รายการอ้างอิง

<u>ภาษาไทย</u>

1. สุภร ศิริคุรุรัตน์ . <u>การประยุกต์แบบจำลองทางคณิตศาสตร์ในการออกแบบผิวจราจรยืดหยุ่น</u>. วิทยานิพนธ์ปริญญามหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย, 2523.

<u>ภาษาอังกฤษ</u>

1. Carl, Erik Froberg. <u>Introduction To Numerical Analysis</u>. 2 nd ed. Massachusetts: Addision-wesley Publiching Company., 1974.
2. Clessen, A.I.M., Edwward, J.M., Sommer, P., and Uge, P. Aspalt Pavement Design - The Shell Method. <u>Proceeding of the Forth International Conference on The Structural Design of Asphalt pavement</u>, pp. 39-73. University of Michigan, 1977.
3. Deacon, J.A. Load Equivalency in Flexible Pavement. <u>Proceeding Association of Asphalt Paving Technologists</u> 38 (1969): 465-494.
4. Deacon, J.A. Equivalant Passages of Aircraft with Respect to Fatigue Distress of Flexible Airfield Pavement. <u>Proceeding Association of Asphalt Paving Technologist</u> 40 (1971): 465-494.
5. Edward, J.M. and Valkering, C.P. <u>Structural Design of Asphalt Pavement for Heavy Aircraft.</u> London: Shell International Petroleum Company Ltd., 1971.

6. Harr, M.E. Foundation of Theorectical Soil Mechanics. New York: Mc Graw-Hill Book Co., 1951.

7. Hammond, R.H., Rogers,W.B., and Crittenden, J.B. Introduction to Fortran 77 and The Personal Computer. New York: Mc Graw - Hill Book Co, 1987.

8. Jones, A. Table of Stress in Three-Layer Elastic System. Highway Research Board, 342 (1962): 176-214.

9. Kingham, R.IAN. Failure Criteria Developed From AASHO Road Test. Proceeding of The Third International Conference on The Strutural Design of Asphalt pavement, pp. 657-669. University of Michigan, 1972.

10. Schiffman, R.L. General Analysis of Stress And Displacement in Layered Elastic System. Proceeding of The Forth International Conference on The Structural Design of Asphalt pavement, pp. 365-375, University of Micigan, 1977.

11. Sneddon, I.N. Fourier Transform. New York: Mc Graw-hill Book Co., 1951.

12. Shook, J.F., and Kallas, B.F. Factors Influencing Dynamic Modulus of Asphalt Concrete. Proceedings Association of Asphalt Paving Technologists 38 (1969): 140-178.

13. The Asphalt Institue. A Computer Program Supplement to Thickness Desing-Asphalt Pavements For Air Carrier Airports. Computer Program No.3 (CP-3). Maryland: The Asphalt Institute, 1987.

14. The Asphalt Institue. Computer Program DAMA. User's Manual (CP-1). Maryland: The Asphalt Institute, 1983.

15. The Asphalt Institute., Full-Depth Asphalt Pavements for Air Carrier Airports. Manual Series No.11 (MS-11) Maryland : The Asphalt Institute, 1973

16. The Asphalt Institute. Research and Development of The Asphalt Institute's Thickness Design Manual (MS-1). Ninth Edition, Research report No.82-2. Maryland: The Asphalt Institute, 1982.

17. The Asphalt Institute. Soil Manual For The Design of Asphal Pavement Structures. Manual Series No.10 (MS-10). Maryland: The Aspalt Institute, 1986.

18. The Asphalt Institute. Thickness Design-Aspalt Pavements for Hightway and Street. Manual Series No.1 (MS-1). Maryland: the Asphalt Institute, 1981.

19. Witczak, M.W. Design of Full Depth Asphalt Airfield Pavement. Proceeding of The Third International Conference of Asphalt Pavement, pp. 550-576. University of Michigan, 1972

19. Witczak, M.W. Prediction of Equivalent Damage Repetitions from Aircraft Traffic Mixture for Full Depth Asphalt Airfield Pavement. Proceedings Association of Asphalt Pavement Technologists 40 (1971): 194-226.

20. Yoder, E.J., and Witczak, M.W. Principle of Pavement Design. 2 nd ed. New York: John Wiley&Sons, inc., 1975.

ภาคผนวก

ภาคผนวก ก.

SOURCE PROGRAM

1 ´ โปรแกรมออกแบบ A (Fatigue Cracking)
ประกอบด้วย

1) Main Program
2) Subroutine PART
3) Subroutine BESSEL
4) Subroutine COEF
5) Subroutine SSR (วิเคราะห์ความเค้น)
6) Subroutine EIGEN
7) Subroutine LEASQ

```
$DEBUG
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C PROGRAM A  -THICKNESS DESIGN OF AIRPORT FLEXIBLE PAVEMENT
C            -DESIGN TO LIMIT MAX. PRINCIPLE TENSILE STRAIN
C             AT BOTTOM OF ASPHALT CONCRETE LAYER
C             (CONSIDER FATIGUE CRACKING OF ASPHALT CONCRETE)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      COMMON  ITN,BZ(51),H(2),RR(6),E(3),V(3),AZ(6,200),
     *        RJ0(6,200),RJ1(6,200),AJ(6,200),L,LI,J,K,
     *        X(25),Y(25),Z,R(12),X0(6),Y0(6),TP(12),
     *        SSZ(6),SSX(6),SSY(6),SSXY(6),SSXZ(6),SSYZ(6),
     *        NTEST(6),A(200,3),B(200,3),C(200,3),D(200,3)
      DIMENSION HB(10),HA(10),NPS(12),NT(12),DT(12),DD(12),NM(12)
      DIMENSION DX(2,12),WT(12),SFT(3,61),FRC(12)
      DIMENSION FF(81),F1(12,61),HH(5)
      DIMENSION P1(6,280),TR(3,3),XPTSN(12,5),PTSN(25)
      DIMENSION TSZ(25),TSX(25),TSY(25),TSXY(25),TSXZ(25),TSYZ(25)
      DIMENSION NPT(5),NAT(5),PT(5),AT(5),TF(12),NNT(61)
      INTEGER TN(12,31)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      1) MATERIAL PROPERTIES
C         EI     =  EXPERIMENTALLY DETERMINED COEF. OF GRANULAR BASE(K1)
C         E(N)   =  ELASTIC MODULUS OF N-TH LAYER
C         V(N)   =  POISSON'S RATIO OF N-TH LAYER
C         VB     =  VOLUME OF BITUMEN IN PERCENT
C         VV     =  VOLUME OF VOID IN PERCENT
C         P200   =  PERCENT AGGREGATE PASSING NO 200 SIEVE
C         VS     =  ABSOLUTE VISCOSITY AT 70 F. , poisex10^6
C      2) AVT    =  AVERAGE MEAN ANNUAL AIR TEMPERATURE
C      3) AIRCRAFT CHARACTERISTICS
C         NA          =  TOTAL TYPE OF DESIGN PERIOD
C         MODEL(J)    =  J-TH AIRCRAFT MODEL
C         NPS(J)      =  NUMBER OF J-TH PASSES OF THE J-TH AIRCRAFT
C         NT(J)       =  NUMBER OF TRAILING TIRE ROWS IN MAIN GEAR
C         NM(J)       =  1 (4,8 MAIN GEAR TIRE)
C                     =  2 (16 MAIN GEAR TIRE)
C         DT(I,J)     =  LATERAL DISTANCE BETWEEN MAIN GEAR AND AIRCRAFT
C                        CENTERLINE
C         WT(J)       =  MAXIMUM WEIGHT PER TIRE (MAIN GEAR)
C         TP(J)       =  TIRE PRESSURE (MAIN GEAR)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
 8000 FORMAT(' AIRCRAFT CHARACTERISTICS',/)
 8030 FORMAT(5X,I2,'.','AIRCRAFT MODEL',2X,A12)
 8040 FORMAT(5X,I2,'.','STANDARD AIRCRAFT DC-8-63F')
 8060 FORMAT(10X,'MOVEMENTS IN DESIGN PERIOD =',I5)
 8090 FORMAT(10X,'TYPE OF GEAR ASSEMBLIES    = SINGLE-TRICYCLE')
 8120 FORMAT(10X,'TYPE OF GEAR ASSEMBLIES    = DOUBLE-TRICYCLE')
 8150 FORMAT(10X,'TYPE OF TIRE ASSEMBLIES    = TWIN')
 8180 FORMAT(10X,'TYPE OF TIRE ASSEMBLIES    = TWIN-TANDEM')
 8210 FORMAT(10X,'TYPE OF TIRE ASSEMBLIES    = TRIPLE-TANDEM')
 8240 FORMAT(10X,'TIRE SPACING(in.)          =',F4.1)
 8270 FORMAT(10X,'TIRE SPACING(in.X in.)     =',F4.1,'X',F4.1)
 8300 FORMAT(10X,'X1 (in.)                   =',F5.1)
 8360 FORMAT(10X,'X2 (in.)                   =',F5.1)
 8390 FORMAT(10X,'MAX. WEIGHT PER TIRE(lb)   =',F7.1)
 8420 FORMAT(10X,'TIRE PRESSURE (psi)        =',F5.1,/)
```

```
 9000  FORMAT (A)
 9010  FORMAT (A40)
 9020  FORMAT (F4.1)
 9025  FORMAT (F5.1)
 9030  FORMAT (A35,A8)
 9035  FORMAT (3(1X,F5.2),1X,F7.1)
 9040  FORMAT (F8.1)
 9050  FORMAT (3(F4.2,1X))
 9055  FORMAT (I2)
 9060  FORMAT (A10,1X,I6,2(1X,I1),2(1X,F4.1),2(1X,F5.2),1X,F7.1,1X,
      *F5.1)
 9061  FORMAT(2(1X,I1),2(1X,F4.1),2(1X,F5.2),1X,F7.1,1X,F5.1)
 9065  FORMAT (80('$'))
 9070  FORMAT (25X,'THICKNESS DESIGN')
 9075  FORMAT (20X,'TENSILE STRAIN CRITERION')
 9076  FORMAT (5X,'DETERMINE TENSILE STRAIN AT BOTTOM OF ASPHALT',
      *' CONCRETE LAYER')
 9080  FORMAT (20X,19HSTRUCTURAL RESPONSE)
 9085  FORMAT ('+',19X,19('_'))
 9090  FORMAT (1X,'DESIGN  AVERAGE MEAN ANNUAL AIR TEMPERATURAL(F)=',
      *F4.1)
 9092  FORMAT(1X,'EXPERIMENTALLY DETERMINED COEF. OF GRANULAR BASE',
      *'(psi) =',F8.1)
 9095  FORMAT (1X,'DESIGN SUBGRADE RESILIENT MODULUS (psi)=',F8.1)
 9100  FORMAT (1X,'GRANULAR BASE THICKNESS(in.)=',F5.1,/)
 9200  FORMAT(2X,I2,'.','LAYER AND MATERIAL PROPERTIES')
 9210  FORMAT ('+',2X,31('_'))
 9220  FORMAT (5X,'LAYER   MATERIAL   POISSON`S   THICKNESS')
 9230  FORMAT (5X,'NUMBER    TYPE       RATIO      (in.)',/)
 9240  FORMAT (6X,' 1    ASPH. CONC.     ',F4.2,7X,F5.1)
 9250  FORMAT (6X,' 2    GRANU. BASE     ',F4.2,7X,F5.1)
 9255  FORMAT (6X,' 3    SUBGR. SOIL     ',F4.2,/)
 9260  FORMAT (2X,'**',I2,'.',I2,'.',A10,1X,'MOVEMENTS IN  DESIGN',
      *' PEROID=',I5)
 9264  FORMAT('+',4X,49('_'),/)
 9265  FORMAT (2X,'**',I2,'.',I2,'.',1X,'STANDARD AIRCRAFT DC-8-63F')
 9290  FORMAT (2X,'ELASTIC MODULUS OF ASPHAL CONCRETE(psi)=',F8.1)
 9300  FORMAT (2X,'ELASTIC MODULUS OF GRANULAR BASE(psi)   =',F8.1)
 9330  FORMAT (80('_'))
 9340  FORMAT (' COMPUTATIONAL ',25X,'STRESS(psi)')
 9350  FORMAT(' POINT (IN)',3X,66('_'))
 9355  FORMAT ('    X      Y    ','     Z            Y            X       ',
      *'  XZ           XY          YZ')
 9361  FORMAT (2(1X,F5.2),6(E10.4,1X))
 9362  FORMAT(' PRINCIPAL TENSILE STRAIN(in/in)=',E13.7,/)
 9370  FORMAT ('  MAXIMUM PRINCIPAL TENSILE STRAIN =',E13.7,/)
 9385  FORMAT (//,30X,23HPREDICTED TRAFFIC VALUE)
 9386  FORMAT ('+',29X,23('_'),//)
 9400  FORMAT (' AIRCRAFT MODEL      NUMBER OF STRAIN REPETITION AT ',
      *'DISTANCE X(ft) FROM CENTERLINE')
 9405  FORMAT (14X,67('_'))
 9410  FORMAT (13X,'  1     3     6     10     12     15     18     20',
      *'     22     25   ')
 9420  FORMAT (A14,10I6)
 9430  FORMAT (' TOTAL',8X,10I6)
 9445  FORMAT (//,30X,'ALLOWABLE TRAFFIC VALUE')
```

```
D Line# 1      7                                      Microsoft FORTRAN77 V3.30 March 1985
      9447  FORMAT ('+',29X,23('_'),//)
      9450  FORMAT (' NUMBER OF STRAIN REPETITION(TENSILE STRAIN CRITERIA)',
           * '=',I10,/)
      9530  FORMAT (' NUMBER OF ALLOWABLE TENSILE STRAIN REPETITION',I10)
      9600  FORMAT(80('='),//)
      9800  FORMAT(' THE FINAL THICKNESS IS ',F5.2,' in.')
            CHARACTER*50 FILEIN,FILEOT,TEMP,MAT,STAC,MODEL(12)
            WRITE (*,'(A\)') '  NAME OF DATA FILE:  '
            READ (*,9000) FILEIN
            WRITE(*,'(A\)') '  NAME OF OUTPUT FILE: .
            READ(*,9000) FILEOT
            OPEN(UNIT=4,FILE=FILEIN,STATUS='OLD')
            OPEN(UNIT=8,FILE=FILEOT,STATUS='NEW')
      C READ IN AIRCRAFT CHARACTERISTICS AND MATERRIAL PROPERTIES
            READ(4,9010) TEMP
            READ(4,9020) AVT
            READ(4,9030) MAT
            READ(4,9035)P200,VB,VV,VS
            READ(4,9040) EI
            READ(4,9040) E(3)
            READ(4,9050) V(1),V(2),V(3)
            READ(4,9055)ITN
            READ(4,9055) NA
            READ(4,9030) AAIRCH
            DO 10 J = 1,NA
1              READ(4,9060) MODEL(J),NPS(J),NT(J),NM(J),DT(J),
1          *                    DD(J),DX(1,J),DX(2,J),WT(J),TP(J)
1              R(J)=SQRT(WT(J)/(TP(J)*3.141592654))
1     10    CONTINUE
      20    NA1 = NA+1
      C ****READ IN STANDARD AIRCRAFT CHARACTERISTICS
            OPEN(UNIT=5,FILE = 'STR AIR.DAT',STATUS='OLD')
            READ(5,9030) STAC,MODEL(NA1)
            READ(5,9061)NT(NA1),NM(NA1),DT(NA1),
           *DD(NA1),DX(1,NA1),DX(2,NA1),WT(NA1),TP(NA1)
            R(NA1)=SQRT(WT(NA1)/(TP(NA1)*3.141592654))
      cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
      C ****DETERMINE TRANSVERSDE DISTRIBUTION FACTOR
      C     F1(J,MX)=TRANSVERSE DISTRIBUTION FACTOR AT DISTANCE MX(ft)
      C                 FROM RUNWAY CENTERLINE
      cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
            DO 500 J =1,NA
1           W1 = DD(J)
1           DO 501 I = 1,NM(J)
2              S = DX(I,J)
2              DO 555 MX=-30,30
3                 B1=0.5*0.57735027+MX
3                 B2=-0.5*0.57735027+MX
3                 FQ1=EXP((-.5)*(((B1-S)/3.5)**2))/(3.5*SQRT(2.0*3.1415927))
3                 FQ2=EXP((-.5)*(((B2-S)/3.5)**2))/(3.5*SQRT(2.0*3.1415927))
3                 FF(MX)=0.5*(FQ1+FQ2)
3     555      CONTINUE
2              W=W1/12.0
2              IF(W.LE.0.5) N=0
2              IF(W.GT.0.5) N=INT(W+0.5)
2              RD=W-0.5-AINT(W-0.5)
```

```
D Line# 1      7
2                  DO 502 MX = -30,30
3                     SUMFT=0.0
3                     N1=(-1)*N
3                     DO 503 MI = N1,N
4                        IF(ABS(MX+MI).GT.30)GOTO 511
4                        IF(ABS(MI).EQ.N)SUMFT=SUMFT+FF(MX+MI)*RD
4                        IF(ABS(MI).LT.N)SUMFT=SUMFT+FF(MX+MI)
4        511          CONTINUE
4        503       CONTINUE
3                  SFT(I,MX)=SUMFT
3
3        502       CONTINUE
2        501       CONTINUE
1                  DO 527 MX = -30,30
2                     TOTF1 = 0.0
2                     DO 507 I = 1,NM(J)
3                        TOTF1 = TOTF1+SFT(I,MX)+SFT(I,-MX)
3        507          CONTINUE
2                     F1(J,MX)=TOTF1
2        527       CONTINUE
1        500    CONTINUE
         cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
         c****ANALYSE STRESS AND STRAIN
         cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
         C ****COMPUTE ZEROS OF J1(X) AND J0(X) SET UP GAUSS CONSTANT
                ITN4 = ITN*4
                K =ITN+1
                BZ(1) = 0.0
                BZ(2) = 1.0
                BZ(3) = 2.4048
                BZ(4) = 3.8317
                BZ(5) = 5.5201
                BZ(6) = 7.0156
                DO 700 I = 7,K,2
1                  T = I/2
1                  TD=4.0*T-1.
1                  BZ(I)=3.1415927*(T-0.25+0.050661/TD-0.053041/TD**3+
1             *          0.262051/TD**5)
1        700    CONTINUE
                DO 710 I = 8,ITN,2
1                  T = (I-2)/2
1                  TD = 4.0*T+1.
1                  BZ(I) =3.1415927*(T+0.25-0.151982/TD+0.015399/TD**3
1             *             -0.245270/TD**5)
1        710    CONTINUE
         C ****PRINT OUT INPUT DATA
                WRITE(8,9065)
                WRITE(8,9070)
                WRITE(8,9075)
                WRITE(8,9076)
                WRITE(8,9065)
                WRITE(8,9090) AVT
                WRITE(8,9092) EI
                WRITE(8,9095) E(3)
         C ***DETERMINE AVERAGE MEAN ANNUAL PAVEMENT TEMPERATURE FROM AIR TEMP.
                AVT=1.05*AVT+5.0
```

```
            WRITE(8,8000)
      C ****PRINT OUT AIRCRAFT CHARACTERISTICS
            DO 699 J=1,NA1
1              IF(J.LT.NA1)WRITE(8,8030)J,MODEL(J)
1              IF(J.EQ.NA1)WRITE(8,8040)J
1              IF(J.LT.NA1)WRITE(8,8060)NPS(J)
1              IF(NM(J).EQ.1)WRITE(8,8090)
1              IF(NM(J).EQ.2)WRITE(8,8120)
1              IF(NT(J).EQ.1)WRITE(8,8150)
1              IF(NT(J).EQ.2)WRITE(8,8180)
1              IF(NT(J).EQ.3)WRITE(8,8210)
1              IF(NT(J).EQ.1)WRITE(8,8240)DD(J)
1              IF(NT(J).GT.1)WRITE(8,8270)DD(J),DT(J)
1              WRITE(8,8300)DX(1,J)
1              IF(NM(J).EQ.2)WRITE(8,8360)DX(2,J)
1              WRITE(8,8390)WT(J)
1              WRITE(8,8420)TP(J)
1     699   CONTINUE
            IMH = 0
            DO 601 MH=1,10
1           IMH=IMH+1
1     C ****READ IN A.C. SURFACE  AND GRANULAR BASE THICKNESS
1           WRITE(*,'(A\)') ' INPUT GARNULAR BASE THICKNESS(ins.)=?'
1           READ(*,9025)HB(MH)
1           H2 = HB(MH)
1           WRITE(8,9100)H2
1           DO 602 M = 1,5
2              IF(M.GE.2)WRITE(*,*)'NO.    H1        Na         Np'
2              DO  444 III=1,(M-1)
3                 IF(M.GE.2)WRITE(*,2010)III,HH(III),NAT(III),NPT(III)
3     2010        FORMAT (1X,I2,2X,F5.2,2X,I8,2X,I8)
3     444      CONTINUE
2              WRITE(8,9200) M
2              WRITE(8,9210)
2              WRITE(8,9220)
2              WRITE(8,9230)
2     C ****START NEW A.C. SURFACE THICKNESS
2              WRITE(*,'(A\)')' INPUT A.C. SURFACE THICKNESS(ins.)=?'
2              READ (*,9025) HH(M)
2     C ****PRINT OUT LAYER AND MATERIAL PROPERTIES
2              WRITE(8,9240) V(1),HH(M)
2              WRITE(8,9250) V(2),H2
2              WRITE(8,9255) V(3)
2     C *****SET DEPTH OF EACH LAYER
2              H(1) = HH(M)
2              H(2) = H(1)+H2
2     C *****PRINT OUT AIRCRAFT CHARACTERISTICS
2              DO 604 J = 1,NA1
3                 IF(J.EQ.NA1)WRITE(8,9265) M,J
3                 IF(J.LT.NA1)WRITE(8,9260) M,J,MODEL(J),NPS(J)
3                 WRITE(8,9264)
3                 XPTSN(J,M) = 0.0
3     C ******SET LOADING FREQUENCY(FRC(J)) OF J-TH AIRCRAFT
3                 IF(NT(J).EQ.1)FRC(J)=4.0
3                 IF(NT(J).EQ.2)FRC(J)=2.0
3                 IF(NT(J).EQ.3)FRC(J)=1.0
```

```
C *****DETERMINE ELASTIC MODULUS OF A.C. LAYER
              B5=1.3+0.49825*LOG10(FRC(J))
              B4=VB
              B3=0.553833+0.028829*P200/(FRC(J)**0.1703)-0.03476*VV+
     *           0.070377*VS+0.93175/(FRC(J)**0.02774)
              B2=(B4**0.5)*(AVT**B5)
              B1=B3+0.000005*B2-0.00189*B2/(FRC(J)**1.1)
              E(1)=100000.0*(10.0**B1)
C ****SET UP COORDINATE OF CENTER OF TIRES AND COMPUTATIONAL POINT
              IF(NT(J).EQ.3)THEN
                NR = 6
                KK = 5
                X0(1) = 0.0
                X0(2) =0.0
                X0(3) = DD(J)
                X0(4) = DD(J)
                X0(5) = DD(J)
                X0(6) = 0.0
                Y0(1) = 0.0
                Y0(2) = -DT(J)
                Y0(3) = -DT(J)
                Y0(4) = 0.0
                Y0(5) = DT(J)
                Y0(6) = DT(J)
              ENDIF
              IF(NT(J).EQ.2)THEN
                NR = 4
                KK = 5
                X0(1) = 0.0
                X0(2) = DD(J)
                X0(3) = DD(J)
                X0(4) = 0.0
                Y0(1) = 0.0
                Y0(2) = 0.0
                Y0(3) = DT(J)
                Y0(4) = DT(J)
              ENDIF
                IF(NT(J).EQ.1)THEN
                NR = 2
                KK = 3
                X0(1) = 0.0
                X0(2) = DD(J)
                Y0(1)=0.0
                Y0(2) = 0.0
                X(2)=R(J)
                Y(2)=0.0
                X(3)=DD(J)/2.0
                Y(3)=0.0
              ENDIF
              X(1)=0.0
              Y(1)=0.0
              IF(NT(J).EQ.2.OR.NT(J).EQ.3)THEN
              IF(DD(J).LT.DT(J))THEN
                X(2)=R(J)
                Y(2)=0.0
                X(3)=DD(J)/2.0
```

```
3                     Y(3)=0.0
3                  ENDIF
3                  IF(DD(J).GT.DT(J))THEN
3                     X(2)=0.0
3                     Y(2)=R(J)
3                     X(3)=0.0
3                     Y(3)=DT(J)/2.0
3                  ENDIF
3                  ENDIF
3     C*COMPUTE DISTANCE BET. CENTER OF N-TH TIRE AND K-TH COMPUTATIONAL POIN
3                  DO 606 K = 1,3
4                     DO 1234 NN=1,NR
5                        RR(NN) = SQRT((X(K)-X0(NN))**2+(Y(K)-Y0(NN))**2)
5     1234            CONTINUE
4     C*** COMPUTE ELASTICS MODULUS OF GRANULAR BASE
4                     E(2) = (10.447)*(E(3)**0.287)*(EI**0.868)/
4          *          ((HH(M)**0.471)*(H2**0.041)*(E(1)**0.139))
4                     IF(K.EQ.1) THEN
4                        WRITE(8,9080)
4                        WRITE(8,9085)
4                     ENDIF
4     C*** SET DEPTH (Z) WHERE STRESS AND STRAIN ARE REQ'D
4                     Z = H(1)
4                     TSZ(K) = 0.0
4                     TSX(K) = 0.0
4                     TSY(K) = 0.0
4                     TSXY(K) = 0.0
4                     TSXZ(K) = 0.0
4                     TSYZ(K) = 0.0
4                     DO 660 N = 1,NR
5     C*** CALCULATE THE PARTITION
5                        CALL PART(N)
5                        DO 670 LI = 1,ITN4
6                           P = AZ(N,LI)
6     C*** CALCULATE THE COEFFICIENTS
6                           CALL COEF(N,P)
6                           IF(RR(N))115,115,110
6     110                   PR = P*RR(N)
6                           CALL BESSEL(0,PR,Y1)
6                           RJ0(N,LI) = Y1
6                           CALL BESSEL(1,PR,Y1)
6                           RJ1(N,LI) = Y1
6     115                   PA = P*R(J)
6                           CALL BESSEL(1,PA,Y1)
6                           AJ(N,LI) = Y1
6
6     670                CONTINUE
5     C*** SET LAYER WHERE STRESS AND STRAIN ARE REQ'D
5                        L = 1
5     C*** CALCULATE STRESS FROM N-TH TIRE
5                        CALL SSR(N)
5                        TSZ(K) = TSZ(K)+SSZ(N)
5                        TSX(K) = TSX(K)+SSX(N)
5                        TSY(K) = TSY(K)+SSY(N)
5                        TSXY(K) = TSXY(K)+SSXY(N)
5                        TSXZ(K) = TSXZ(K)+SSXZ(N)
```

```
D Line# 1      7                         Microsoft FORTRAN77 V3.30 March 1985
5                        TSYZ(K) = TSYZ(K)+SSYZ(N)
5       660           CONTINUE
4       C*** CALCULATE PRINCIPAL TENSILE STRAIN
4                     TR(1,1)  = (1/E(L))*(TSZ(K)-V(L)*(TSX(K)+TSY(K)))
4                     TR(2,2)  = (1/E(L))*(TSX(K)-V(L)*(TSZ(K)+TSY(K)))
4                     TR(3,3)  = (1/E(L))*(TSY(K)-V(L)*(TSZ(K)+TSX(K)))
4                     TR(1,2) = ((1+V(L))/E(L))*TSXY(K)
4                     TR(1,3) = ((1+V(L))/E(L))*TSXZ(K)
4                     TR(2,3) = ((1+V(L))/E(L))*TSYZ(K)
4                     CALL EIGEN(TR,PRE)
4                     PTSN(K) = PRE
4                     IF(XPTSN(J,M)-PTSN(K) .GT. 0.000001)  GO TO 661
4                     XPTSN(J,M) = PTSN(K)
4       661           CONTINUE
4       C*** PRINT OUT COMPUTATIONAL POINTS , STRESS AND STRAIN
4                     IF(K.EQ.1) THEN
4                       WRITE(8,9290) E(1)
4                       WRITE(8,9300) E(2)
4                       WRITE(8,9330)
4                       WRITE(8,9340)
4                       WRITE(8,9350)
4                       WRITE(8,9355)
4                       WRITE(8,9330)
4                     ENDIF
4                     WRITE(8,9361)X(K),Y(K),TSZ(K),TSX(K),TSY(K),TSXZ(K),
4            *                     TSXY(K),TSYZ(K)
4                     WRITE(8,9362)PTSN(K)
4       606        CONTINUE
3                  WRITE(8,9330)
3                  WRITE(8,9370) XPTSN(J,M)
3       604     CONTINUE
2               WRITE(8,9385)
2               WRITE(8,9386)
2               WRITE(8,9330)
2               WRITE(8,9400)
2               WRITE(8,9405)
2               WRITE(8,9410)
2               WRITE(8,9330)
2               NPT(M) = 0
2       CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2       CDETERMINE NUMBER OF PREDICTED EQUIVALENT DC-8-63F STRAIN REPETITION(Np)
2       CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2              DO 916 NX = 0,30
3                 NNT(NX) = 0
3                 DO 912 J = 1,NA
4                    TF(J) = ((XPTSN(J,M)/XPTSN(NA1,M))**4.995)
4                    TN(J,NX) = NPS(J)*TF(J)*F1(J,NX)
4                    NNT(NX) = NNT(NX)+TN(J,NX)
4       912       CONTINUE
3                 IF(NPT(M) .GT. NNT(NX)) GO TO 916
3                 NPT(M) = NNT(NX)
3       916    CONTINUE
2              DO 902 J=1,NA
3                 WRITE(8,9420)MODEL(J),TN(J,1),TN(J,3),TN(J,6),TN(J,10),
3            *    TN(J,12),TN(J,15),TN(J,18),TN(J,20),TN(J,22),TN(J,25)
3       902    CONTINUE
```

```
D       1      7
2                  WRITE(8,9330)
2                  WRITE(8,9430)NNT(1),NNT(3),NNT(6),NNT(10),NNT(12),NNT(15),
2            *                  NNT(18),NNT(20),NNT(22),NNT(25)
2                  WRITE(8,9330)
2                  PT(M)=LOG10(FLOAT(NPT(M)))
2                  WRITE(8,9450) NPT(M)
2                  WRITE(8,9330)
2                  WRITE(8,9445)
2                  WRITE(8,9447)
2       CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2       C DETERMINE NUMBER OF ALLOWABLE DC-8-63F STRAIN REPETITION(Na)
2       CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2                  NAT(M) =INT(1.86351E-17*(1.01996**(AVT**1.45))/
2            *                 (XPTSN(NA1,M)**4.995))
2                  WRITE(8,9530)NAT(M)
2                  AT(M)=LOG10(FLOAT(NAT(M)))
2       602    CONTINUE
1       CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
1       C DETERMINE RELATIONSHIP OF
1       C      - A.C. SURFACE THICKNESS (h1) AND Na
1       C      - A.C. SURFACE THICKNESS (h1) AND Np
1       C BY LEAST SQUARE METHOD
1       C AND DETERMINE THE INTERSECTION OF h1-Np CURVE AND h1-Na CUVRE
1       C NEEDED TO SATISFY THE ASPHALT CONCRETE TENSILE STRAIN CRITERION
1       CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
1              CALL LEASQ(HH,PT,AT,DHT,DNT)
1              WRITE(8,9600)
1              NTD=INT(10.0**(DNT))
1              WRITE(*,9800)DHT
1              WRITE(8,9800)DHT
1              WRITE(8,*)'
1              WRITE(8,*)'#################### END ########################'
1              WRITE(8,*)'
1              HA(MH)=DHT
1              WRITE(*,*)'PRESS 1 => SELECT NEW GRANULAR BASE THICKNESS'
1              WRITE(*,*)'      2 => QUIT    '
1              WRITE(*,'(A\)')'  SELECT ============> '
1              READ(*,*)NNN
1              IF(NNN.EQ.1) GOTO 601
1              IF(NNN.GT.1)GOTO 311
1       601    CONTINUE
        311    CONTINUE
               WRITE(*,*) '    NO.     h2(ins.)    h1(ins.)  '
               DO 312 I=1,IMH
1       1          FORMAT(4X,I2,7X,F6.2,5X,F6.2)
1                  WRITE(*,1) I,HB(I),HA(I)
1       312    CONTINUE
               STOP
               END
```

FILE STR_AIR.DAT

```
STANDARD AIRCRAFT                         DC-8-63F
 2 1 55.0 32.0 17.50 00.00 43000.0 196.0 4.0
```

```
  $DEBUG
        SUBROUTINE PART(N)
  CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
  C       COMPUTE POINTS FOR LEGENDRE-GAUSS INTREGRATION
  CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        COMMON ITN,BZ(71),H(2),RR(6),E(3),V(3),AZ(6,280),
       *        RJ0(6,280),RJ1(6,280),AJ(6,280),L,LI,J,K,
       *        X(25),Y(25),Z,R(12),X0(6),Y0(6),TP(12),
       *        SSZ(6),SSX(6),SSY(6),SSXY(6),SSXZ(6),SSYZ(6),
       *        NTEST(6),A(280,3),B(280,3),C(280,3),D(280,3)
        DATA G1/8.6113631E-1/,G2/3.3998104E-1/
        MX = N
        NI = J
        ZF = R(NI)
        NTEST(MX) = 2
        IF (RR(MX))8,8,9
  9     CONTINUE
        NTEST (MX) =R(NI)/RR(MX)+0.0001
        IF (NTEST(MX))6,6,5
  6     CONTINUE
        NTEST(MX) = RR(MX)/R(NI)+0.0001
        ZF  = RR(MX)
  5     CONTINUE
        NTEST(MX) = NTEST(MX)+1
        IF (NTEST(MX)-10)8,8,7
  7     CONTINUE
        NTEST(MX) =10
  8     CONTINUE
        MK =1
        ZF = 2*ZF
        SZ2=0.0
        DO 28 N1 = 1,ITN
1          SZ1 =SZ2
1          SZ2 = BZ(N1+1)/ZF
1          SF = SZ2-SZ1
1          PP = SZ2+SZ1
1          SG1 = SF*G1
1          SG2 = SF*G2
1          AZ(MX,MK)=PP-SG1
1          AZ(MX,MK+1) = PP-SG2
1          AZ(MX,MK+2)=PP+SG2
1          AZ(MX,MK+3) = PP+SG1
1          MK =MK+4
1 28    CONTINUE
        RETURN
        END
```

```
D Line# 1     7
  $DEBUG
              SUBROUTINE BESSEL(NI,XI,Y)
  CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
  C           CALCULATE BESSEL FUNTION ORDER 1 AND 0
  CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
              DIMENSION D(20),PZ(6),QZ(6),P1(6),Q1(6)
              DOUBLE PRECISION T4,X,PI
              DATA PZ/1.0E0,-1.125E-4,2.8710938E-7,-2.3449658E-9,
       *             3.9806841E-11,-1.1536133E-12/
              DATA QZ/-5.0E-3,4.6875E-6,-2.3255859E-8,2.8307087E-10,
       *             -6.3912096E-12,2.3124704E-12/
              DATA P1/1.0E0,1.875E-4,-3.6914063E-7,2.7713232E-9,
       *            -4.5114421E-11,1.2750463E-12/
              DATA Q1/1.5E-2,-6.5625E-6,2.8423828E-8,
       *            -3.2662024E-10,7.1431166E-12,-2.5327056E-13/
              PI=3.1415927
  9           N =NI
              X = XI
              IF (X-7.0) 10,10,160
  10          X2=X/2.0
              FAC = -X2*X2
              IF(N) 11,11,14
  11          C=1.0
              Y = C
              DO 13 I=1,34
1             T = I
1             C = FAC*C/(T*T)
1             TEST = ABS (C) - 10.0**(-8)
1             IF (TEST) 17,17,12
1 12          Y = Y+C
1 13          CONTINUE
  14          C = X2
              Y=C
              DO 16 I = 1,34
1             T = I
1             C = FAC*C/(T*(T+1.0))
1             TEST = ABS(C)-10.0**(-8)
1             IF(TEST)17,17,15
1 15          Y = Y+C
1 16          CONTINUE
  17          RETURN
  160         IF(N)161,161,164
  161         DO 162 I = 1,6
1             D(I) = PZ(I)
1             D(I+10) = QZ(I)
1 162         CONTINUE
              GOTO 163
  164         DO 165 I =1,6
1             D(I) = P1(I)
1             D(I+10) = Q1(I)
1 165         CONTINUE
  163         CONTINUE
              T1 = 25.0/X
              T2 = T1*T1
              P = D(6)*T2+D(5)
              DO 170 I =1,4
```

```
D Line# 1      7
1                J = 5-I
1                P = P*T2+D(J)
1          170   CONTINUE
                 Q = D(16)*T2+D(15)
                 DO 171 I =1,4
1                J = 5-I
1                Q = Q*T2+D(J+10)
1          171   CONTINUE
                 Q = Q*T1
           C
                 T4 = DSQRT(X*PI)
                 T6 = SIN(X)
                 T7 = COS(X)
           C
                 IF(N) 180,180,185
           180   T5 = ((P-Q)*T6+(P+Q)*T7)/T4
                 GOTO 99
           185   T5 = ((P+Q)*T6-(P-Q)*T7)/T4
           99    Y = T5
                 RETURN
                 END
```

```
D Line# 1     7                               Microsoft FORTRAN77 V3.30 March 1985
        $DEBUG
              SUBROUTINE  COEF(N,P)
        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        C     USE FOR ALL PROBLEM, UP TO MAX. DIMENSION OF 3-LAYER
        C     CALCULATE COEF. A,B,C AND D OF 3-LAYER ELASTIC SYSTEM
        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
              COMMON ITN,BZ(71),H(2),RR(6),E(3),V(3),AZ(6,280),
             *       RJ0(6,280),RJ1(6,280),AJ(6,280),L,LI,J,K,
             *       X(25),Y(25),Z,R(12),X0(6),Y0(6),TP(12),
             *       SSZ(6),SSX(6),SSY(6),SSXY(6),SSXZ(6),SSYZ(6),
             *       NTEST(6),A(280,3),B(280,3),C(280,3),D(280,3)
              DIMENSION PM(4,4,4),XP(5,4,4),FM(4),SC(4)
              REAL *4 Q(2,2)
              LC = LI
        C**** SET UP MATRIX  X=DI*MI*KI*K*M*D
        C     COMPUTE THE MATRICES X(K)
              DO 10 N1 = 1,2
1                T1 = E(N1)*(1.+V(N1+1))/(E(N1+1)*(1.+V(N1)))
1                T1M = T1-1.
1                PH = P*H(N1)
1                PH2 = PH*2.
1                VK2 = 2.*V(N1)
1                VKP2 = 2.*V(N1+1)
1                VK4=2.0*VK2
1                VKP4=2.0*VKP2
1                VKK8 = 8.*V(N1)*V(N1+1)
1       C
1                XP(N1,1,1) = VK4-3.-T1
1                XP(N1,2,1) = 0.
1                XP(N1,3,1) = T1M*(PH2-VK4+1.)
1                XP(N1,4,1) = -2.*T1M*P
1       C
1                T3 = PH2*(VK2-1.)
1                T4 = VKK8+1.-3.*VKP2
1                T5 = PH2*(VKP2-1.)
1                T6 = VKK8+1.-3.0*VK2
1       C
1                XP(N1,1,2) = (T3+T4-T1*(T5+T6))/P
1                XP(N1,2,2) = T1*(VKP4-3.)-1.
1                XP(N1,4,2) = T1M*(1.-PH2-VKP4)
1       C
1                XP(N1,3,4) = (T3-T4-T1*(T5-T6))/P
1       C
1                T3 = PH2*PH-VKK8+1.
1                T4 = PH2*(VK2-VKP2)
1       C
1                XP(N1,1,4) = (T3+T4+VKP2-T1*(T3+T4+VK2))/P
1                XP(N1,3,2)=(-T3+T4-VKP2+T1*(T3-T4+VK2))/P
1       C
1                XP(N1,1,3) = T1M*(1.-PH2-VK4)
1                XP(N1,2,3) = 2.*T1M*P
1                XP(N1,3,3) = VK4-3.-T1
1                XP(N1,4,3) = 0.
1       C
1                XP(N1,2,4) = T1M*(PH2-VKP4+1.)
1                XP(N1,4,4) = T1*(VKP4-3.)-1.
```

```
1    10    CONTINUE
     C**** COMPUTE THE PRODUCT MATRICES PM
           SC(2) = 4.*(V(2)-1.)
           SC(1) = SC(2)*4.*(V(1)-1.)
           Q(1,1) = 1.
           Q(2,2) = 1.
           Q(1,2) = 0.
           QQ = P*2.*H(2)
           IF(QQ-80.) 15,15,16
     15    CONTINUE
           Q(1,2) = EXP(-QQ)
     C     Q(2,1) IS NOT NEEDED FOR INITIALIZING THE PM MATRIX
     16    CONTINUE
     C**** 20 LOOP INITIALIZES PM(N,,)
           DO 20 MX = 1,4
1             LL = (MX+1)/2
1             PM(2,MX,4) = XP(2,MX,4)*Q(LL,2)
1             PM(2,MX,3) = XP(2,MX,3)*Q(LL,2)
1    20    CONTINUE
           QQ = P*2.*H(1)
           IF (QQ-80.)22,22,23
     22    CONTINUE
           Q(2,1) = EXP(QQ)
           Q(1,2) = 1./Q(2,1)
           GOTO 24
     23    CONTINUE
           Q(1,2) = 0.
           Q(2,1) = 1.E20
     24    CONTINUE
           DO 25 MX = 1,4
1          LL = (MX+1)/2
1          DO 25 MI = 3,4
2             PM(1,MX,MI) = (XP(1,MX,1)*PM(2,1,MI)+XP(1,MX,2)*PM(2,2,MI))*
2         *   Q(LL,1)+(XP(1,MX,3)*PM(2,3,MI)+XP(1,MX,4)*PM(2,4,MI))*Q(LL,2)
2    25    CONTINUE
     C**** SOLVE FOR C(3) AND D(3) OF SUBGRADE
           T3 = 2.*V(1)
           T4 = T3-1.0
           FM(1) = P*(PM(1,1,3)+PM(1,3,3))+T3*(PM(1,2,3)-PM(1,4,3))
           FM(2) = P*(PM(1,1,3)-PM(1,3,3))+T4*(PM(1,2,3)+PM(1,4,3))
           FM(3) = P*(PM(1,1,4)+PM(1,3,4)) + T3*(PM(1,2,4)-PM(1,4,4))
           FM(4) = P*(PM(1,1,4)-PM(1,3,4))+T4*(PM(1,2,4)+PM(1,4,4))
           WRITE(*,*) FM(4)
           DFAC = SC(1)/((FM(1)*FM(4)-FM(3)*FM(2))*P*P)
           A(LC,3) = 0.
           B(LC,3) = 0.
           C(LC,3) = -FM(3)*DFAC
           D(LC,3) = FM(1)*DFAC
     C**** BACKSOLVE FOR THE OTHER A,B,C,D
           DO 100 N1 = 1,2
1             A(LC,N1)=(PM(N1,1,3)*C(LC,3)+PM(N1,1,4)*D(LC,3))/SC(N1)
1             B(LC,N1)=(PM(N1,2,3)*C(LC,3)+PM(N1,2,4)*D(LC,3))/SC(N1)
1             C(LC,N1)=(PM(N1,3,3)*C(LC,3)+PM(N1,3,4)*D(LC,3))/SC(N1)
1             D(LC,N1)=(PM(N1,4,3)*C(LC,3)+PM(N1,4,4)*D(LC,3))/SC(N1)
1    100   CONTINUE
           RETURN
```

```
D Line# 1     7
        $DEBUG
                SUBROUTINE SSR(N)
        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        C       THIS SUBROUTINE COMPUTE THE STRESS IN CYLINDRICAL COORDINATE
        C       AND TRANSFORM TO CARTESIAN SYSTEM BY USING TRANSFORMATION
        C       FORMULAS
        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
                COMMON ITN,BZ(71),H(2),RR(6),E(3),V(3),AZ(6,280),
             *         RJ0(6,280),RJ1(6,280),AJ(6,280),L,LI,J,K,
             *         X(25),Y(25),Z,R(12),X0(6),Y0(6),TP(12),
             *         SSZ(6),SSX(6),SSY(6),SSXY(6),SSXZ(6),SSYZ(6),
             *         NTEST(6),A(280,3),B(280,3),C(280,3),D(280,3)
              DIMENSION W(4),ARP(21),TEST(11)
              DATA  W/0.34785485,0.65214515,0.65214515,0.34785485/
              MI=J
              MX=N
              VL=2.0*V(L)
              EL=(1.0+V(L))/E(L)
              VL1 = 1.0-VL
        C**** COMPUTE THE COMPONENT OF STRESS IN CYLINDRICAL COORDINATE SYSTEM
              CSZ = 0.0
              CST = 0.0
              CSR = 0.0
              CTR = 0.0
              ARP(MI) = R(MI)*TP(MI)
              DO 1001 N1 = 1,ITN
1       C******* INITIALIZE THE SUB-INTEGRALS
1                 RSZ = 0.0
1                 RST = 0.0
1                 RSR = 0.0
1                 RTR = 0.0
1                 NK = 4*(N1-1)
1       C******* COMPUTE THE SUB-INTEGRALS
1                 DO 1002 NJ = 1,4
2                    J1 = NK + NJ
2                    P = AZ(MX,J1)
2                    IF(P*Z.LE.80.0)EP = EXP(P*Z)
2                    IF(P*Z.GT.80.0)EP = 1.0E20
2                    T1 = B(J1,L)*EP
2                    T2 = D(J1,L)/EP
2                    T1P = T1+T2
2                    T1M = T1-T2
2                    T1 = (A(J1,L)+B(J1,L)*Z)*EP
2                    T2 = (C(J1,L)+D(J1,L)*Z)/EP
2                    T2P = P*(T1+T2)
2                    T2M = P*(T1-T2)
2                    WA = AJ(MX,J1)*W(NJ)
2                    IF (RR(MX).EQ.0.0) GO TO 20
2                    BJ1 = RJ1(MX,J1)*P
2                    BJ0 = RJ0(MX,J1)*P
2                    RSZ = RSZ+WA*P*BJ0*(VL1*T1P-T2M)
2                    RTR = RTR+WA*P*BJ1*(VL*T1M+T2P)
2                    RSR=RSR+WA*(P*BJ0*((1.0+VL)*T1P+T2M)-BJ1*(T1P+T2M)/RR(MX))
2                    RST = RST+WA*(VL*BJ0*P*T1P+BJ1*
2            *             (T1P+T2M)/RR(MX))
2                    GOTO 1002
```

```
D Line# 1      7
2       C********** SPECIAL ROUTINE FOR RR(N)= ZERO
2       20            PP = P*P
2                     RSZ = RSZ+WA*PP*(VL1*T1P-T2M)
2                     RST = RST+WA*PP*((VL+0.5)*T1P+0.5*T2M)
2                     RSR = RST
2       1002      CONTINUE
1                 SF = (AZ(MX,NK+4)-AZ(MX,NK+1))/1.7222726
1                 CSZ = CSZ+RSZ*SF
1                 CST = CST+RST*SF
1                 CSR = CSR+RSR*SF
1                 CTR = CTR + RTR*SF
1       1001  CONTINUE
              CSZ = CSZ*ARP(MI)
              CST = CST*ARP(MI)
              CTR = CTR*ARP(MI)
              CSR = CSR*ARP(MI)
        C***** TRANSFORM THE COMPONENT OF STRESS IN THE CYLINDRICAL COORDINATE
        C       SYSTEM TO THE CARTESIAN COORDINATE SYSTEM
              SSZ(MX)=CSZ
              IF(RR(N))70,70,80
        70    SSX(MX)=CSR
              SSY(MX)=CST
              SSXY(MX)=0.0
              SSXZ(MX)=0.0
              SSYZ(MX)=0.0
              GOTO 90
        80    SSX(MX)=0.5*(CSR+CST)+0.5*(CSR-CST)*((X(K)-X0(N))**2
             *          -(Y(K)-Y0(N))**2)/((X(K)-X0(N))**2+(Y(K)-Y0(N))**2)
                          SSY(MX) = 0.5*(CSR+CST)-0.5*(CSR-CST)*((X(K)-X0(N))**2
             *          -(Y(K)-Y0(N))**2)/((X(K)-X0(N))**2+(Y(K)-Y0(N))**2)
                SSXY(MX)  =  (CSR-CST)*(X(K)-X0(N))*(Y(K)-Y0(N))/((X(K)-
             *                X0(N))**2+(Y(K)-Y0(N))**2)
                SSXZ(MX)  =   CTR*(X(K)-X0(N))/SQRT((X(K)-X0(N))**2+(Y(K)-
             *                Y0(N))**2)
                SSYZ(MX)  =  CTR*(Y(K)-Y0(N))/SQRT((X(K)-X0(N))**2+(Y(K)-
             *                Y0(N))**2)
        90      CONTINUE
                RETURN
                END
```

```
                                                              Page    1
                                                              03-12-91
                                                              06:38:32
D Line# 1      7                     Microsoft FORTRAN77 V3.30 March 1985
     $DEBUG
     CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
     C  SUBROUTINE LEASQ -DETERMINE RELATIONSHIP BETWEEN
     C                    1) A.C. SURFACE THICKNESS (hl) AND Na
     C                    2) A.C. SURFACE THICKNESS (hl) AND Np
     C                    BY LEAST SQUARE METHOD (POLYNOMIAL DEGREE=4)
     C                  - DETERMIN INTERSECTION BET. hl-Na AND hl-Np
     C                    BY  NEWTON-RAPHSON ITERATIVE METHOD
     CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
           SUBROUTINE LEASQ(HL,PTL,ATL,DHT,DNT)
           DIMENSION HL(5),ATL(5),PTL(5)
           DIMENSION A(5,6),X(6)
           DIMENSION SL(9),SS(5)
           REAL*4 L1,L2,L3,L4,N1,N
           REAL*4 LA(4),LB(4),LC(4),LD(4),LE(4)

              DO 901 K=1,2
1                SL(1) = 5.0
1                DO 1000 KK = 2,9
2                   SL(KK)=0.0
2    1000        CONTINUE
1                DO 1010 KK = 1,5
2                   SS(KK)=0.0
2    1010        CONTINUE
1                L1 = 0.0
1                L2 = 0.0
1                L3 = 0.0
1                L4 = 0.0
1                L5 = 0.0
1                LA(K) = 0.0
1                LB(K) = 0.0
1                LC(K) = 0.0
1                LD(K) = 0.0
1                LE(K) = 0.0
1                IF (K.EQ.1) THEN
1                DO 1020 KK = 2,9
2                   DO 1020  M = 1,5
3                      SL(KK)=SL(KK)+PTL(M)**(KK-1)
3    1020           CONTINUE
1                   DO 906  KK = 1,5
2                       DO 906 M = 1,5
3                          SS(KK)=SS(KK)+HL(M)*PTL(M)**(KK-1)
3    906                CONTINUE
1                ENDIF
1                IF (K.EQ.2) THEN
1                DO 1030 KK=2,9
2                DO 1030 M = 1,5
3                   SL(KK)=SL(KK)+ATL(M)**(KK-1)
3    1030        CONTINUE
1                DO 908  KK = 1,5
2                DO 908 M = 1,5
3                   SS(KK)=SS(KK)+HL(M)*ATL(M)**(KK-1)
3    908         CONTINUE
1                ENDIF
1                DO 1060 KK = 1,5
2                DO 1060 KM = 1,6
```

```
                                                              Page    2
                                                              03-12-91
                                                              06:38:32
                                 Microsoft FORTRAN77 V3.30 March 1985
D Line# 1      7
3                       IF(KM.EQ.6) A(KK,KM)=SS(KK)
3                       IF(KM.LT.6)A(KK,KM)= SL(KK+KM-1)
3        1060   CONTINUE
1                 DO 960 NK=1,5
2                 DO 962 NP=NK,5
3                   D=A(NP,NK)
3                   DO 962 NJ=NK,6
4                     A(NP,NJ)=A(NP,NJ)/D
4        962        CONTINUE
2                   IF(NK.EQ.5)GOTO 960
2                   DO 966 NP=NK+1,5
3                     DO 966 NJ=NK,6
4                     A(NP,NJ)=A(NP,NJ)-A(NK,NJ)
4        966        CONTINUE
2        960      CONTINUE
1                 X(6)=0.0
1                 DO 970 NK=5,1,-1
2                   X(NK)=A(NK,6)
2                   DO 970 NW=NK,5
3                     X(NK)=X(NK)-A(NK,NW+1)*X(NW+1)
3        970        CONTINUE
1                   LA(K)=X(1)
1                   LB(K)=X(2)
1                   LC(K)=X(3)
1                   LD(K)=X(4)
1                   LE(K)=X(5)
1        901      CONTINUE
                  K =1
                  NN=0
                  N = (ATL(2)+PTL(2))/2.0
         920      CONTINUE
                  FN=(LA(K)-LA(K+1))+(LB(K)-LB(K+1))*N+(LC(K)-LC(K+1))*N**2
               *       +(LD(K)-LD(K+1))*N**3+(LE(K)-LE(K+1))*N**4
                  FDN=(LB(K)-LB(K+1))+2.0*(LC(K)-LC(K+1))*N+
               *        3.0*(LD(K)-LD(K+1))*N**2+4.0*(LE(K)-LE(K+1))*N**3
                  N1=N-FN/FDN
                  H1=LA(K)+LB(K)*N+LC(K)*N**2+LD(K)*N**3+LE(K)*N**4
                  H2=LA(K+1)+LB(K+1)*N+LC(K+1)*N**2+LD(K+1)*N**3+LE(K+1)*N**4
                  IF(ABS(H1-H2).GT.0.001.OR.NN.LT.100)GOTO 921
                  GO TO 923
         921      CONTINUE
                  NN=NN+1
                  N=N1
                  GOTO 920
         923      CONTINUE
                  DHT = H1
                  DNT = N

                RETURN
                END
```

2 โปรแกรมออกแบบ B (Rutting)

ประกอบด้วย

1) Main Program
2) Subroutine PART
3) Subroutine BESSEL
4) Subroutine SSR (วิเคราะห์ความเค้น)
5) Subroutine LEASQ

```
$DEBUG
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C PROGRAM B -THICKNESS DESIGN OF AIRPORT 3-LAYER FLEXIBLE PAVEMENT
C           -DESIGN TO LIMIT MAX. VERTICAL COMPRESSIVE STRAIN AT
C            THE TOP OF SUBGRADE
C            (CONSIDER RUTTING OF SUBGRADE)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      COMMON  ITN,BZ(51),H(2),RR(6),E(3),V(3),AZ(6,200),
     *         RJ0(6,200),RJ1(6,200),AJ(6,200),L,LI,J,K,
     *         X(25),Y(25),Z,R(12),X0(6),Y0(6),TP(12),
     *         SSZ(6),SSX(6),SSY(6),SSXY(6),SSXZ(6),SSYZ(6),
     *         NTEST(6),A(200,3),B(200,3),C(200,3),D(200,3)
      DIMENSION HA(10),HB(10),NPS(12),NT(12),DT(12),DD(12),NM(12)
      DIMENSION DX(2,12),XWT(12),SFC(3,61),FRC(12)
      DIMENSION FF(81),F2(12,61),HH(5)
      DIMENSION P1(6,280),XVCSN(12,5),VCSN(25),TR(3,3)
      DIMENSION CZ(25),CX(25),CY(25),NPC(5),CF(12)
      DIMENSION NAC(5),PC(5),AC(5),NC(12,31),NNC(61)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C 1) MATERIAL PROPERTIES
C     EI   = EXPERIMENTALLY DETERMINED COEF. OF N-TH LAYER
C     E(N) = ELASTIC MODULUS OF N-TH LAYER
C     V(N) = POISSON'S RATIO OF N-TH LAYER
C     VB   = VOLUME OF BITUMEN IN PERCENT
C     VV   = VOLUME OF VOID IN PERCENT
C     VS   = ABSOLUTE VISCOSITY AT 70 F.,poisex10**6
C 2) AVT  = AVERAGE MEAN ANNUAL AIR TEMPERATURE
C 3)  AIRCRAFT CHARACTERISTICS
C     NA       = TOTAL TYPE OF DESIGN AIRCRAFT
C     MODEL(J) = J-TH AIRCRAFT MODEL
C     NPS(J)   = NUMBER OF RUNWAY PASSES OF J-TH AIRCRAFT
C     NT(J)    = NUMBER OF TRAILING TIRE ROWS IN MAIN GEAR
C     NM(J)    = 1 (4,8 MAIN GEAR TIRE)
C              = 2 (16 MAIN GEAR TIRE)
C     DT(I,J)  = LATERAL DISTANCE BETWEEN MAIN GEAR AND AIRCRAFT CENTERLI
C     XWT(J)   = MAXIMUM WEIGHT PER TIRE(MAIN GEAR)
C     TP(J)    = TIRE PRESSURE (MAIN GEAR)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
 8000 FORMAT(' AIRCRAFT CHARACTERISTICS',/)
 8030 FORMAT(5X,I2,'.','AIRCRAFT MODEL',2X,A10)
 8040 FORMAT(5X,I2,'.','STANDARD AIRCRAFT DC-8-63F')
 8060 FORMAT(10X,'MOVEMENTS IN DESIGN PERIOD = ',I6)
 8090 FORMAT(10X,'TYPE OF GEAR ASSEMBLIES     = SINGLE-TRICYCLE')
 8120 FORMAT(10X,'TYPE OF GEAR ASSEMBLIES     = DOUBLE-TRICYCLE')
 8150 FORMAT(10X,'TYPE OF TIRE ASSEMBLIES     = TWIN')
 8180 FORMAT(10X,'TYPE OF TIRE ASSEMBLIES     = TWIN-TANDEM')
 8210 FORMAT(10X,'TYPE OF TIRE ASSEMBLIES     = TRIPLE-TANDEM')
 8240 FORMAT(10X,'TIRE SPACING (in.)          =',F4.1)
 8270 FORMAT(10X,'TIRE SPACING (in.X in.)     =',F4.1,'X',F4.1)
 8300 FORMAT(10X,'X1 (in.)                    =',F5.1)
 8360 FORMAT(10X,'X2 (in.)                    =',F5.1)
 8390 FORMAT(10X,'MAX. WEIGHT PER TIRE (lb)   =',F7.1,/)
 8420 FORMAT(10X,'TIRE PRESSURE (psi)         =',F5.1)
 9000 FORMAT (A)
 9010 FORMAT (A40)
 9020 FORMAT (F4.1)
```

```
D Line# 1     7
      9025  FORMAT (F5.1)
      9030  FORMAT (A35,A8)
      9035  FORMAT (3(1X,F5.2),1X,F7.1)
      9040  FORMAT (F8.1)
      9050  FORMAT (3(F4.2,1X))
      9055  FORMAT (I2)
      9060  FORMAT (A10,1X,I6,2(1X,I1),2(1X,F4.1),2(1X,F5.2),1X,F7.1,1X,
           *F5.1)
      9061  FORMAT(2(1X,I1),2(1X,F4.1),2(1X,F5.2),1X,F7.1,1X,F5.1)
      9065  FORMAT (80('$'))
      9070  FORMAT (25X,'THICKNESS  DESIGN')
      9075  FORMAT (20X,'COMPRESSIVE STRAIN CRITERION')
      9076  FORMAT (10X,'DETERMINE COMPRESSIVE STRAIN AT TOP OF',
           *' SUBGRADE')
      9080  FORMAT (15X,19HSTRUCTURAL RESPONSE)
      9085  FORMAT ('+',14X,19('_'))
      9090  FORMAT (1X,'DESIGN  AVERAGE MEAN ANNUAL AIR TEMPERATURAL(F)=',
           *F4.1)
      9092  FORMAT(1X,'EXPERIMENTALLY DETERMINED COEF. OF GRANULAR BASE',
           *'(psi) =',F8.1)
      9095  FORMAT (' DESIGN SUBGRADE RESILIENT MODULUS (psi)=',F8.1)
      9100  FORMAT (' GRANULAR BASE THICKNESS(in.)=',F5.1,/)
      9200  FORMAT(2X,I2,'.','LAYER AND MATERIAL PROPERTIES')
      9210  FORMAT ('+',2X,31('_'))
      9220  FORMAT (5X,'LAYER    MATERIAL   POISSON`S   THICKNESS')
      9230  FORMAT (5X,'NUMBER     TYPE        RATIO       (in.)',/)
      9240  FORMAT (6X,' 1     ASPH. CONC.       ',F4.2,7X,F5.1)
      9250  FORMAT (6X,' 2     GRANU. BASE       ',F4.2,7X,F5.1)
      9255  FORMAT (6X,' 3     SUBGR. SOIL       ',F4.2,/)
      9260  FORMAT(2X,'**',I2,'.',I2,'.',A10,1X,'MOVEMENTS IN  DESIGN'
           *' PEROID=',I6)
      9264  FORMAT ('+',4X,48('_'),/)
      9265  FORMAT (2X,'**',I2,'.',I2,'.',3X,'STANDARD AIRCRAFT DC-8-63F')
      9290  FORMAT (2X,'ELASTIC MODULUS OF ASPHALTIC CONCRETE(psi)=',F8.1)
      9300  FORMAT (2X,'ELASTIC MODULUS OF GRANULAR BASE(psi)    =',F8.1)
      9330  FORMAT (44(' '))
      9340  FORMAT (' COMPUTATIONAL ',8X,'STRESS(psi)')
      9350  FORMAT(' POINT (in)',3X,30('_'))
      9355  FORMAT('    X      Y   ','    Z              Y             X      ')
      9363  FORMAT (2(1X,F5.2),3(E10.4,1X))
      9365  FORMAT(' VERTICAL COMPRESSIVE STRAIN(in/in)=',E13.7,/)
      9380  FORMAT ('  MAXIMUM VERTICAL COMPRESSIVE STRAIN =',E13.7,//)
      9385  FORMAT (//,30X,23HPREDICTED TRAFFIC VALUE)
      9386  FORMAT ('+',29X,23('_'),//)
      9387  FORMAT (80(' '))
      9400  FORMAT (' AIRCRAFT MODEL     NUMBER OF STRAIN REPETITION AT',
           *' DISTANCE X(ft) FROM CENTERLINE')
      9405  FORMAT (14X,67('_'))
      9410  FORMAT (13X,'     1      3      6      10     12     15',
           *'     18     20     22     25'  )
      9420  FORMAT (A14,10I6)
      9430  FORMAT (' TOTAL',8X,10I6)
      9445  FORMAT (//,30X,'ALLOWABLE TRAFFIC VALUE')
      9447  FORMAT ('+',29X,23('_'),//)
      9460  FORMAT (' NUMBER OF STRAIN REPETITION(COMP. STRAIN CRITERIA)='
           *I9,/)
```

```
                                                              Page    3
                                                              05-02-91
                                                              09:34:37
D Line# 1     7                            Microsoft FORTRAN77 V3.30 March 1985
        9540  FORMAT(' NUMBER OF ALLOWABLE COMPRESIVRE STRAIN REPETITION',I9,//)
        9600  FORMAT(80('='),/)
        9800  FORMAT(' THE FINAL A.C. THICKNESS IS ',F5.2,' in. ')
              CHARACTER*50 FILEIN,FILEOT,TEMP,MAT,STAC,MODEL(12)
              WRITE (*,'(A\)') '  NAME OF DATA FILE:  '
              READ (*,9000) FILEIN
              WRITE(*,'(A\)') '  NAME OF OUTPUT FILE:  '
              READ(*,9000) FILEOT
              OPEN(UNIT=4,FILE=FILEIN,STATUS='OLD')
              OPEN(UNIT=8,FILE=FILEOT,STATUS='NEW')
        C**** READ IN AIRCRAFT CHARACTERISTICS AND MATERIAL PROPERTIES
              READ(4,9010) TEMP
              READ(4,9020) AVT
              READ(4,9030) MAT
              READ(4,9035)P200,VB,VV,VS
              READ(4,9040) EI
              READ(4,9040) E(3)
              READ(4,9050) V(1),V(2),V(3)
              READ(4,9055)ITN
              READ(4,9055) NA
              READ(4,9030) AAIRCH
              DO 10 J = 1,NA
1                READ(4,9060) MODEL(J),NPS(J),NT(J),NM(J),DT(J),
1            *                DD(J),DX(1,J),DX(2,J),XWT(J),TP(J)
1                R(J)=SQRT(XWT(J)/(TP(J)*3.141592654))
1
1       10    CONTINUE
        20    NA1 = NA+1
        C**** READ IN STANDARD AIRCRAFT CHARACTERISTICS
              OPEN(UNIT=5,FILE = 'STR_AIR.DAT',STATUS='OLD')
              READ(5,9030) STAC,MODEL(NA1)
              READ(5,9061)NT(NA1),NM(NA1),DT(NA1),
             *DD(NA1),DX(1,NA1),DX(2,NA1),XWT(NA1),TP(NA1)
              R(NA1)=SQRT(XWT(NA1)/(TP(NA1)*3.141592654))
        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        C DETERMINE TRANSVERSE DISTRIBUTION FACTOR
        C (VERTICAL COMPRESSIVE STRAIN CRITERIA)
        C F(J,MX)=TRANSVERSE DISTRIBUTION FACTOR
        C        OF J-TH AIRCRAFT AT DISTANCE MX(ft)
        C        FROM RUNWAY CENTERLINE
        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
              DO 500 J =1,NA
1                W2 = DD(J)/2.0+(SQRT(XWT(J)/(TP(J)*1.452)/2.0))
1                DO 501 I = 1,NM(J)
2                   S = DX(I,J)
2                   DO 555 MX=-30,30
3                      B1=0.5*0.57735027+MX
3                      B2=-0.5*0.57735027+MX
3                     FQ1=EXP((-.5)*(((B1-S)/3.5)**2))/(3.5*SQRT(2.0*3.1415927))
3                     FQ2=EXP((-.5)*(((B2-S)/3.5)**2))/(3.5*SQRT(2.0*3.1415927))
3                      FF(MX)=0.5*(FQ1+FQ2)
3       555         CONTINUE
2                   W = W2/12.0
2                   IF(W.LE.0.5) N=0
2                   IF(W.GT.0.5) N=INT(W+0.5)
2                   RD=W-0.5-AINT(W-0.5)
```

```
D Line# 1     7
2
2                    DO 505 MX = -30,30
3                       SUMFC=0.0
3                       N1=(-1)*N
3                       DO 504 MI = N1,N
4                          IF(ABS(MX+MI).GT.30) GO TO 566
4                          IF(ABS(MI).EQ.N )SUMFC=SUMFC+FF(MX+MI)*RD
4                          IF(ABS(MI).LT.N) SUMFC=SUMFC+FF(MX+MI)
4          566             CONTINUE
4          504          CONTINUE
3                       SFC(I,MX)=SUMFC
3          505       CONTINUE
2          501    CONTINUE
1                 DO 527 MX = -30,30
2                    TOTF2 = 0.0
2                    DO 507 I = 1,NM(J)
3                       TOTF2 = TOTF2 +(SFC(I,MX)+SFC(I,-MX))*NT(J)
3          507       CONTINUE
2                    F2(J,MX)=TOTF2
2          527    CONTINUE
1          500  CONTINUE
           CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
           C ****ANALYSE STRESS AND STRAIN
           CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
           C*****COMPUTE ZEROS OF J1(X) AND J0(X) SET UP GAUSS CONSTANT
                ITN4 = ITN*4
                K =ITN+1
                BZ(1) = 0.0
                BZ(2) = 1.0
                BZ(3) = 2.4048
                BZ(4) = 3.8317
                BZ(5) = 5.5201
                BZ(6) = 7.0156
                DO 700 I = 7,K,2
1                  T = I/2
1                  TD=4.0*T-1.
1                  BZ(I)=3.1415927*(T-0.25+0.050661/TD-0.053041/TD**3+
1               *          0.262051/TD**5)
1          700  CONTINUE
                DO 710 I = 8,ITN,2
1                  T = (I-2)/2
1                  TD = 4.0*T+1.
1                  BZ(I) =3.1415927*(T+0.25-0.151982/TD+0.015399/TD**3
1               *          -0.245270/TD**5)
1          710  CONTINUE
           C**** PRINT OUT INPUT DATA
                WRITE(8,9065)
                WRITE(8,9070)
                WRITE(8,9075)
                WRITE(8,9076)
                WRITE(8,9065)
                WRITE(8,9090) AVT
                WRITE(8,9092) EI
                WRITE(8,9095) E(3)
           C**** DETERMINE AVERAGE MEAN ANNUAL PAVEMENT TEMPERATURE FROM AIR TEMP.
                AVT=AVT*1.05+5.0
```

```
        WRITE(8,8000)
  C**** PRINT OUT AIRCRAFT CHARACTERISTIC
        DO 699 J=1,NA1
1       IF(J.LT.NA1) WRITE(8,8030) J,MODEL(J)
1       IF(J.EQ.NA1) WRITE(8,8040) J
1       IF(J.LT.NA1) WRITE(8,8060) NPS(J)
1       IF(NM(J).EQ.1)WRITE(8,8090)
1       IF(NM(J).EQ.2)WRITE(8,8120)
1       IF(NT(J).EQ.1)WRITE(8,8150)
1       IF(NT(J).EQ.2)WRITE(8,8180)
1       IF(NT(J).EQ.3)WRITE(8,8210)
1       IF(NT(J).EQ.1)WRITE(8,8240) DD(J)
1       IF(NT(J).GT.1)WRITE(8,8270) DD(J),DT(J)
1       WRITE(8,8300) DX(1,J)
1       IF(NM(J).EQ.2)WRITE(8,8360) DX(2,J)
1       WRITE(8,8420) TP(J)
1       WRITE(8,8390) XWT(J)
1 699   CONTINUE
  C**** READ IN A.C. SURFACE AND BASE THICKNESS
         IMH=0
         DO 601 MH=1,10
1        IMH=IMH+1
1       WRITE(*,'(A\)') ' INPUT GARNULAR BASE THICKNESS(ins.)=?'
1       READ(*,9025)HB(MH)
1       H2=HB(MH)
1       WRITE(8,9100)H2
1       DO 602 M = 1,5
2          IF(M.GE.2)WRITE(*,*)'NO.   H1       Na        Np '
2           DO 444 III=1,(M-1)
3              IF(M.GT.1)WRITE(*,2010)III,HH(III),NAC(III),NPC(III)
3 2010         FORMAT(1X,I2,2X,F5.2,2X,I8,2X,I8)
3 444       CONTINUE
2           WRITE(8,9200) M
2           WRITE(8,9210)
2           WRITE(8,9220)
2           WRITE(8,9230)
2 C**** START NEW A.C. SURFACE THICKNESS
2           WRITE(*,'(A\)')' INPUT A.C. SURFACE THICKNESS(ins.)=?'
2           READ (*,9025) HH(M)
2 C**** PRINT OUT LAYER AND MATERIAL PROPERTIES
2           WRITE(8,9240) V(1),HH(M)
2           WRITE(8,9250) V(2),H2
2           WRITE(8,9255) V(3)
2 C**** SET DEPTH OF EACH LAYER
2           H(1) = HH(M)
2           H(2) = H(1)+H2
2 C**** PRINT OUT AIRCRAFT CHARACTERISTICS
2           DO 604 J = 1,NA1
3              IF(J.EQ.NA1)WRITE(8,9265) M,J
3              IF(J.LT.NA1)WRITE(8,9260) M,J,MODEL(J),NPS(J)
3              WRITE (8,9264)
3              XVCSN(J,M) = 0.0
3 C***** SET LOADING FREQUENCY
3              IF(NT(J).EQ.1)FRC(J)=4.0
3              IF(NT(J).EQ.2)FRC(J)=2.0
3              IF(NT(J).EQ.3)FRC(J)=1.0
```

```
3   C**** DETERMINE ELASTIC MODULUS OF A.C. LAYER
3                   B5=1.3+0.49825*LOG10(FRC(J))
3                   B4=VB
3                   B3=0.553833+0.028829*P200/(FRC(J)**0.1703)-0.03476*VV+
3        *             0.070377*VS+0.93175/(FRC(J)**0.02774)
3                   B2=(B4**0.5)*(AVT**B5)
3                   B1=B3+0.000005*B2-0.00189*B2/(FRC(J)**1.1)
3                   E(1)=100000.0*(10.0**B1)
3   C**** SET UP CENTER OF TIRES AND COMPUTATIONAL POINTS
3                   IF(NT(J).EQ.3)THEN
3                     NR = 6
3                     KK = 5
3                     X0(1) = 0.0
3                     X0(2) =0.0
3                     X0(3) = DD(J)
3                     X0(4) = DD(J)
3                     X0(5) = DD(J)
3                     X0(6) = 0.0
3                     Y0(1) = 0.0
3                     Y0(2) = -DT(J)
3                     Y0(3) = -DT(J)
3                     Y0(4) = 0.0
3                     Y0(5) = DT(J)
3                     Y0(6) = DT(J)
3                   ENDIF
3                   IF(NT(J).EQ.2)THEN
3                     NR = 4
3                     KK = 5
3                     X0(1) = 0.0
3                     X0(2) = DD(J)
3                     X0(3) = DD(J)
3                     X0(4) = 0.0
3                     Y0(1) = 0.0
3                     Y0(2) = 0.0
3                     Y0(3) = DT(J)
3                     Y0(4) = DT(J)
3                   ENDIF
3                   IF(NT(J).EQ.1)THEN
3                     NR = 2
3                     KK =3
3                     X0(1) = 0.0
3                     X0(2) = DD(J)
3                     Y0(1)=0.0
3                     Y0(2) = 0.0
3                     X(2)=R(J)
3                     Y(2)=0.0
3                     X(3)=DD(J)/2.0
3                     Y(3)=0.0
3                    ENDIF
3                    IF(NT(J).EQ.2.OR.NT(J).EQ.3)THEN
3                      IF(DD(J).LT.DT(J))THEN
3                      X(2)=R(J)
3                      Y(2)=0.0
3                      X(3)=DD(J)/2.0
3                      Y(3)=0.0
3                    ENDIF
```

```
D Line# 1      7
3                        IF(DD(J).GE.DT(J))THEN
3                          X(2)=0.0
3                          Y(2)=R(J)
3                          X(3)=0.0
3                          Y(3)=DT(J)/2.0
3                        ENDIF
3                        ENDIF
3                        X(1)=0.0
3                        Y(1)=0.0
3         C**** COMPUTE DISTANCE BET. CENTER OF N-TH TIRE AND K-TH COMPUTATIONAL P
3                        DO 606 K = 1,3
4                            DO 1234 NN=1,NR
5                                RR(NN) = SQRT((X(K)-X0(NN))**2+(Y(K)-Y0(NN))**2)
5         1234               CONTINUE
4         C**** DETERMINE ELASTIC MODULUS OF GRANULAR BASE
4                            E(2) = (10.447)*(E(3)**0.287)*(EI**0.868)/
4             *              ((HH(M)**0.471)*(H2**0.041)*(E(1)**0.139))
4                            IF(K.EQ.1) THEN
4                            WRITE(8,9080)
4                            WRITE(8,9085)
4                            ENDIF
4                            IF(K.EQ.1) THEN
4                            WRITE(8,9290) E(1)
4                            WRITE(8,9300) E(2)
4                            WRITE(8,9330)
4                            WRITE(8,9340)
4                            WRITE(8,9350)
4                            WRITE(8,9355)
4                            WRITE(8,9330)
4                            ENDIF
4         C**** SET DEPTH WHERE STRESS AND STRAIN ARE REQ'D
4                            Z = H(2)
4                            CZ(K) = 0.0
4                            CX(K) = 0.0
4                            CY(K) = 0.0
4                            DO 810 N=1,NR
5         C**** CALCULATE THE PARTITION
5                                CALL PART(N)
5                                DO 777 LI=1,ITN4
6                                    P=AZ(N,LI)
6         C**** CALCULATE THE COEFICIENTS
6                                    CALL COEF(N,P)
6                                    IF(RR(N))115,115,110
6         110                        PR=P*RR(N)
6                                    CALL BESSEL(0,PR,Y1)
6                                    RJ0(N,LI)=Y1
6                                    CALL BESSEL(1,PR,Y1)
6                                    RJ1(N,LI)=Y1
6         115                        PA=P*R(J)
6                                    CALL BESSEL(1,PA,Y1)
6                                    AJ(N,LI)=Y1
6
6         777                    CONTINUE
5         C**** SET LAYER (L) WHERE STRESS AND STRAIN ARE REQ'D
5                                L = 3
5         C**** CALCULATE STRESS FROM N-TH TIRE
```

```
2              WRITE(8,9460) NPC(M)
2              WRITE(8,9387)
2              WRITE (8,9445)
2              WRITE (8,9447)
2   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2   C DETERMINE NUMBER OF ALLOWABLE DC-8-63F STRAIN REPETITION(Na)
2   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2              NAC(M) = INT(10.0**(248.14810177-485.45078636*XN
2        *                +364.20477903*(XN**2)-122.003*(XN**3)
2        *                +15.29*(XN**4)))
2              WRITE(8,9540)NAC(M)
2              AC(M)=LOG10(FLOAT(NAC(M)))
2   602     CONTINUE
1   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
1   C DETERMINE RELATIONSHIP BETWEEN
1   C  -A.C. SURFACE THICKNESS(hl) AND LOG(Np)
1   C  -A.C. SURFACE THICKNESS(hl) AND LOG(Na)
1   C BY LEAST SQUARE METHOD
1   C AND DETERMINE INTERSECTION OF hl-LOG(Na) AND hl-LOG(Np)
1   C NEEDED TO SATISFY THE ASPHALT CONCRETE COMPRESSIVE STRAIN CRITION
1   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
1         CALL LEASQ(HH,PC,AC,DHC,DNC)
1         WRITE(8,9600)
1         NCD=INT(10.0**(DNC))
1         WRITE(8,9800)DHC
1         WRITE(8,*)'   '
1         WRITE(8,*)'################################### END ################
1        *##############'
1         WRITE(8,*)'   '
1         HA(MH)=DHC
1         WRITE(*,*)'PRESS  1 => SELECT NEW GRANULAR BASE THICKNESS'
1         WRITE(*,*)'       2 => QUIT'
1         WRITE(*,'(A\)')' SELECT ============> '
1         READ(*,*)NNN
1         IF(NNN.EQ.1)GOTO 601
1         IF(NNN.GT.1)GOTO 311
1   601   CONTINUE
    311   CONTINUE
          WRITE(*,*)'    NO.      h2(ins.)      hl(ins.)'
          DO 312 I=1,IMH
1   1        FORMAT(4X,I2,7X,F6.2,5X,F6.2)
1            WRITE(*,1)I,HB(I),HA(I)
1   312   CONTINUE
          STOP
          END
```

```
D Line# 1     7
5                     CALL SSR(N)
5                     CZ(K) = CZ(K) + SSZ(N)
5                     CX(K) = CX(K) + SSX(N)
5                     CY(K) = CY(K) + SSY(N)
5       810        CONTINUE
4       C**** CALCULATE VERTICAL COMPRESSIVE STRAIN
4                  VCSN(K) = (1/E(L))*(CZ(K)-V(L)*(CX(K)+CY(K)))
4                  IF(XVCSN(J,M)-VCSN(K).LT. 0.0000001)GO TO 811
4                  XVCSN(J,M) = VCSN(K)
4       811        CONTINUE
4       C**** PRINT OUT COMPUTATIONAL POINT,STRESS AND STRAIN
4                  WRITE(8,9363)X(K),Y(K),CZ(K),CX(K),CY(K)
4                  WRITE(8,9365)VCSN(K)
4       606     CONTINUE
3               WRITE(8,9330)
3               WRITE(8,9380) XVCSN(J,M)
3       604  CONTINUE
2            WRITE(8,9385)
2            WRITE (8,9386)
2            WRITE(8,9387)
2            WRITE(8,9400)
2            WRITE(8,9405)
2            WRITE(8,9410)
2            WRITE(8,9387)
2            NPC(M) = 0
2       CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2       C DETERMINE NO. OF PREDICTED EQUIVALENT DC-8-63F STRAIN REPETITIONS(Np)
2       CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2               DO 916 NX = 0,30
3                  NNC(NX) = 0
3                  DO 912 J = 1,NA
4       C**** CALCULATE EQUIVALENT TRAFFIC VALUE
4                    XV=ABS(XVCSN(J,M))*1000.0
4                    XN=ABS(XVCSN(NA1,M))*1000.0
4            C1=248.14810177-485.45078636*XV+364.20477903*(XV**2)-122.003*
4           *    (XV**3)+15.29*(XV**4)
4            C2=248.14810177-485.45078636*XN+364.20477903*(XN**2)-122.003*
4           *    (XN**3)+15.29*(XN**4)
4                    CF(J)=(10.0**C2)/(10.0**C1)
4                    NC(J,NX) = NPS(J)*CF(J)*F2(J,NX)
4                    NNC(NX) = NNC(NX)+NC(J,NX)
4
4
4       912          CONTINUE
3                    IF(NPC(M) .GT. NNC(NX)) GO TO 916
3                    NPC(M) = NNC(NX)
3       916       CONTINUE
2                 DO 902 J=1,NA
3                    WRITE(8,9420)MODEL(J),NC(J,1),NC(J,3),NC(J,6),NC(J,10),
3           *        NC(J,12),NC(J,15),NC(J,18),NC(J,20),NC(J,22),NC(J,25)
3       902       CONTINUE
2                 WRITE(8,9387)
2                 WRITE(8,9430)NNC(1),NNC(3),NNC(6),NNC(10),NNC(12),NNC(15),
2           *     NNC(18),NNC(20),NNC(22),NNC(25)
2                 WRITE(8,9387)
2                 PC(M)=LOG10(FLOAT(NPC(M)))
```

FILE STR_AIR.DAT

```
STANDARD AIRCRAFT                        DC-8-63F
 2 1 55.0 32.0 17.50 00.00 43000.0 196.0 4.0
```

```
D Line# 1      7
        $DEBUG
              SUBROUTINE PART(N)
        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        C       COMPUTE POINTS FOR LEGENDRE-GAUSS INTREGRATION
        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
              COMMON ITN,BZ(71),H(2),RR(6),E(3),V(3),AZ(6,280),
             *       RJ0(6,280),RJ1(6,280),AJ(6,280),L,LI,J,K,
             *       X(25),Y(25),Z,R(12),X0(6),Y0(6),TP(12),
             *       SSZ(6),SSX(6),SSY(6),SSXY(6),SSXZ(6),SSYZ(6),
             *       NTEST(6),A(280,3),B(280,3),C(280,3),D(280,3)
              DATA G1/8.6113631E-1/,G2/3.3998104E-1/
              MX = N
              NI = J
              ZF = R(NI)
              NTEST(MX) = 2
              IF (RR(MX))8,8,9
        9     CONTINUE
              NTEST (MX) =R(NI)/RR(MX)+0.0001
              IF (NTEST(MX))6,6,5
        6     CONTINUE
              NTEST(MX) = RR(MX)/R(NI)+0.0001
              ZF  = RR(MX)
        5     CONTINUE
              NTEST(MX) = NTEST(MX)+1
              IF (NTEST(MX)-10)8,8,7
        7     CONTINUE
              NTEST(MX) =10
        8     CONTINUE
              MK =1
              ZF = 2*ZF
              SZ2=0.0
              DO 28 N1 = 1,ITN
1                SZ1 =SZ2
1                SZ2 = BZ(N1+1)/ZF
1                SF = SZ2-SZ1
1                PP = SZ2+SZ1
1                SG1 = SF*G1
1                SG2 = SF*G2
1                AZ(MX,MK)=PP-SG1
1                AZ(MX,MK+1) = PP-SG2
1                AZ(MX,MK+2)=PP+SG2
1                AZ(MX,MK+3) = PP+SG1
1                MK =MK+4
1       28    CONTINUE
              RETURN
              END
```

```
D Line# 1      7
    $DEBUG
          SUBROUTINE BESSEL(NI,XI,Y)
    CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
    C     CALCULATE BESSEL FUNTION ORDER 1 AND 0
    CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
          DIMENSION D(20),PZ(6),QZ(6),P1(6),Q1(6)
          DOUBLE PRECISION T4,X,PI
          DATA PZ/1.0E0,-1.125E-4,2.8710938E-7,-2.3449658E-9,
         *         3.9806841E-11,-1.1536133E-12/
          DATA QZ/-5.0E-3,4.6875E-6,-2.3255859E-8,2.8307087E-10,
         *         -6.3912096E-12,2.3124704E-12/
          DATA P1/1.0E0,1.875E-4,-3.6914063E-7,2.7713232E-9,
         *         -4.5114421E-11,1.2750463E-12/
          DATA Q1/1.5E-2,-6.5625E-6,2.8423828E-8,
         *        -3.2662024E-10,7.1431166E-12,-2.5327056E-13/
          PI=3.1415927
    9     N =NI
          X = XI
          IF (X-7.0) 10,10,160
    10    X2=X/2.0
          FAC = -X2*X2
          IF(N) 11,11,14
    11    C=1.0
          Y = C
          DO 13 I=1,34
1         T = I
1         C = FAC*C/(T*T)
1         TEST = ABS (C) - 10.0**(-8)
1         IF (TEST) 17,17,12
1   12    Y = Y+C
1   13    CONTINUE
    14    C = X2
          Y=C
          DO 16 I = 1,34
1         T = I
1         C = FAC*C/(T*(T+1.0))
1         TEST = ABS(C)-10.0**(-8)
1         IF(TEST)17,17,15
1   15    Y = Y+C
1   16    CONTINUE
    17    RETURN
    160   IF(N)161,161,164
    161   DO 162 I = 1,6
1         D(I) = PZ(I)
1         D(I+10) = QZ(I)
1   162   CONTINUE
          GOTO 163
    164   DO 165 I =1,6
1         D(I) = P1(I)
1         D(I+10) = Q1(I)
1   165   CONTINUE
    163   CONTINUE
          T1 = 25.0/X
          T2 = T1*T1
          P = D(6)*T2+D(5)
          DO 170 I =1,4
```

```
D Line# 1      7
1                J = 5-I
1                P = P*T2+D(J)
1         170    CONTINUE
                 Q = D(16)*T2+D(15)
                 DO 171 I =1,4
1                J = 5-I
1                Q = Q*T2+D(J+10)
1         171    CONTINUE
                 Q = Q*T1
          C
                 T4 = DSQRT(X*PI)
                 T6 = SIN(X)
                 T7 = COS(X)
          C
                 IF(N) 180,180,185
          180    T5 = ((P-Q)*T6+(P+Q)*T7)/T4
                 GOTO 99
          185    T5 = ((P+Q)*T6-(P-Q)*T7)/T4
          99     Y = T5
                 RETURN
                 END
```

```
D Line# 1       7                          Microsoft FORTRAN77 V3.30 March 1985
$DEBUG
      SUBROUTINE  COEF(N,P)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     USE FOR ALL PROBLEM, UP TO MAX. DIMENSION OF 3-LAYER
C     CALCULATE COEF. A,B,C AND D OF 3-LAYER ELASTIC SYSTEM
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      COMMON ITN,BZ(71),H(2),RR(6),E(3),V(3),AZ(6,280),
     *        RJ0(6,280),RJ1(6,280),AJ(6,280),L,LI,J,K,
     *        X(25),Y(25),Z,R(12),X0(6),Y0(6),TP(12),
     *        SSZ(6),SSX(6),SSY(6),SSXY(6),SSXZ(6),SSYZ(6),
     *        NTEST(6),A(280,3),B(280,3),C(280,3),D(280,3)
      DIMENSION PM(4,4,4),XP(5,4,4),FM(4),SC(4)
      REAL *4 Q(2,2)
      LC = LI
C**** SET UP MATRIX  X=DI*MI*KI*K*M*D
C     COMPUTE THE MATRICES X(K)
      DO 10 N1 = 1,2
         T1 = E(N1)*(1.+V(N1+1))/(E(N1+1)*(1.+V(N1)))
         T1M = T1-1.
         PH = P*H(N1)
         PH2 = PH*2.
         VK2 = 2.*V(N1)
         VKP2 = 2.*V(N1+1)
         VK4=2.0*VK2
         VKP4=2.0*VKP2
         VKK8 = 8.*V(N1)*V(N1+1)
C
         XP(N1,1,1) = VK4-3.-T1
         XP(N1,2,1) = 0.
         XP(N1,3,1) = T1M*(PH2-VK4+1.)
         XP(N1,4,1) = -2.*T1M*P
C
         T3 = PH2*(VK2-1.)
         T4 = VKK8+1.-3.*VKP2
         T5 = PH2*(VKP2-1.)
         T6 = VKK8+1.-3.0*VK2
C
         XP(N1,1,2) = (T3+T4-T1*(T5+T6))/P
         XP(N1,2,2) = T1*(VKP4-3.)-1.
         XP(N1,4,2) = T1M*(1.-PH2-VKP4)
C
         XP(N1,3,4) = (T3-T4-T1*(T5-T6))/P
C
         T3 = PH2*PH-VKK8+1.
         T4 = PH2*(VK2-VKP2)
C
         XP(N1,1,4) = (T3+T4+VKP2-T1*(T3+T4+VK2))/P
         XP(N1,3,2)=(-T3+T4-VKP2+T1*(T3-T4+VK2))/P
C
         XP(N1,1,3) = T1M*(1.-PH2-VK4)
         XP(N1,2,3) = 2.*T1M*P
         XP(N1,3,3) = VK4-3.-T1
         XP(N1,4,3) = 0.
C
         XP(N1,2,4) = T1M*(PH2-VKP4+1.)
         XP(N1,4,4) = T1*(VKP4-3.)-1.
```

```
D       1      7
1       10     CONTINUE
        C****  COMPUTE THE PRODUCT MATRICES PM
               SC(2) = 4.*(V(2)-1.)
               SC(1) = SC(2)*4.*(V(1)-1.)
               Q(1,1) = 1.
               Q(2,2) = 1.
               Q(1,2) = 0.
               QQ = P*2.*H(2)
               IF(QQ-80.) 15,15,16
        15     CONTINUE
               Q(1,2) = EXP(-QQ)
        C      Q(2,1) IS NOT NEEDED FOR INITIALIZING THE PM MATRIX
        16     CONTINUE
        C****  20 LOOP INITIALIZES PM(N,,)
               DO 20 MX = 1,4
1                 LL = (MX+1)/2
1                 PM(2,MX,4) = XP(2,MX,4)*Q(LL,2)
1                 PM(2,MX,3) = XP(2,MX,3)*Q(LL,2)
1       20     CONTINUE
               QQ = P*2.*H(1)
               IF (QQ-80.)22,22,23
        22     CONTINUE
               Q(2,1) = EXP(QQ)
               Q(1,2) = 1./Q(2,1)
               GOTO 24
        23     CONTINUE
               Q(1,2) = 0.
               Q(2,1) = 1.E20
        24     CONTINUE
               DO 25 MX = 1,4
1              LL = (MX+1)/2
1              DO 25 MI = 3,4
2                 PM(1,MX,MI) = (XP(1,MX,1)*PM(2,1,MI)+XP(1,MX,2)*PM(2,2,MI))*
2            *    Q(LL,1)+(XP(1,MX,3)*PM(2,3,MI)+XP(1,MX,4)*PM(2,4,MI))*Q(LL,2)
2       25     CONTINUE
        C****  SOLVE FOR C(3) AND D(3) OF SUBGRADE
               T3 = 2.*V(1)
               T4 = T3-1.0
               FM(1) = P*(PM(1,1,3)+PM(1,3,3))+T3*(PM(1,2,3)-PM(1,4,3))
               FM(2) = P*(PM(1,1,3)-PM(1,3,3))+T4*(PM(1,2,3)+PM(1,4,3))
               FM(3) = P*(PM(1,1,4)+PM(1,3,4)) + T3*(PM(1,2,4)-PM(1,4,4))
               FM(4) = P*(PM(1,1,4)-PM(1,3,4))+T4*(PM(1,2,4)+PM(1,4,4))
               WRITE(*,*) FM(4)
               DFAC = SC(1)/((FM(1)*FM(4)-FM(3)*FM(2))*P*P)
               A(LC,3) = 0.
               B(LC,3) = 0.
               C(LC,3) = -FM(3)*DFAC
               D(LC,3) = FM(1)*DFAC
        C****  BACKSOLVE FOR THE OTHER A,B,C,D
               DO 100 N1 = 1,2
1                 A(LC,N1)=(PM(N1,1,3)*C(LC,3)+PM(N1,1,4)*D(LC,3))/SC(N1)
1                 B(LC,N1)=(PM(N1,2,3)*C(LC,3)+PM(N1,2,4)*D(LC,3))/SC(N1)
1                 C(LC,N1)=(PM(N1,3,3)*C(LC,3)+PM(N1,3,4)*D(LC,3))/SC(N1)
1                 D(LC,N1)=(PM(N1,4,3)*C(LC,3)+PM(N1,4,4)*D(LC,3))/SC(N1)
1       100    CONTINUE
               RETURN
```

```
D Line# 1      7
   $DEBUG
         SUBROUTINE SSR(N)
   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
   C     THIS SUBROUTINE COMPUTE THE STRESS IN CYLINDRICAL COORDINATE
   C     AND TRANSFORM TO CARTESIAN SYSTEM BY USING TRANSFORMATION
   C     FORMULAS
   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
         COMMON ITN,BZ(71),H(2),RR(6),E(3),V(3),AZ(6,280),
        *          RJ0(6,280),RJ1(6,280),AJ(6,280),L,LI,J,K,
        *          X(25),Y(25),Z,R(12),X0(6),Y0(6),TP(12),
        *          SSZ(6),SSX(6),SSY(6),SSXY(6),SSXZ(6),SSYZ(6),
        *          NTEST(6),A(280,3),B(280,3),C(280,3),D(280,3)
        DIMENSION W(4),ARP(21),TEST(11)
        DATA  W/0.34785485,0.65214515,0.65214515,0.34785485/
        MI=J
        MX=N
        VL=2.0*V(L)
        EL=(1.0+V(L))/E(L)
        VL1 = 1.0-VL
   C**** COMPUTE THE COMPONENT OF STRESS IN CYLINDRICAL COORDINATE SYSTEM
        CSZ = 0.0
        CST = 0.0
        CSR = 0.0
        CTR = 0.0
        ARP(MI) = R(MI)*TP(MI)
        DO 1001 N1 = 1,ITN
 1 C******* INITIALIZE THE SUB-INTEGRALS
 1            RSZ = 0.0
 1            RST = 0.0
 1            RSR = 0.0
 1            RTR = 0.0
 1            NK = 4*(N1-1)
 1 C******* COMPUTE THE SUB-INTEGRALS
 1            DO 1002 NJ = 1,4
 2               J1 = NK + NJ
 2               P = AZ(MX,J1)
 2               IF(P*Z.LE.80.0)EP = EXP(P*Z)
 2               IF(P*Z.GT.80.0)EP = 1.0E20
 2               T1 = B(J1,L)*EP
 2               T2 = D(J1,L)/EP
 2               T1P = T1+T2
 2               T1M = T1-T2
 2               T1 = (A(J1,L)+B(J1,L)*Z)*EP
 2               T2 = (C(J1,L)+D(J1,L)*Z)/EP
 2               T2P = P*(T1+T2)
 2               T2M = P*(T1-T2)
 2               WA = AJ(MX,J1)*W(NJ)
 2               IF (RR(MX).EQ.0.0) GO TO 20
 2               BJ1 = RJ1(MX,J1)*P
 2               BJ0 = RJ0(MX,J1)*P
 2               RSZ = RSZ+WA*P*BJ0*(VL1*T1P-T2M)
 2               RTR = RTR+WA*P*BJ1*(VL*T1M+T2P)
 2               RSR=RSR+WA*(P*BJ0*((1.0+VL)*T1P+T2M)-BJ1*(T1P+T2M)/RR(MX))
 2               RST = RST+WA*(VL*BJ0*P*T1P+BJ1*
 2      *              (T1P+T2M)/RR(MX))
 2               GOTO 1002
```

```
D  1      7
2  C********** SPECIAL ROUTINE FOR RR(N)= ZERO
2  20             PP = P*P
2                 RSZ = RSZ+WA*PP*(VL1*T1P-T2M)
2                 RST = RST+WA*PP*((VL+0.5)*T1P+0.5*T2M)
2                 RSR = RST
2  1002       CONTINUE
1             SF = (AZ(MX,NK+4)-AZ(MX,NK+1))/1.7222726
1             CSZ = CSZ+RSZ*SF
1             CST = CST+RST*SF
1             CSR = CSR+RSR*SF
1             CTR = CTR + RTR*SF
1  1001   CONTINUE
          CSZ = CSZ*ARP(MI)
          CST = CST*ARP(MI)
          CTR = CTR*ARP(MI)
          CSR = CSR*ARP(MI)
   C***** TRANSFORM THE COMPONENT OF STRESS IN THE CYLINDRICAL COORDINATE
   C       SYSTEM TO THE CARTESIAN COORDINATE SYSTEM
          SSZ(MX)=CSZ
          IF(RR(N))70,70,80
   70     SSX(MX)=CSR
          SSY(MX)=CST
          SSXY(MX)=0.0
          SSXZ(MX)=0.0
          SSYZ(MX)=0.0
          GOTO 90
   80     SSX(MX)=0.5*(CSR+CST)+0.5*(CSR-CST)*((X(K)-X0(N))**2
         *          -(Y(K)-Y0(N))**2)/((X(K)-X0(N))**2+(Y(K)-Y0(N))**2)
                     SSY(MX) = 0.5*(CSR+CST)-0.5*(CSR-CST)*((X(K)-X0(N))**2
         *          -(Y(K)-Y0(N))**2)/((X(K)-X0(N))**2+(Y(K)-Y0(N))**2)
           SSXY(MX)  =  (CSR-CST)*(X(K)-X0(N))*(Y(K)-Y0(N))/((X(K)-
         *               X0(N))**2+(Y(K)-Y0(N))**2)
           SSXZ(MX)  =   CTR*(X(K)-X0(N))/SQRT((X(K)-X0(N))**2+(Y(K)-
         *               Y0(N))**2)
           SSYZ(MX)  =  CTR*(Y(K)-Y0(N))/SQRT((X(K)-X0(N))**2+(Y(K)-
         *               Y0(N))**2)
   90      CONTINUE
           RETURN
           END
```

```
D Line# 1     7
   $DEBUG
            SUBROUTINE EIGEN(H,PRE)
   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
   C        DETERMINE PRINCIPAL STRAIN
   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
            DIMENSION H(3,3),W(3),IQ(3)
            H(1,2) = 0.
            H(2,1) =0.
            H(2,3) =0.
            H(3,1) = H(1,3)
            H(3,2) = 0.
   60       CONTINUE
            DO 90 I =1,2
1           W(I) = 0.
1           II = I+1
1           DO 80 J =II,3
2              IF (W(I)-ABS(H(I,J))) 70,70,80
2  70          W(I) = ABS(H(I,J))
2              IQ(I) = J
2  80       CONTINUE
1           XMAX = 0.
1  90       CONTINUE
   100      DO 120 I = 1,2
1              IF (I.EQ.1) GO TO 110
1              IF(XMAX.GE.W(I)) GO TO 120
1  110         XMAX = W(I)
1              IP = I
1              JP = IQ(I)
1  120      CONTINUE
            IF (XMAX-1.E-12/3.)170,170,130
   130      Z = H(IP,IP)-H(JP,JP)
            Y = 2.*H(IP,JP)
            TA = Y/(ABS(Z)+SQRT(Z*Z+Y*Y))
            IF(Z.LT.0.) TA= -TA
            CO = 1./SQRT(1.+TA*TA)
            SI = TA*CO
            HII = H(IP,IP)
            HJJ = H(JP,JP)
            HIJ = H(IP,JP)
            DO 140 K =1,3
1           HTE = H(K,IP)
1           H(K,IP) = H(K,IP)*CO+H(K,JP)*SI
1           H(K,JP) = H(K,JP)*CO+HTE*SI
1           H(IP,K) = H(K,JP)
1  140      CONTINUE
            H(IP,JP) = 0.
            H(JP,IP) = 0.
            AA = HIJ*TA
            H(IP,IP) = HII+AA
            H(JP,JP) = HJJ-AA
            GOTO 60
   170      CONTINUE
            PRE = H(1,1)
            IF(PRE.GT.H(2,2)) GO TO 180
            PRE = H(2,2)
   180      IF (PRE.GT.H(3,3)) GO TO 185
```



```
D Line# 1     7
            PRE = H(3,3)
   185      RETURN
            END
```

```
D Line# 1     7                          Microsoft FORTRAN77 V3.30 March 1985
    $DEBUG
    CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
    C   SUBROUTINE LEASQ -DETERMINE RELATIONSHIP BETWEEN
    C                      1) A.C. SURFACE THICKNESS (hl) AND Na
    C                      2) A.C. SURFACE THICKNESS (hl) AND Np
    C                      BY LEAST SQUARE METHOD (POLYNOMIAL DEGREE=4)
    C                    - DETERMIN INTERSECTION BET. hl-Na AND hl-Np
    C                      BY  NEWTON-RAPHSON ITERATIVE METHOD
    CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
              SUBROUTINE LEASQ(HL,PTL,ATL,DHT,DNT)
              DIMENSION HL(5),ATL(5),PTL(5)
              DIMENSION A(5,6),X(6)
              DIMENSION SL(9),SS(5)
              REAL*4 L1,L2,L3,L4,N1,N
              REAL*4 LA(4),LB(4),LC(4),LD(4),LE(4)

                 DO 901 K=1,2
1                   SL(1) = 5.0
1                   DO 1000 KK = 2,9
2                      SL(KK)=0.0
2   1000            CONTINUE
1                   DO 1010 KK = 1,5
2                      SS(KK)=0.0
2   1010            CONTINUE
1                   L1 = 0.0
1                   L2 = 0.0
1                   L3 = 0.0
1                   L4 = 0.0
1                   L5 = 0.0
1                   LA(K) = 0.0
1                   LB(K) = 0.0
1                   LC(K) = 0.0
1                   LD(K) = 0.0
1                   LE(K) = 0.0
1                   IF (K.EQ.1) THEN
1                   DO 1020 KK = 2,9
2                      DO 1020  M = 1,5
3                         SL(KK)=SL(KK)+PTL(M)**(KK-1)
3   1020               CONTINUE
1                      DO 906  KK = 1,5
2                         DO 906 M = 1,5
3                            SS(KK)=SS(KK)+HL(M)*PTL(M)**(KK-1)
3   906                   CONTINUE
1                   ENDIF
1                   IF (K.EQ.2) THEN
1                   DO 1030 KK=2,9
2                   DO 1030 M = 1,5
3                      SL(KK)=SL(KK)+ATL(M)**(KK-1)
3   1030            CONTINUE
1                   DO 908  KK = 1,5
2                   DO 908 M = 1,5
3                      SS(KK)=SS(KK)+HL(M)*ATL(M)**(KK-1)
3   908             CONTINUE
1                   ENDIF
1                   DO 1060 KK = 1,5
2                   DO 1060 KM = 1,6
```

```
D Line# 1      7
3                         IF(KM.EQ.6) A(KK,KM)=SS(KK)
3                         IF(KM.LT.6)A(KK,KM)= SL(KK+KM-1)
3        1060       CONTINUE
1                     DO 960 NK=1,5
2                     DO 962 NP=NK,5
3                       D=A(NP,NK)
3                       DO 962 NJ=NK,6
4                         A(NP,NJ)=A(NP,NJ)/D
4        962            CONTINUE
2                       IF(NK.EQ.5)GOTO 960
2                       DO 966 NP=NK+1,5
3                         DO 966 NJ=NK,6
4                         A(NP,NJ)=A(NP,NJ)-A(NK,NJ)
4        966            CONTINUE
2        960          CONTINUE
1                     X(6)=0.0
1                     DO 970 NK=5,1,-1
2                       X(NK)=A(NK,6)
2                       DO 970 NW=NK,5
3                         X(NK)=X(NK)-A(NK,NW+1)*X(NW+1)
3        970            CONTINUE
1                       LA(K)=X(1)
1                       LB(K)=X(2)
1                       LC(K)=X(3)
1                       LD(K)=X(4)
1                       LE(K)=X(5)
1        901          CONTINUE
                      K =1
                      NN=0
                      N = (ATL(2)+PTL(2))/2.0
         920          CONTINUE
                      FN=(LA(K)-LA(K+1))+(LB(K)-LB(K+1))*N+(LC(K)-LC(K+1))*N**2
                *         +(LD(K)-LD(K+1))*N**3+(LE(K)-LE(K+1))*N**4
                      FDN=(LB(K)-LB(K+1))+2.0*(LC(K)-LC(K+1))*N+
                *         3.0*(LD(K)-LD(K+1))*N**2+4.0*(LE(K)-LE(K+1))*N**3
                      N1=N-FN/FDN
                      H1=LA(K)+LB(K)*N+LC(K)*N**2+LD(K)*N**3+LE(K)*N**4
                      H2=LA(K+1)+LB(K+1)*N+LC(K+1)*N**2+LD(K+1)*N**3+LE(K+1)*N**4
                      IF(ABS(H1-H2).GT.0.001.OR.NN.LT.100)GOTO 921
                      GO TO 923
         921          CONTINUE
                      NN=NN+1
                      N=N1
                      GOTO 920
         923          CONTINUE
                      DHT = H1
                      DNT = N

                    RETURN
                    END
```

3. โปรแกรมวิเคราะห์โครงสร้างชั้นทาง (D)
ประกอบด้วย

1) Main Program
2) Subroutine PART
3) Subroutine BESSEL
4) Subroutine COEF
5) Subroutine SSR (วิเคราะห์ความเค้น)
6) Subroutine EIGEN

```
$DEBUG
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C  PROGRAM D COMPUTE STRESS AND STRAIN OF AIRPORT 3-LAYER PAVEMENT
C               IN CARTESIAN COORDINATE SYSTEM
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      COMMON  ITN,BZ(71),H(2),RR(6),E(3),V(3),AZ(6,280),
     *          RJ0(6,280),RJ1(6,280),AJ(6,280),L,LI,K,
     *          X(40),Y(40),Z,R,X0(6),Y0(6),TP,
     *          SSZ(6),SSX(6),SSY(6),SSXY(6),SSXZ(6),SSYZ(6),
     *          NTEST(6),A(280,3),B(280,3),C(280,3),D(280,3)
      DIMENSION P1(6,280),TR(3,3),PTSN(40),VCSN(40)
      DIMENSION TSZ(40),TSX(40),TSY(40),TSXY(40),TSXZ(40),TSYZ(40)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     E(N)          = ELASTIC MODULUS OF N-TH LAYER
C     V(N)          = POISSON'S RATIO OF N-TH LAYER
C     X0(N),Y0(N)   = CENTER OF N-TH TIRE
C     X(K),Y(K)     = COORDINATE OF COMPUTATIONAL COMPUTATIONAL POINT
C     TP            = TIRE PRESSURE
C     WT            = WEIGHT PER TIRE
C     Z             = DEPTH WHERE STRESS AND STRAIN ARE CACULATED
C     L             = NO. OF LAYER WHERE STRESS &STRAIN ARE CALCULATED
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
 9000 FORMAT (A)
 9020 FORMAT (' AIRCRAFT MODEL     ',A15)
 9080 FORMAT (20X,'STRUCTURAL RESPONSE')
 9085 FORMAT ('+',19X,19('_'))
 9200 FORMAT ('     LAYER AND MATERIAL PROPERTIES')
 9210 FORMAT ('+',2X,31('_'))
 9220 FORMAT(5X,'LAYER   MATERIAL   ELAS.  MOD.  POISSON'S   THICKNESS')
 9230 FORMAT (5X,'NUMBER     TYPE       (kN/m^2)       RATIO         (m.)',/)
 9240 FORMAT (6X,' 1    ASPH. CONC. ',F9.1,7X,F4.2,7X,F5.2)
 9250 FORMAT (6X,' 2    GRANU. BASE ',F9.1,7X,F4.2,7X,F5.2)
 9255 FORMAT (6X,' 3    SUBGR. SOIL ',F9.1,7X,F4.2,/)
 9330 FORMAT (80(' '))
 9340 FORMAT (' COMPUTATIONAL ',25X,'STRESS(kN/m^2)')
 9350 FORMAT(' POINT (m)',3X,66('_'))
 9355 FORMAT ('     X        Y    ','     Z            Y          X        ',
     *'  XZ          XY          YZ')
 9361 FORMAT (2(1X,F5.2),6(E10.4,1X))
 9362 FORMAT(' PRINCIPAL TENSILE STRAIN(cm/cm)=',E13.7)
 9363 FORMAT (' VERTICAL COMPRESSIVE STRAIN (cm/cm) =',E13.7,/)
 9370 FORMAT ('  MAXIMUM PRINCIPAL TENSILE STRAIN        =   ',E13.7,/)
 9375 FORMAT ('  MAXIMUM VERTICAL COMPRESSIVE STRAIN    =    ',E13.7,/)
 9400 FORMAT (F9.1)
 9420 FORMAT (' WEIGHT PER TIRE (kN)          =',F9.1)
 9450 FORMAT (F8.1)
 9480 FORMAT (' TIRE PRESSURE (kN/m^2)      =',F8.1)
 9510 FORMAT (3(1X,F9.1),3(1X,F4.2))
 9515 FORMAT(2(1X,F4.1))
 9520 FORMAT(2(1X,I2))
 9521 FORMAT(' NO. OF TIRES =',I2,'  NO. OF COMPUTATIONAL POINTS =',I2)
 9525 FORMAT(2(1X,F4.2))
 9526 FORMAT( ' CENTER OF TIRE NO.',I2,' X0(',I2,') =',F4.2,' m. Y0(',I2
     *') =',F4.2,' m.')
 9535 FORMAT(F4.2,1X,I1)
 9536 FORMAT(' COMPUTE STRESS AND STRAIN AT DEPTH ',F4.2,' m.')
```

```
   C***  READ IN OUTPUT AND INPUT FILE
         CHARACTER*50 FILEIN,FILEOT,MODEL
         WRITE (*,'(A\)') '  NAME OF DATA FILE:  '
         READ (*,9000) FILEIN
         WRITE(*,'(A\)') '  NAME OF OUTPUT FILE:  '
         READ(*,9000) FILEOT
         OPEN(UNIT=4,FILE=FILEIN,STATUS='OLD')
         OPEN(UNIT=8,FILE=FILEOT,STATUS='NEW')
   C**** PRINT HEADING
         WRITE(8,*)'@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@'
         WRITE(8,*)'       DETERMINE  STRESS AND STRAIN'
         WRITE(8,*)'~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
   C**** READ IN INPUT DATA
   1111  FORMAT(I2)
         READ(4,1111)ITN
         READ(4,9000) MODEL
         WRITE (8,9020) MODEL
         READ(4,9400) WT
         WRITE (8,9420) WT
         READ(4,9450)     TP
         WRITE(8,9480) TP
            R=SQRT(WT/(TP*3.141592654))
         READ(4,9510)E(1),E(2),E(3),V(1),V(2),V(3)
         READ(4,9515) H1,H2
         READ(4,9520) NR,KK
         WRITE(8,9521)NR,KK
         DO 7000 N=1,NR
1           READ(4,9525)XO(N),YO(N)
1           WRITE(8,9526)N,N,XO(N),N,YO(N)
1  7000  CONTINUE
         DO 7010 K=1,KK
1           READ(4,9525)X(K),Y(K)
1  7010  CONTINUE
         READ(4,9535)Z,L
         WRITE(8,9536) Z
         IF(L.EQ.1)WRITE(8,*)'IN A.C. LAYER'
         IF(L.EQ.2)WRITE(8,*)'IN GARNULAR BASE'
         IF(L.EQ.3)WRITE(8,*)'IN SUBGRADE'
   C**** COMPUTE ZEROS OF J1(X) AND J0(X) SET UP GAUSS CONSTANT
         ITN4 = ITN*4
         K =ITN+1
         BZ(1) = 0.0
         BZ(2) = 1.0
         BZ(3) = 2.4048
         BZ(4) = 3.8317
         BZ(5) = 5.5201
         BZ(6) = 7.0156
         DO 700 I = 7,K,2
1           T = I/2
1           TD=4.0*T-1.
1           BZ(I)=3.1415927*(T-0.25+0.050661/TD-0.053041/TD**3+
1       *          0.262051/TD**5)
1  700   CONTINUE
         DO 710 I = 8,ITN,2
1           T = (I-2)/2
1           TD = 4.0*T+1.
```

```
D Line# 1      7
1             BZ(I) =3.1415927*(T+0.25-0.151982/TD+0.015399/TD**3
1         *            -0.245270/TD**5)
1     710     CONTINUE
      C****   PRINT OUT LAYER AND MATERIAL PROPERTIES
              WRITE(8,9200)
              WRITE(8,9210)
              WRITE(8,9220)
              WRITE(8,9230)
              WRITE(8,9240)E(1), V(1),H1
              WRITE(8,9250)E(2), V(2),H2
              WRITE(8,9255)E(3), V(3)
      C****   SET DEPTH OF EACH LAYER
              H(1) = H1
              H(2) = H(1)+H2
              XPTSN = 0.0
              XVCSN =0.0
      C*****START ON A NEW COMPUTATIONAL POINT
              DO 606 K = 1,KK
1             IF(K.EQ.1) THEN
1             WRITE(8,9080)
1             WRITE(8,9085)
1             ENDIF
1     C ***   COMPUTE DISTANCE BET. CENTER OF N-TH TIRE
1     C       AND K-TH COMPUTATIONAL POINT
1             DO 1234 N=1,NR
2                RR(N)=SQRT((X(K)-X0(N))**2+(Y(K)-Y0(N))**2)
2     1234    CONTINUE
1     C****   INITIALIZE THE COMPONENT OF STRESS
1             TSZ(K) = 0.0
1             TSX(K) = 0.0
1             TSY(K) = 0.0
1             TSXY(K) = 0.0
1             TSXZ(K) = 0.0
1             TSYZ(K) = 0.0
1     C****   START ON A NEW TIRE
1             DO 660 N = 1,NR
2     C****   CALCULATE THE PARTTIION
2                  CALL PART(N)
2                  DO 670 LI = 1,ITN4
3                       P = AZ(N,LI)
3     C******CALCULATE THE COEFFICIENTS AND BESSEL FUNCTION
3                       CALL COEF(N,P)
3                       IF(RR(N))115,115,110
3     110               PR = P*RR(N)
3                       CALL BESSEL(0,PR,Y1)
3                       RJ0(N,LI) = Y1
3                       CALL BESSEL(1,PR,Y1)
3                       RJ1(N,LI) = Y1
3     115               PA = P*R
3                       CALL BESSEL(1,PA,Y1)
3                       AJ(N,LI) = Y1
3     670          CONTINUE
2     C ****  CALCULATE STRESS OF N-TH TIRE
2                  CALL SSR(N)
2                  TSZ(K) = TSZ(K)+SSZ(N)
2                  TSX(K) = TSX(K)+SSX(N)
```

```
2                TSY(K) = TSY(K)+SSY(N)
2                TSXY(K) = TSXY(K)+SSXY(N)
2                TSXZ(K) = TSXZ(K)+SSXZ(N)
2                TSYZ(K) = TSYZ(K)+SSYZ(N)
2        660   CONTINUE
1        C**** CALCULATE STRAIN IN CARTESIAN COORDINATE SYSTEM
1              TR(1,1)  = (1/E(L))*(TSZ(K)-V(L)*(TSX(K)+TSY(K)))
1        C**** CALCULATE VERTICAL COMPRESSIVE STRAIN
1              VCSN(K)=TR(1,1)
1              TR(2,2)  = (1/E(L))*(TSX(K)-V(L)*(TSZ(K)+TSY(K)))
1              TR(3,3)  = (1/E(L))*(TSY(K)-V(L)*(TSZ(K)+TSX(K)))
1              TR(1,2) = ((1+V(L))/E(L))*TSXY(K)
1              TR(1,3) = ((1+V(L))/E(L))*TSXZ(K)
1              TR(2,3) = ((1+V(L))/E(L))*TSYZ(K)
1        C**** CALCULATE PRINCIPAL TENSILE STRAIN
1              CALL EIGEN(TR,PRE)
1              PTSN(K) = PRE
1              IF(XPTSN-PTSN(K) .LT. 0.0)  XPTSN=PTSN(K)
1              IF(XVCSN- VCSN(K).GT.0.0)   XVCSN=VCSN(K)
1              IF(K.EQ.1) THEN
1              WRITE(8,9330)
1              WRITE(8,9340)
1              WRITE(8,9350)
1              WRITE(8,9355)
1              WRITE(8,9330)
1              ENDIF
1              WRITE(8,9361)X(K),Y(K),TSZ(K),TSX(K),TSY(K),TSXZ(K),
1             *             TSXY(K),TSYZ(K)
1              WRITE(8,9362)PTSN(K)
1              WRITE(8,9363)VCSN(K)
1        606   CONTINUE
               WRITE(8,9330)
               WRITE(8,9370) XPTSN
               WRITE(8,9375)XVCSN
               STOP
               END
```

```
D Line# 1     7
        $DEBUG
              SUBROUTINE PART(N)
        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        C     COMPUTE POINT FOR LEGENDRE-GAUSS INTREGRATION
        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
              COMMON ITN,BZ(71),H(2),RR(6),E(3),V(3),AZ(6,280),
             *       RJ0(6,280),RJ1(6,280),AJ(6,280),L,LI,K,
             *       X(40),Y(40),Z,R,X0(6),Y0(6),TP,
             *       SSZ(6),SSX(6),SSY(6),SSXY(6),SSXZ(6),SSYZ(6),
             *       NTEST(6),A(280,3),B(280,3),C(280,3),D(280,3)
              DATA G1/8.6113631E-1/,G2/3.3998104E-1/
              MX = N
              DO 800 IT=1,5
1       800   CONTINUE
              ZF = R
              NTEST(MX) = 2
              IF (RR(MX))8,8,9
        9     CONTINUE
              NTEST (MX) =R/RR(MX)+0.0001
              IF (NTEST(MX))6,6,5
        6     CONTINUE
              NTEST(MX) = RR(MX)/R+0.0001
              ZF  = RR(MX)
        5     CONTINUE
              NTEST(MX) = NTEST(MX)+1
              IF (NTEST(MX)-10)8,8,7
        7     CONTINUE
              NTEST(MX) =10
        8     CONTINUE
              MK =1
              ZF = 2*ZF
              SZ2=0.0
              DO 28 N1 = 1,ITN
1                SZ1 =SZ2
1                SZ2 = BZ(N1+1)/ZF
1                SF = SZ2-SZ1
1                PP = SZ2+SZ1
1                SG1 = SF*G1
1                SG2 = SF*G2
1                AZ(MX,MK)=PP-SG1
1                AZ(MX,MK+1) = PP-SG2
1                AZ(MX,MK+2)=PP+SG2
1                AZ(MX,MK+3) = PP+SG1
1                MK =MK+4
1       28    CONTINUE
              RETURN
              END
```

```
D Line# 1      7
   $DEBUG
              SUBROUTINE BESSEL(NI,XI,Y)
   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
   C          CALCULATE BESSEL FUNCTION ORDER 0 AND 1
   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
              DIMENSION D(20),PZ(6),QZ(6),P1(6),Q1(6)
              DOUBLE PRECISION T4,X,PI
              DATA PZ/1.0E0,-1.125E-4,2.8710938E-7,-2.3449658E-9,
         *         3.9806841E-11,-1.1536133E-12/
              DATA QZ/-5.0E-3,4.6875E-6,-2.3255859E-8,2.8307087E-10,
         *         -6.3912096E-12,2.3124704E-12/
              DATA P1/1.0E0,1.875E-4,-3.6914063E-7,2.7713232E-9,
         *         -4.5114421E-11,1.2750463E-12/
              DATA Q1/1.5E-2,-6.5625E-6,2.8423828E-8,
         *         -3.2662024E-10,7.1431166E-12,-2.5327056E-13/
              PI=3.1415927
   9          N =NI
              X = XI
              IF (X-7.0) 10,10,160
   10         X2=X/2.0
              FAC = -X2*X2
              IF(N) 11,11,14
   11         C=1.0
              Y = C
              DO 13 I=1,34
1             T = I
1             C = FAC*C/(T*T)
1             TEST = ABS (C) - 10.0**(-8)
1             IF (TEST) 17,17,12
1  12         Y = Y+C
1  13         CONTINUE
   14         C = X2
              Y=C
              DO 16 I = 1,34
1             T = I
1             C = FAC*C/(T*(T+1.0))
1             TEST = ABS(C)-10.0**(-8)
1             IF(TEST)17,17,15
1  15         Y = Y+C
1  16         CONTINUE
   17         RETURN
   160        IF(N)161,161,164
   161        DO 162 I = 1,6
1             D(I) = PZ(I)
1             D(I+10) = QZ(I)
1  162        CONTINUE
              GOTO 163
   164        DO 165 I =1,6
1             D(I) = P1(I)
1             D(I+10) = Q1(I)
1  165        CONTINUE
   163        CONTINUE
              T1 = 25.0/X
              T2 = T1*T1
              P = D(6)*T2+D(5)
              DO 170 I =1,4
```

```
D Line# 1       7
1                 J = 5-I
1                 P = P*T2+D(J)
1          170    CONTINUE
                  Q = D(16)*T2+D(15)
                  DO 171 I =1,4
1                 J = 5-I
1                 Q = Q*T2+D(J+10)
1          171    CONTINUE
                  Q = Q*T1
           C
                  T4 = DSQRT(X*PI)
                  T6 = SIN(X)
                  T7 = COS(X)
           C
                  IF(N) 180,180,185
           180    T5 = ((P-Q)*T6+(P+Q)*T7)/T4
                  GOTO 99
           185    T5 = ((P+Q)*T6-(P-Q)*T7)/T4
           99     Y = T5
                  RETURN
                  END
```

```
D Line# 1     7
 $DEBUG
       SUBROUTINE  COEF(N,P)
 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
 C     USE FOR ALL PROBLEM , UP TO MAX. DIMENSION OF 3 LAYER
 C     CALCULATE COEF. A,B,C AND D OF 3-LAYER ELASTIC SYSTEM
 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
       COMMON ITN,BZ(71),H(2),RR(6),E(3),V(3),AZ(6,280),
      *        RJ0(6,280),RJ1(6,280),AJ(6,280),L,LI,K,
      *        X(40),Y(40),Z,R,X0(6),Y0(6),TP,
      *        SSZ(6),SSX(6),SSY(6),SSXY(6),SSXZ(6),SSYZ(6),
      *        NTEST(6),A(280,3),B(280,3),C(280,3),D(280,3)
       DIMENSION PM(4,4,4),XP(5,4,4),FM(4),SC(4)
       REAL *4 Q(2,2)
       LC = LI
 C*****SET UP MATRIX X=DI*MI*KI*K*M*D
 C     COMPUTE THE MATRICES X(K)
       DO 10 N1 = 1,2
1         T1 = E(N1)*(1.+V(N1+1))/(E(N1+1)*(1.+V(N1)))
1         T1M = T1-1.
1         PH = P*H(N1)
1         PH2 = PH*2.
1         VK2 = 2.*V(N1)
1         VKP2 = 2.*V(N1+1)
1         VK4=2.0*VK2
1         VKP4=2.0*VKP2
1         VKK8 = 8.*V(N1)*V(N1+1)
1 C
1         XP(N1,1,1) = VK4-3.-T1
1         XP(N1,2,1) = 0.
1         XP(N1,3,1) = T1M*(PH2-VK4+1.)
1         XP(N1,4,1) = -2.*T1M*P
1 C
1         T3 = PH2*(VK2-1.)
1         T4 = VKK8+1.-3.*VKP2
1         T5 = PH2*(VKP2-1.)
1         T6 = VKK8+1.-3.0*VK2
1 C
1         XP(N1,1,2) = (T3+T4-T1*(T5+T6))/P
1         XP(N1,2,2) = T1*(VKP4-3.)-1.
1         XP(N1,4,2) = T1M*(1.-PH2-VKP4)
1 C
1         XP(N1,3,4) = (T3-T4-T1*(T5-T6))/P
1 C
1         T3 = PH2*PH-VKK8+1.
1         T4 = PH2*(VK2-VKP2)
1 C
1         XP(N1,1,4) = (T3+T4+VKP2-T1*(T3+T4+VK2))/P
1         XP(N1,3,2)=(-T3+T4-VKP2+T1*(T3-T4+VK2))/P
1 C
1         XP(N1,1,3) = T1M*(1.-PH2-VK4)
1         XP(N1,2,3) = 2.*T1M*P
1         XP(N1,3,3) = VK4-3.-T1
1         XP(N1,4,3) = 0.
1 C
1         XP(N1,2,4) = T1M*(PH2-VKP4+1.)
1         XP(N1,4,4) = T1*(VKP4-3.)-1.
```

```
D        1     7
1       10     CONTINUE
           C*****COMPUTE THE PRODUCT MATRIX PM
               SC(2) = 4.*(V(2)-1.)
               SC(1) = SC(2)*4.*(V(1)-1.)
               Q(1,1) = 1.
               Q(2,2) = 1.
               Q(1,2) = 0.
               QQ = P*2.*H(2)
               IF(QQ-80.) 15,15,16
        15     CONTINUE
               Q(1,2) = EXP(-QQ)
           C**** Q(2,1) IS NOT NEEDED FOR INITIALIZING THE PM MATRIX
        16     CONTINUE
           C***** 20 LOOP INITIALIZES PM(N,,)
               DO 20 MX = 1,4
1                 LL = (MX+1)/2
1                 PM(2,MX,4) = XP(2,MX,4)*Q(LL,2)
1                 PM(2,MX,3) = XP(2,MX,3)*Q(LL,2)
1       20     CONTINUE
               QQ = P*2.*H(1)
               IF (QQ-80.)22,22,23
        22     CONTINUE
               Q(2,1) = EXP(QQ)
               Q(1,2) = 1./Q(2,1)
               GOTO 24
        23     CONTINUE
               Q(1,2) = 0.
               Q(2,1) = 1.E20
        24     CONTINUE
               DO 25 MX = 1,4
1              LL = (MX+1)/2
1              DO 25 MI = 3,4
2                 PM(1,MX,MI) = (XP(1,MX,1)*PM(2,1,MI)+XP(1,MX,2)*PM(2,2,MI))*
2             *    Q(LL,1)+(XP(1,MX,3)*PM(2,3,MI)+XP(1,MX,4)*PM(2,4,MI))*Q(LL,2)
2       25     CONTINUE
           C**** SOLVE FOR C(3) AND D(3)
               T3 = 2.*V(1)
               T4 = T3-1.0
               FM(1) = P*(PM(1,1,3)+PM(1,3,3))+T3*(PM(1,2,3)-PM(1,4,3))
               FM(2) = P*(PM(1,1,3)-PM(1,3,3))+T4*(PM(1,2,3)+PM(1,4,3))
               FM(3) = P*(PM(1,1,4)+PM(1,3,4)) + T3*(PM(1,2,4)-PM(1,4,4))
               FM(4) = P*(PM(1,1,4)-PM(1,3,4))+T4*(PM(1,2,4)+PM(1,4,4))

               DFAC = SC(1)/((FM(1)*FM(4)-FM(3)*FM(2))*P*P)
               A(LC,3) = 0.
               B(LC,3) = 0.
               C(LC,3) = -FM(3)*DFAC
               D(LC,3) = FM(1)*DFAC
           C**** BACKSOLVE FOR THE OTHER A,B,C,D
               DO 100 N1 = 1,2
1                 A(LC,N1)=(PM(N1,1,3)*C(LC,3)+PM(N1,1,4)*D(LC,3))/SC(N1)
1                 B(LC,N1)=(PM(N1,2,3)*C(LC,3)+PM(N1,2,4)*D(LC,3))/SC(N1)
1                 C(LC,N1)=(PM(N1,3,3)*C(LC,3)+PM(N1,3,4)*D(LC,3))/SC(N1)
1                 D(LC,N1)=(PM(N1,4,3)*C(LC,3)+PM(N1,4,4)*D(LC,3))/SC(N1)
1      100     CONTINUE
               RETURN
```

```
D Line# 1     7                          Microsoft FORTRAN77 V3.30 March 1985
      $DEBUG
            SUBROUTINE SSR(N)
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      C   THIS SUBROUTINE COMPUTE THE COMPONENT OF STRESS IN THE CYLINDRICAL
      C   COORDINATE SYSTEM AND TRANSFORM TO THE CARTESIAN COORDINATE
      C   SYSTEM BY USING TRANSFORMATION FORMULAS
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
            COMMON ITN,BZ(71),H(2),RR(6),E(3),V(3),AZ(6,280),
           *        RJ0(6,280),RJ1(6,280),AJ(6,280),L,LI,K,
           *        X(40),Y(40),Z,R,X0(6),Y0(6),TP,
           *        SSZ(6),SSX(6),SSY(6),SSXY(6),SSXZ(6),SSYZ(6),
           *        NTEST(6),A(280,3),B(280,3),C(280,3),D(280,3)
            DIMENSION W(4)
            DATA  W/0.34785485,0.65214515,0.65214515,0.34785485/
            MX=N
            VL=2.0*V(L)
            EL=(1.0+V(L))/E(L)
            VL1 = 1.0-VL
      C* COMPUTE THE COMPONENT OF STRESS IN THE CYLINDRICAL COORDINATE SYSTEM
            CSZ = 0.0
            CST = 0.0
            CSR = 0.0
            CTR = 0.0
            ARP = R*TP
            DO 1001 N1 = 1,ITN
1     C**** INITIALIZE THE SUB-INTEGRALS
1               RSZ = 0.0
1               RST = 0.0
1               RSR = 0.0
1               RTR = 0.0
1               NK = 4*(N1-1)
1     C***** COMPUTE THE SUB-INTEGRALS
1               DO 1002 NJ = 1,4
2                  J1 = NK + NJ
2
2                  P = AZ(MX,J1)
2                  IF(P*Z.LE.80.0)  EP = EXP(P*Z)
2                  IF(P*Z.GT.80.0)  EP = 1.E20
2                  T1 = B(J1,L)*EP
2                  T2 = D(J1,L)/EP
2                  T1P = T1+T2
2                  T1M = T1-T2
2                  T1 = (A(J1,L)+B(J1,L)*Z)*EP
2                  T2 = (C(J1,L)+D(J1,L)*Z)/EP
2                  T2P = P*(T1+T2)
2                  T2M = P*(T1-T2)
2                  WA = AJ(MX,J1)*W(NJ)
2                  IF (RR(MX).EQ.0.0) GO TO 20
2                  BJ1 = RJ1(MX,J1)*P
2                  BJ0 = RJ0(MX,J1)*P
2                  RSZ = RSZ+WA*P*BJ0*(VL1*T1P-T2M)
2                  RTR = RTR+WA*P*BJ1*(VL*T1M+T2P)
2                  RSR=RSR+WA*(P*BJ0*((1.0+VL)*T1P+T2M)-BJ1*(T1P+T2M)/RR(MX))
2                  RST = RST+WA*(VL*BJ0*P*T1P+BJ1*
2          *            (T1P+T2M)/RR(MX))
2                  GOTO 1002
```

```
D  1     7
2  C********     SPECIAL ROUTINE FOR RR(N)=ZERO
2  20            PP = P*P
2                RSZ = RSZ+WA*PP*(VL1*T1P-T2M)
2                RST = RST+WA*PP*((VL+0.5)*T1P+0.5*T2M)
2                RSR = RST
2  1002       CONTINUE
1             SF = (AZ(MX,NK+4)-AZ(MX,NK+1))/1.7222726
1             CSZ = CSZ+RSZ*SF
1             CST = CST+RST*SF
1             CSR = CSR+RSR*SF
1             CTR = CTR + RTR*SF
1  CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
1  C          TRANSFORM THE COMPONENT OF STRESS IN THE CYLINDRICAL COORDINATE
1  C          SYSTEM TO THE CARTESIAN COORDINATE SYSTEM
1  CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
1  1001          CONTINUE
                 CSZ = CSZ*ARP
                 CST = CST*ARP
                 CTR = CTR*ARP
                 CSR = CSR*ARP
                 SSZ(MX)=CSZ
                 IF(RR(N))70,70,80
   70            SSX(MX)=CSR
                 SSY(MX)=CST
                 GOTO 90
   80       SSX(MX)=0.5*(CSR+CST)+0.5*(CSR-CST)*((X(K)-X0(N))**2
          *        -(Y(K)-Y0(N))**2)/((X(K)-X0(N))**2+(Y(K)-Y0(N))**2)
                   SSY(MX) = 0.5*(CSR+CST)-0.5*(CSR-CST)*((X(K)-X0(N))**2
          *        -(Y(K)-Y0(N))**2)/((X(K)-X0(N))**2+(Y(K)-Y0(N))**2)
            SSXY(MX)  =  (CSR-CST)*(X(K)-X0(N))*(Y(K)-Y0(N))/((X(K)-
          *              X0(N))**2+(Y(K)-Y0(N))**2)
            SSXZ(MX)  =   CTR*(X(K)-X0(N))/SQRT((X(K)-X0(N))**2+(Y(K)-
          *              Y0(N))**2)
            SSYZ(MX)  =  CTR*(Y(K)-Y0(N))/SQRT((X(K)-X0(N))**2+(Y(K)-
          *              Y0(N))**2)
   90       CONTINUE
            RETURN
            END
```

```
$DEBUG
      SUBROUTINE EIGEN(H,PRE)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     DETERMINE PRINCIPAL STRAIN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DIMENSION H(3,3),W(3),IQ(3)
      H(1,2) = 0.
      H(2,1) =0.
      H(2,3) =0.
      H(3,1) = H(1,3)
      H(3,2) = 0.
60    CONTINUE
      DO 90 I =1,2
         W(I) = 0.
         II = I+1
         DO 80 J =II,3
            IF (W(I)-ABS(H(I,J))) 70,70,80
70          W(I) = ABS(H(I,J))
            IQ(I) = J
80       CONTINUE
         XMAX = 0.
90    CONTINUE
100   DO 120 I = 1,2
      IF (I.EQ.1) GO TO 110
      IF(XMAX.GE.W(I)) GO TO 120
110   XMAX = W(I)
      IP = I
      JP = IQ(I)
120   CONTINUE
      IF (XMAX-1.E-12/3.)170,170,130
130   Z = H(IP,IP)-H(JP,JP)
      Y = 2.*H(IP,JP)
      TA = Y/(ABS(Z)+SQRT(Z*Z+Y*Y))
      IF(Z.LT.0.) TA= -TA
      CO = 1./SQRT(1.+TA*TA)
      SI = TA*CO
      HII = H(IP,IP)
      HJJ = H(JP,JP)
      HIJ = H(IP,JP)
      DO 140 K =1,3
      HTE = H(K,IP)
      H(K,IP) = H(K,IP)*CO+H(K,JP)*SI
      H(K,JP) = H(K,JP)*CO+HTE*SI
      H(IP,K) = H(K,JP)
140   CONTINUE
      H(IP,JP) = 0.
      H(JP,IP) = 0.
      AA = HIJ*TA
      H(IP,IP) = HII+AA
      H(JP,JP) = HJJ-AA
      GOTO 60
170   CONTINUE
      PRE = H(1,1)
      IF(PRE.GT.H(2,2)) GO TO 180
      PRE = H(2,2)
180   IF (PRE.GT.H(3,3)) GO TO 185

      PRE = H(3,3)
185   RETURN
      END
```

ภาคผนวก ข.

คุณลักษณะของเครื่องบินที่ใช้ในการออกแบบ

Lockheed (cont'd)																
37. L-649	(A)	94.0	(1)	15	4.7	2.4	75	(1)	28	44.7	22.4	80	14'0"	NA	39'3"	NA
38. L-49E	(A)	98.0	(1)	15	4.9	2.5	80	(1)	28	46.6	23.3	85	14'0"	NA	39'5"	NA
39. L-49D	(A)	96.0	(1)	15	4.8	2.4	80	(1)	28	45.6	22.8	85	14'0"	NA	39'5"	NA
40. L-49B/C	(A)	93.0	(1)	15	4.7	2.3	80	(1)	28	44.2	22.1	85	14'0"	NA	39'5"	NA
41. L-49A	(A)	90.0	(1)	15	4.5	2.3	80	(1)	28	42.6	21.3	85	14'0"	NA	39'5"	NA
42. L-49	(A)	86.3	(1)	15	4.3	2.2	80	(1)	28	41.0	20.5	85	14'0"	NA	39'5"	NA
IV. McDonnell-Douglas																
43. DC-10-30CF	(B)	533.0	(1)	25	42.8	21.4	180	(2)	54x64	201.2	50.3	185	17'6"	NA	72'5"	NA
				(This line entry refers to TM_3)[3]				(1)	38	87.8	43.9	170	0'0"	NA	74'11"	NA
44. DC-10-30	(B)	533.0	(1)	25	42.8	21.4	175	(2)	54x64	201.2	50.3	185	17'6"	NA	72'5"	NA
				(This line entry refers to TM_3)[3]				(1)	38	87.8	43.9	170	0'0"	NA	74'11"	NA
45. DC-10-20CF	(B)	533.0	(1)	25	42.8	21.4	180	(2)	54x64	201.2	50.3	185	17'6"	NA	72'5"	NA
				(This line entry refers to TM_3)[3]				(1)	38	87.8	43.9	170	0'0"	NA	74'11"	NA
46. DC-10-20	(B)	533.0	(1)	25	42.8	21.4	175	(2)	54x64	201.2	50.3	185	17'6"	NA	72'5"	NA
				(This line entry refers to TM_3)[3]				(1)	38	87.8	43.9	170	0'0"	NA	74'11"	NA
47. DC-10-10	(A)	413.0	(1)	24	24.8	12.4	155	(2)	54x64	194.1	48.5	175	17'6"	NA	72'5"	NA
48. DC-8-63F	(A)	358.0	(1)	18.5	13.8	6.9	143	(2)	32x55	172.1	43.0	196	10'5"	NA	77'6"	NA
49. DC-8-63	(A)	358.0	(1)	18.5	13.8	6.9	143	(2)	32x55	172.1	43.0	196	10'5"	NA	77'6"	NA
50. DC-8-62F	(A)	353.0	(1)	18.5	17.4	8.7	168	(2)	32x55	167.8	42.0	191	10'5"	NA	60'10"	NA
51. DC-8-62	(A)	353.0	(1)	18.5	23.4	11.7	168	(2)	32x55	164.8	41.2	187	10'5"	NA	60'10"	NA
52. DC-8-61F	(A)	331.0	(1)	18.5	13.2	6.6	118	(2)	30x55	158.9	39.7	190	10'5"	NA	77'6"	NA
53. DC-8-61	(A)	328.0	(1)	18.5	12.8	6.4	117	(2)	30x55	157.6	39.4	188	10'5"	NA	77'6"	NA
54. DC-8-55F	(A)	328.0	(1)	18.5	17.4	8.7	171	(2)	30x55	155.3	38.8	186	10'5"	NA	57'6"	NA
55. DC-8-55	(A)	328.0	(1)	18.5	17.8	8.9	171	(2)	30x55	155.1	38.8	186	10'5"	NA	57'6"	NA
56. DC-8-43	(A)	318.0	(1)	18.5	22.0	11.0	162	(2)	30x55	148.0	37.0	177	10'5"	NA	57'6"	NA
57. DC-7C	(A)	140.0	(0)	NA	14.2	14.2	70	(1)	29.8	67.0	33.5	126	17'4"	NA	48'1"	NA
58. DC-7B	(A)	126.0	(0)	NA	14.3	14.3	85	(1)	29.8	60.0	30.0	127	12'4"	NA	44'9"	NA
59. DC-7	(A)	122.2	(0)	NA	14.3	14.3	85	(1)	29.8	58.4	29.2	123	12'4"	NA	44'9"	NA
60. DC-9-41	(A)	115.0	(1)	14	7.4	3.7	130	(1)	26	53.8	26.9	163	8'2½"	NA	56'2"	NA
61. DC-9-32	(A)	109.0	(1)	14	8.2	4.1	130	(1)	25	50.4	25.2	152	8'2½"	NA	53'2"	NA
62. DC-9-21	(A)	101.0	(1)	14	5.6	2.8	122	(1)	25	47.7	23.8	144	8'2½"	NA	43'8"	NA
63. DC-9-15	(A)	91.5	(1)	14	6.7	3.4	105	(1)	24	42.4	21.2	127	8'2½"	NA	43'8"	NA
64. DC-6A/B	(A)	107.0	(0)	NA	12.8	12.8	70	(1)	30.8	51.4	25.7	106	12'4"	NA	36'2"	NA
65. DC-6	(A)	97.2	(0)	NA	12.1	12.1	70	(1)	30.8	45.8	22.9	95	12'4"	NA	36'2"	NA
66. DC-4	(A)	73.0	(0)	NA	10.4	10.4	60	(1)	30	35.4	17.7	75	12'4"	NA	36'0"	NA
V. SAF-BAC																
67. Concorde	(A)	388.0	(1)	21.1	19.4	9.7	174	(2)	26.4x66	184.3	46.1	184	12'8"	NA	59'8"	NA
VI. BAC																
68. BAC 1-11-500	(A)	100.0	(1)	13.7	5.0	2.5	115	(1)	21	47.5	23.8	174	7'1½"	NA	48'5"	NA
69. Viscount 810	(A)	72.5	(1)	13.7	3.6	1.8	95	(1)	19	34.4	17.2	138	11'11"	NA	39'6"	NA
70. Viscount 745D	(A)	64.5	(1)	13.7	3.2	1.6	92	(1)	19	30.6	15.3	122	11'11"	NA	35'8"	NA
VII. SUD																
71. Caravelle SE-210-6R[6]	(A)	110.2	(1)	16	7.4	3.7	90	(2)	16.8x48.7	51.4	12.9	155	8'6½"	NA	55'6"	NA
72. Caravelle SE-210-1	(A)	95.9	(1)	16	6.4	3.2	90	(2)	16.8x48.7	44.7	11.2	155	8'6½"	NA	55'6"	NA

NOTES: (1) See Figure IX-1 for aircraft assembly type legend.

(2) See Figure IX-2 for gear assembly type legend.

(3) See Figure IX-3 for explanation of X,Y values of aircraft gear layout.

(4) Dimensions shown for tire spacings in Table IX-1 are center-to-center of the tires or axles, depending on direction. Dimensions are listed in the alphabetical order noted in Figure IX-2 for the respective assembly type.

(5) Nose and Main gear weights are based on *static* condition. Maximum nose gear weights during braking conditions generally are in excess of those noted in Table IX-1.

(6) Nominal tire spacing values used for Caravelle are due to different fore and aft lateral dimensions in dual tandem main gear. Maximum tire pressure is used for all tires of main gear.

Metric unit conversion factors applicable to this table are:

Pound × 0.45359 = Kilogram (kg)

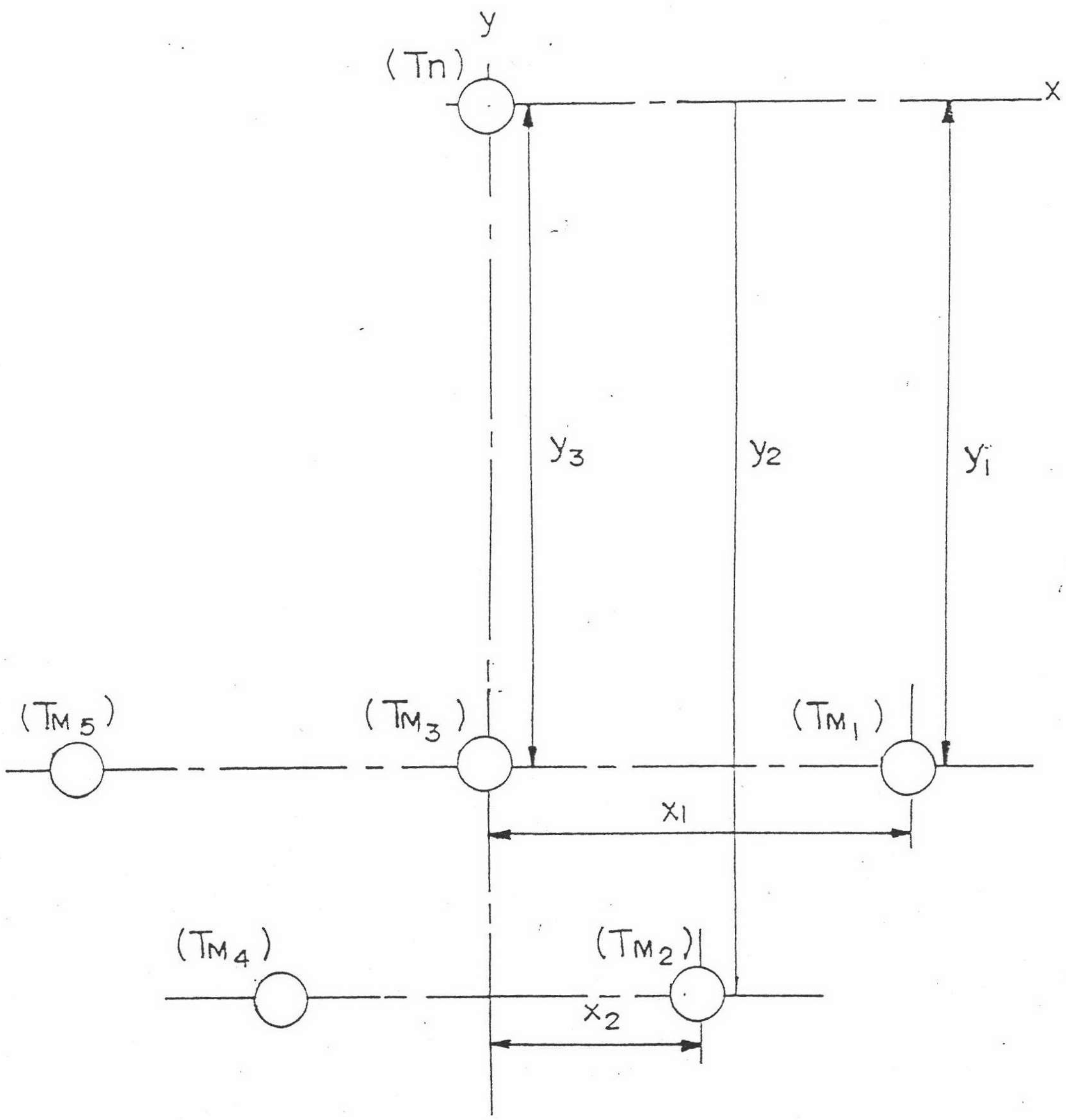
Inch × 0.0254 = Meter (m)

Foot × 0.3048 = Meter (m)

psi × 6.894757 = Kilonewton per square meter (kN/m^2)

ตารางที่ ข-1 คุณลักษณะของเครื่องบินที่ใช้ออกแบบ

Col. (a)	Col. (b)	Col. (c)	Col. (d)	Col. (e)	Col. (f)	Col. (g)	Col. (h)	Col. (i)	Col. (j)	Col. (k)	Col. (l)	Col. (m)	Col. (n)	Col. (o)	Col. (p)	Col. (q)
Aircraft Identification			Nose Gear Assembly					Main Truck Assembly								
Aircraft Company/Model	Landing Gear Assembly Type[1]	Maximum Gross Wt. lb. x 10^3	Assembly Type[2]	Tire Spacing Inch[4]	Maximum Gear Wt.[5] lb. x 10^3	Max. Wt. Per Tire lb. x 10^3	Tire Pressure psi	Assembly Type[2]	Tire Spacing Inch[4]	Maximum Truck Wt.[5] lb. x 10^3	Max. Wt. Per Tire lb. x 10^3	Tire Press. psi	X_1[3]	X_2	Y_1	Y_2
I. Boeing																
1. B-747-F	(C)	778.0	(1)	36	50.8	25.4	180	(2)	44x58	181.8	45.5	185	18'½"	6'3"	78'11½"	89'½"
2. B-747-C	(C)	778.0	(1)	36	50.8	25.4	180	(2)	44x58	181.8	45.5	185	18'½"	6'3"	78'11½"	89'½"
3. B-747-B	(C)	778.0	(1)	36	50.8	25.4	180	(2)	44x58	181.8	45.5	185	18'½"	6'3"	78'11½"	89'½"
4. B-747	(C)	713.0	(1)	36	47.0	23.5	165	(2)	44x58	166.5	41.6	204	18'½"	6'3"	78'11½"	89'½"
5. B-707-320C	(A)	336.0	(1)	22	22.0	11.0	115	(2)	34.5x56	157.0	39.3	180	11'½"	NA	59'0"	NA
6. B-707-320B	(A)	328.0	(1)	22	26.0	13.0	115	(2)	34.5x56	151.0	37.8	180	11'½"	NA	59'0"	NA
7. B-707-320/420	(A)	316.0	(1)	22	25.0	12.5	115	(2)	34.5x56	145.5	36.4	180	11'½"	NA	59'0"	NA
8. B-707-120B	(A)	258.0	(1)	22	16.8	8.4	90	(2)	34x56	120.6	30.2	170	11'½"	NA	52'4"	NA
9. B-720-B	(A)	230.0	(1)	22	12.0	6.0	100	(2)	32x49	109.0	27.3	145	10'11½"	NA	50'8"	NA
10. B-720	(A)	230.0	(1)	22	12.0	6.0	100	(2)	32x49	109.0	27.3	145	10'11½"	NA	50'8"	NA
11. B-727-200	(A)	173.0	(1)	24	13.2	6.6	100	(1)	34	79.9	39.9	168	9'4½"	NA	63'3"	NA
12. B-727-100C	(A)	170.0	(1)	24	17.2	8.6	100	(1)	34	76.4	38.2	166	9'4½"	NA	53'3"	NA
13. B-727-100	(A)	170.0	(1)	24	16.2	8.1	100	(1)	34	76.9	38.5	166	9'4½"	NA	53'3"	NA
14. B-737-200C	(A)	111.0	(1)	15	8.0	4.0	135	(1)	30.5	51.5	25.8	148	8'7"	NA	37'4"	NA
15. B-737-200	(A)	111.0	(1)	15	8.0	4.0	135	(1)	30.5	51.5	25.8	148	8'7"	NA	37'4"	NA
16. B-737-100	(A)	111.0	(1)	15	8.0	4.0	135	(1)	30.5	51.5	25.8	148	8'7"	NA	37'4"	NA
II. Convair																
17. Cv-990	(A)	255.0	(1)	17	13.6	6.8	200	(2)	24x46.5	120.7	30.2	186	9'11½"	NA	57'3"	NA
18. Cv-880M	(A)	193.5	(1)	17	8.7	4.4	140	(2)	21.5x45	92.4	23.1	155	9'5½"	NA	53'1"	NA
19. Cv-880	(A)	185.0	(1)	17	11.8	5.9	136	(2)	21.5x45	87.1	21.8	150	9'5½"	NA	53'1"	NA
III. Lockheed																
20. L-500	(C)	861.5	(4)	30.8 x 30.8 x 30.8	53.5	13.4	130	(2)	53x72	202.0	50.5	185	12'9¼"	12'9¼"	63'9¼"	82'1"
21. L-1011-8	(A)	598.0	(1)	25	30.0	15.0	182	(3)	42x56x56	284.0	47.3	196	16'9"	NA	71'½"	NA
22. L-1011-1	(A)	411.0	(1)	24	21.0	10.5	179	(2)	52x70	195.0	48.8	175	18'0"	NA	70'0"	NA
23. L-1649A	(A)	160.0	(1)	15	8.0	4.0	75	(1)	30	76.0	38.0	140	19'2½"	NA	54'4"	NA
24. L-100-30	(A)	155.0	(1)	22	9.8	4.9	60	(7)	60.5	75.3	37.7	110	7'1½"	NA	40'4½"	NA
25. L-100-20	(A)	155.0	(1)	22	9.5	4.8	60	(7)	60.5	75.1	37.5	110	7'1½"	NA	37'¾"	NA
26. L-100-10	(A)	155.0	(1)	22	11.0	5.5	60	(7)	60.5	74.3	37.1	110	7'1½"	NA	32'¾"	NA
27. L-1049H	(A)	142.0	(1)	15	7.1	3.6	70	(1)	28	67.5	33.7	132	14'0"	NA	49'11"	NA
28. L-1049G	(A)	137.5	(1)	15	6.9	3.4	70	(1)	28	65.3	32.7	128	14'0"	NA	49'11"	NA
29. L-1049C/D	(A)	133.0	(1)	15	6.7	3.3	70	(1)	28	63.2	31.6	123	14'0"	NA	49'11"	NA
30. L-1049	(A)	120.0	(1)	15	6.0	3.0	70	(1)	28	57.0	28.5	110	14'0"	NA	49'11"	NA
31. L-188C	(A)	116.0	(1)	16	5.8	2.9	90	(1)	26	55.1	27.6	145	15'7"	NA	48'3"	NA
32. L-188A	(A)	113.0	(1)	16	5.7	2.8	90	(1)	26	53.7	26.8	140	15'7"	NA	48'3"	NA
33. L-749A	(A)	107.0	(1)	15	5.4	2.7	80	(1)	28	50.8	25.4	85	14'0"	NA	39'3"	NA
34. L-749	(A)	102.0	(1)	15	5.1	2.6	80	(1)	28	48.5	24.2	85	14'0"	NA	39'3"	NA
35. L-149	(A)	100.0	(1)	15	5.0	2.5	80	(1)	28	47.5	23.8	85	14'0"	NA	39'5"	NA
36. L-649A	(A)	98.0	(1)	15	4.9	2.5	80	(1)	28	46.6	23.3	85	14'0"	NA	39'3"	NA



T_N = NOSE GEAR ASSEMBLY

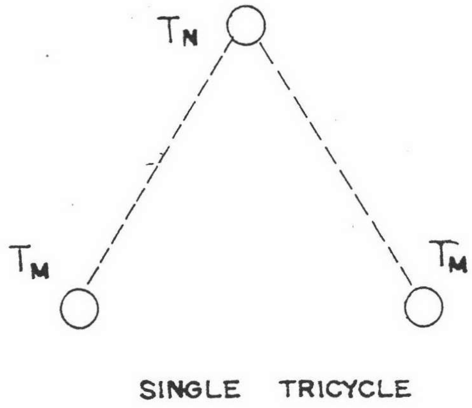
T_M = MAIN TRUCK (GEAR) ASSEMBLY

NOTE All x,y distances taken from nose gear centerline to the geometric centerline of the main truck gear

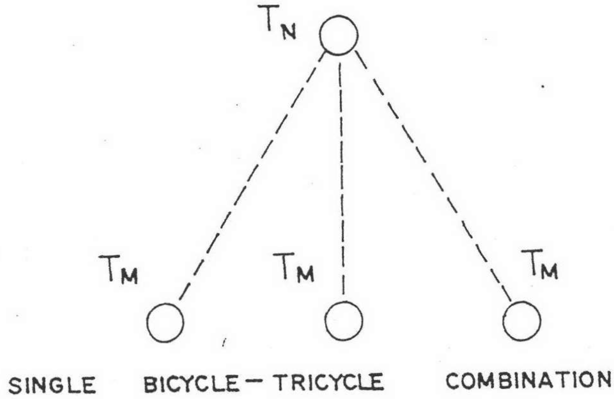
รูปที่ ข-1 การระบุตำแหน่งกลุ่มล้อของเครื่องบิน

สัญญลักษณ์ **ชนิด**

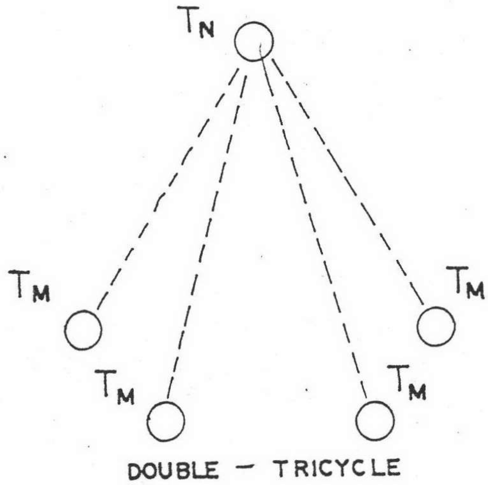
(A)



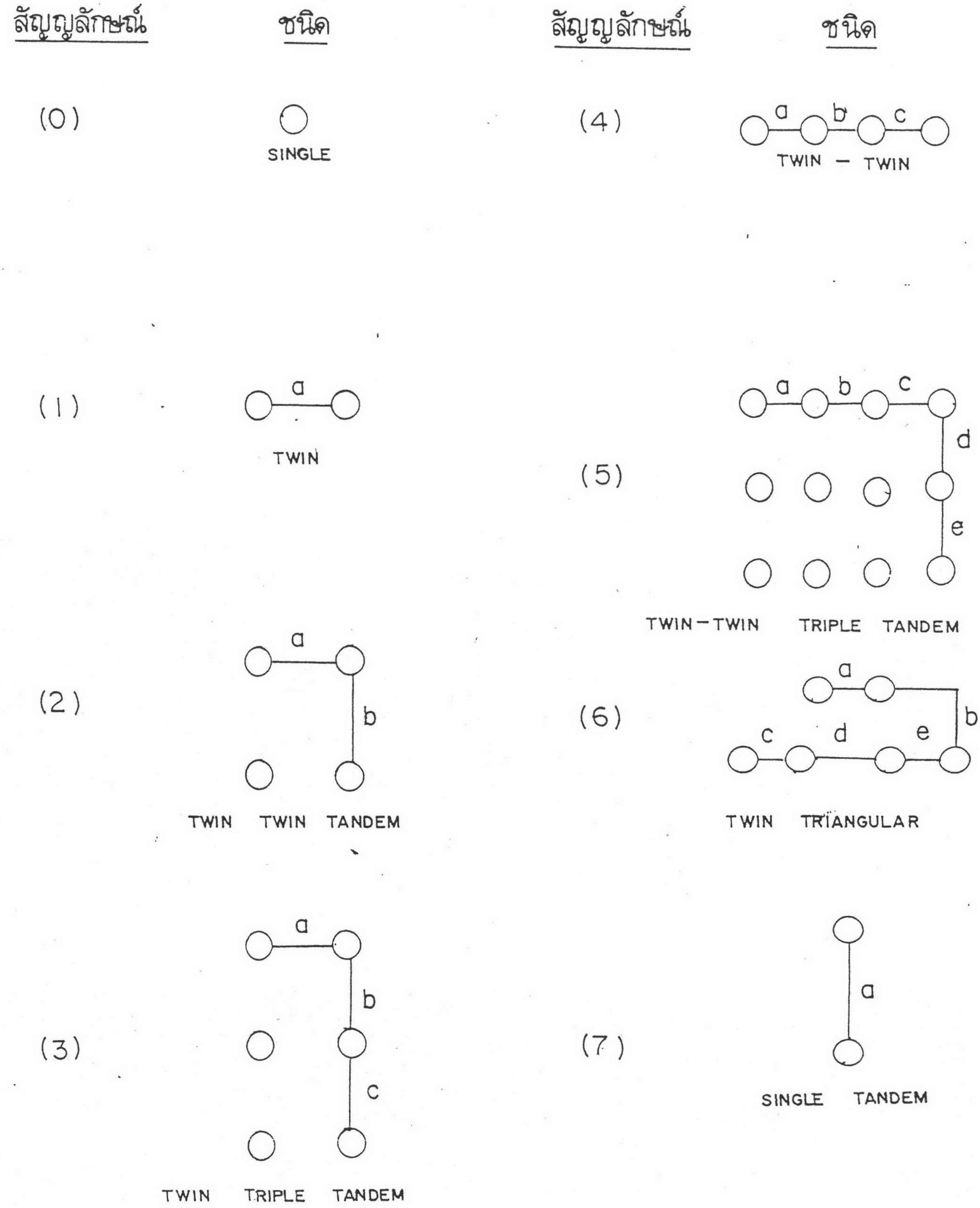
(B)



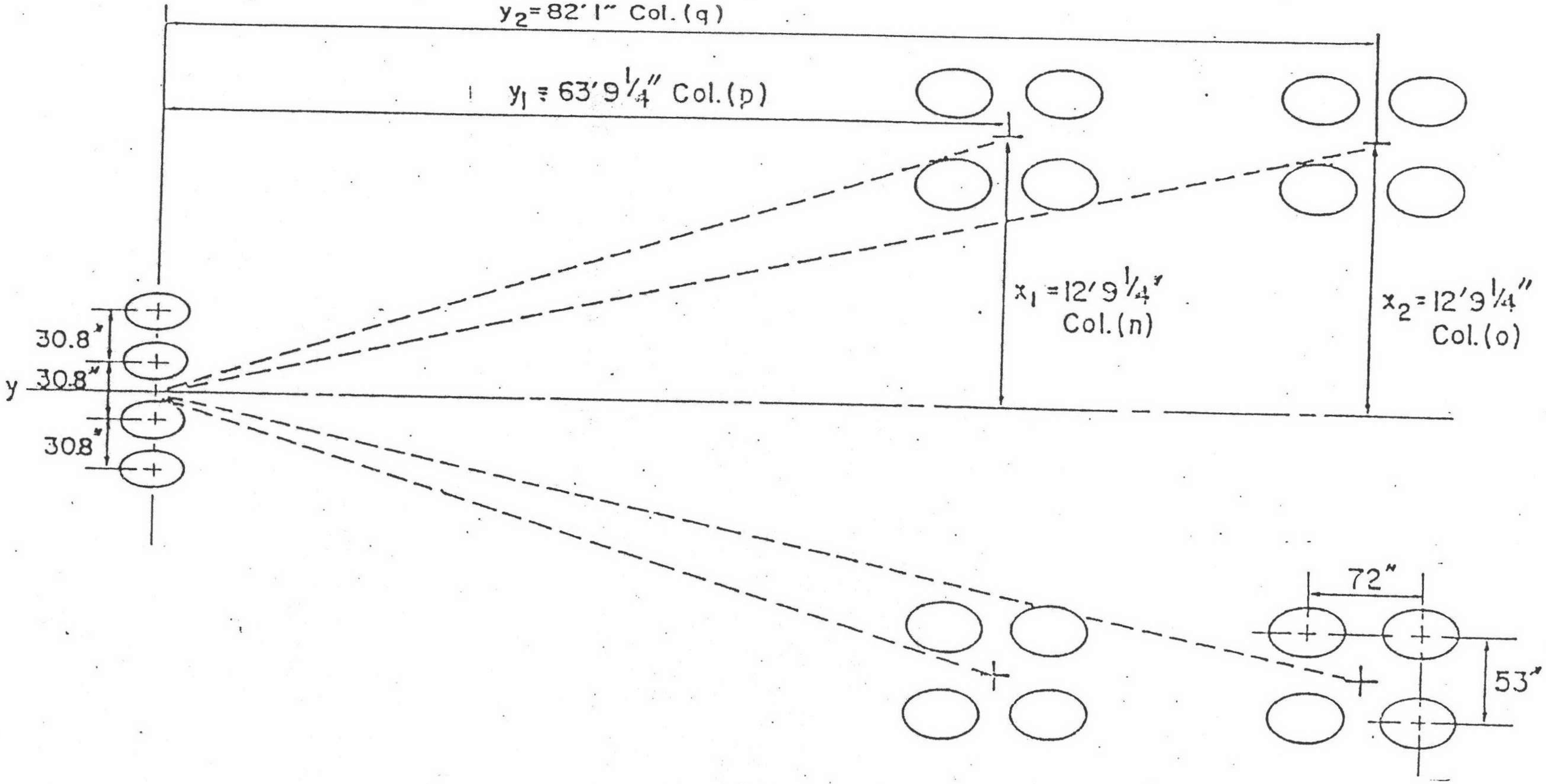
(C)



รูปที่ ข–2 การประกอบกลุ่มล้อชนิดต่าง ๆ



รูปที่ ข-3 การประกอบชนิดต่าง ๆ ของล้อ



AIRCRAFT CHARACTERISTICS

Type: L-500 Col. (a)
Assembly Type: (C) Col. (b)
Max. Gr. Wt.: 861,500 lb. Col. (c)

NOTE: See Table IX-1 for Column references.
See "Notes", Table IX-1 for metric unit conversion functions.

	NOSE GEAR ASSEMBLY	MAIN TRUCK ASSEMBLY
Assembly Type:	(4) Col. (d)	(2) Col. (i)
Tire Spacing:	30.8" x 30.8" x 30.8" Col. (e)	53" x 72" Col. (j)
Max. Gear Wt.:	53,500 lb. Col. (f)	202,000 lb. Col. (k)
Max. Wt. per Tire:	13,400 lb. Col. (g)	50,500 lb. Col. (l)
Tire Pressure:	130 psi Col. (h)	185 psi Col. (m)

รูปที่ ข-4 สรุปคุณลักษณะของเครื่องบิน

ภาคผนวก ค.
วิธีการใช้โปรแกรม

โปรแกรมออกแบบและวิเคราะห์ที่เขียนขึ้นใช้ภาษาฟอร์แทรน 77 Source Program ในภาคผนวก ก. สามารถเขียนโดยใช้ Line Editor (Edlin) ซึ่งมีอยู่ในโปรแกรม DOS (Disk Operating System) หรืออาจใช้โปรแกรมประมวลคำ (Word Processing) ต่าง ๆ

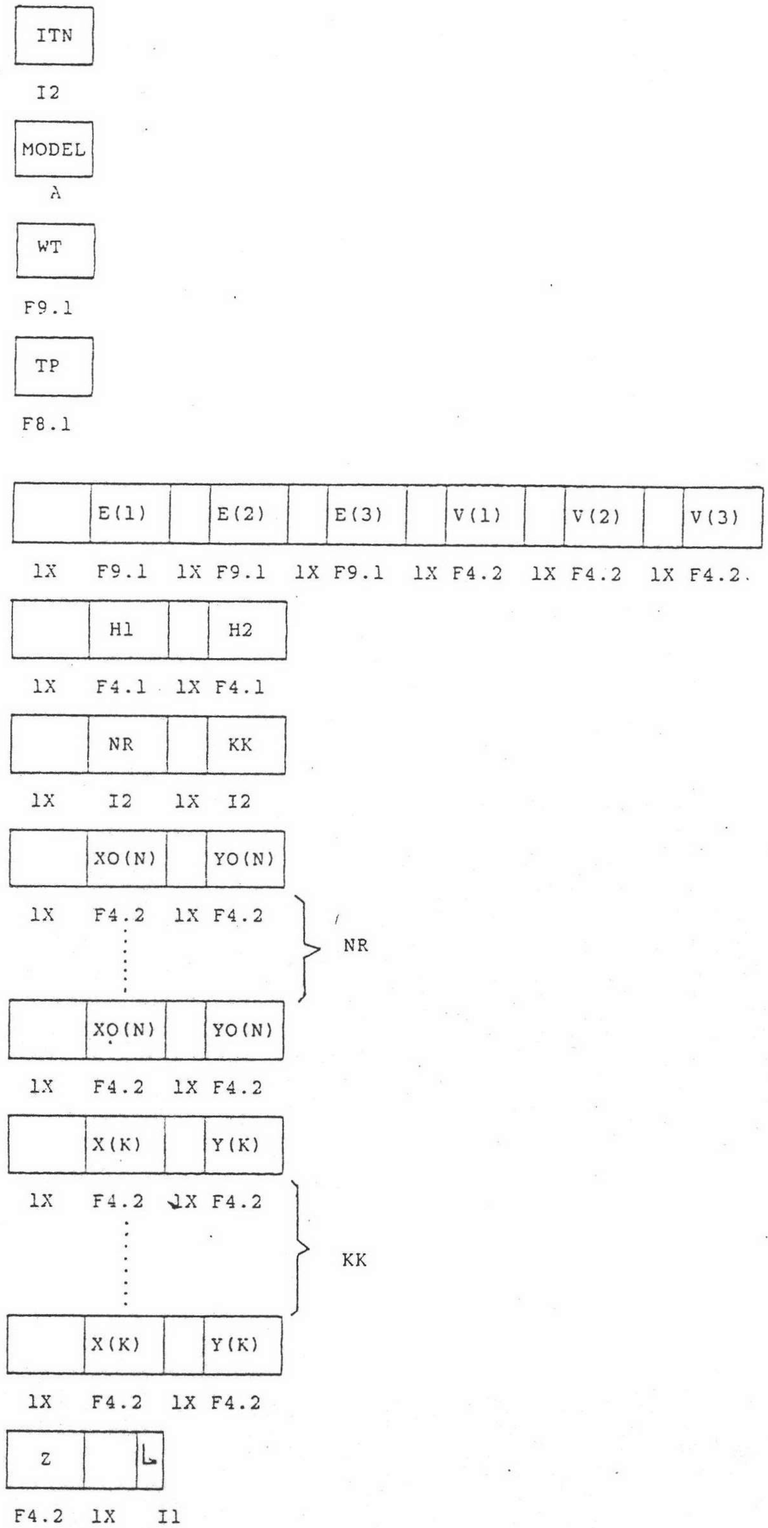
Source Program ที่เขียนขึ้นมานี้ถูกแปลเป็นภาษาเครื่องโดยใช้โปรแกรม Compiler ชื่อ Microsoff (MS) Fortran V 3.3 ของ Microsoft Corpovation of Bellevue, Washington. การแปลและประมวลโปรแกรมใช้เครื่อง IBM-PC ชนิด XT (ไมโครโปรเซสเซอร์ 8088) ภายใต้โปรแกรมจัดระบบงาน DOS. V 2.1 ขึ้นไป

ในกรณีที่ใช้เครื่อง XT (8088) หรือ AT (80286) ซึ่งอาจติด Math - coprocessor (8087/80287) เพื่อทำให้การคำนวณเร็วขึ้นนั้น อาจแปลโดยใช้ MS-Fortran V.4.0 ซึ่งสามารถใช้งานได้ดีกว่า V.3.3 โดยการนำ Source โปรแกรมมาแปลใหม่แล้วเลือก Option ให้ถูกกับชนิดของเครื่องและหน่วยความจำที่ต้องใช้

1. วิธีการใช้โปรแกรม D

การใช้โปรแกรม D มีขั้นตอนดังต่อไปนี้

1.1 สร้าง Text File พร้อมตั้งชื่อของ File ไว้ โดยใช้คำสั่ง EDLIN ซึ่งเป็นคำสั่ง Line Editor มีอยู่ในแผ่น DOS รูปแบบ Format สำหรับการสร้าง Text File สำหรับโปรแกรม D อยู่ในรูปที่ ค-1 ลักษณะ Format ต่าง ๆ มีความหมายตามตารางที่ ค-1



รูปที่ ค-1 รูปแบบ FORMAT การป้อนและอ่านข้อมูลโปรแกรม D

ตารางที่ ค-1 ความหมายของ Format ในภาษา fortran

รูปแบบ	ความหมาย
rIw	เลขจำนวนเต็ม w ตำแหน่ง ซ้ำ r ครั้ง
rFw.d	เลขทศนิยม w ตำแหน่งรวมทั้งจุดทศนิยม ค่า d เป็นจำนวนตัวเลขหลังจุดทศนิยม ซ้ำ r ครั้ง
rX	ช่องว่าง r ตำแหน่ง
Aw	ตัวอักษรจำนวน w ตำแหน่ง

สำหรับความหมายของตัวแปรในรูปที่ ค.1 มีดังนี้

1) ITN ระบุจำนวนช่วงในการหาค่าอินติเกรทของความเค้นเป็นจำนวนเต็ม 2 ตำแหน่ง
2) MODEL ระบุชนิดของเครื่องบิน
3) WT น้ำหนักต่อล้อของกลุ่มล้อหลัก MAIN GEAR (kN)
4) TP แรงดันลมในล้อของกลุ่มล้อหลัก (kN/m^2)
5) E (n) อีลาสติกโมดูลัสของชั้นทางที่ n (kN/M^2)
6) V (n) POISSON'S RATIO ของชั้นทางที่ n
7) H1 ความหนาชั้นผิวทาง A.C (ชั้นที่ 1) หน่วยเป็นเมตร
8) H2 ความหนาชั้นพื้นทาง (ชั้นที่ 2) หน่วยเป็นเมตร
9) NR จำนวนของล้อในกลุ่มล้อหลัก
10) KK จำนวนจุดที่วิเคราะห์
11) XO (n), YO (n) ตำแหน่งของล้อในพิกัดแนวราบ (m)
12) X (k), Y (k) ตำแหน่งที่วิเคราะห์หา STRESS ในแนวราบ (m)

13) Z ความลึกที่วิเคราะห์จากผิวทางหน่วยเป็นเมตร

14) L หมายเลขชั้นทางที่วิเคราะห์ มีความจำเป็นต้องระบุเนื่องจากรอยต่อระหว่างชั้นไม่สามารถระบุเพียงพอด้วยค่า X (k), Y (k) และ Z

<u>ตัวอย่างของ Text File</u>

```
46
B-747-F
185.0
1406.6
1500000.0  200000.0  50000.0 0.35 0.35 0.35
0.40 0.70
4  3
0.00 0.00
1.12 0.00
1.12 1.47
0.00 1.47
0.40 0.75
0.50 0.75
0.55 0.75
1.10 3
```

1.2 เมื่อสร้าง Text File ของข้อมูลเสร็จตาม Format รูปที่ ค-1 โปรแกรม D ตามด้วย ENTER เมื่อเครื่องคอมพิวเตอร์อยู่ในสภาพเตรียมพร้อม

A>d

สักครู่จะปรากฏข้อความดังนี้

NAME OF DATA FILE:

ให้ใส่ชื่อ Text File ของข้อมูลที่สร้างขึ้นตามข้อ 1 แล้วกด ENTER จากนั้นจะปรากฏข้อความบนจอภาพถัดไปคือ

NAME OF DATA FILE: des.dat
NAME OF OUTPUT FILE:

ให้พิมพ์ชื่อ File ของผลลัพธ์ที่ได้ ถ้าต้องการให้แสดงออกทางเครื่องพิมพ์ให้พิมพ์ PRN จากนั้นกด ENTER เครื่องคอมพิวเตอร์ก็จะเริ่มคำนวณให้

สำหรับ DATA FILE และ OUTPUT FILE ให้ใส่ชื่อ DRIVE ของไฟล์ดังกล่าวไว้ข้างหน้าชื่อ FILE ด้วย เช่น A: DES.DAT หรือ B: OUT.DAT เป็นต้น

2. วิธีการใช้โปรแกรม A และ B

2.1 สร้าง Text File ของข้อมูลตามรูป ค-2 โดยมีความหมายของตัวแปรดังนี้

1) TEMP เมื่อชุดอักขระระบุข้อมูลอุณหภูมิให้พิมพ์ดังนี้ DESIGN AVERAGE MEAN ANNUAL AIR TEMP. (F)

2) AVT เป็นค่าเฉลี่ยของอุณหภูมิอากาศประจำปี ของบริเวณทางวิ่งของสนามบินที่ออกแบบ บริเวณข้างเคียงควรหาจากค่าเฉลี่ยหลาย ๆ ปี ตามคู่มือ MS - 11 กำหนดให้ใช้ 30 ปี ค่านี้จะนำมาหาค่าอุณหภูมิของชั้น A.C. เพื่อในการหาค่าอีลาสติกโมดูลัส หน่วย (°F)

3) MAT เป็นชุดอักขระระบุข้อมูลสมบัติของวัสดุให้พิมพ์ดังนี้ MATERIAL PROPERTIES สำหรับส่วนผสมของวัสดุ A.C, และการทดสอบตามคู่มือ MS-11

4) P200 เปอร์เซ็นต์ของมวลรวมที่ผ่านตระแกรงเบอร์ 200

5) VB % ยางแอสฟัลท์ที่เทียบกับส่วนผสมทั้งหมด

6) VV เปอร์เซ็นต์ของฟองอากาศ (Percent of Air Voids)

7) VS ความหนืดสมบูรณ์ (Absolute Viscosity) ที่ 70°F

หน่วย Poise$\times 10^6$

อาจประมาณค่าได้จาก

$VS = 29508.2 \text{ pen}^{-2.1939}$

โดยที่ pen = ค่า penetration ที่อุณหภูมิ 77°F

8) EI ค่าคงที่ k_1 ของชั้นพื้นทางดูจากตารางที่ 2.2 หรือจากการทดสอบสำหรับวัสดุพื้นทางชนิดต่าง ๆ ใช้ในการคำนวณค่าอีลาสติกโมดูลัสของชั้นพื้นทาง E (2)

9) E (3) อีลาสติกโมดูลัสาของชั้นดินเดิม (Subgrade) วิธีการหาอยู่ในหัวข้อ 2.4.2

TEMP
A40

AVT
F4.1

MAT
A35

	P200		VB		VV		VB
1X	F5.2	1X	F5.2	1X	F5.2	1X	F7.1

EI
F8.1

E(3)
F8.1

V(1)		V(2)		V(3)
F4.2	1X	F4.2	1X	F4.2

ITN
I2

NA
I2

AAIRCH
A35

MODEL(J)		NPS(J)		NT(J)		NM(J)		DT(J)		DD(J)		DX(1,J)		DX(2,J)		WT(J)		TP(J)
A10	1X	I5	1X	I1	1X	I1	1X	F4.1	1X	F4.1	1X	F5.2	1X	F5.2	1X	F7.1	1X	F5.1

⋮ N

MODEL(J)		NPS(J)		NT(J)		NM(J)		DT(J)		DD(J)		DX(1,J)		DX(2,J)		WT(J)		TP(J)
A10	1X	I5	1X	I1	1X	I1	1X	F4.1	1X	F4.1	1X	F5.2	1X	F5.2	1X	F7.1	1X	F5.1

รูปที่ ค-2 รูปแบบ FORMAT การป้อนและอ่านข้อมูลโปรแกรม A,B

ตัวอย่างวิธีการหาจากค่า CBR

- ผลการทดสอบค่า CBR จากชั้นดิน Subgrade ของทางวิ่ง 20 ค่าดังนี้

  CBR = 4,4,4,4, 5,5,5,5,5, 6,6,6,6,6, 7,7, 8,8, 10 และ 11

- หา % ของค่า CBR ที่มากกว่าหรือเท่ากับค่า CBR ต่าง ๆ

ค่า CBR	จำนวนที่มากกว่าหรือเท่ากับ	%ที่มากกว่าหรือเท่ากับ
4	20	100
5	16	80
6	11	55
7	6	30
8	4	20
10	2	10
11	1	5

E (3) = 1,500 CBR = 1500 x 4.7 = 7,050 psi

นอกจากนี้ค่า E (3) ยังหาได้จากค่าจากค่า plate Bearing Value ดังรูป 2.11 หรือจากการจำแนกชนิดดินของ FAA ดังนี้

การจำแนกดินตามวิธี FAA	อีลาสติกโมดูลัสของชั้นดินเดิม (psi)	อีลาสติกโมดูลัสของชั้นดินเดิม (kN/m**2)
F10	5,500	37,900
F9	6,500	44,800
F8	7,700	53,100
F7	8,900	61,400
F6	10,800	74,500
F5	12,600	86,900
F4	14,600	100,700
F3	16,600	114,500
F2	19,900	137,200
F1	22,700	156,500
Fa	31,000	213,700

10) V (N) เป็นค่า Poisson's Ratio ของชั้นทางที่ N

11) ITN เป็นจำนวนช่วงการอินติเกรทหาค่า STRESS กำหนดให้ 70 ช่วงมากที่สุด

12) NA จำนวนชนิดของเครื่องบินที่ใช้ออกแบบ กำหนดให้มากที่สุด 10 ชนิด ชนิดและคุณลักษณะของเครื่องบินดูได้จากภาคผนวก ข.

13) AAIRCH ชุดอักขระระบุข้อมูลคุณลักษณะของเครื่องบินที่ใช้ออกแบบให้พิมพ์ AIRCRAFT CHARACTERISTICS

14) MODEL(J) ชื่อชนิดของเครื่องบินหลายเลข J ที่ออกแบบ

15) NPS(J) จำนวนเที่ยวของเครื่องบิน J ที่ผ่านทางวิ่งในช่วงเวลาที่ออกแบบ

16) NT (J) จำนวนแถวของล้อในกลุ่มล้อหลัก

17) NM (J) จำนวนแถวของกลุ่มล้อหลัก

18) DT (J) ระยะระหว่างแถวของล้อขนานกับเส้นกึ่งกลางเครื่องบิน (ศูนย์กลางถึงศูนย์กลางหน่วยเป็นนิ้ว)

19) DD (J) ระยะระหว่างล้อในแถวเดียวกันตามขวางเครื่องบิน (ศูนย์กลางถึงศูนย์กลาง) หน่วยเป็นนิ้ว

20) DX(1,J) ระยะระหว่าง ศ.ก. ของกลุ่มล้อหลักกลุ่มแรกถึงเส้นกึ่งกลางเครื่องบินหน่วยเป็นฟุต

21) DX(2,J) ระยะระหว่าง ศ.ก. ของกลุ่มหลักกลุ่มหลักถึงเส้นกึ่งกลางเครื่องบินหน่วยเป็นฟุต

22) WT (J) น้ำหนักต่อล้อ ของกลุ่มล้อหลัก (16)

23) TP (J) แรงดันลมในล้อในกลุ่มล้อหลัก (16)

<u>ตัวอย่างของ Text File</u>

```
 DESIGN AVERAGE MEAN ANNUAL AIR TEMP.(F)
80.0
MATERIAL PROPERTIES
  5.00  5.31  4.00      1.5
  8000.0
  7000.0
0.50 0.30 0.30
46
 2
AIRCRAFT CHARACTERISTICS
 B-747-F     70000 2 2 58.0 48.0 18.50  6.25 45500.0 185.0
 B-727-200   95000 1 1 00.0 34.0  9.38  0.00 39900.0 168.0
```

2.2 เมื่อสร้าง Text File ของข้อมูลเสร็จตาม Format ในรูปที่ ค 2 ให้พิมพ์ A หรือ B แล้วแต่จะเลือกประมวลผลโปรแกรมไหนก่อน จากนั้นกด ENTER สักครู่จะปรากฏข้อความบนจอภาพดังนี้

```
C:\>a

 NAME OF DATA FILE:  des2.dat
```

ให้ใส่ชื่อ Text File ข้อมูลแล้วกด ENTER จะปรากฏข้อความบนจอภาพดังนี้

```
NAME OF DATA FILE:  des2.dat
NAME OF OUTPUT FILE:
```

ให้พิมพ์ชื่อ File ข้อมูลที่จะแสดงผล ถ้าให้แสดงผลออกทางเครื่องพิมพ์ให้พิมพ์ PRN แล้วกด ENTER

```
NAME OF DATA FILE:  des2.dat
NAME OF OUTPUT FILE:  prn
```

ทั้งนี้ให้ระบุ Drive ของ Data File และ Output File ด้วย

2.3 หลังจากพิมพ์ชื่อ OUTPUT FILE แล้วกด ENTER แล้วจอภาพจะปรากฏข้อความให้ป้อนความหนาชั้นพื้นทางดังนี้

```
INPUT GARNULAR BASE THICKNESS(ins.)=?
```

ให้ป้อนขอ้มูลความหนาชั้นพื้นทางเข้าไป จากนั้นจึงป้อนความหนาชั้น A.C. หลังข้อความต่อไปที่ปรากฏบนจอคือ

```
INPUT GARNULAR BASE THICKNESS(ins.)=?10.0
INPUT A.C. SURFACE THICKNESS(ins.)=?
```

เมื่อป้อนความหนาชั้น A.C (h_1) แล้วเครื่องก็จะคำนวณหาค่า N_a และ N_p จนเสร็จ ก็ต้องป้อนค่า h_1 ใหม่ หลังจากที่จอภาพปรากฏข้อความแสดงค่า h_1 , N_a และ N_p พร้อมกันให้ป้อนค่า h_1 ใหม่

ต้องป้อนค่า h_1 จำนวนทั้งหมด 5 ครั้ง ควรเลือกค่าที่เหมาะสมที่จะทำให้เส้นความสัมพันธ์ h_1-N_a กับ h_1-N_p ตัด กันเพื่อที่จะหาค่า h_1 ที่เหมาะสมซึ่งให้ค่า $N_a = N_p$ ได้

การป้อนค่า h_1 มากหรือน้อยจนเกินไปจะทำให้เกิดความผิดพลาดและโปรแกรมหยุดทำงาน เนื่องจากตัวเลขมากเกินเครื่องจะรับได้ (Overflow) หรือเกินค่าLog ของศูนย์ เป็นต้น ทำให้ต้องประมวลผลโปรแกรมใหม่

เมื่อหาค่าความหนา h_1 ที่ความหนาชั้นพื้นทางใด ๆ ได้ จอภาพจะปรากฏ

```
INPUT GARNULAR BASE THICKNESS(ins.)=?10.0
INPUT A.C. SURFACE THICKNESS(ins.)=?12.0
NO.   H1        Na        Np
 1  12.00     32750     70447
INPUT A.C. SURFACE THICKNESS(ins.)=?13.0
NO.   H1        Na        Np
 1  12.00     32750     70447
 2  13.00     49697     71106
INPUT A.C. SURFACE THICKNESS(ins.)=?14.0
NO.   H1        Na        Np
 1  12.00     32750     70447
 2  13.00     49697     71106
 3  14.00     74454     71444
INPUT A.C. SURFACE THICKNESS(ins.)=?15.0
NO.   H1        Na        Np
 1  12.00     32750     70447
 2  13.00     49697     71106
 3  14.00     74454     71444
 4  15.00    109921     71461
INPUT A.C. SURFACE THICKNESS(ins.)=?16.0
THE FINAL THICKNESS IS 13.88 in.
PRESS 1 => SELECT NEW GRANULAR BASE THICKNESS
      2 => QUIT
 SELECT ============>
```

ให้เลือก 1 เมื่อต้องการหาความหนา h_1 ที่ความหนาของชั้นพื้นทางใหม่

```
NO.   Hl        Na        Np
 1  12.00      32750     70447
INPUT A.C. SURFACE THICKNESS(ins.)=?13.0
NO.   Hl        Na        Np
 1  12.00      32750     70447
 2  13.00      49697     71106
INPUT A.C. SURFACE THICKNESS(ins.)=?14.0
NO.   Hl        Na        Np
 1  12.00      32750     70447
 2  13.00      49697     71106
 3  14.00      74454     71444
INPUT A.C. SURFACE THICKNESS(ins.)=?15.0
NO.   Hl        Na        Np
 1  12.00      32750     70447
 2  13.00      49697     71106
 3  14.00      74454     71444
 4  15.00     109921     71461
INPUT A.C. SURFACE THICKNESS(ins.)=?16.0
THE FINAL THICKNESS IS 13.88 in.
PRESS 1 => SELECT NEW GRANULAR BASE THICKNESS
      2 => QUIT
 SELECT ============> 1
INPUT GARNULAR BASE THICKNESS(ins.)=?
```

ให้เลือก 2 เมื่อต้องการสิ้นสุดการประมวลผล

```
NO.   Hl        Na        Np
 1  12.00      34048     70256
 2  10.00      14611     68093
INPUT A.C. SURFACE THICKNESS(ins.)=?13.0
NO.   Hl        Na        Np
 1  12.00      34048     70256
 2  10.00      14611     68093
 3  13.00      51532     70958
INPUT A.C. SURFACE THICKNESS(ins.)=?14.0
NO.   Hl        Na        Np
 1  12.00      34048     70256
 2  10.00      14611     68093
 3  13.00      51532     70958
 4  14.00      77040     71341
INPUT A.C. SURFACE THICKNESS(ins.)=?15.0
THE FINAL THICKNESS IS 13.81 in.
PRESS 1 => SELECT NEW GRANULAR BASE THICKNESS
      2 => QUIT
 SELECT ============> 2
    NO.      h2(ins.)    hl(ins.)
    1         10.00       13.88
    2         11.00       13.81
Stop - Program terminated.
```

ประวัติผู้เขียน

นายกิจธวัช ตั้งบุญธินา เกิดเมื่อวันที่ 20 กุมภาพันธ์ พ.ศ. 2506 ที่จังหวัดนครศรีธรรมราช สำเร็จการศึกษาระดับปริญญาตรีวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมโยธา จาก มหาวิทยาลัยสงขลานครินทร์ เมื่อปี พ.ศ. 2528 จากนั้นจึงเข้าเรียนต่อในบัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย ในสาขาวิศวกรรมโยธา เมื่อปี พ.ศ. 2530 ปัจจุบันปฏิบัติราชการอยู่ที่ฝ่ายบูรณะซ่อมบำรุงทางและโครงสร้าง กองบูรณะและซ่อมบำรุง สำนักงานเร่งรัดพัฒนาชนบท