LEVEL II

AFGL-TR-78-0258

DESIGN AND ANALYSIS OF SINGLE CELL BALLOONS

James L. Rand

Texas A&M University
Department of Aerospace Engineering
College Station, Texas 77843

Scientific Report No. 1

August 1978

AIR FORCE GEOPHYSICS LABORATORY
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
HANSCOM AFB, MASSACHUSETTS 01731

APPENDIX A

Balloon Design Program

BALDE

```
LEVEL  21                    MAIN                DATE = 79101          12/54/45
C*****************************************************************************
C                              - B A L D E -
C                          BALLOON DESIGN PROGRAM
C*****************************************************************************
C      THIS PROGRAM CALCULATES THE SHAPE OF A FREE BALLOON GIVEN A SET OF
C      INPUT CONDITIONS.  IT IS FOR FULLY TAILORED BALLOONS ONLY. WITH OR
C      WITHOUT AN ENDCAP.  CONVERGENCE OF THE SOLUTION IS OBTAINED BY
C      MATCHING THE TOP LOAD REACTION TO THE TOP LOAD.  THIS PROGRAM
C      INCLUDES THE EFFECTS OF THE LOAD TAPES IN THE CALCULATIONS.
C*****************************************************************************
C      THIS PROGRAM IS BASED ON THE PROGRAMS "BALLOON" AND "FLATBALL",
C      AFCRL-65-92, "DETERMINATION OF THE SHAPE OF A FREE BALLOON", BY
C      J. H. SMALLEY.
C*****************************************************************************
C      DESCRIPTION OF DATA DECK:
C        *  EACH SET OF DATA CONSISTS OF THREE CARDS LISTING THE VALUES
C           IN THE READ STATEMENTS 500 AND 501
C           ALL DATA ARE READ IN 10 WIDE FIELDS OF THE APPROPRIATE TYPE.
C         1 "FIRST" CARD INPUTS ARE AS FOLLOWS:
C              PAYLOAD IN POUNDS
C              ALTITUDE OPTION: 1 = ALT IN FT, 2 = ALT IN MB
C              ALTITUDE IN FEET OR MILLIBARS AS DESIRED
C              FILM TYPE:  1 = POLYETHYLENE, 2 = MYLAR
C              FILM THICKNESS IN INCHES
C              LOAD TAPE TYPE:  1 = POLYESTER, 2 = KEVLAR
C              TAPE LOAD RATING IN POUNDS
C              NUMBER OF LOAD TAPES
C         2 "SECOND" CARD INPUTS ARE AS FOLLOWS:
C              TOP LOAD IN POUNDS, (+) UP, (-) DOWN
C              STRESS CONSTANT TAU0 (USUALLY 0.00)
C              STRESS CONSTANT TAU1 (USUALLY 0.00)
C              SUPERPRESSURE (0.00 FOR NATURAL SHAPE)
C              PRINT INCREMENT N (0 FOR STANDARD OUTPUT)
C              NON-DIMENSIONAL GORE INCREMENT DSO
C              NON-DIMENSIONAL GORE LENGTH TO CAP STARTING LOCATION CSTART
C              FILM THICKNESS INCLUDING CAP IN INCHES
C         3 "THIRD" CARD INPUTS ARE AS FOLLOWS:
C              OUTPUT CONTROL (KEY2=2 FOR PUNCHED DECK OF SHAPE & WEIGHT
C                              KEY2=1 FOR DISK FILE OF SHAPE & WEIGHT)
C              OUTPUT CONTROL (MPT IS NUMBER OF POINTS IN LOAD-ALTITUDE)
C              IDENTIFY LIFTING GAS (FOR HELIUM, IGAS=1)
C              MINIMUM RECOMMENDED PAYLOAD IN POUNDS
C              MAXIMUM RECOMMENDED PAYLOAD IN POUNDS
C        *    IF NO ENDCAP, MAKE SURE THAT CSTART IS GREATER THAN ANY
C           ANTICIPATED GORE LENGTH.
C        *    THE PROGRAM REQUIRES A LAST DATA SET WITH P = 0.0 TO
C           TERMINATE
C*****************************************************************************
C      DESCRIPTION OF OUTPUT:
C           FOR EACH CASE THE INPUT DATA WILL FIRST BE PRINTED, FOLLOWED
C           BY A RECORD OF THE ITERATIONS REQUIRED FOR CONVERGENCE OF THE
C           SOLUTION.  THIS RECORD CONSISTS OF A DISPLAY OF INITIAL AND
C           FINAL ANGLES OF THE GORE WRT THE VERTICAL AXIS OF THE BALLOON.
C           DURING THE FINAL ITERATION, PERTINENT VALUES ALONG THE GORE ARE
C           PRINTED OUT CONTINUOUSLY, FOLLOWED BY A LISTING OF FINAL VALUES.
C*****************************************************************************
C      APPROPRIATE ERROR MESSAGES ARE SUPPLIED IF THE SOLUTION DOES NOT
C      CONVERGE
```

```
EVEL  21                    MAIN                DATE = 78226          21/29/42
*****************************************************************************

*****************************************************************************
**********                  MAIN PROGRAM                  **********************
*****************************************************************************
      REAL K,KT,KTAPE,LAMBDA,IAM
      INTEGER CODEF,CODET
      DIMENSION GP(100),GW(100),GWT(100),GT(100)
  100 WRITE(6,600)
  600 FORMAT('1')
      READ(5,500) P,KEY,CONST,CODEF,FTHICK,CODET,TLR,NT
  500 FORMAT(F10.0,I10,F10.0,I10,F10.4,I10,F10.0,I10)
      READ(5,501) TL,TAU0,TAU1,ALPHA,N,DSO,CSTART,CAP
  501 FORMAT(F10.2,F10.2,F10.2,F10.2,I10,F10.2,F10.2,F10.4)
C-----------------------------------------------------------------------
C     IF NO MORE DATA, EXIT.
      IF(P) 200,300,200
C-----------------------------------------------------------------------
  200 CONTINUE
      READ(5,502)KEY2,MPT,IGAS,PMIN,PMAX
  502 FORMAT(3I10,2F10.0)
      CALL DESIGN (P,KEY,CONST,CODEF,FTHICK,CODET,TLR,NT,TL,TAU0,TAU1,
     $ALPHA,N,DSO,CSTART,CAP,GP,GW,WBAL,KEY2,S)
C
C              COMPUTE LOAD - ALTITUDE CURVE
C
      WRITE(6,600)
      WRITE(6,601)
  601 FORMAT(50X,'LOAD - ALTITUDE DATA',/42X,'GROSS AIRBORN',/37X,'
     1WEIGHT - (KG)',10X,'ALTITUDE - (KM)')
      GPMIN=1.+WBAL
      GPMAX=GPMIN+(PMAX-PMIN)/P
      DG=(GPMAX-GPMIN)/MPT
      GP1=GPMIN
  250 CONTINUE
      CALL GLNGTH(GP1,SLMDA,1)
      B1=P/(S/SLMDA)**3/.06243
      CALL BOYNCY(B1,PA,IGAS,TCHIGH)
      CALL ATMOS2(PA,H,TA,RHOA,B,IAM,GXPAN,TOHIGH)
      GROS=GP1*P*.454
      H=H/1000.
      WRITE(6,602)GROS,H
  602 FORMAT(42X,E15.7,8X,E15.7)
      GP1=GP1+DG
      IF(GP1.GT.GPMAX)GO TO 100
      GO TO 250
  300 CONTINUE
      STOP
      END
```

```
 EVEL  21                    DESIGN            DATE = 78226          21/29/42

      SUBROUTINE DESIGN (P,KEY,CONST,CODEF,FTHICK,CODET,TLR,NT,TL,TAU0,
     $TAU1,ALPHA,N,DSC,CSTART,CAP,GP,GW,WBAL,KEY2,S)
.*************************************************************************
.*****************      SUBROUTINE DESIGN      ************************
:*************************************************************************
      REAL K,KT,KTAPE,NU
      REAL LAMBDA
      INTEGER CODEF,CODET
      DIMENSION Y(6),G(6),G(6),E(4),C(4),D(4),WORD1(3),WORD2(3)
      DIMENSION GP(100),GW(100),GWT(100),GT(100)
      DIMENSION SSTORE(300),GSTORE(300),WSTORE(300),TSTORE(300)
      REAL F1(3)/'POLY','ETHY','LENE'/
      REAL F2(3)/'MYLA','R   ','    '/
      REAL T1(3)/'POLY','ESTE','R   '/
      REAL T2(3)/'KEVL','AR  ','    '/
      REAL K1/'  FT'/
      REAL K2/'  MB'/
C
C     B, C, AND D ARE CONSTANTS USED IN GILL'S MODIFIED R-K METHOD FOR
C     SOLVING THE GOVERNING DIFFERENTIAL EQUATIONS.
      DATA B/2.0,1.0,1.0,2.0/, C/0.5,0.29289,1.7071,0.5/, D/0.5,
     $0.29289,1.7071,0.16667/
C
      CALL MATL(P,KEY,CONST,CODEF,FTHICK,CODET,TLP,NT,CAP,SIGMA,CSIGMA,
     $TSIGMA,KT,EM,EC,EMC,LAMBDA,TSIGME)
      NU=EMC/EM
      OMEGA1=SIGMA/1.8453
      OMEGA2=OMEGA1*(CSIGMA/SIGMA)
      OMEGA=OMEGA1+TSIGMA/1.8453
      TOMEGA=TSIGMA/6.2832
C
C     EMPIRICAL CURVE FITS FROM SMALLEY'S REPORT, TO GET GOOD FIRST ESTIMAT
C     OF THETA0.  GOOD FOR SIGMA BETWEEN 0.0 AND 0.8 .
      THETA0=1.347
      IF(OMEGA.LT.0.43354) THETA0=0.72*OMEGA+1.1015
      IF(OMEGA.LT.0.35225) THETA0=1.15*OMEGA+0.9507
      IF(OMEGA.LT.0.27096) THETA0=1.46*OMEGA+0.8687
      OMEGA=OMEGA1
C
      IQUIT = 0
      LSTRUN=1
      IF(CODEF.EQ.2) GO TO 52
      DO 51   I=1,3
   51 WORD1(I)=F1(I)
      GO TO 54
   52 DO 53   I=1,3
   53 WORD1(I)=F2(I)
   54 IF(CODET.EQ.2) GO TO 56
      DO 55   I=1,3
   55 WORD2(I)=T1(I)
      GO TO 58
   56 DO 57   I=1,3
   57 WORD2(I)=T2(I)
   58 IF(KEY.EC.2) GO TO 59
      WORD3=K1
      GO TO 60
   59 WORD3=K2
   60 CONTINUE
```

```
EVEL  21                    DESIGN              DATE = 78226          21/29/42

      WRITE(6,601) (WCRD1(I),I=1,3),FTHICK,SIGMA,TAU0,(WORD2(I),I=1,3),
     $EM,TAU1,TLR,EC,ALPHA
  601 FORMAT(5X,37('*')/5X,'*',35X,'*',15X,'NON-DIMENSIONAL QUANTITIES',
     18X,'MISC. PARAMETERS'/5X,'*  FILM TYPE',8X,': ',A4,A4,A4,'   *'/5X,
     2'*  FILM THICKNESS   : ',F7.4,' IN',4X,'*',17X,'SIGMA FILM =',F9.4
     3,12X,'TAUC  =',F6.3/5X,'*  TAPE TYPE',8X,': ',A4,A4,A4,'   *',17X,
     4'EM',9X,'=',F8.3,13X,'TAU1  =',F6.3/5X,'*  TAPE LOAD RATING : ',
     5F6.0,'  LBS    *',17X,'EC',9X,'=',F8.3,13X,'ALPHA =',F6.3)
      WRITE(6,602) NT,EMC,DSC,P,TSIGMA,CSTART,TL,KT,N,CONST,WORD3
  602 FORMAT(' ',4X,'*  NUMBER OF TAPES   : ',I5,9X,'*',17X,'EMC',8X,
     1'=',F8.3,13X,'DS0   =',F6.3/5X,'*  PAYLOAD',10X,': ',F6.0,'  LBS
     2 *',17X,'SIGMA TAPE =',F9.4,12X,'CSTART=',F5.2/5X,'*  (INCL TOPLOA
     3D CF): ',F6.0,'  LBS    *',17X,'KT',9X,'=',F6.1,15X,'N      =',I2/5X
     4,'*  DESIGN ALTITUDE  : ',F7.0,A4,'    *'/5X,'*',35X,'*'/5X,37('*')
     5,///5X, 'ITERATION RECORD (INITIAL AND FINAL ANGLES)'//)
C-----------------------------------------------------------------------
C     IF THE SOLUTION IS NOT CONVERGING, EXIT
   21 IGUIT = IGUIT+1
      IF(IGUIT.LT.20) GO TO 22
      WRITE(6,603)
  603 FORMAT('-',' *** SOLUTION IS NOT CONVERGING - MAXIMUM NUMBER OF IT
     $ERATIONS HAS BEEN EXCEEDED *** ')
      GO TO 5
C
C     LSTRUN = 0 IS THE KEY FOR LAST ITERATION.  IT IS AN OUTPUT CONTROL.
   22 IF(LSTRUN.NE.0)GO TO 23
C-----------------------------------------------------------------------
      WRITE(6,604)
  604 FORMAT('0',95X,27HMANUFACTURER'S TABLE LAYOUT//10X,'S',13X,'R',
     113X,'Z',13X,'T',11X,'TAUM',9X,'TAUC',8X,'GORE POSITION (FT)',5X,
     2'HALF GORE WIDTH (IN)'/)
   23 DO 24 I=1,6
   24 Q(I)=0.
  231 DO 241 I=1,300
      SSTORE(I)=0.0
      WSTORE(I)=0.
      TSTORE(I)=0.
  241 GSTORE(I)=0.0
      JCOUNT=1
      DS = DS0
      SSTOP = CSTART
      OMEGA=OMEGA1
      TOMEGA=TSIGMA/6.2832
      TW=0.
      ITAG=0
      RC=0.
      GW2=0.
      S0=0.
      S0DIM=0.
      R=0.
      S=0.
      Z=0.
      WSUM=0.
      ASUM=0.
      TAUM=0.
      TAUC=TAU0
      THETA=THETAC
      RT0=1./6.2832/COS(THETA)
```

```
EVEL  21                    DESIGN              DATE = 78226          21/29/42

      Y(1)=THETA
      Y(2)=R
      Y(3)=Z+ALPHA
      Y(4)=((F+TL)/P)/6.2832/COS(THETA)
      Y(5)=C.C
      Y(6)=C.C
      SSTORE(JCCUNT) = SODIM
      GSTORE(JCCUNT) = GW2
    2 T=6.2832*Y(4)
      IF(LSTRUN)34,32,34
   32 WRITE(6,605) S,R,Z,T,TAUM,TAUC,SODIM,GW2
  605 FORMAT(4(4X,F10.5),4X,F8.3,5X,F8.3,12X,F7.1,17X,F7.2)
      IF(KEY2.EQ.1)WRITE(3,650)Y(2),Y(3),Y(4),Y(1),Y(6)
      IF(KEY2.EQ.2)WRITE(7,650)Y(2),Y(3),Y(4),Y(1),Y(6)
  650 FORMAT(5E15.7)
C-----------------------------------------------------------------------
C     ICOUNT AND N ARE OUTPUT CONTROLS USED DURING THE LAST ITERATION.
C     N IS THE NUMBER OF INCREMENTS THAT WILL BE SKIPPED IN THE PRINTOUT.
C
   34 ICOUNT=N
C-----------------------------------------------------------------------
C     ITAG SENDS THE PROGRAM INTO ITERATIONS AND RELEASES IT WHEN THE TOP
C     OF THE BALLOON HAS BEEN REACHED
      IF(ITAG-1)4,3,4
C-----------------------------------------------------------------------
    3 IF(LSTRUN.EQ.C)GO TO 35
      DEGO = THETAO*57.2956
      THETAD = THETA*57.2956
      DS = DSO
      WRITE(6,611) DEGO,THETAD
  611 FORMAT(' ',5X,F10.5,5X,F10.5)
C-----------------------------------------------------------------------
C     CONVERGENCE TEST: TOP LOAD REACTION WITHIN .5 LB OF APPLIED TOP LOAD.
      F = -6.2832*Y(4)*COS(THETA)*P
      IF(ABS(F+TL).LE.0.5C) LSTRUN=0
C
C     COMPUTE CORRECTIVE TERM ON THETAO
C     DELTA(THETAO)/DELTA(THETA) IS GENERALLY APPROX.= (+1/-2)
C     DTHETA IS THE DESIRED VALUE OF THETA TO MATCH THE TOP LOAD REACTION
C     TO THE APPLIED TOP LOAD.
      DTHETA = -(ARCOS(TL/(6.2832*P*Y(4))))
      DELTA = DTHETA - THETA
      CORR = -DELTA/2.
      THETAO = THETAO+CORR
C
      GO TO 21
C-----------------------------------------------------------------------
C
C     COMPUTE AND PRINT FINAL QUANTITIES
   35 VOL = 3.1416*Y(6)*(LAMEDA**3)
      ASUM = (ASUM+6.2832*Y(5))*(LAMBDA**2)
      WSUM = (WSUM+6.2832*UMEGA*Y(5))*P
      TW=TW*6.2832*P
      WEIGHT = WSUM+TW
      F = -6.2832*Y(4)*COS(THETA)*P
      THETAD=57.2956*THETA
      DEGO = THETAO*57.2956
      JMAX = JCCUNT
```

```
      PTOP=-TL/F
      WBAL=WEIGHT/P
      WRITE(6,606) LAMBDA,VOL,DEGO,ASUM,THETAD,WSUM,F,TW,WEIGHT
  606 FORMAT('-',4X,'FINAL QUANTITIES (LAMBDA =',F7.1,' FT):'//5X,'VOLUM
     1E',6X,'=',E10.3,' FT3',10X,'INITIAL ANGLE',5X,'=',F7.3,' DEG'/5X,
     2'AREA',8X,'=',E10.3,' FT2',10X,'FINAL ANGLE',7X,'=',F7.3,' DEG'
     3/5X,'FILM WEIGHT =',F7.0,'     LBS',10X,'TOP LOAD REACTION =',F7.3,
     4' LBS'/5X,'TAPE WEIGHT =',F7.0,'     LBS'/5X,'TOT. WEIGHT =',F7.0,
     5'     LBS'/)
      IF(KEY2.EQ.1)WRITE(3,650)F,PTOP,WBAL,S,LAMBDA
      IF(KEY2.EQ.2)WRITE(7,650)F,PTOP,WBAL,S,LAMBDA
      IF(KEY2.EQ.1)WRITE(3,651)NU,KT,EM,NT
      IF(KEY2.EQ.2)WRITE(7,651)NU,KT,EM,NT
  651 FORMAT(3E15.7,I15)
C
C     GENERATE TRANSFER MATRICES ... 100 EVENLY SPACED, UNDEFORMED
C     CORE COORDINATES
C
      DSNEW = SSTORE(JMAX)/99.
      J=1
      SNEW=0.0
      GP(J)=0.0
      GW(J)=0.0
      GWT(J)=TOMEGA
      GT(J)=TOMEGA*6.2832/NT
      WRITE(6,600) J,GP(J),J,GW(J),J,GWT(J),J,GT(J)
  100 SNEW = SNEW+DSNEW
      J = J+1
      DO 101 I=1,JMAX
      QUAN = SNEW-SSTORE(I)
      IF(QUAN.LT.0.0) GO TO 102
  101 CONTINUE
  102 GP(J) = SNEW
      GW(J) = GSTORE(I-1)+((SNEW-SSTORE(I-1))*(GSTORE(I)-GSTORE(I-1))/
     $(SSTORE(I)-SSTORE(I-1)))
      GWT(J)= WSTORE(I-1)+((SNEW-SSTORE(I-1))*(WSTORE(I)-WSTORE(I-1))/
     $(SSTORE(I)-SSTORE(I-1)))
      GT(J) = TSTORE(I-1)+((SNEW-SSTORE(I-1))*(TSTORE(I)-TSTORE(I-1))/
     $(SSTORE(I)-SSTORE(I-1)))
      WRITE(6,600) J,GP(J),J,GW(J),J,GWT(J),J,GT(J)
  600 FORMAT(' ',5X,'GP(',I3,') = ',F10.3,10X,'GW(',I3,') =',F10.3,10X,
     1'GWT(',I3,') = ',F10.3,10X,'GT(',I3,') = ',F10.3)
  652 FORMAT(4E15.7)
      IF(J.EQ.100) GO TO 103
      GO TO 100
  103 CONTINUE
      DO 300 J=1,JMAX
      IF(KEY2.EQ.1)WRITE(3,652)SSTORE(J),GSTORE(J),WSTORE(J),TSTORE(J)
      IF(KEY2.EQ.2)WRITE(7,652)SSTORE(J),GSTORE(J),WSTORE(J),TSTORE(J)
  300 CONTINUE
C
      GO TO 5
C-----------------------------------------------------------------------
C     COMPUTATIONAL PORTION
C-----------------------------------------------------------------------
C
C     CHECK IF OVER THE SHOULDER OF THE BALLOON
    4 IF(SIN(THETA))6,8,8
```

```
 EVEL  21                    DESIGN              DATE = 78226          21/29/42

      IF SO, CHECK IF WITHIN DS OF THE TOP
    6 IF(R-CS)7,7,8

      IF SO, REDEFINE DS, KEY WITH ITAG
    7 DS=R/ABS(SIN(THETA))
      ITAG=1
C-----------------------------------------------------------------------
      PROGRAM SOLVING STARTS HERE
C
C     SET UP FOR GILL'S METHOD
    8 DO 10 J=1,4
      G(2)=SIN(Y(1))
      G(3)=COS(Y(1))
      G(1)=(TAUC*G(3)-Y(2)*Y(3)-G(2)*Y(2)*OMEGA-TOMEGA*G(2))/Y(4)
      G(4)=TAUC*G(2)+G(3)*(OMEGA*Y(2)+TOMEGA)
      G(5)=Y(2)
      G(6)=Y(2)*Y(2)*G(3)
C
C     GILL'S METHOD
      DO 9 I=1,6
      P1 = B(J)*(G(I)-B(J)*Q(I))
      Y(I) = Y(I)+DS*P1
    9 Q(I) = Q(I)+3.0*P1-C(J)*G(I)
C
   10 CONTINUE
C-----------------------------------------------------------------------
      TW=TW+TOMEGA*DS
      THETA=Y(1)
      R=Y(2)
      Z=Y(3)-ALPHA
      S=S+DS
      JCOUNT = JCOUNT+1
      IF(R) 161,161,16
  161 R=1.E-10
   16 TAUC=TAU0+(TAU1*(Y(4)-RTO))/R
      EPSM=(Y(4)/R-EMC*TAUC/EC)/(EM-EMC*EMC/EC+KT/R/6.2832)
      EPST=EPSM
      TAUM=(Y(4)-KT*EPST/6.2832)/R
      RO=R/(1.+(TAUC-(EMC*EPSM)/EC))
      GW2=3.1416*RO*LAMBDA*12./NT
      SO=SO+(DS/(1.+EPSM))
      SODIM=SO*LAMBDA
      SSTORE(JCOUNT) = SODIM
      GSTORE(JCOUNT) = GW2
      WSTORE(JCOUNT)=Y(2)*OMEGA+TOMEGA
      TSTORE(JCOUNT)=TOMEGA*6.2832/NT
C-----------------------------------------------------------------------
C     IF TAUC < 0, BUCKLING OCCURS, SO SET VALUES TO 0
C
      IF(TAUC)31,17,17
   31 TAUC=0.
      TAU0=0.0
      TAU1=0.0
C-----------------------------------------------------------------------
C     CHECKS FOR IMPOSSIBLE BALLOONS.
C     IF |THETA|>PI OR S>10.0, PRINT AN APPROPRIATE ERROR MESSAGE AND
C     PROCEED TO THE NEXT SET OF DATA
```

```
EVEL  21                    DESIGN              DATE = 78226          21/29/42

   17 IF((ABS(THETA)-3.1416).LT.0.0) GO TO 12
      WRITE(6,607)
  607 FORMAT('-',' *** SOLUTION IS NOT CONVERGING - IMPOSSIBLE BALLOON :
     $ |THETA| > PI *** ')
      GO TO 5
   12 IF((S-10.0).LT.0.0) GO TO 30
      WRITE(6,608)
  608 FORMAT('-',' *** SOLUTION IS NOT CONVERGING - IMPOSSIBLE BALLOON :
     $ NON-DIM S > 10.0 *** ')
      GO TO 5
C-----------------------------------------------------------------------
C     THIS PORTION ALLOWS FOR THE CHANGE IN FILM THICKNESS DUE TO THE CAP
C
   30 IF(S-SSTOP)14,13,13
   13 T=6.2832*Y(4)
  131 ASUM=ASUM+6.2832*Y(5)
      WSUM=WSUM+6.2832*OMEGA*Y(5)
      Y(5)=0.
      IF(LSTRUN.NE.0) GO TO 132
  609 FORMAT('0',3X,'CAP STARTS AT S =',F6.2,  ' ... SIGMA CAP =',F7.4/)
      WRITE(6,609) CSTART,CSIGMA
  132 OMEGA = OMEGA2
      TOMEGA=TSIGMB/6.2832
      SSTOP = 10.0
   14 IF(ICOUNT)2,2,15
   15 ICOUNT=ICOUNT-1
      IF(ITAG)2,4,2
    5 CONTINUE
      S=S*LAMBDA
      RETURN
      END
```

```
EVEL  21                    MATL                    DATE = 78226          21/29/42

      SUBROUTINE MATL(P,KEY,CONST,CODEF,FTHICK,CODET,TLR,NT,CAP,SIGMA,
     $CSIGMA,TSIGMA,KT,EM,EC,EMC,LAMECA,TSIGMB)
C*************************************************************************
C     THIS SUBROUTINE CALCULATES MATERIAL PROPERTIES GIVEN INFORMATION
C     AS FOLLOWS:
C        INPUTS:  PAYLOAD, ALTITUDE OPTION KEY, ALTITUDE, FILM TYPE AND
C                 THICKNESS, TAPE TYPE AND LOAD RATING, NUMBER OF TAPES, AND
C                 CAP THICKNESS
C        OUTPUTS:  NON-DIMENSIONAL FILM AND TAPE WEIGHTS, TAPE STIFF-
C                 NESS, AND FILM MODULI.
C*************************************************************************
C     ALL EMPIRICAL RELATIONSHIPS USED ARE SUBJECT TO REVISION WHEN
C     ADDITIONAL TEST DATA BECOME AVAILABLE.
C*************************************************************************
C     THIS SUBROUTINE ASSUMES, FOR LACK OF A GOOD TEMPERATURE MODEL, THAT
C     THE BALLOON MATERIAL TEMPERATURE IS ATMOSPHERIC TEMPERATURE.
C     THIS ASSUMPTION MAY NOT BE ADEQUATE.
C*************************************************************************
      REAL K,KT,KTAPE,LAMBDA
      REAL MG,MA,LF,IAM
      INTEGER CODEF,CODET
C-------------------------------------------------------------------------
C     100 SERIES - DEFINE NECESSARY VALUES
C-------------------------------------------------------------------------
      IF(KEY.EQ.2) GO TO 101
  100 H=CONST
C
      H=H*0.3048
C     CONVERSION OF H FROM FEET TO METERS FOR SUBRT ATMOS
C
      CALL ATMOS(H,TA,PA,RHOA,B,IAM,GXPAN,TOHIGH)
      GO TO 102
  101 PA=CONST
      CALL ATMOS2(PA,H,TA,RHOA,B,IAM,GXPAN,TOHIGH)
C
  102 B=B*6.243E-02
C     CONVERSION OF B FROM KG/M3 TO LBM/FT3
C
      T=TA-273.16
C     CONVERSION OF TEMPERATURE FROM DEG(K) TO DEG(C)
C
      LAMBDA = (P/B)**(1./3.)
      K = 1./((6.283185)**(1./3.))
C-------------------------------------------------------------------------
C     200 SERIES - COMPUTE FILM PROPERTIES
C-------------------------------------------------------------------------
      IF(CODEF.EQ.2) GO TO 202
  201 WFILM = 58.06*(FTHICK/12.)
      EFILM = (0.02*(T**2)-1.60*(T)+41.95)*1000.
      GO TO 203
  202 WFILM = 87.08*(FTHICK/12.)
C
C     EFILM = ?????      MYLAR FILM DATA IS NECESSARY
C
  203 SIGMA = WFILM/(K*B*LAMBDA)
      CSIGMA=SIGMA*(CAP/FTHICK)
C
C     THIS PROGRAM ASSUMES A POISSON'S RATIO OF 0.50
```

```
C
      EM = (EFILM*FTHICK*12.)/(B*(LAMBCA**2))
      EC = EM
      EMC = EM/2.
C-----------------------------------------------------------------------
C     300 SERIES - COMPUTE TAPE PROPERTIES
C-----------------------------------------------------------------------
      IF(CODET.EQ.2) GO TO 302
  301 WTAPE = (6.1E-06)*TLR+0.0038+.3024*(.003+FTHICK)
      WTAPE2=WTAPE+.3024*(CAP-FTHICK)
      KTAPE = (TLR*10.)*(1.-(6.67E-03)*(T-20.))
      GO TO 303
  302 WTAPE = (2.0E-06)*TLR+0.0021
      KTAPE = 26.*TLR
C
C     NO INFORMATION ON LOW TEMPERATURE EFFECTS CURRENTLY AVAILABLE
C
  303 TSIGMA = (NT*WTAPE)/(E*(LAMBDA**2))
      TSIGMB=TSIGMA*WTAPE2/WTAPE
      KT = (NT*KTAPE)/(B*(LAMBDA**3))
      RETURN
      END
```

```
LEVEL  21                    ATMOS                    DATE = 78226          21/29/42

      SUBROUTINE ATMOS(H,TA,PA,RHOA,B,IAM,GXPAN,TOHIGH)
C
C ..... SUBROUTINE FOR SOLVING FOR THE VALUES OF TEMPERATURE,
C ..... PRESSURE, DENSITY, SPECIFIC BOUYANCY, INTEGRATED AIR
C ..... MASS, AND "GAS EXPANSION" FOR ANY GIVEN ALTITUDE BELOW
C ..... 61000. METERS (200131. FEET).  ALL EQUATIONS HAVE BEEN
C ..... DERIVED ACCORDING TO THE U.S. STANDARD ATMOSPHERE, 1962.
C ..... --- ALL UNITS ARE IN THE SI SYSTEM, I.E.
C ..... TEMPERATURE(DEG KELVIN), PRESSURE(MB), DENSITY(KG/M3),
C ..... SPECIFIC BOUYANCY(KG/M3) INTEGRATED AIR MASS(KG/M2),
C ..... GAS EXPANSION(DIMENSIONLESS).--- LIFTS WERE BASED ON
C ..... GRADE A HELIUM AND PURE HYDROGEN.
C
C         FOR SPECIFIC BOUYANCY, TO CONVERT FROM KG/M3 TO LB/FT3,
C         MULTIPLY BY 0.06243
C
C         H IS INPUT IN METERS
C
      REAL MG,MA,LP,IAM
      TOHIGH =1
      MG=4.0026
      MA=28.9644
      R=8.31432E03
      GY=9.80665
      RHOSTP=1.2250
      IGAS=1
C     IF HYDROGEN GAS IS USED INSTEAD OF HELIUM,IGAS MUST BE CHANGED TO
C     SOMETHING OTHER THAN 1
C
      IF(H.LT.61000.0) GO TO 10
      WRITE(6,155) H
  155 FORMAT('0',5X,'***** ALTITUDE H OUTSIDE RANGE OF SUBROUTINE ATMOS.
     1  H =',E14.7,' METERS. ******')
      TOHIGH=-1.
      GO TO 28
   10 IF (H.LT.52000.0)GO TO 12
      HB=52000.0
      TB=270.65
      LP=-0.0028
      PB=0.590
      RHOB=0.0007594
      GO TO 22
   12 IF(H.LT.47000.0) GO TO 14
      HB=47000.0
      TB=270.65
      LP=0.0
      PB=1.105
      RHOB=0.0014275
      GO TO 22
   14 IF(H.LT.32000.0) GO TO 16
      HB=32000.0
      TB=228.65
      LP=0.0028
      PB=8.680
      RHOB=0.013225
      GO TO 22
   16 IF(H.LT. 20000.0 )GO TO 18
      HB=20000.0
```

```
      TB=216.65
      LP=0.0010
      PB=54.749
      RHOB=0.088035
      GO TO 22
   18 IF (H.LT.11000.0)GO TO 20
      HB=11000.0
      TB=216.65
      LP=0.0
      PB=226.320
      RHOB=0.36392
      GO TO 22
   20 HB=0.0
      TB=288.15
      LP=-0.0065
      PB=1013.250
      RHOB=1.2250
   22 TA=TB+LP*(H-HB)
      IF(LP.EQ.0)GO TO 24
      PA=PB/(LP*(H-HB)/TB+1.0)**(GY*MA/LP/R)
      GO TO 26
   24 PA=PB/EXP(GY*MA/R/TB*(H-HB))
   26 RHOA= PA*MA/R/TA*100.0
      IF(IGAS.NE.1) MG=2.0159
      B=RHOA*(1.0-MG/MA)
      IAM=10.0*PA*(1.03751-0.00527*ALOG10(PA))
      GXPAN=RHOSTP/RHOA
   28 RETURN
      END
```

```
LEVEL  21                    ATMOS2              DATE = 78226          21/29/42

      SUBROUTINE ATMOS2(PA,H,TA,RHOA,B, IAM,GXPAN,TOHIGH)
C
C ..... SUBROUTINE FOR SOLVING FOR THE VALUES OF TEMPERATURE,
C ..... ALTITUDE, DENSITY, SPECIFIC BOUYANCY, INTEGRATED AIR
C ..... MASS, AND "GAS EXPANSION" FOR ANY GIVEN ALTITUDE BELOW
C ..... 61000. METERS (200131. FEET).  ALL EQUATIONS HAVE BEEN
C ..... DERIVED ACCORDING TO THE U.S. STANDARD ATMOSPHERE, 1962.
C ..... --- ALL UNITS ARE IN THE SI SYSTEM, I.E.
C ..... TEMPERATURE(DEG KELVIN), PRESSURE(MB), DENSITY(KG/M3),
C ..... SPECIFIC BOUYANCY(KG/M3) INTEGRATED AIR MASS(KG/M2),
C ..... GAS EXPANSION(DIMENSIONLESS).--- LIFTS WERE BASED ON
C ..... GRADE A HELIUM AND PURE HYDROGEN.
C
C        FOR SPECIFIC BOUYANCY, TO CONVERT FROM KG/M3 TO LB/FT3,
C        MULTIPLY BY 0.06243
C
C        PA IS INPUT IN MB
C
      REAL MG,MA,LP,IAM
      TOHIGH=1
      MG=4.0026
      MA=28.9664
      R=8.31432E03
      GY=9.80665
      RHOSTP=1.2250
      IGAS=1
      IF(PA.LT.226.32) GO TO 10
      PB=1013.25
      HB=0.0
      TB=288.15
      LP=-0.0065
      RHOB=1.225
      GO TO 22
   10 IF(PA.LT.54.749) GO TO 12
      PB=226.32
      HB=11000.
      TB=216.65
      LP=0.0
      RHOB=0.36392
      GO TO 22
   12 IF(PA.LT.8.680) GO TO 14
      PB=54.749
      HB=20000.
      TB=216.65
      LP=0.001
      RHOB=0.088035
      GO TO 22
   14 IF(PA.LT.1.109) GO TO 16
      PB=8.680
      HB=32000.
      TB=228.65
      LP=0.0028
      RHOB=0.013225
      GO TO 22
   16 IF(PA.LT.0.590) GO TO 18
      PB=1.109
      HB=47000.
      TB=270.65
```

```
.EVEL  21                    ATMOS2              DATE = 78226          21/29/42

      LP=0.0
      RHOB=0.0014275
      GO TO 22
   18 IF(PA.LT.0.1828) GO TO 20
      PB=0.59
      HB=52000.
      TB=270.65
      LP=-0.002
      RHOB=0.0007594
      GO TO 22
   20 WRITE(6,155) PA
      WRITE(6,155) PA
  155 FORMAT('0',5X,'***** ATMOSPHERIC PRESSURE PA OUTSIDE RANGE OF SUBR
     1OUTINE ATMOS2.  PA =',E14.7,' MB. *****')
      TCHIGH=-1.
      GO TO 28
   22 IF(LP.EQ.0.0) GO TO 24
      H=HB+TB/LP*((PB/PA)**(LP*R/GY/MA)-1.0)
      GO TO 26
   24 H=ALOG(PB/PA)*R*TE/GY/MA+HB
   26 TA=TB+LP*(H-HB)
      RHOA=PA*MA/R/TA*100.0
      IF(IGAS.NE.1) MG=2.0159
      B=RHOA*(1.0-MG/MA)
      IAM=10.0*PA*(1.03751-0.00527*ALOG10(PA))
      GXPAN=RHOSTP/RHOA
   28 RETURN
      END
```

```
.EVEL  21                    BOYNCY              DATE = 78226          21/29/42

      SUBROUTINE BOYNCY(B,PA,IGAS,TCHIGH)
C
C
C     ..... SUBROUTINE TO SOLVE FOR ATMOSPHERIC PRESSURE (PA) GIVEN THE
C     ..... CORRESPONDING GAS BOUYANCY (B) AT ANY ALTITUDE BELOW 61,000
C     ..... METERS (200,131 FEET).  ALL EQUATIONS HAVE BEEN DERIVED ACCORDING
C     ..... TO THE U.S. STANDARD ATMOSPHERE, 1962.
C     ..... ----- ALL UNITS ARE IN THE SI SYSTEM:
C     .....
C     ..... SPECIFIC BOUYANCY (B) IS THE INPUT IN KG/M3
C     ..... ATMOSPHERIC PRESSURE (PA) IS THE OUTPUT IN MB
C     .....
C     .....CALCULATIONS ARE BASED ON PURE HYDROGEN AND GRADE A HELIUM.
C     ..... FOR HYDROGEN, INPUT IGAS AS 0.
C     ..... FOR HELIUM, INPUT IGAS AS 1.
C     .....
C     ..... TCHIGH WILL BE SET TO  -1  IF BOUYANCY IS OUTSIDE THE PARAMETERS
C     ..... OF THE STANDARD ATMOSPHERE.
C
C
C
      REAL MW
      TCHIGH=1
      MW=4.0026
      B1=1.0557
      B2=0.31355
      B3=0.075858
      B4=0.011396
      B5=1.2308E-3
      B6=6.5444E-4
      B7=2.3635E-4
      IF(IGAS.EQ.1)GO TO 90
      MW=2.0159
      B1=1.1397
      B2=0.33853
      B3=0.081899
      B4=0.012303
      B5=1.3288E-3
      B6=7.0656E-4
      B7=2.5517E-4
   90 IF(B.LT.B1)GO TO 1
      WRITE(6,101) B
  101 FORMAT('0','********** VALUE OF B IS OUTSIDE LIMITS OF STANDARD AT
     1MOSPHERE.  B = ',E12.5,' (KG/M3) **********')
      TCHIGH=-1.
      GO TO 40
    1 IF(B.LE.B2)GO TO 2
      TB=288.15
      XLP=-0.0065
      PB=1013.250
      GO TO 9
    2 IF(B.LE.B3)GO TO 3
      TB=216.65
      XLP=0.0
      PB=226.320
      GO TO 9
    3 IF(B.LE.B4)GO TO 4
      TB=216.65
```

```
      XLP=0.0010
      PB=54.749
      GO TO 9
    4 IF(B.LE.85)GO TO 5
      TB=228.65
      XLP=0.0028
      PB=8.680
      GO TO 9
    5 IF(B.LE.86)GO TO 6
      TB=270.65
      XLP=0.0
      PB=1.109
      GO TO 9
    6 IF(B.LE.87)GO TO 7
      TB=270.65
      XLP=-0.0020
      PB=0.590
      GO TO 9
    7 WRITE(6,101) B
      TCHIGH=-1.
      GO TO 40
    9 ITTER=0
      PNEW=PB
   10 POLD=PNEW
      ITTER=ITTER+1
      FPA=POLD-(2.87C531*B*TB/(1.-MW/28.9644))*(1.+((PB/POLD)**(29.2713*
     1XLP)-1.))
      GPA=1.+(84.0241*B*TB*XLP*PB/((1.-MW/28.9644)*POLD**2))*(PB/POLD)**
     1(29.2713*XLP-1.)
      PNEW=POLD-FPA/GPA
      IF(ABS((PNEW-POLD)/PNEW).LE..00001)GO TO 20
      IF(ITTER.GE.50)GO TO 30
      GO TO 10
   30 DELPA=PNEW-POLD
      WRITE(6,100) DELPA
  100 FORMAT('0',10X,'SOLUTION DID NOT CONVERGE AFTER 50 ITTERATIONS.'
     1/5X,'DIFFERENCE BETWEEN NEW AND OLD PRESSURE ON LAST ITTERATION WA
     2S',2X,E10.3)
      TCHIGH=-1.
   20 PA=PNEW
   40 RETURN
      END
```

```
LEVEL  21                    GLNGTH              DATE = 78226          21/29/42

      SUBROUTINE GLNGTH(GP,SLMDA,INDIC)
C
C
C ..... SUBROUTINE TO SOLVE FOR THE GORE LENGTH RELATIONS OF ZERO-PRESSUR
C ..... NATURAL-SHAPE, FLAT-TOP BALLOONS.  TO OBTAIN GORE LENGTH/LAMBDA
C ..... (SLMDA) FROM GROSS/PAYLOAD (GP) INITIALIZE INDIC TO 1.  TO OBTAIN
C ..... THE REVERSE OF THE ABOVE, INTIALIZE INDIC TO ZERO.
C .....
C ..... THE FOLLOWING IS A CURVE-FIT OF FIGURE 5 IN NCAR-TN/IA-99, SECTIO
C
      IF(INDIC.EQ.0)GO TO 60
      IF(GP.GE.1.)GO TO 10
   40 WRITE(6,100) GP
  100 FORMAT('0','********** VALUE OF GROSS/PAYLOAD OUTSIDE RANGE OF SUB
     1ROUTINE GLNGTH.  GROSS/PAYLOAD(GP) = ',E12.5,'**********')
      INDIC=-1
      GO TO 95
   10 IF(GP.GE.1.1458)GO TO 11
      A=1.99442
      B=0.274845
      GO TO 50
   11 IF(GP.GE.1.3205)GO TO 12
      A=1.99287
      B=0.280545
      GO TO 50
   12 IF(GP.GE.1.5306)GO TO 13
      A=1.989503
      B=0.286635
      GO TO 50
   13 IF(GP.GE.1.7838)GO TO 14
      A=1.98507
      B=0.291881
      GO TO 50
   14 IF(GP.GE.2.0897)GO TO 15
      A=1.97853
      B=0.297581
      GO TO 50
   15 IF(GP.GE.2.4586)GO TO 16
      A=1.97175
      B=0.302239
      GO TO 50
   16 IF(GP.GE.2.9035)GO TO 17
      A=1.96328
      B=0.307023
      GO TO 50
   17 IF(GP.GE.3.4380)GO TO 18
      A=1.95482
      B=0.311076
      GO TO 50
   18 IF(GP.GE.4.0778)GO TO 19
      A=1.94521
      B=0.315067
      GO TO 50
   19 IF(GP.GE.4.8380)GO TO 20
      A=1.93701
      B=0.318071
      GO TO 50
   20 IF(GP.GE.5.7363)GO TO 21
```

```
      A=1.92864
      B=0.320819
      GO TO 50
   21 IF(GP.GE.6.7898)GO TO 22
      A=1.92034
      B=0.323288
      GO TO 50
   22 IF(GP.GE.8.0155)GO TO 23
      A=1.91325
      B=0.325220
      GO TO 50
   23 IF(GP.GT.10.)GO TO 40
      A=1.90667
      B=0.326875
   50 SLMDA=A*GP**B
      GO TO 99
   60 IF(SLMDA.GE.1.99442)GO TO 70
   91 WRITE(6,200) SLMDA
  200 FORMAT('0','********** VALUE OF GORE LENGTH/LAMBDA OUTSIDE RANGE O
     1F SUBROUTINE GLNGTH.  SLMDA = ',E12.5,'**********')
      INDIC=-1
      GO TO 99
   70 IF(SLMDA.GE.2.07044)GO TO 71
      A=1.99442
      B=0.274845
      GO TO 95
   71 IF(SLMDA.GE.2.15453)GO TO 72
      A=1.99287
      B=0.280545
      GO TO 95
   72 IF(SLMDA.GE.2.24767)GO TO 73
      A=1.989503
      B=0.286635
      GO TO 95
   73 IF(SLMDA.GE.2.35038)GO TO 74
      A=1.98507
      B=0.291881
      GO TO 95
   74 IF(SLMDA.GE.2.46373)GO TO 75
      A=1.97853
      B=0.297581
      GO TO 95
   75 IF(SLMDA.GE.2.58781)GO TO 76
      A=1.97175
      B=0.302239
      GO TO 95
   76 IF(SLMDA.GE.2.72339)GO TO 77
      A=1.96328
      B=0.307023
      GO TO 95
   77 IF(SLMDA.GE.2.87037)GO TO 78
      A=1.95482
      B=0.311076
      GO TO 95
   78 IF(SLMDA.GE.3.02894)GO TO 79
      A=1.94521
      B=0.315067
      GO TO 95
```

```
LEVEL  21                          GLNGTH                DATE = 78226      21/29/42

   79 IF(SLMDA.GE.3.19819)GO TO 80
      A=1.93701
      B=0.318071
      GC TC 95
   80 IF(SLMDA.GE.3.37780)GC TC 81
      A=1.92864
      B=0.320819
      GO TO 95
   81 IF(SLMDA.GE.3.56703)GO TO 82
      A=1.92034
      B=0.323288
      GC TC 95
   82 IF(SLMDA.GE.3.76484)GC TC 83
      A=1.91325
      B=0.325220
      GO TO 95
   83 IF(SLMDA.GT.4.0)GO TO 91
      A=1.90667
      B=0.326875
   95 GP=(SLMDA/A)**(1./E)
   99 RETURN
      END
```

```
//$OPTIONS
C*****************************************************************************
C                          - B A L A N -
C                      BALLOON ANALYSIS PROGRAM
C*****************************************************************************
C        THIS PROGRAM IS INTENDED TO COMPUTE THE SHAPE AND STRESS
C        DISTRIBUTIONS OF A BALLOON AT ANY ALTITUDE WHERE THE FILM
C        MAY BE CONSIDERED TO LOBE BETWEEN TAPES.  THE MANUFACTURED
C        RADIUS AND WEIGHT DISTRIBUTIONS ARE REQUIRED INPUTS.  THE
C        DESIGN SHAPE PARAMETERS ARE NEEDED TO START THE ITERATION
C        PROCESS.  THE OUTPUT CONSISTS OF THE DEFORMED SHAPE PARA-
C        METERS, MERIDIONAL STRESS, CIRCUMFERENTIAL STRESS, SHEAR
C        STRESS, PRINCIPAL STRESSES, AND TENSION FIELD PATTERNS
C        WHERE APPLICABLE.
C        ALL QUANTITIES ARE NONDIMENSIONALIZED WITH RESPECT TO
C        SMAX, THE UNDEFORMED GORE LENGTH, AND PD, THE DESIGN
C        PAYLOAD.
C*****************************************************************************
      1       IMPLICIT REAL*8(A-H,O-Z)
      2       DIMENSION A(300),C(1801),F(1801),X(1801),RO(300),WT
     1(300),GP(300),WTAPE(300),SMP(300),SCP(300)
      3       REAL*8   NU,KBAR,LAMBDA
      4       ATAN(Z)=DATAN(Z)
      5       ARSIN(Z)=DARSIN(Z)
      6       ARCOS(Z)=DARCOS(Z)
      7       SIN(Z)=DSIN(Z)
      8       COS(Z)=DCOS(Z)
      9       TAN(Z)=DTAN(Z)
     10       READ(5,800) PD,PTOP,WBAL,SMAX,LAMBDA
     11       READ(5,801) NU,KBAR,EBAR,N
     12       READ(5,805) KMAX,ABAR,BBAR
     13       READ(5,802) (GP(K),RO(K),WT(K),WTAPE(K),K=1,KMAX)
     14       SMAX=GP(KMAX)
     15       SOL=SMAX/LAMBDA
     16       IMAX=6*KMAX+1
     17       DO 10 K=1,KMAX
     18       L=6*(K-1)+1
     19       READ(5,803) (X(L+I), I=2,6)
     20       X(L+1)=0.0
     21    10 CONTINUE
     22       PI=4.*ATAN(1.0D0)
     23       PON=PI/N
     24       X(1)=1.-PTOP
     25       BBAR=BBAR*SOL**3
     26       KBAR=KBAR/2./PI
     27       EBAR=EBAR*SOL
     28       VTOP=X(IMAX)
     29       DO 20 K=1,KMAX
     30       L=6*(K-1)+1
     31       X(L+2)=X(L+2)/SOL
     32       X(L+3)=X(L+3)/SOL
     33       X(L+6)=(VTOP-X(L+6))/SOL**3*PI
     34       GP(K)=GP(K)/SMAX
     35       RO(K)=RO(K)*N/12./PI/SMAX
     36       WT(K)=WT(K)*SOL
     37       WTAPE(K)=WTAPE(K)*SOL
     38    20 CONTINUE
     39       WRITE(6,903)(K,GP(K),RO(K),WT(K),WTAPE(K),K=1,KMAX)
     40       SPON=SIN(PON)
     41       CPON=COS(PON)
```

```
      S2PON=SPON*SPON
      DS=1./(KMAX-1)
      ZPA=ABAR
      DO 3 MNN=1,5
  100 DO 200 K=1,KMAX
      IF(K.EQ.50)EBAR=4.6*EBAR
      EM=EBAR/(1.-NU**2)
      EC=EM
      EMC=EC*NU
      D=EM*EC-EMC**2
      DM=EC/D
      DC=EM/D
      DMC=-EMC/D
      L=6*(K-1)+1
      STH=SIN(X(L+5))
      CTH=COS(X(L+5))
      SB=SIN(X(L+1))
      CB=COS(X(L+1))
      ALFA=ARSIN(SPON*CTH)
      IF(ZPA.LT.0.)ALFA=-ALFA
      IF(CTH.LT.0.)ALFA=-ALFA
      ABET=ALFA+X(L+1)
      PHI=2.*ABET
      SBPON=SIN(ABET)
      F1R=RO(K)*DC/DMC+(DC*DM/DMC-DMC)*KBAR
      IF(K.EQ.1)THP=(X(12)-X(6))/DS
      IF(K.EQ.KMAX)THP=(X(L+5)-X(L-1))/DS
      IF(K.EQ.1) GO TO 102
      IF(K.EQ.KMAX) GO TO 102
      THP=(X(L+1)-X(L-1))/2./DS
  102 CONTINUE
      ZPA=X(L+3)+ABAR
      SM=(X(L+4)-KBAR*DMC*ABET/PON*(RO(K)*BBAR*ZPA+STH*(WT(K)
     1-WTAPE(K)/PON/2.)))/(RO(K)+KBAR*DM+KBAR*DMC*ABET/PON*
     2RO(K)*THP)
      EF=X(L+4)-RO(K)*SM
      IF(EF.LT.0.)SM=X(L+4)/RO(K)
      IF(EF.LT.0.)EF=0.
      FTB=SPON*CB*CTH+SB*(CPON*CTH*CTH-1.+STH*STH)
      SC=ABET/PON*(RO(K)*SM*THP                    +RO(K)*BBAR
     1*ZPA+STH*(WT(K)-WTAPE(K)/PON/2.))
      SMP(K)=SM
      A(K)=(X(L+4)-RO(K)*SM)*2.*PON
  105 CONTINUE
      IF(SC.LT.0.)SC=0.
      SCP(K)=SC
      EMP1=1.+DM*SM+DMC*SC
      ECP1=1.+DMC*SM+DC*SC
      F1B=SPON*CB+SB*CTH*(CPON-1.)
      F2B=FTB/PON
      IF(EF.EQ.0.)BET=ARSIN(-PON*STH*WTAPE(K)/SC)
      IF(EF.EQ.0.)STUFF=TAN(BET)
      IF(EF.EQ.0.)GO TO 104
      RATIO=RO(K)*SM/(X(L+4)-RO(K)*SM)
      TOP=BBAR*ZPA*X(L+2)*EMP1*SPON+PON*(WT(K)-WTAPE(K)
     1/PON/2.)*STH-SC*SPON*CTH*CB-RATIO*WTAPE(K)*STH/2.
      BOT=CB*(RATIO*SC+SC*(CPON*CTH*CTH+STH*STH))
      IF(SC.EQ.0.)STUFF=99999999
      IF(SC.EQ.0.)GO TO 104
      STUFF=TOP/BOT
  104 CONTINUE
```

```
      ASEG=(PHI-SIN(PHI))*(SPUN/SHPUN)**2*CTH
      IF(ZPA.EQ.0.)SIGN=1.
      IF(ZPA.EQ.0.) GO TO 101
      SIGN=ZPA/DABS(ZPA)
  101 ASUBN=N*(SIN(2*PUN)+SIGN*ASEG)
      NN=1
      JJ=0
      IF(K.EQ.1) GO TO 170
      IF(K.EQ.KMAX)GO TO 180
      IF(K.EQ.2) GO TO 160
  159 IF(K.GE.2)JJ=6
      IF(K.GE.2)NN=2
  160 CONTINUE
      C(L+1)=ATAN(STUFF)
      IF(K.EQ.KMAX-1)GO TO 161
      C(L+2+JJ)=X(L-4)+NN*DS*STH*EMP1
      C(L+3+JJ)=X(L-3)+NN*DS*CTH*EMP1
      C(L+4+JJ)=X(L-2)+NN*DS*(WT(K)*CTH+SC*STH/PUN*F1B)
      IF(JJ.EQ.6)GO TO 161
      IF(K.EQ.2) GO TO 159
  161 C(L+5-JJ)=X(L+11)+NN*DS/X(L+4)*(WT(K)*STH+BBAR*ZPA*X(L+2)*EMP1
     1*SPUN/PUN-SC/PUN*FTB)
      C(L+6-JJ)=X(L+12)+NN*DS*EMP1*X(L+2)*X(L+2)*ASUBN*CTH/2.
      IF(JJ.EQ.0) GO TO 200
      IF(K.EQ.KMAX-1)NN=1
      IF(K.EQ.KMAX-1)JJ=0
      IF(K.EQ.KMAX-1) GO TO 161
      GO TO 200
  170 CONTINUE
      C(1)=BBAR*X(7)-PTOP-WBAL
      C(2)=ATAN(STUFF)
      C(3)=RO(1)
      C(4)=0.
      C(5)=X(1)/CTH/2./PI
      C(6)=X(12)+DS/X(5)*(STH*WT(K)+BBAR*ZPA*X(3)*EMP1*SPUN/PUN
     1-SC*F2B)
      C(7)=X(13)+DS*EMP1*X(3)*X(3)/2.*CTH*ASUBN
      GO TO 200
  180 CONTINUE
      C(L+1)=ATAN(STUFF)
      C(L+2)=RO(KMAX)
      C(L+3)=X(L-3)+DS*CTH*EMP1
      C(L+4)=X(L-2)+DS*(WT(K)*CTH+SC*STH/PUN*F1B)
      C(L+5)=-PI/2.
      PR=BBAR*ZPA*RO(KMAX)**2*PI
      PSUP=PTOP-PR
      IF(PSUP.GT.0.)C(L+5)=-ARCOS((PR-PTOP)/2./PI/X(L+4))
      IF(PSUP.LT.0.)C(L+5)=ARCOS((-PR+PTOP)/2./PI/X(L+4))
      C(L+6)=0.
  200 CONTINUE
      EBAR=EBAR/4.0
      RMS=0.
      SIGMA=.50
      DO 210 I=1,IMAX
      F(I)=C(I)-X(I)
      X(I)=X(I)+SIGMA*F(I)
      RMS=RMS+F(I)*F(I)
  210 CONTINUE
    2 CONTINUE
      DO 1 KL=1,KMAX
      IF(KL.EQ.1) WRITE(6,13) KL,X(KL),F(KL)
```

```
158           WRITE(6,7) KL,(X(I+6*KL-5),I=1,5),(F(I+6*KL-5),I=1,5),
             1SMP(KL),SCP(KL)
159          13 FORMAT(I5,2F10.6)
160           7 FORMAT(I5,12F10.6)
161           1 CONTINUE
162             CHECK=DABS(F(1)/X(1))
163             RMS=RMS**.5
164             WRITE(6,900)RMS
165             HT=BBAR/SOL**3
166             VOL=X(7)*SMAX**3
167             PAY=X(1)*PD
168           3 CONTINUE
169             WRITE(6,908) PAY,LAMBDA,EBAR,VOL,NU,ABAR,SMAX,
             1KBAR,HT,N,KMAX
170             CALL PRINT (KMAX,A,SMP,SCP,X,WTAPE,DS)
C            FORMAT STATEMENTS FOR INPUT
C
171        800 FORMAT(5E15.7)
172        801 FORMAT(3E15.7,I15)
173        802 FORMAT(4E15.7)
174        803 FORMAT(5E15.7)
175        805 FORMAT(I15,2E15.7)
C
C            FORMAT STATEMENTS FOR OUTPUT
C
176        900 FORMAT(1X,'RMS VALUE = ',E20.13)
177        903 FORMAT(14,4E15.7)
178        908 FORMAT(1H1,10X,'DESIGN VALUES',/,10X,'PAYLOAD=',6X,F7.2,
             1' LBS',10X,'LAMBDA=',F4.0,5X,'EBAR=',F7.2,/,10X,'VOLUME=',
             2E14.4,' FT(3)',8X,'NU=',F8.2,5X,'ABAR=',F5.2,/,10X,'CORE LENGTH=',
             3F9.2,' FT',11X,'KBAR=',F6.2,5X,'BBAR=',F5.2,/,10X,
             4'NUMBER OF TAPES=',I5,/,10X,'NUMBER OF POINTS=',I4,//)
179        999 STOP
180             END

181             SUBROUTINE PRINT (KMAX,F,SMP,SCP,X,WTAPE,DS)
182             IMPLICIT REAL*8(A-H,O-Z)
183             DIMENSION F(300),SCP(300),SMP(300),X(1801),WTAPE(300)
184             IMAX=6*KMAX+1
185             WRITE(6,801)
186             WRITE(6,800) (X(I),I=2,IMAX)
187             WRITE(6,900)
188             PI=4.*DATAN(1.D0)
189             DO 105 K=1,KMAX
C              CALCULATE SHEAR STRESS
C
190             CTH=DCOS(X(6*K))
191             IF(K.EQ.1) GO TO 101
192             IF(K.EQ.KMAX) GO TO 102
193             TAU=(F(K+1)-F(K-1))/4./DS-WTAPE(K)*CTH/2.
194             GO TO 103
195        101  TAU=(-3.*F(K)+4.*F(K+1)-F(K+2))/4./DS-WTAPE(K)*CTH/2.
196             GO TO 103
197        102  TAU=(F(K-2)-4.*F(K-1)+3.*F(K))/4./DS-WTAPE(K)*CTH/2.
198        103  SP=SMP(K)+SCP(K)
C              CHECK TO SEE IF A TENSION FIELD EXISTS
C
199             IF(SCP(K).LE.0.) GO TO 104
200             TAUMAX=DSQRT(((SMP(K)-SCP(K))/2.D0)**2+TAU**2)
201             S1=SP/2.+TAUMAX
202             S2=SP/2.-TAUMAX
```

```
      ANG=DATAN(2.D0*TAU/(SMP(K)-SCP(K)))/2.
      IF(SCP(K).GT.SMP(K)) ANG=ANG+PI/2.D0
      GO TO 105
  104 ANG=-DATAN(SMP(K)/TAU)-PI/2.D0
      S1=SMP(K)/DCOS(ANG)/DCOS(ANG)
      S2=0.
      TAUMAX=SMP(K)/2.D0
  105 WRITE(6,901) K,F(K),SMP(K),SCP(K),TAU,S1,S2,TAUMAX,ANG
C
C       FORMAT STATEMENTS FOR OUTPUT
C
  800 FORMAT(8F10.4)
  801 FORMAT(//,4X,'BETA',7X,'R',9X,'Z',9X,'T',8X,'THETA'
     1,5X,'VOL',//)
  900 FORMAT(1H1,43X,'MER.',6X,'CIR.',6X,'MER.',4X,'PRINCIPAL STRESSES'
     1,4X,'MAX',/,29X,'K',5X,'F',7X,'STRESS',4X,'STRESS',4X,'SHEAR'
     2,7X,'S1',8X,'S2',6X,'SHEAR',5X,'ANGLE',//)
  901 FORMAT(26X,I4,8F10.4)
      RETURN
      END
```