA,QEAT,QECST,NTBA
      NAMELIST/OUTPUT/ YAMA,U,F,ETAV,DETA,YAM1,YAM2,ITRY
    1 READ(5,INPUT)
      NTIME = 1
      NVAR = 1
      DO 11 I=1,7
   11 UINT(I)=U(I)
    2 YAMA=0.
      WRITE(6,INPUT)
      NTBL=0
      IX=0
      NV=7
C
C     FIND LAMDA FOR THETA PRIME CALCULATION
C
      CALL LAMDA
      U(7)=4.*SQRT(2.*EKS1)*(CM/ROMR/UE*SQRT(BK*ETE/SM))*(2.+.5*ALOG(SM/
     1(2.*PI*EM)))*U(6)**1.5
      CALL RKPB(DERIV,ATEMP,ETAV,DETA,U,F,NV)
C
C     ETAV IS THE INDEPENDENT VARIABLE
C     DETA IS THE DELTA ETA
C     U IS THE DEPENDENT VARIABLE ARRAY
C     F IS THE DERIVATIVE ARRAY
C     ETAFL IS THE FINAL ETA
C
    5 KFIRST=0
C
C     KFIRST IS A CONTROL TO SAVE YAM1 INITIALLY IN DERIV
C
      ETAPR=ETAV
      ITRY=0
      DO 10 IA=1,7
   10 USV(IA)=U(IA)
   15 CALL RKPB1
      ITRY=ITRY+1
      IF (ITRY.LT.4) GO TO 16
      WRITE(6,OUTPUT)
   16 IF(ITRY.EQ.2) GO TO 50
C
C     TEST FOR FINAL ETA
C
```

```
      IF(ETAV.GE.ETAFL) GO TO 3                                         30
      IF(ITRY.EQ.4) GO TO 5                                             33
      CALL RKPB2                                                        36
      GO TO 15                                                          37
C
C     RE-CALCULATE LAMDA PRIME
C
   50 YAMA=(YAM2-YAM1)/DETA                                             38
C
C     RESET VALUES TO PREVIOUS POINT
C
      ETAV=ETAPR                                                        39
      DO 60 IA=1,7                                                      40
   60 U(IA)=USV(IA)                                                     41
      GO TO 15                                                          43
    3 IF (NVAR-2) 100,101,102                                           44
  100 IF (NTIME-20) 103,103,104                                         45
  104 WRITE (6,105)                                                     46
  105 FORMAT (34H1FAILED TO CONVERGE GO TO NEXT CASE)                   48
      GO TO 1                                                           48
  103 IF (ABS(U(2)-1.)-.001) 101,101,107                                49
  107 A = (1.-U(2))/ABS(U(2)-1.)                                        50
      IF (NTIME.GT.1) GO TO 108                                         51
      DELU = .02*A*UINT(3)                                              54
      GO TO 109                                                         55
  108 DELU = .5*A*UINT(3)*ABS(UOLD(2)-U(2))/UOLD(2)                     56
  109 UINT(3) = UINT(3)+DELU                                            57
      DO 7 I=1,7                                                        58
      UOLD(I)=U(I)                                                      59
    7 U(I) = UINT(I)                                                    60
      NTIME = NTIME+1                                                   62
      ETAV = 0.                                                         63
      GO TO 2                                                           64
  101 NVAR = 2                                                          65
      IF (NTIME-20) 111,111,104                                         66
  111 IF (ABS(U(4)-1.)-.001) 102,102,110                                67
  110 A = (1.-U(4))/ABS(U(4)-1.)                                        68
      IF (NTIME.GT.1) GO TO 115                                         69
      DELU = .05*A*UINT(5)                                              72
      GO TO 116                                                         73
  115 DELU = .5*A*UINT(5)*ABS(UOLD(4)-U(4))/UOLD(4)                     74
  116 UINT(5) = UINT(5)+DELU                                            75
      DO 8 I=1,7                                                        76
      UOLD(I) = U(I)                                                    77
    8 U(I) = UINT(I)                                                    78
      NTIME = NTIME+1                                                   80
      ETAV = 0.                                                         81
      GO TO 2                                                           82
  102 NVAR =3                                                           83
      IF (NTIME-20) 112,112,104                                         84
  112 IF (ABS(U(6)-1.)-.001) 1,1,113                                    85
  113 A = (U(6)-1.)/ABS(U(6)-1.)                                        86
```

```
      IF (NTIME.GT.1) GO TO 117
      DELU = .01*A*UINT(6)
      GO TO 118
  117 DELU = .5*A*UINT(6)*ABS(UOLD(6)-U(6))/UOLD(6)
  118 UINT(6) = UINT(6)+DELU
      DO 9 I=1,7
      UOLD(I) = U(I)
    9 U(I) = UINT(I)
      NTIME = NTIME+1
      ETAV = 0.
      GO TO 2
      END
```

```
C BLOCK      BLOCK PROGRAM
      BLOCK DATA
      COMMON /CCOM1/ PIZ,CP,ETE,UE,CM,FM,TF,C1,C2,PER,PRN,EKS1,AK0,EC,
     1QEAQAA,ROMR,BK,PRES,PI,SM,RHOE
      DATA PIZ/6.22E-19/,CP/515./,ETE/1920./,UE/395.6/,CM/6.67E-26/,
     1FM/9.107E-31/,TF/1920./,C1/2.420E21/,C2/4.49E4/,PER/.01/,
     2PRN/.666667/,EKS1/.01/,AK0/8.854E-12/,EC/1.602E-19/,QEAQAA/.1/,
     3ROMR/449.5775E-7/,BK/1.38E-23/,PRES/164000./,PI/3.1416/,
     4SM/2.2E-25/,RHOE/.5/
      END
```

```
C LAMDA      LAMDA CALCULATION
      SUBROUTINE LAMDA
      COMMON /CCOM1/ PIZ,CP,ETE,UE,CM,EM,TE,C1,C2,PER,PRN,EKS1,AK0,EC,
     1QEAQAA,ROMR,BK,PRES,PI,SM,RHOE
      COMMON/CCOM2/ U(7),F(7),YAM1,YAM2,YAMA,KFIRST
      COMMON /CCOM3/ BPSP,ET,TEST,S,ENE,NTBL
      BPSP=PER*PRES
      S=C1*ETE**(1.5)*U(6)**(1.5)*EXP(-C2/(ETE*U(6)))
      TEST=4.*BPSP/(U(4)*TE*S*BK)
      IF(TEST.GT..01) GO TO 15
      ENE=TEST*S/4.*(1.-TEST/4.)
      GO TO 20
   15 ENE=S/2.*(SQRT(1.+TEST)-1.)
   20 ALAM1=1.24E7*(ETE*U(6))**1.5/ENE**.5
      ALAM2=1.8E5*ETE*U(6)/ENE**(1./3.)
      GAMMA=BK*TE*ENE*U(4)/PRES
      AKSKA=(7.5E-7*(U(6)*ETE)**2.5/(TE*U(4))**.75)/(.25*ALOG(55.*ALAM1
     1**4+ALAM2**4))
      AKEKA=AKSKA/(1.+1.414214*1./GAMMA*AKSKA*QEAQAA*SQRT((EM/CM)*(U(4)*
     1TE)/(U(6)*ETE)))
      YAM2=AKEKA*U(4)**(-.25)/PRN
  500 RETURN
      END
```

```
C DERIV     DERIVATIVE ROUTINE
      SUBROUTINE DERIV
      COMMON /CCOM1/ PIZ,CP,ETE,UE,CM,EM,TE,C1,C2,PER,PRN,EKS1,AK0,EC,
     1QEAQAA,ROMR,BK,PRES,PI,SM,RHOE
      COMMON/CCOM2/ U(7),F(7),YAM1,YAM2,YAMA,KFIRST
      COMMON /CCOM3/ BPSP,ET,TEST,S,ENE,NTBL
      COMMON /CCOM4/ TEMPRA,QEAT,QECST,NTBA
      DIMENSION TEMPRA(50),QEAT(50),QECST(50)
      CALL LAMDA
      IF(KFIRST.EQ.0) YAM1=YAM2
      KFIRST=1
      ET=U(6)*ETE
   10 CALL TABLE(TEMPRA,QEAT,QECST,ET,NTBA,NTBL,QEA,QECS)
   15 NTBL=1
C
C     CALCULATE VARIABLES
C
   20 S1=S*(3./(2.*ETE*U(6))+C2/(ETE*U(6))**2)*(-.5+(S/2.+TEST*S/4.)/(S*
     1SQRT(1.+TEST)))
      TERM1=1./(U(4)*BK)
      S2=((4.*CM*BPSP/(5.*TE**2))*TERM1)*(TERM1/(SQRT(1.+TEST)))
      TERM1=PI*U(6)**(1.5)*(((EC/(16.*BK))*(EC/(AK0*ETE))*(1./U(6)))**2
      TERM2=ALOG((((32./EC)*(BK/EC)*(ETE*AK0/EC))*((SQRT(BK))/(SQRT(ENE)
     1)))*(SQRT(AK0*ETE)))
   30 QEI=TERM1*TERM2
      TERM1=(QEA/CM)*(PRES/(BK*TE))*(1./U(4))+(QECS/SM)*((BPSP/(BK*TE))*
     1(1./U(4))-ENE)+((QEI/SM)*ENE)
   40 SNUS=TERM1*SQRT(((8.*BK*ETE)/(PI*EM))*U(6))
C
C     CALCULATE DERIVATIVES
C
      F(1)=U(2)
      F(2)=U(3)
      F(3)=U(4)**.25*U(3)*(.25*U(4)**(-1.25)*U(5)-U(1))
      F(4)=U(5)
      CSNT1=BK*ETE*U(6)
      TERM1=YAMA*U(7)+(U(1)*U(4)*U(7)/(RHOE*CP))*(1.5*BK*ENE*(1.5*CSNT1+
     1PIZ)*S1)
      TERM2=U(5)*U(1)/(RHOE*CP*ETE)*((1.5*CSNT1+PIZ)*S2+ENE*(2.5*CSNT1+
     1PIZ))
      TERM3=(ENE*U(4)*2.*EKS1/(ROMR*UE**2))*(3.*BK*ETE/(RHOE*CP*TE))*(EM
     1*SNUS)*(U(6)-TE*U(4)/ETE)
   50 F(7)=-1./YAM2*(TERM1-TERM2-TERM3)
      TERM1=U(1)*U(5)-(U(4)**(-1.25)*U(5)**2/(4.*PRN))+(UE*U(3))**2*U(4)
     1**(-.25)/(CP*TE)
      TERM2=ETE/TE*(U(7)*YAMA+YAM2*F(7))*PIZ*((ETE/(RHOE*CP*TE))*S1*U(1)
     1*U(4)*U(7)-S2*U(1)*U(4)*U(5)/RHOE+ENE*U(1)*U(5)/(RHOE*CP*TE))
      F(5)=-PRN*U(4)**(.25)*(TERM1*TERM2)
      F(6)=U(7)
  500 RETURN
      END
```