REPORT NO. FRA-OR&D-76-135.II

FREQUENCY DOMAIN COMPUTER PROGRAMS FOR PREDICTION AND ANALYSIS OF RAIL VEHICLE DYNAMICS
Volume II: Appendixes

A. B. Perlman
F. P. DiMasi

DECEMBER 1975
FINAL REPORT

Prepared for

U.S. DEPARTMENT OF TRANSPORTATION
FEDERAL RAILROAD ADMINISTRATION
Office of Research and Development
Washington DC 20590

NOTICE

This document is disseminated under the sponsorship of the Department of Transportation in the interest of information exchange. The United States Government assumes no liability for its contents or use thereof.

NOTICE

The United States Government does not endorse products or manufacturers. Trade or manufacturers' names appear herein solely because they are considered essential to the object of this report.

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
FRA-OR&S-76-135.II		

4. Title and Subtitle	5. Report Date
FREQUENCY DOMAIN COMPUTER PROGRAMS FOR PREDICTION AND ANALYSIS OF RAIL VEHICLE DYNAMICS - Volume II: Appendixes	December 1975
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
A.B. Perlman and F.P. DiMasi	DOT-TSC-FRA-75-16.II
9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
U.S. Department of Transportation Transportation Systems Center Kendall Square Cambridge MA 20142	RR515/R6321
	11. Contract or Grant No.
12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
U.S. Department of Transportation Federal Railroad Administration Office of Research and Development Washington DC 20590	Final Report July 1973 - May 1975
	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

Frequency domain computer programs developed or acquired by TSC for the analysis of rail vehicle dynamics are described in two volumes.

Volume I defines the general analytical capabilities required for computer programs applicable to single rail vehicles and presents a detailed description of frequency domain programs developed at TSC in terms of analytical capabilities, model description, equations of motion, solution procedure, input/output parameters, and program control logic. Descriptions of programs FULL, FLEX, LATERAL, and HALF are presented. TSC programs for assessing lateral dynamic stability of single rail vehicles are also described.

Volume II contains program listings including subroutines for the four TSC frequency domain programs described in Volume I.

17. Key Words	18. Distribution Statement
Rail Vehicle Dynamics, Computer Modeling Frequency, Domain Computer Programs, Lateral Dynamic Stability, Computer Programs	DOCUMENT IS AVAILABLE TO THE PUBLIC THROUGH THE NATIONAL TECHNICAL INFORMATION SERVICE, SPRINGFIELD, VIRGINIA 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	102	

Form DOT F 1700.7 (8-72)

TABLE OF CONTENTS

Volume II

APPENDIX A
PROGRAM FULL INPUT PARAMETERS AND LISTINGS

The Program FULL input deck consists of the following:

<u>Card 1</u> UNITS: MPH

|F7.3|
V

V = Vehicle velocity

<u>Card 2</u> UNITS: FEET

|F7.3|
L

L = Axle spacing (Truck wheelbase)

<u>Card 3</u> UNITS: FEET

|F7.3|
2a

2a = Truck spacing (Vehicle wheelbase)

<u>Card 4</u> UNITS: LBS/IN

|FR.4|
K1

K1 = Primary suspension vertical stiffness (Equalizer spring constant)

<u>Card 5</u> UNITS: LBS

|F12.4|
W1

W1 = Truck frame assembly (not including wheelsets)

<u>Card 6</u> UNITS: LBS/IN

|F12.4|
K2

K2 = Secondary suspension vertical stiffness (Bolster spring constant)

Card 7 UNITS: LBS

|F12.4|
W2

W2 = Car-body weight

Cards 8, 9, 10 (Frequency Control Variables)

(8) UNITS: NONE

|I2 | I2
IFREQ NDF

(9) UNITS: NONE

|7I4|
DF(N), N=1, NDF FOR IFREQ = 1
DF(1) FOR IFREQ = 2

(10)

|8F10.4| UNITS: HERTZ
FL(N) N=1, NDF+1 FOR IFREQ = 1
FL(1) FOR IFREQ = 2

IFREQ selects one of two methods for specifying the frequency points at which the response will be found. With IFREQ = 1, logarithmically evenly-spaced points are generated for a set of frequency ranges. NDF gives the number of frequency ranges (maximum of seven). DF(N) specifies the number of points for the range FL(N) to FL(N+1). FL(N) gives the lower limit of the Nth frequency range, and FL(N+1) is the upper limit for the Nth range. FL(NDF+1) gives the upper limit for the last frequency range.

With IFREQ=2, logarithmically evenly-spaced points are generated for a set of octaves. NDF specifies the number of octaves. DF(1) gives the number of points per octave, and FL(1) gives the starting frequency of the first octave. Note that the remainders of the DF and FL fields, are left blank.

Card 11 UNITS: NONE

|I2|
N1

N1 = The number of BETAS (BETA is the damping ratio associated with the secondary suspension) to be used in the response computation of vehicle response. (A maximum of 7 values)

Card 12

|7F10.4|
B(J)

UNITS: NONE

B(J) = Values of BETA (secondary suspension damping ratio)

Card 13

UNITS: FEET

|F12.4|
RHO

RHO = Radius of gyration of car-body

Card 14

UNITS: NONE

| L6 | L6 | L6|
OPT1 OPT2 OPT3

OPT 1 controls displacement plots

OPT1 = TRUE for plots

OPT1 = FALSE for no plots

OPT2 controls acceleration plots

OPT2 = TRUE for plots

OPT2 = FALSE for no plots

OPT3 controls acceleration response due to track irregularity of the form $v_o = a_1\lambda$ (amplitude promotional to wavelength)

OPT3 = TRUE for plots

OPT3 = FALSE for no plots

Card 15

UNITS: NONE

|L6 | L6 | L6|
OPT1 OPT2 OPT3

OPT1, OPT2, and OPT3 have logical values identical to those of Card 14, and control printing or non-printing of results for displacement, acceleration and acceleration spectra responses.

The following pages contain the listing for Program FULL.

```
C********************************
C** FULL (7-29-75)
C********************************
C** FOLLOWING TEXT PRINTED FROM FILE DSKB:FULLPS.F4 [4351,533] 29-JUL-75
C********************************
      DIMENSION SAVEM(200,40),IITYPE(40)
      DATA IITYPE/ 25*1,15*2/
      LOGICAL PRINT
      LOGICAL OPT1,OPT2,OPT3
      DIMENSION BB (5 )
      DIMENSION  B (5 )
      DIMENSION F(200)
      DIMENSION PF(200)
      DIMENSION LAM(200)
      INTEGER DF(7)
      DIMENSION FL(8)
      DIMENSION Z2DDV(200,5)
      DIMENSION S(200)
      DIMENSION APDDV(200,5)
      DIMENSION Z2A(200,5)
      DIMENSION APA(200,5)
      DIMENSION ZADDV(200, 5),Z2APA(200, 5)
      DIMENSION ZZV(200, 5),AAPV(200, 5),ZZVN(200,5 ),AAPVN(200,5)   ,
     1ZZAA(200,5)
      REAL M
      REAL LAM
      EQUIVALENCE (SAVEM(1,1),ZZV),(SAVEM(1,6),AAPV),
     X(SAVEM(1,11),ZZVN),(SAVEM(1,16),AAPVN),(SAVEM(1,21),ZZAA),
     X(SAVEM(1,26),Z2DDV),(SAVEM(1,31),APDDV),(SAVEM(1,36),ZADDV)
      REAL K1, K2
      CALL PLOTS(IBUF,360,16)
      CALL PLOT(0.,-11.,-3)
      CALL PLOT(0.,1.1,-3)
      PI=3.14159265
C FIRST DATA CARD IS VELOCITY IN MPH
      READ(20,1) V1
    1 FORMAT(F7.3)
      V=V1*5280.0/3600.0
C SECOND DATA CARD IS L IN FEET
      READ(20,2) XL
    2 FORMAT(F7.3)
C THIRD DATA CARD IS LL IN FEET
      READ(20,3) XLL
    3 FORMAT(F7.3)
C FOURTH DATA CARD IS K1
      READ(20,4) K1
C FIFTH DATA CARD IS W1
      READ(20,4) W1
C SIXTH DATA CARD IS K2
      READ(20,4) K2
```

```
C********************************
C** FULL (7-29-75)
C********************************
C SEVENTH DATA CARD IS W2
      READ(20,4) W2
    4 FORMAT(F12.4)
      W1Z=SQRT(K1*386./W1)
      W2Z=SQRT(K2*386./W2)
      M=W2/W1
C EIGHTH DATA CARD
      READ(20,10) IFREQ,NDF
   10 FORMAT(2I2)
C NINTH DATA CARD
      READ(20,20) DF
   20 FORMAT(7I4)
C TENTH DATA CARD
      READ(20,30) FL
   30 FORMAT(8F10.4)
C ELEVENTH DATA CARD IS NUMBER OF BETAS
      READ(20,35) N1
   35 FORMAT(I2)
C TWELFTH DATA CARD IS VALUES FOR BETA
      READ(20,40) (  B(J),J=1,N1)
   40 FORMAT (7F10.4)
C THIRTEENTH DATA CARD IS RHO
      READ(20,4) XRH
      G=(XL/(2.*XRH))**2
C FORTEENTH DATA CARD IS LOGICAL VALUES FOR PLOTTING OPTIONS
      READ(20,404) OPT1,OPT2,OPT3
  404 FORMAT(3L6)
C FIFTEENTH DATA CARD IS A LOGICAL VALUE FOR PRINTIN7 OR NOT PRINTING
C THE RESULTS
      READ (20,404) PRINT
      WRITE(6,1600)
 1600 FORMAT(1H1,5X,'INPUT DATA')
      WRITE(6,1601)
 1601 FORMAT(6X,'----- ----')
      WRITE(6,41) V
   41 FORMAT(1H0,5X,4HV = ,F7.3)
      WRITE(6,42) XL
   42 FORMAT(1H0,5X,5HXL = ,F7.3)
      WRITE(6,43) XLL
   43 FORMAT(1H0,5X,6HXLL = ,F7.3)
      WRITE(6,44) M
   44 FORMAT(1H0,5X,4HM = ,F10.5)
      WRITE(6,45) W1Z
   45 FORMAT(1H0,5X,6HW1Z = ,F10.5)
      WRITE(6,46) W2Z
   46 FORMAT(1H0,5X,6HW2Z = ,F10.5)
      WRITE(6,47) N1
```

```
C******************************
C** FULL (7-29-75)
C******************************
   47 FORMAT(1H0,5X,18HNUMBER OF BETAS = ,I2)
C COMPUTE FREQUENCIES
      DO 55 J=1,N1
      WRITE(6,48)   B(J)
   48 FORMAT(1H0,5X,7HBETA = ,F10.4)
      BB(J)=B(J)
   55 CONTINUE
      SQ2=SQRT(2.)
      DO 407 J=1,N1
      B  (J)=B(J)/SQ2
  407 CONTINUE
      WRITE(6,511) IFREQ,NDF
  511 FORMAT(/5X,6HIFREQ=,I2,5X,4HNDF=,I2)
      WRITE(6,513) DF
  513 FORMAT(/5X,4HDF =,7I4)
      WRITE(6,515) FL
  515 FORMAT(/5X,8HFL(HZ) =,10F10.4)
      WRITE (6,516) XRH
  516 FORMAT(1H0,5X,6HRHO = ,F12.4)
      WRITE (6,406) OPT1,OPT2,OPT3
      WRITE (6,408) PRINT
  406 FORMAT(1H0,5X,7HOPT1 = ,L3,11H ,  OPT2 = ,L3,11H ,  OPT3 = ,L3)
  408 FORMAT(1H0,5X,8HPRINT = ,L6)
      CALL FTABG(IFREQ,NDF,DF,FL,F,NF)
C COMPUTE LAMBDAS
      DO 80 N=1,NF
      I=N
      LAM(I)=V/F(N)
   80 CONTINUE
C COMPUTE Z2/VM FOR EVERY S FOR ALL BETAS
      DO 100 J=1,N1
      DO 100 I=1,NF
C COMPUTE S FOR EVERY LAMBDA
      S(I)=2.0*PI*F(I)
      C1=2.0*B(J)/W2Z
      C2=W1Z**2+(2.0+M)*W2Z**2
      C3=(2.0+M)*B(J)
      C4=2.0*(W1Z**2)*(W2Z**2)
      C5=W2Z*W1Z**2
      A1=C2/C4
      A2=C3/C5
      A3=1.0/C4
C REAL TERMS IN DENOMINATOR
      DR=1.0-(A1*S(I)**2)+A3*S(I)**4
C IMAGINARY TERMS IN DENOMINATOR
      DI=C1*S(I)-(A2*S(I)**3)
      DM=DR**2+DI**2
```

```
C******************************
C** FULL (7-29-75)
C******************************
C REAL TERMS
      Z2VR=(DR+C1*S(I)*DI)/DM
C IMAGINARY TERMS
      Z2VI=(-DI+C1*S(I)*DR)/DM
C Z2/VM EQUALS
      Z2VM=SQRT(Z2VR**2+Z2VI**2)
      ZZV(I,J)=Z2VM
C COMPUTE APHI/V DELTA FOR EVERY S FOR ALL BETAS
      X1=2.0*B(J)
      X2=W1Z**2+((2.0*G+M)*W2Z**2)
      X3=2.0*G*(W1Z**2)*(W2Z**2)
      X4=(2.0*G+M)*B(J)
      X5=G*W2Z*(W1Z**2)
      Y1=X1/W2Z
      Y2=X2/X3
      Y3=X4/X5
      Y4=1.0/X3
C REAL TERMS IN DENOMINATOR
      YR=1.0-(Y2*S(I)**2)+(Y4*S(I)**4)
C IMAGINARY TERMS IN DENOMINATOR
      YI=(Y1*S(I))-(Y3*S(I)**3)
      YM=(YR**2)+(YI**2)
C REAL TERMS
      APVR=(-YR-(YI*Y1*S(I)))/YM
C IMAGINARY TERMS
      APVI=(YI-YR*(Y1*S(I)))/YM
C APHI/V DELTA EQUALS
      APVD=SQRT(APVR**2+APVI**2)
      AAPV(I,J)=APVD
      FB=2.0*PI/LAM(I)
C COMPUTE Z2/V0 FOR EVERY LAMBDA FOR ALL BETAS
      FBLL=FB*XLL
      FBL=FB*XL
      CFBLL=COS(FBLL)
      CFBL=COS(FBL)
      SFBLL=SIN(FBLL)
      SFBL=SIN(FBL)
      A1=1.0+CFBLL+CFBL+CFBLL*CFBL-SFBLL*SFBL
      B1=SFBLL+SFBL+SFBLL*CFBL+CFBLL*SFBL
C Z2/V0 EQUALS
      Z2VN=(SQRT(A1**2+B1**2)*Z2VM)/4.0
      ZZVN(I,J)=Z2VN
C COMPUTE APHI/V0 FOR EVERY LAMBDA FOR ALL BETAS
      A2=1.0+CFBLL-CFBL-CFBLL*CFBL+SFBLL*SFBL
      B2=SFBLL-SFBL-SFBLL*CFBL-CFBLL*SFBL
C APHI/V0 EQUALS
      APVN=(SQRT(A2**2+B2**2)*APVD)/4.0
```

```
C*******************************
C** FULL (7-29-75)
C*******************************
      AAPVN(I,J)=APVN
C COMPUTE Z2/V0 + A PHI/V0
      P1=Z2VR*A1
      P2=Z2VI*B1
      P3=Z2VI*A1
      P4=Z2VR*B1
      ZAR1=P1-P2
      ZAI1=P3+P4
      P5=APVR*A2
      P6=APVI*B2
      P7=APVR*B2
      P8=APVI*A2
      ZAR2=P5-P6
      ZAI2=P7+P8
      ZAR3=ZAR1+ZAR2
      ZAI3=ZAI1+ZAI2
      ZAVN=SQRT(ZAR3**2+ZAI3**2)/4.0
      ZZAA(I,J)=ZAVN
      PF(I)=((2.0*PI*F(I))**2)/386.
      Z2DDV(I,J)=PF(I)*Z2VN
      APDDV(I,J)=PF(I)*APVN
      ZADDV(I,J)=ZAVN*PF(I)
      APA(I,J)=APDDV(I,J)*LAM(I)*12.
      Z2APA(I,J)=ZADDV(I,J)*LAM(I)*12.
      Z2A(I,J)=Z2DDV(I,J)*LAM(I)*12.
  100 CONTINUE
      IF (.NOT.PRINT) GO TO 405
  601 FORMAT(1H0,4X,9HFREQUENCY,10X,7HAPHI/V0,4X,12H(Z2+APHI)/V0)
  603 FORMAT(1H0,3(2X,E13.6,2X))
  600 FORMAT(1H0,4X,9HFREQUENCY,9X,9HAPHIDD/A1,11X,5HZ2/VB,10X,7HAPHI/VD
     X)
  125 FORMAT(1H0,4X,9HFREQUENCY,5X,16H(Z2DD+APHIDD)/V0,1X,16H(Z2DD+APHID
     XD)/A1,5X,5HZ2/V0)
  112 FORMAT(1H0,4(2X,E13.6,2X))
  110 FORMAT(1H0,4X,9HFREQUENCY,9X,7HZ2DD/V0,9X,9HAPHIDD/V0,9X,7HZ2DD/A1
     X)
   94 FORMAT(1H1,3X,5HBETA=,F6.4)
      DO 120 J=1,N1
      WRITE(6,94) BB(J)
      WRITE(6,110)
      DO 119 I=1,NF
      WRITE(6,112) F(I),Z2DDV(I,J),APDDV(I,J),Z2A(I,J)
  119 CONTINUE
  120 CONTINUE
      DO 156 J=1,N1
      WRITE(6,94) BB(J)
      WRITE(6,600)
```

```
C*******************************
C** FULL (7-29-75)
C*******************************
      DO 157 I=1,NF
      WRITE(6,112) F(I),APA(I,J),ZZV(I,J),AAPV(I,J)
  157 CONTINUE
  156 CONTINUE
      DO 140 J=1,N1
      WRITE(6,94) BB(J)
      WRITE(6,125)
      DO 130 I=1,NF
      WRITE(6,112) F(I),ZADDV(I,J),Z2APA(I,J),ZZVN(I,J)
  130 CONTINUE
  140 CONTINUE
      DO 166 J=1,N1
      WRITE(6,94) BB(J)
      WRITE(6,601)
      DO 167 I=1,NF
      WRITE(6,603) F(I),AAPVN(I,J),ZZAA(I,J)
  167 CONTINUE
  166 CONTINUE
  405 CONTINUE
C
C     TEST FOR PSD PLOTS
C
      IPSD=0
      READ(20,810,END=815) PSD
 810  FORMAT(A5)
      IF(PSD .EQ. 'RAILP') IPSD=1
 815  CONTINUE
      IF(IPSD .EQ. 0) GO TO 820
      CALL RAILPL(F,V*12.,NF,40,IITYPE,SAVEM,200,20,6)
 820  CONTINUE
      R=.0001
      R1=.00001
      R2=.001
      DO 200 I=1,NF
      F(I)=ALOG10(F(I))
      DO 300 J=1,N1
      IF(Z2DDV(I,J).LT.R) Z2DDV(I,J)=R
      Z2DDV(I,J)=ALOG10(Z2DDV(I,J))
      IF(APDDV(I,J).LT.R) APDDV(I,J)=R
      APDDV(I,J)=ALOG10(APDDV(I,J))
      IF(Z2A(I,J).LT.R2) Z2A(I,J)=R2
      Z2A(I,J)=ALOG10(Z2A(I,J))
      IF(APA(I,J).LT.R2) APA(I,J)=R2
      APA(I,J)=ALOG10(APA(I,J))
      IF(Z2APA(I,J).LT.R2) Z2APA(I,J)=R2
      Z2APA(I,J)=ALOG10(Z2APA(I,J))
      IF(ZADDV(I,J).LT.R) ZADDV(I,J)=R
```

```
C*******************************
C** FULL (7-29-75)
C*******************************
      ZADDV(I,J)=ALOG10(ZADDV(I,J))
      IF(ZZV(I,J).LT.R1) ZZV(I,J)=R1
      ZZV(I,J)=ALOG10(ZZV(I,J))
      IF(AAPV(I,J).LT.R1) AAPV(I,J)=R1
      AAPV(I,J)=ALOG10(AAPV(I,J))
      IF(ZZVN(I,J).LT.R1) ZZVN(I,J)=R1
      ZZVN(I,J)=ALOG10(ZZVN(I,J))
      IF(AAPVN(I,J).LT.R1) AAPVN(I,J)=R1
      AAPVN(I,J)=ALOG10(AAPVN(I,J))
      IF(ZZAA(I,J).LT.R1) ZZAA(I,J)=R1
      ZZAA(I,J)=ALOG10(ZZAA(I,J))
  300 CONTINUE
  200 CONTINUE
      F(NF+1)=-1.
      F(NF+2)=.5
      FDEL=F(NF+1)
       FSCA=F(NF+2)
      G1=-4.
      G2=-3.
      G11=-5.
      DO 500 J=1,N1
      Z2DDV(NF+1,J)=G1
      Z2DDV(NF+2,J)=1.
      APDDV(NF+1,J)=G1
      APDDV(NF+2,J)=1.
      Z2A(NF+1,J)=G2
      Z2A(NF+2,J)=1.
      APA(NF+1,J)=G2
      APA(NF+2,J)=1.
      Z2APA(NF+1,J)=G2
      Z2APA(NF+2,J)=1.
      ZADDV(NF+1,J)=G1
      ZADDV(NF+2,J)=1.
      ZZV(NF+1,J)=G11
      ZZV(NF+2,J)=1.
      AAPV(NF+1,J)=G11
      AAPV(NF+2,J)=1.
      ZZVN(NF+1,J)=G11
      ZZVN(NF+2,J)=1.
      AAPVN(NF+1,J)=G11
      AAPVN(NF+2,J)=1.
      ZZAA(NF+1,J)=G11
      ZZAA(NF+2,J)=1.
  500 CONTINUE
      IF(OPT1) GO TO 401
      IF(OPT2) GO TO 402
      IF(OPT3) GO TO 403
```

```
C*******************************
C** FULL (7-29-75)
C*******************************
      GO TO 400
  401 CONTINUE
      CALL LAXIS(0.,0.,3,9HFREQUENCY,9,6.,-1,2)
      CALL LAXIS(0.,0.,7,13HZ2/VB (IN/IN),13,7.,-5,1)
      CALL HVGRID(0.,0.,2.,1.,3,7)
      CALL SYMBOL(0.5,8.75,0.14,36H     VERTICAL TRANSMISSIBILITY      ,
     X0.0,36)
      DO 1070 I=1,N1
      CALL CLINE(F,ZZV(1,I),NF,FDEL,FSCA,ZZV(NF+1,I),ZZV(NF+2,I),1)
      XP=(F(NF)-F(NF+1))/F(NF+2)
      YP=(ZZV(NF,I)-ZZV(NF+1,I))/ZZV(NF+2,I)
      CALL SYMBOL(XP,YP,.07,3HB =,0.,3)
      CALL NUMBER(999.,999.,  .07,BB(I),0.,2)
 1070 CONTINUE
      CALL GPR(V1,XLL,XL,XRH,W1,W2,K1,K2,V)
      CALL PLOT(12.,0.,-3)
      CALL LAXIS(0.,0.,3,9HFREQUENCY,9,6.,-1,2)
      CALL LAXIS(0.,0.,7,15HAPHI/VD (IN/IN),15,7.,-5,1)
      CALL HVGRID(0.,0.,2.,1.,3,7)
      CALL SYMBOL(0.5,8.75,0.14,38HROTATIONAL (PITCHING) TRANSMISSIBILIT
     XY,0.0,38)
      DO 1080 I=1,N1
      CALL CLINE(F,AAPV(1,I),NF,FDEL,FSCA,AAPV(NF+1,I),AAPV(NF+2,I),1)
      XP=(F(NF)-F(NF+1))/F(NF+2)
      YP=(AAPV(NF,I)-AAPV(NF+1,I))/AAPV(NF+2,I)
      CALL SYMBOL(XP,YP,.07,3HB =,0.,3)
      CALL NUMBER(999.,999.,  .07,BB(I),0.,2)
 1080 CONTINUE
      CALL GPR(V1,XLL,XL,XRH,W1,W2,K1,K2,V)
      CALL PLOT(12.,0.,-3)
      CALL LAXIS(0.,0.,3,9HFREQUENCY,9,6.,-1,2)
      CALL LAXIS(0.,0.,7,13HZ2/V0 (IN/IN),13,7.,-5,1)
      CALL HVGRID(0.,0.,2.,1.,3,7)
      CALL SYMBOL(0.5,8.75,0.14,36H  VERTICAL CAR-CENTER DISPLACEMENT   ,
     X0.0,36)
      CALL SYMBOL(0.5,8.55,0.14,36HDUE TO SINUSOIDAL TRACK IRREGULARITY,
     X0.0,36)
      DO 1090 I=1,N1
      CALL CLINE(F,ZZVN(1,I),NF,FDEL,FSCA,ZZVN(NF+1,I),ZZVN(NF+2,I),1)
      YP=(ZZVN(NF,I)-ZZVN(NF+1,I))/ZZVN(NF+2,I)
      XP=(F(NF)-F(NF+1))/F(NF+2)
      CALL SYMBOL(XP,YP,.07,3HB =,0.,3)
      CALL NUMBER(999.,999.,  .07,BB(I),0.,2)
 1090 CONTINUE
      CALL GPR(V1,XLL,XL,XRH,W1,W2,K1,K2,V)
      CALL PLOT(12.,0.,-3)
      CALL LAXIS(0.,0.,3,9HFREQUENCY,9,6.,-1,2)
```

```
C*******************************
C** FULL (7-29-75)
C*******************************
      CALL LAXIS(0.,0.,7,15HAPHI/V0 (IN/IN),15,7.,-5,1)
      CALL HVGRID(0.,0.,2.,1.,3,7)
      CALL SYMBOL(0.5,8.75,0.14,36H        PITCHING RESPONSE TO        ,
     X0.,0,36)
      CALL SYMBOL(0.5,8.55,0.14,36H   SINUSOIDAL TRACK IRREGULARITY    ,
     X0.,0,36)
      DO 2000 I=1,N1
      CALL CLINE(F,AAPVN(1,I),NF,FDEL,FSCA,AAPVN(NF+1,I),AAPVN(NF+2,I),1
     1)
      XP=(F(NF)-F(NF+1))/F(NF+2)
      YP=(AAPVN(NF,I)-AAPVN(NF+1,I))/AAPVN(NF+2,I)
      CALL SYMBOL(XP,YP,.07,3HB =,0.,3)
      CALL NUMBER(999.,999.,  .07,BB(I),0.,2)
 2000 CONTINUE
      CALL GPR(V1,XLL,XL,XRH,W1,W2,K1,K2,V)
      CALL PLOT(12.,0.,-3)
      CALL LAXIS(0.,0.,3,9HFREQUENCY,9,6.,-1,2)
      CALL LAXIS(0.,0.,7,20H(Z2+APHI)/V0 (IN/IN),20,7.,-5,1)
      CALL HVGRID(0.,0.,2.,1.,3,7)
      CALL SYMBOL(0.5,8.75,0.14,38HVERTICAL CAR DISPLACEMENT (OVER TRUCK
     X),0.,0,38)
      CALL SYMBOL(0.5,8.55,0.14,36HDUE TO SINUSOIDAL TRACK IRREGULARITY,
     X0.,0,36)
      DO 2010 I=1,N1
      CALL CLINE(F,ZZAA(1,I),NF,FDEL,FSCA,ZZAA(NF+1,I),ZZAA(NF+2,I),1)
      XP=(F(NF)-F(NF+1))/F(NF+2)
      YP=(ZZAA(NF,I)-ZZAA(NF+1,I))/ZZAA(NF+2,I)
      CALL SYMBOL(XP,YP,.07,3HB =,0.,3)
      CALL NUMBER(999.,999.,  .07,BB(I),0.,2)
 2010 CONTINUE
      CALL GPR(V1,XLL,XL,XRH,W1,W2,K1,K2,V)
      CALL PLOT(12.,0.,-3)
      IF(OPT2) GO TO 402
      IF(OPT3) GO TO 403
      GO TO 400
  402 CONTINUE
      CALL LAXIS(0.,0.,3,9HFREQUENCY,9,6.,-1,2)
      CALL LAXIS(0.,0.,7,16HZ2DD/V0 (G'S/IN),16,7.,-4,1)
      CALL HVGRID(0.,0.,2.,1.,3,7)
      CALL SYMBOL(0.5,8.75,0.14,36H  VERTICAL CAR-CENTER ACCELERATION  ,
     X0.,0,36)
      CALL SYMBOL(0.5,8.55,0.14,36HDUE TO SINUSOIDAL TRACK IRREGULARITY,
     X0.,0,36)
      DO 1010 I=1,N1
      CALL CLINE(F,Z2DDV(1,I),NF,FDEL,FSCA,Z2DDV(NF+1,I),Z2DDV(NF+2,I),1
     1)
      XP=(F(NF)-F(NF+1))/F(NF+2)
```

```
C*******************************
C** FULL (7-29-75)
C*******************************
      YP=(Z2DDV(NF,I)-Z2DDV(NF+1,I))/Z2DDV(NF+2,I)
      CALL SYMBOL(XP,YP,.07,3HB =,0.,3)
      CALL NUMBER(999.,999.,  .07,BB(I),0.,2)
 1010 CONTINUE
      CALL GPR(V1,XLL,XL,XRH,W1,W2,K1,K2,V)
      CALL PLOT(12.,0.,-3)
      CALL LAXIS(0.,0.,3,9HFREQUENCY,9,6.,-1,2)
      CALL LAXIS(0.,0.,7,18HAPHIDD/V0 (G'S/IN),18,7.,-4,1)
      CALL HVGRID(0.,0.,2.,1.,3,7)
      CALL SYMBOL(0.5,8.75,0.14,36H  ANGULAR (PITCHING) ACCELERATION   ,
     X0.0,36)
      CALL SYMBOL(0.5,8.55,0.14,36HDUE TO SINUSOIDAL TRACK IRREGULARITY,
     X0.0,36)
      DO 1020 I=1,N1
      CALL CLINE(F,APDDV(1,I),NF,FDEL,FSCA,APDDV(NF+1,I),APDDV(NF+2,I),1
     1)
      XP=(F(NF)-F(NF+1))/F(NF+2)
      YP=(APDDV(NF,I)-APDDV(NF+1,I))/APDDV(NF+2,I)
      CALL SYMBOL(XP,YP,.07,3HB =,0.,3)
      CALL NUMBER(999.,999.,  .07,BB(I),0.,2)
 1020 CONTINUE
      CALL GPR(V1,XLL,XL,XRH,W1,W2,K1,K2,V)
      CALL PLOT(12.,0.,-3)
      CALL LAXIS(0.,0.,3,9HFREQUENCY,9,6.,-1,2)
      CALL LAXIS(0.,0.,7,25H(Z2DD+APHIDD)/V0 (G'S/IN),25,7.,-4,1)
      CALL HVGRID(0.,0.,2.,1.,3,7)
      CALL SYMBOL(0.5,8.75,0.14,38HVERTICAL CAR ACCELLERATION (OVER TRUCK
     X),0.0,38)
      CALL SYMBOL(0.5,8.55,0.14,36HDUE TO SINUSOIDAL TRACK IRREGULARITY,
     X0.0,36)
      DO 1050 I=1,N1
      CALL CLINE(F,ZADDV(1,I),NF,FDEL,FSCA,ZADDV(NF+1,I),ZADDV(NF+2,I),1
     1)
      XP=(F(NF)-F(NF+1))/F(NF+2)
      YP=(ZADDV(NF,I)-ZADDV(NF+1,I))/ZADDV(NF+2,I)
      CALL SYMBOL(XP,YP,.07,3HB =,0.,3)
      CALL NUMBER(999.,999.,  .07,BB(I),0.,2)
 1050 CONTINUE
      CALL GPR(V1,XLL,XL,XRH,W1,W2,K1,K2,V)
      CALL PLOT(12.,0.,-3)
      IF(OPT3) GO TO 403
      GO TO 400
  403 CONTINUE
      CALL LAXIS(0.,0.,3,9HFREQUENCY,9,6.,-1,2)
      CALL LAXIS(0.,0.,7,13HZ2DD/A1 (G'S),13,7.,-3,1)
      CALL HVGRID(0.,0.,2.,1.,3,7)
      CALL SYMBOL(0.5,9.00,0.14,36H  VERTICAL CAR-CENTER ACCELERATION   ,
```

```
C*******************************
C** FULL (7-29-75)
C*******************************
     X0,0,36)
      CALL SYMBOL(0.5,8.80,0.14,37HDUE TO TRACK IRREGULARITY OF THE FORM
     X,0.0,37)
      CALL SYMBOL(0.5,8.60,0.14,36H       V0 = A1 * WAVE LENGTH         ,
     X0,0,36)
      DO 1030 I=1,N1
      CALL CLINE(F,Z2A(1,I),NF,FDEL,FSCA,Z2A(NF+1,I),Z2A(NF+2,I),1)
      XP=(F(NF)-F(NF+1))/F(NF+2)
      YP=(Z2A(NF,I)-Z2A(NF+1,I))/Z2A(NF+2,I)
      CALL SYMBOL(XP,YP,.07,3HB =,0.,3)
      CALL NUMBER(999.,999.,  .07,BB(I),0.,2)
 1030 CONTINUE
      CALL GPR(V1,XLL,XL,XRH,W1,W2,K1,K2,V)
      CALL PLOT(12.,0.,-3)
      CALL LAXIS(0.,0.,3,9HFREQUENCY,9,6.,-1,2)
      CALL LAXIS(0.,0.,7,15HAPHIDD/A1 (G'S),15,7.,-3,1)
      CALL HVGRID(0.,0.,2.,1.,3,7)
      CALL SYMBOL(0.5,9.00,0.14,36H  ANGULAR (PITCHING) ACCELERATION   ,
     X0,0,36)
      CALL SYMBOL(0.5,8.80,0.14,37HDUE TO TRACK IRREGULARITY OF THE FORM
     X,0.0,37)
      CALL SYMBOL(0.5,8.60,0.14,36H       V0 = A1 * WAVE LENGTH         ,
     X0,0,36)
      DO 1040 I=1,N1
      CALL CLINE(F,APA(1,I),NF,FDEL,FSCA,APA(NF+1,I),APA(NF+2,I),1)
      XP=(F(NF)-F(NF+1))/F(NF+2)
      YP=(APA(NF,I)-APA(NF+1,I))/APA(NF+2,I)
      CALL SYMBOL(XP,YP,.07,3HB =,0.,3)
      CALL NUMBER(999.,999.,  .07,BB(I),0.,2)
 1040 CONTINUE
      CALL GPR(V1,XLL,XL,XRH,W1,W2,K1,K2,V)
      CALL PLOT(12.,0.,-3)
      CALL LAXIS(0.,0.,3,9HFREQUENCY,9,6.,-1,2)
      CALL LAXIS(0.,0.,7,22H(Z2DD+APHIDD)/A1 (G'S),22,7.,-3,1)
      CALL HVGRID(0.,0.,2.,1.,3,7)
      CALL SYMBOL(0.5,9.00,0.14,38HVERTICAL CAR ACCELERATION (OVER TRUCK
     X),0.0,38)
      CALL SYMBOL(0.5,8.80,0.14,37HDUE TO TRACK IRREGULARITY OF THE FORM
     X,0.0,37)
      CALL SYMBOL(0.5,8.60,0.14,36H       V0 = A1 * WAVE LENGTH         ,
     X0,0,36)
      DO 1060 I=1,N1
      CALL CLINE(F,Z2APA(1,I),NF,FDEL,FSCA,Z2APA(NF+1,I),Z2APA(NF+2,I),1
     1)
      XP=(F(NF)-F(NF+1))/F(NF+2)
      YP=(Z2APA(NF,I)-Z2APA(NF+1,I))/Z2APA(NF+2,I)
      CALL SYMBOL(XP,YP,.07,3HB =,0.,3)
```

```
C*******************************
C** FULL (7-29-75)
C*******************************
      CALL NUMBER(999.,999.,  .07,BB(I),0.,2)
 1060 CONTINUE
      CALL GPR(V1,XLL,XL,XRH,W1,W2,K1,K2,V)
  400 CALL PLOT(10.,0.,-3)
      STOP
      END
```

```
      SUBROUTINE GPR(V1,XLL,XL,XRH,W1,W2,K1,K2,V)
      REAL K1,K2
      I=0
      CALL PLFORM(.50,8.00,.07,V1,I,0,1,21H(4HV   =,F12.2,4H MPH),20,0.)
      CALL PLFORM(.50,7.85,.07,V,I,0,1,24H(4HV   =,F12.2,7H FT/SEC),23,
     10.)
      CALL PLFORM(.50,7.70,.07,XLL,I,0,1,20H(4HLL =,F12.2,3H FT),19,0.)
      CALL PLFORM(.50,7.55,.07,XL,I,0,1,20H(4H2A =,F12.2,3H FT),19,0.)
      CALL PLFORM(.50,7.40,.07,XRH,I,0,1,20H(4HRHO=,F12.2,3H FT),19,0.)
      CALL PLFORM(3.5,8.00,.07,W1,I,0,1,20H(4HW1 =,F12.2,3H LB),19,0.)
      CALL PLFORM(3.5,7.85,.07,W2,I,0,1,20H(4HW2 =,F12.2,3H LB),19,0.)
      CALL PLFORM(3.5,7.70,.07,K1,I,0,1,25H(4HK1 =,F12.2,8H LB/INCH),24
     1,0.)
      CALL PLFORM(3.5,7.55,.07,K2,I,0,1,25H(4HK2 =,F12.2,8H LB/INCH),24
     1,0.)
      RETURN
      END
```

```
      SUBROUTINE PLFORM(X,Y,HT,ARRAY,IARRAY,IORF,N,WFORM,NCHAR,ANG)
```

SUBROUTINE PLFORM
(REAL/INTEGER VERSION)
****

THIS SUBROUTINE ALLOWS THE USER TO PLOT A MIXED STRING OF REAL/INTEGER NUMBERS AND HOLLERITH TEXT WITH JUST ONE CALL PER LINE. THE MODE, REAL OR INTEGER, CANNOT BE MIXED DURING ANY ONE CALL. ORDINARILY, IF ONE WISHES TO PLOT SEVERAL NUMBERS ON A LINE, WITH HOLLERITH IDENTIFIERS, HE HAS TO MAKE AT LEAST ONE CALL TO SYMBOL AND AS MANY CALLS TO NUMBER AS HE HAS NUMBERS TO PLOT. THIS ROUTINE ELIMINATES ALL THOSE CALLS.

THE CALLING SEQUENCE IS

```
CALL PLFORM(X,Y,HEIGHT,ARRAY,IARRAY,IORF,N,FORMAT,NCHAR,ANGLE)
```

X, Y — ARE THE COORDINATES, IN INCHES, OF THE LOWER LEFT HAND CORNER (BEFORE ROTATION) OF THE FIRST CHARACTER TO BE PRODUCED.

HEIGHT — IS THE HEIGHT, IN INCHES, OF THE CHARACTERS TO BE PLOTTED.

```
C*****************************************
C** FULL (7-29-75)
C*****************************************
C
C   ARRAY    IS THE ARRAY CONTAINING THE REAL NUMBERS TO BE PLOTTED,
C            SINCE THE MODE OF THE NUMBERS BEING PRINTED CANNOT BE
C            MIXED IN ANY ONE CALL, THIS IS A DUMMY ARGUMENT IF
C            INTEGERS ARE BEING PRINTED ON THE PLOT.
C
C   IARRAY   IS THE ARRAY CONTAINING THE INTEGER NUMBERS TO BE PRINTED
C            ON THE PLOT.  THIS IS DUMMY IF REALS ARE BEING PRINTED ON
C            THE PLOT.
C
C   IORF     IS 1 IF INTEGERS ARE BEING PRINTED ON THE PLOT -- IT IS
C            NOT 1 IF REALS ARE BEING PRINTED.
C
C   N        IS THE NUMBER OF VALUES DESIRED TO BE PLOTTED FROM ARRAY.
C
C   FORMAT   IS THE FORMAT OF THE LINE TO BE PLOTTED.  THE ARGUMENT IS
C            IN AN H-TYPE FORMAT.  FOR EXAMPLE, IF ARRAY CONTAINS 1.0
C            AND 2.0 AND THESE ARE TO BE PLOTTED ON A LINE IN THE FORM
C                     X=1.0  Y=2.0
C            FORMAT WOULD BE
C                     22H(2HX=F3.1,2X,2HY=F3.1)
C            WHERE THE FORMAT IS ENCLOSED IN PARENTHESES, PRECEDED BY
C            AN H, PRECEDED BY THE NUMBER OF CHARACTERS BETWEEN AND
             INCLUDING THE PARENTHESES (IN THIS CASE 22).

C   NCHAR    IS THE NUMBER OF CHARACTERS, INCLUDING BLANKS, TO BE
C            PRINTED ON THE PLOT LINE.  THE NUMBER MUST NOT EXCEED
C            130.  THE EXACT NUMBER OF CHARACTERS IS DETERMINED BY THE
C            USER BY COUNTING THE NUMBER OF HOLLERITH CHARACTERS
C            (INCLUDING BLANKS SUCH AS 2X) AND SPACES SPECIFIED FOR
C            NUMBERS (SUCH AS 3 SPACES FOR F3.1) IN FORMAT.  IN THE
C            ABOVE FORMAT EXAMPLE, NCHAR=12.
C
C   ANGLE    THE ANGLE, IN DEGREES AT WHICH THE LINE IS TO BE PLOTTED.
C
C
C   DIMENSION ARRAY(1),IARRAY(1),WFORM(26),RFORM(26),RITE(26),RDFORM(1
C    1)
      REAL*8 RFORM,RDFORM
      DATA RFORM/6H(  A5),6H( 2A5),6H( 3A5),6H( 4A5),6H( 5A5),6H( 6A5),
     *           6H( 7A5),6H( 8A5),6H( 9A5),6H(10A5),6H(11A5),6H(12A5),
     *           6H(13A5),6H(14A5),6H(15A5),6H(16A5),6H(17A5),6H(18A5),
     *           6H(19A5),6H(20A5),6H(21A5),6H(22A5),6H(23A5),6H(24A5),
     *           6H(25A5),6H(26A5)/
      IF(IORF .EQ. 1) WRITE( 1,WFORM) (IARRAY(I),I=1,N)
      IF(IORF .NE. 1) WRITE( 1,WFORM) (ARRAY(I),I=1,N)
      NWORDS=NCHAR/5
      IF(MOD(NCHAR,5) .GT. 0) NWORDS=NWORDS+1
      RDFORM(1)=RFORM(NWORDS)
      REWIND 1
      READ( 1,RDFORM) (RITE(I),I=1,NWORDS)
        REWIND 1
      CALL SYMBOL(X,Y,HT,RITE,ANG,NCHAR)
      RETURN
      END
```

```
C*******************************
C** FULL (7-29-75)
C*******************************
      SUBROUTINE HVGRID(X,Y,XS,YS,M,N)
      YØ=YS
      XF=X+XS*FLOAT(M)
      XØ=X
      CALL PLOT(XØ,YØ,3)
      DO 5 I=1,N
      CALL PLOT(XØ,YØ,3)
      CALL PLOT(XF,YØ,2)
      YØ=YØ+YS
      XT=XF
      XF=XØ
    5 XØ=XT
      XØ=XS
      YØ=Y
      XF=Y+YS*FLOAT(N)
      CALL PLOT(XØ,YØ,3)
      DO 10 I=1,M
      CALL PLOT(XØ,YØ,3)
      CALL PLOT(XØ,XF,2)
      XØ=XØ+XS
      XT=XF
      XF=YØ
   10 YØ=XT
      RETURN
      END
```

```
C******************************
C** FULL (7-29-75)
C******************************
      SUBROUTINE FTABG(IFREQ,NDF,DF,FL,F,NF)
C
C     THIS SUBROUTINE COMPUTES THE FREQUENCY TABLE
C
C     IFREQ=1 RANGE INPUTS, =2 OCTAVE INPUTS
C     NDF  - NUMBER OF RANGES OR OCTAVES
C     DF   - FREQUENCY POINTS VECTOR
C     FL   - FREQUENCY LOWER LIMITS VECTOR
C     F    - FREQUENCY TABLE RETURNED
C     NF   - NUMBER OF POINTS IN FREQUENCY TABLE
C
      INTEGER DF(7)
      DIMENSION FL(8),F(200)
      IF(IFREQ .EQ. 2) GO TO 500
C
C     RANGE INPUTS
C
      NF=0
      DO 200 I=1,NDF
      C=EXP(ALOG(FL(I+1)/FL(I))/DF(I))
      NF=NF+1
      F(NF)=FL(I)
      DO 100 J=1,DF(I)
      FT=C*F(NF)
      IF(FT .GE. FL(I+1)-0.000001) GO TO 200
      NF=NF+1
      F(NF)=FT
100   CONTINUE
200   CONTINUE
      RETURN
C
C     OCTAVE INPUTS
C
500   C=2**(1./DF(1))
      NF=1
      F(NF)=FL(1)
      DO 700 I=1,NDF
      DO 700 J=1,DF(1)
      NF=NF+1
      F(NF)=F(NF-1)*C
700   CONTINUE
      RETURN
      END
```

APPENDIX B
PROGRAM FLEX INPUT PARAMETERS AND LISTINGS

The Program FLEX input data deck is arranged in 21-card groups. Most of the groups do not have a fixed number of cards - the actual number will depend upon the user's choices. In some cases a group will have no cards at all for a particular run. The following is a card-by-card description of the input deck:

Card Group 1 UNITS: NONE

No. of Cards = 1 Required

I2	I2	I2	I2
PRINT	DISP	ACC	SPEC

These variables determine which of the possible outputs are produced. The displacement responses are always calculated and printed. All other output is controlled by these codes:

PRINT = { 0, Intermediate results not printed.
1, Intermediate results printed.

DISP = { 0, Displacement responses not plotted.
1, Displacement responses plotted.

ACC = { 0, Acceleration responses not calcualted.
1, Acceleration responses calculated, printed, and plotted.

SPEC = { 0, Acceleration spectra not calculated.
1, Acceleration spectra calculated, printed and plotted.

Card Group 2 UNITS: FEET

No. of Cards = 1 Required

F10.5	F10.5	F10.5	F10.5	F10.5	F10.5
b	$\overline{x}$	L	d	ℓ	a

The following designators describe the physical dimensions of the vehicle. Note that 2*(a+d)=L.

b Position of hanging mass.

$\bar{x}$ Position of car center of gravity.

L Length of car.

d Inset of truck center from each end.

ℓ Truck wheel base.

a One-half of vehicle wheel base.

Card Group 3 UNITS: LBS

No. of Cards = 1 Required

F10.5	F10.5	F10.5
W1	W2	WT

W1, W2, and WT are the weight of the trucks, the weight of the car body, and the weight of the hanging mass, respectively.

Card Group 4 UNITS: NONE

No. of Cards = 1 Required

F10.5	F10.5
β_2	β_T

β_2 is the damping ratio associated with the truck suspension. It can be calculated from the damping constant C_2:

$$\beta_2 = \frac{C_2}{2*M_2}\sqrt{\frac{M_2}{K_2}}$$

β_T is the damping ratio associated with the hanging mass suspension. It can be calculated from the damping constant C_T:

$$\beta_T = \frac{C_T}{2*M_T}\sqrt{\frac{M_T}{K_T}}$$

Card Group 5 UNITS: LBS/IN.

No. of Cards = 1 Required

F10.5	F10.5	F10.5
K1	K2	KT

K1, K2, and KT are the spring constants of the wheel suspension, truck suspension, and hanging mass suspension respectively.

<u>Card Group 6</u> UNITS: NONE

No. of Cards = 3 Required

(1st Card) |I2 | I2| IREQ NDF

(2nd Card) |7I4| DF(N), N=1, NDF FOR IFREQ=1 UNITS: NONE

DF(2) FOR IFREQ=2

(3rd Card) |8F10.4| FL(N), N=1,NDF+1 FOR IFREQ=1 UNITS: HERTZ

FL(1) FOR IFREQ=2

IFREQ selects one of two methods for specifying the frequency points at which the response will be found. With IFREQ = 1, logarithmically evenly-spaced points are generated for a set of frequency ranges. NDF gives the number of frequency ranges (maximum of seven). DF(N) specifies the number of points for the range FL(N) to FL(N+1). FL(N) gives the lower limit of the Nth frequency range, and FL(N+1) is the upper limit for the Nth range. FL(NDF+1) gives the upper limit for the last frequency range.

With IFREQ=2, logarithmically-evenly spaced points are generated for a set of octaves. NDF specifies the number of octaves. DF(1) gives the number of points per octave, and FL(1) gives the starting frequency of the first octave. Note that the remainders of the DF and FL fields are left blank.

<u>Card Group 7</u> UNITS: FEET/SEC^2

No. of Cards = 1 Required

|F10.5| g

g is the gravity constant, usually 32.167 ft./sec^2.

Card Group 8 UNITS: NONE

No. of Cards = 1 Required

I1
ECCEN

ECCEN is the code for wheel eccentricity. Set ECCEN=0 to assume that the wheels are round. In this case, omit groups 9, 10, and 11. Set ECCEN = 1 to enter information about eccentric wheels in groups 9, 10, and 11.

Card Group 9 UNITS: FEET

No. of Cards $= \begin{cases} 0, & \text{if ECCEN} = 0 \\ 1, & \text{if ECCEN} = 1 \end{cases}$

F10.5
R

Card Group 10 UNITS: NONE

No. of Cards $= \begin{cases} 0, & \text{if ECCEN} = 0 \\ 1, & \text{if ECCEN} = 1 \end{cases}$

F10.5	F10.5	F10.5	F10.5
EPS1	EPS2	EPS3	EPS4

Card Group 11 UNITS: DEGREES

No. of Cards $= \begin{cases} 0, & \text{if ECCEN} = 0 \\ 1, & \text{if ECCEN} = 1 \end{cases}$

F10.5	F10.5	F10.5	F10.5
THETA1	THETA2	THETA3	THETA4

R, EPS1, EPS2, EPS3, EPS4, THETA1, THETA2, THETA3, and THETA4 describe the eccentricity of the wheels. These are needed only if ECCEN = 1. R is the wheel radius. EPS1, EPS2, EPS3, and EPS4 give the eccentricity for each wheelset as a ratio of the eccentricity to the amplitude of the sinusoidal input. THETA1, THETA2, THETA3, and THETA4 give the wheel eccentricity phase angle for each wheelset relative to the sinusoidal input.

Card Group 12

No. of Cards = 1 Required

UNITS	
VLTEST	NONE
V	MPH
λ	INCHES/CYCLE

I1	F9.2
VLTEST	VORLAM

$$\text{VORLAM} = \begin{cases} V, & \text{if VLTEST} = 1 \\ \lambda, & \text{if VLTEST} = 2 \end{cases}$$

VLTEST and VORLAM specify either the velocity or wavelength. The user has the option of fixing either the rail vehicle velocity or the wavelength of the track irregularity. Since V=f*λ, for a fixed velocity, the wavelength must be adjusted, in order to sweep frequency. For a fixed wavelength, the velocity must be adjusted. Set VLTEST = 1 and VORLAM = V to specify a constant vehicle velocity. Set VLTEST = 2 and VORLAM = λ to specify a constant track wavelength.

Card Group 13 UNITS: NONE

No. of Cards = 1 Required

I1
INCODE

INCODE determines the manner in which car body flexibility properties will be input. There are four options. INCODE = 1 when the user wants to assume uniform beam characteristics for the car. In this case, only the bending frequency of the car needs to be supplied; the program provides the uniform beam properties. In the other three cases, the weight distribution and car bending mode shape are given by tabular values. Taking the direction of motion as positive, these input values are specified as functions of distance from the rear of the car. The tabular values must be specified at equal intervals along the length of the car. With INCODE = 2, the car bending frequency is still given. If INCODE = 3, the elasticity and moment of inertia are used instead of the bending frequency. With INCODE = 4, the elasticity and a tabular distribution function for the moment of inertia are used in the coefficient calculations. INCODE determines which of the remaining card groups are needed according to this scheme:

If INCODE = 1, use group 14.
If INCODE = 2, use groups 14, 15, 17, and 18.
If INCODE = 3, use groups 15, 16, 17, 18, and 19.
If INCODE = 4, use groups 15, 16, 17, 18, and 20.

Card Group 14 UNITS: HERTZ

No. of Cards = $\begin{cases} 0, & \text{if INCODE = 3,4} \\ 1, & \text{if INCODE = 1,2} \end{cases}$

|F10.5|
FB

FB is the fundamental bending frequency of the car. It is input if INCODE = 1 or 2.

Card Group 15 UNITS: INTDIM NONE, INTDEL FEET

No. of Cards = $\begin{cases} 0, & \text{if INCODE = 1} \\ 1, & \text{otherwise} \end{cases}$

| I10 | F10.2 |
INTDIM INTDEL

INTDIM and INTDEL describe the tabular functions and are needed with INCODE = 2, 3, or 4. The independent variable for the tabular functions is x, ranging from 0 to L, the length of the car body from rear to front. INTDIM is the number of points in the tabular functions (no more than 25). INTDEL is the increment of x between each point of the function. For 20 evenly spaced points with one at each end of the car, there would be 19 intervals, and INTDEL would equal L/19.

Card Group 16 UNITS: LBS/IN^2

No. of Cards = $\begin{cases} 0, & \text{if INCODE = 1,2} \\ 1, & \text{if INCODE = 3,4} \end{cases}$

|F10.5|
E

E is the modulus of elasticity of the car body material and is given only when INCODE = 3 or 4.

Card Group 17 UNITS: LBS/FOOT

No. of Cards = $\begin{cases} 0, & \text{if INCODE = 1} \\ \text{INTDIM/8}, & \text{otherwise} \end{cases}$

|F10.5|F10.5|F10.5|F10.5|F10.5|F10.5|F10.5|F10.5|
M (I), I=1, INTDIM

M (I) is the linear weight distribution of the car body (LBS/FT) and is needed when INCODE = 2, 3, or 4. It is entered 8 values to a card for as many cards as necessary.

Card Group 18 UNITS: NONE

No. of Cards = $\begin{cases} 0, \text{ if INCODE} = 1 \\ \text{INTDIM/8, otherwise} \end{cases}$

|F10.5|F10.5|F10.5|F10.5|F10.5|F10.5|F10.5|F10.5|
W(I), I=1, INTDIM

W(I) is the tabular function which describes the bending mode shape of the car body. It is given when INCODE = 2, 3, or 4 and should be entered 8 values to a card, for as many cards as necessary.

Card Group 19 UNITS: INCHES^4

No. of Cards = $\begin{cases} 0, \text{ if INCODE} = 1,2,4 \\ 1, \text{ if INCODE} = 3 \end{cases}$

|F10.5 |
ICONST

ICONST is the centroidal cross-section moment of inertia of the car. It is needed when INCODE = 3.

Card Group 20 UNITS: INCHES^4

No. of Cards = $\begin{cases} 0, \text{ if INCODE} = 1,2,3 \\ \text{INTDIM/8, if INCODE} = 4 \end{cases}$

|F10.5|F10.5|F10.5|F10.5|F10.5|F10.5|F10.5|F10.5|
I(J), J=1, INTDIM

I(J) is the tabular distribution function for the centroidal cross-section moment of inertia of the car. This should be given only when INCODE = 4, and entered 8 values to a card for as many cards as necessary.

The following pages contain the listing for Program FLEX.

```
C********************************
C** FLEX (7-29-75)
C********************************
C** FOLLOWING TEXT PRINTED FROM FILE DSKB:FLEXPS.F4 [4351,533] 29-JUL-75
C********************************
C
C      ****** MAIN PROGRAM FLEX ******
C
C
C      THIS PROGRAM COMPUTES THE VERTICAL RESPONSE
C      CHARACTERISTICS OF A FLEXIBLE TRAIN CAR SUBJECTED
C      TO A SINUSOIDAL TRACK IRREGULARITY.
C
C      SUBROUTINES REQUIRED:  PLOTIT,DIFF2,INTERW,QSF,
C          MINV,GMPRD,INIPLT,FINPLT,LAXIS,LOGSCA,ALINE,
C          GRID,SYMBOL,NUMBER,PLOT
C
C      INPUT: FOR01.DAT
C
C      OUTPUT:  LINEPRINTER(UNIT 3),PLOT TAPE(UNIT 16)
C
      INTEGER NVAR,IN,OUT,INTDIM,ECCEN,MATSIZ,NVAL,
     XFCOUNT,PLTNUM,PLTEM,FCODE,J,INCODE,FCYCLE,FKVAL
      INTEGER PRINT,DISP,ACC,SPEC,VLTEST,FDEL
      REAL B,XBAR,LL,D,L,A,M1,M2,MT
      REAL C2,CT,K1,K2,KT,LAMBDA,E,MASS,W,I,WDX2
      REAL M,C,K,M11,I2,K11,F,OMEGA,FB,FM
      REAL FL,M11FCT,I2FCT,K11FCT
      REAL WD,WLLD,WB,VORLAM,VV
      REAL TWOPI,CONST1,CONST2,CONST3,CONST4,CONST5
      COMPLEX CE,CZ2,CPHI,CY1,CY2,CZ1
      REAL ALPHA,BETA,INTDEL,UW,ICONST,X,INTERW
      REAL W1,W2,WT,BETA2,BETAT,G,PLTF,CF
      REAL PLTCG,PLTTC,WXBAR,LLDXBA,PLTMZ1
      COMPLEX Q,QVAR,MAT,WORK,DETER,WORKN,ERR
      COMPLEX WORK2(36),IDEN(36)
      REAL ACCG,ACTC,ACZ1,SPCG,SPTC,SPZ1
      REAL R,EPS1,EPS2,EPS3,EPS4,THETA1,THETA2,THETA3,THETA4,FW
      REAL WORKL,WORKM,XAXIS,YAXIS
      DIMENSION M(36),C(36),K(36),Q(6),QVAR(6),ERR(6,100),MAT(36)
      DIMENSION FL(8),FDEL(7),WORK(36),
     XWORKL(6),WORKM(6),WORKN(6)
      DIMENSION M11FCT(25),I2FCT(25),K11FCT(25),MASS(25),
     XW(25),I(25),WDX2(25)
      DIMENSION CF(200),CE(200),CZ2(200),CPHI(200),
     XCY1(200),CY2(200),CZ1(200),PLTNUM(3)
      DIMENSION PLTCG(200),PLTTC(200),PLTMZ1(200),PLTF(200)
      DIMENSION ACCG(200),ACTC(200),ACZ1(200),
     XSPCG(200),SPTC(200),SPZ1(200)
      DIMENSION SAVEM(200,6),IITYPE(6)
```

```
C*******************************
C** FLEX (7-29-75)
C*******************************
      EQUIVALENCE (SAVEM(1,1),PLTCG(1)),(SAVEM(1,2),PLTTC(1)),(SAVEM
     X(1,3),PLTMZ1(1)),(SAVEM(1,4),ACCG(1)),(SAVEM(1,5),ACTC(1)),
     X(SAVEM(1,6),ACZ1(1))
      COMMON/SUB/INTDEL,W
      COMMON /COPLOT/FCYCLE,XAXIS,FKVAL,NVAL,YAXIS
     X,VLTEST,VV,W1,W2,WT,K1,K2,KT,FB,BETA2,BETAT,INCODE
     X,ECCEN,FW
      DATA IITYPE/1,1,1,2,2,2/
      DATA IDEN/(1.,0.),6*(0.,0.),(1.,0.),6*(0.,0.),(1.,0.),
     X6*(0.,0.),(1.,0.),6*(0.,0.),(1.,0.),6*(0.,0.),(1.,0.)/
C
C     DEFINE UW TO BE THE UNIFORM BEAM BENDING MODE FUNCTION
C
      UW(X)=COSH(BETA*X)+COS(BETA*X)-ALPHA*(SINH(BETA*X)
     X+SIN(BETA*X))
C
C     SETUP CONSTANTS. INPUT ON FOR01.DAT. OUTPUT ON LINEPRINTER
C     UNIT 3.  XAXIS AND YAXIS DETERMINE THE SIZE OF THE PLOTS.
C
      MATSIZ=36
      NVAR=6
      IN=1
      OUT=3
      XAXIS=5.5
      YAXIS=8.0
C
C     READ FIXED INPUT DATA
C
      READ(IN,105) PRINT,DISP,ACC,SPEC
105   FORMAT(4I2)
      READ(IN,110) B,XBAR,LL,D,L,A
110   FORMAT(8E10.5)
      READ(IN,110) W1,W2,WT
      READ(IN,110) BETA2,BETAT
      READ(IN,110) K1,K2,KT
      READ(IN,115) IFREQ,FCODE
115   FORMAT(2I2)
      READ(IN,120) FDEL
120   FORMAT(7I4)
      READ(IN,122) FL
122   FORMAT(8F10.4)
      READ(IN,110) G
      READ(IN,115) ECCEN
C
C     GENERATE FREQUENCY TABLE
C
      CALL FTABG(IFREQ,FCODE,FDEL,FL,CF,NFREQ)
```

```
C*******************************
C** FLEX (7-29-75)
C*******************************
      IF(ECCEN .NE. 1) GO TO 125
C
C     READ WHEEL ECCENTRICITY PARAMETERS IF ECCEN = 1
C
      READ(IN,110) R
      READ(IN,110) EPS1,EPS2,EPS3,EPS4
      READ(IN,110) THETA1,THETA2,THETA3,THETA4
125   CONTINUE
C
C     READ VELOCITY OR WAVELENGTH INPUT
C
      READ(IN,130) VLTEST,VORLAM
130   FORMAT(I1,F9.2)
      READ(IN,133) INCODE
133   FORMAT(I1)
C
C     READ CAR BODY PROPERTIES ACCORDING TO INCODE VALUE
C
      IF(INCODE .LT.3) READ(IN,110) FB
      IF(INCODE .EQ.1) GO TO 145
      READ(IN,140) INTDIM,INTDEL
140   FORMAT(I10,F10.2)
      IF(INCODE .GT. 2) READ(IN,110) E
      READ(IN,110) (MASS(J),J=1,INTDIM)
      READ(IN,110) (W(J),J=1,INTDIM)
      IF(INCODE .EQ. 3) READ(IN,110) ICONST
      IF(INCODE .EQ. 4) READ(IN,110) (I(J),J=1,INTDIM)
145   CONTINUE
C
C     CALCULATE FM, THE FREQUENCY OF THE HANGING MASS, SUBSTITUTE
C     THIS LINE FOR THE FOLLOWING ONE TO REMOVE UNIT ASSUMPTIONS,
C     FM=(SQRT(KT*G/WT))/6.28318
C
        FM=(SQRT(KT*G*12./WT))/6.28318
      IF(ECCEN .EQ. 0) GO TO 149
C
C     CALCULATE FW, THE FREQUENCY OF THE ECCENTRICITY
C     CONTRIBUTION. SUBSTITUTE THIS LINE FOR THE
C     FOLLOWING ONE TO REMOVE UNIT ASSUMPTIONS,
C     FW=VORLAM/(6.28318*R)
C
      FW=VORLAM*17.6/(6.28318*R*12.)
C
C     FIND THE POINT IN THE FREQUENCY RANGE CLOSEST TO FW
C
      FXX=1.0E10
      FAB=FW
```

```
C*******************************
C** FLEX (7-29-75)
C*******************************
      DO 148 FCOUNT=1,NFREQ
      F=CF(FCOUNT)
      IF(ABS(F-FAB) .GT. FXX) GO TO 149
      FW=F
148   FXX=ABS(F-FAB)
C
C     WRITE ALL INPUT VALUES, FM, AND FW
C
149   WRITE(OUT,150) B,XBAR,LL,D,L,A
150   FORMAT(1H1/20X,'INPUT VALUES'///10X,'B:  POSITON OF HANGING
     X MASS =',F10.2,' FT'//10X,'XBAR:  POSITION OF CENTER OF GRAVITY
     X OF CAR =',F10.2,' FT'//10X,'LL:  LENGTH OF CAR =',F10.2,' FT'//
     X10X,'D:  INSET OF TRUCK CENTER =',F10.2,' FT'//
     X10X,'L:  TRUCK WHEEL BASE =',F10.2,' FT'//
     X10X,'A:  VEHICLE WHEEL BASE =',F10.2,' FT'/)
      WRITE(OUT,160) W1,W2,WT
160   FORMAT(10X,'W1:  WEIGHT OF TRUCK =',F10.1,' LBS'//
     X10X,'W2:  WEIGHT OF CAR =',F10.1,' LBS'/
     X/10X,'WT:  WEIGHT OF HANGING MASS =',F10.1,' LBS'/)
      WRITE(OUT,170) BETA2,BETAT
170   FORMAT(10X,'BETA2:  SECONDARY SUSPENSION DAMPING RATIO =',
     XF10.3,' DIMENSIONLESS'//
     X10X,'BETAT:  HANGING MASS DAMPING RATIO =',F10.3,
     X' DIMENSIONLESS'/)
      WRITE(OUT,180) K1,K2,KT
180   FORMAT(10X,'K1:  PRIMARY SPRING CONSTANT =',F10.1,' LBS/IN'//
     X10X,'K2:  SECONDARY SPRING CONSTANT =',F10.1,' LBS/IN'//10X,
     X'KT:  HANGING MASS SPRING CONSTANT =',F10.1,' LBS/IN')
      WRITE(OUT,185) FM
185   FORMAT(/10X,'FM:  FREQUENCY OF HANGING MASS =',F11.4,' HZ'/)
      WRITE(OUT,190) FCODE,IFREQ
190   FORMAT(13X,I2,' FREQUENCY RANGES:    IFREQ = ',I2/)
      WRITE(OUT,200) FDEL
200   FORMAT(10X,'FDEL = ',7I4/)
      WRITE(OUT,202) FL
202   FORMAT(10X,'FL = ',8F10.4/)
      WRITE(OUT,205) G
205   FORMAT(/10X,'G:  GRAVITY CONSTANT =',F10.2,' FT/SEC**2'/)
      IF(ECCEN .NE. 1) GO TO 218
      WRITE(OUT,210) R,FW
210   FORMAT(10X,'R:  WHEEL RADIUS =',F10.2,' FT'//10X
     X,'FW:  FREQUENCY OF WHEEL ECCENTRICITY =',F10.4,' HZ'/)
      WRITE(OUT,212) EPS1,EPS2,EPS3,EPS4
212   FORMAT(10X,'EPS1:  WHEEL ECCENTRICITY 1 =',F10.3
     X,' DIMENSIONLESS'//10X,'EPS2:  WHEEL ECCENTRICITY 2 =',F10.3
     X,' DIMENSIONLESS'//10X,'EPS3:  WHEEL ECCENTRICITY 3 =',
     XF10.3,' DIMENSIONLESS'//10X,'EPS4:  WHEEL ECCENTRICITY 4 ='
```

```
C********************************
C** FLEX (7-29-75)
C********************************
     X,F10.3,' DIMENSIONLESS'/)
      WRITE(OUT,215) THETA1,THETA2,THETA3,THETA4
215   FORMAT(10X,'THETA1:  ECCENTRICITY PHASE ANGLE 1 =',
     XF10.3,' DEGREES'//
     X10X,'THETA2:  ECCENTRICITY PHASE ANGLE 2 =',F10.3,' DEGREES'//
     X10X,'THETA3:  ECCENTRICITY PHASE ANGLE 3 =',F10.3,' DEGREES'//
     X10X,'THETA4:  ECCENTRICITY PHASE ANGLE 4 =',F10.3,' DEGREES'/)
218   CONTINUE
      IF(VLTEST .EQ. 1) WRITE(OUT,219) VORLAM
219   FORMAT(10X,'VEL:  VELOCITY =',F11.2,' MPH'/)
      IF(VLTEST .EQ. 2) WRITE(OUT,220) VORLAM
220   FORMAT(10X,'LAMBDA:  WAVELENGTH OF TRACK IRREGULARITY =',
     XF11.2,' IN'/)
      IF(DISP .EQ. 1) WRITE(OUT,222)
222   FORMAT(10X,'DISPLACEMENT RESPONSES WILL BE PLOTTED'/)
      IF(ACC .EQ. 1) WRITE(OUT,223)
223   FORMAT(10X,'ACCELERATION RESPONSES WILL BE
     XCALCULATED AND PLOTTED'/)
      IF(SPEC .EQ. 1) WRITE(OUT,224)
224   FORMAT(10X,'ACCELERATION SPECTRA WILL BE CALCULATED
     X AND PLOTTED'/)
      IF(INCODE .NE. 1) GO TO 230
      WRITE(OUT,225)
225   FORMAT(/10X,'A UNIFORM BEAM WILL BE USED'//)
      GO TO 252
230   WRITE(OUT,233)
233   FORMAT(/10X,'WEIGHT AND BENDING MODE FUNCTIONS:'/
     X/12X,'CAR SEGMENT NO.',5X,'X  FT',10X,'M(X)  LBS/FT',
     X6X,'W(X) DIMENSIONLESS'/)
      X=0.0
      DO 240 J=1,INTDIM
      WRITE(OUT,235) J,X,MASS(J),W(J)
235   FORMAT(10X,I10,3F20.4/)
      X=X+INTDEL
240   CONTINUE
      IF(INCODE .EQ. 2) GO TO 252
      WRITE(OUT,243) E
243   FORMAT(/10X,'E:  MODULUS OF ELASTICITY OF CAR MATERIAL ='
     X,E11.5,' LBS/IN**2'/)
      IF(INCODE .EQ. 4) GO TO 246
      WRITE(OUT,245) ICONST
245   FORMAT(/10X,'ICONST:  CENTROIDAL CROSS-SECTIONAL MOMENT OF
     X INERTIA =',E11.5,' IN**4'/)
      GO TO 256
246   WRITE(OUT,247)
247   FORMAT(/10X,'CENTROIDAL CROSS-SECTIONAL MOMENT OF'/10X,
     X'INERTIA DISTRIBUTION FUNCTION:'//12X,'CAR SEGMENT NO.',5X,
```

```
C********************************
C** FLEX (7-29-75)
C********************************
     X'X  FT',10X,'I(X)  IN**4'/)
      X=0.0
      DO 250 J=1,INTDIM
      WRITE(OUT,235) J,X,I(J)
      X=X+INTDEL
250   CONTINUE
      GO TO 256
252   WRITE(OUT,254) FB
254   FORMAT(/10X,'FB:  FUNDAMENTAL BENDING FREQUENCY OF CAR =',
     XF11.4,' HZ')
256   CONTINUE
C
C     VV IS USED IN THE PLOT ROUTINE
C
      VV=VORLAM
C
C-------------------------------------------------------------
C------- REMOVING THE STATEMENTS BETWEEN THESE LINES -----
C------- (IN ADDITION TO THE TWO CHANGES NOTED ABOVE) -
C------- ALLOWS THE PROGRAM TO ACCEPT ANY CONSISTENT -----
C------- SET OF INPUT UNITS USING SEC.  HOWEVER -----
C------- THE PRINTED LABELS WILL NOT BE CORRECT-----------
C
C     CONVERT INPUT DATA TO IN-LBS-SEC UNITS
C
      B=B*12.
      XBAR=XBAR*12.
      LL=LL*12.
      D=D*12.
      L=L*12.
      A=A*12.
      G=G*12.
      IF(VLTEST .EQ. 1) VORLAM=VORLAM*17.6
      IF(ECCEN .EQ. 1) R=R*12.
      IF(INCODE .EQ. 1) GO TO 257
      INTDEL=INTDEL*12.
      DO 255 J=1,INTDIM
255   MASS(J)=MASS(J)/12.0
257   CONTINUE
C-------------------------------------------------------------
C
C     COMPUTE MASSES FROM WEIGHTS
C
      M1=W1/G
      M2=W2/G
      MT=WT/G
C
```

```
C********************************
C** FLEX (7-29-75)
C********************************
C       CALCULATE DAMPING CONSTANTS
C
        C2=2.*M2*SQRT(K2/M2)*BETA2
        CT=2.*MT*SQRT(KT/MT)*BETAT
        ALPHA=0.7825
        BETA=4.730/LL
        IF(INCODE .LT. 3) OMEGA=6.28318*FB
        IF(INCODE .EQ. 1) GO TO 259
C
C       CONVERT M(X) FROM WEIGHT TO MASS
C
        DO 258 J=1,INTDIM
258     MASS(J)=MASS(J)/G
259     IF(ECCEN .EQ. 0) GO TO 261
C
C       CONVERT ECCENTRICITY PHASE ANGLES TO RADIANS
C
        THETA1=0.0174533*THETA1
        THETA2=THETA2*0.0174533
        THETA3=0.0174533*THETA3
        THETA4=0.0174533*THETA4
261     IF(INCODE .NE. 1) GO TO 260
C
C       COMPUTE VALUES FOR M11, K11, AND I2 (ELEMENTS OF THE
C       MASS MATRIX) UNDER UNIFORM BEAM ASSUMPTIONS
C
        M11=M2
        K11=(OMEGA**2)*M11
        I2=M2*(LL**2)/12.0
        GO TO 299
C
C       FOR INCODE= 3, 4 SUBROUTINE DIFF2 COMPUTES THE
C       SECOND DERIVATIVE OF W(X)
C
260     IF(INCODE .NE. 2) CALL DIFF2(W,WDX2,INTDIM,INTDEL)
        X=0.0
C
C       M11, K11, AND I2 ARE CALCULATED USING INTEGRATION
C       SUBROUTINE QSF ACCORDING TO THE VALUE OF INCODE
C
        DO 270 J=1,INTDIM
        M11FCT(J)=MASS(J)*(W(J)**2)
        I2FCT(J)=MASS(J)*((X-XBAR)**2)
        IF(INCODE .EQ. 4) K11FCT(J)=I(J)*(WDX2(J)**2)
        IF(INCODE .EQ. 3) K11FCT(J)=WDX2(J)**2
        X=X+INTDEL
270     CONTINUE
```

```
C********************************
C** FLEX (7-29-75)
C********************************
      CALL QSF(INTDEL,M11FCT,M11FCT,INTDIM)
      CALL QSF(INTDEL,I2FCT,I2FCT,INTDIM)
      IF(INCODE .NE. 2) CALL QSF(INTDEL,K11FCT,K11FCT,INTDIM)
      M11=M11FCT(INTDIM)
      I2=I2FCT(INTDIM)
      IF(INCODE .EQ. 4) K11=K11FCT(INTDIM)*E
      IF(INCODE .EQ. 2) K11=(OMEGA**2)*M11
      IF(INCODE .EQ. 3) K11=K11FCT(INTDIM)*E*ICONST
299   CONTINUE
C
C     ZERO THE COEFFICIENT MATRICES
C
      DO 300 J=1,36
      M(J)=0.0
      C(J)=0.0
300   K(J)=0.0
C
C     SET MASS MATRIX VALUES
C
      M(1)=M11
      M(8)=M2
      M(15)=I2
      M(22)=M1
      M(29)=M1
      M(36)=MT
C
C     COMPUTE CONSTANTS FOR MATRICES. FUNCTION INTERW IS
C     USED TO INTERPOLATE W(X) FOR INCODE = 2,3,4
C
      LLDXBA=LL-D-XBAR
      IF(INCODE .NE. 1) GO TO 305
      WD=UW(D)
      WLLD=UW(LL-D)
      WB=UW(B)
      WXBAR=UW(XBAR)
      GO TO 310
305   WD=INTERW(D)
      WLLD=INTERW(LL-D)
      WB=INTERW(B)
      WXBAR=INTERW(XBAR)
310   CONTINUE
C
C     SETUP DAMPING AND STIFFNESS COEFFICIENT MATRICES
C
      C(1)=C2*(WD**2+WLLD**2)+CT*(WB**2)
      C(7)=C2*(WD+WLLD)+CT*WB
      C(8)=2.*C2+CT
```

```
C******************************
C** FLEX (7-29-75)
C******************************
      C(13)=C2*((LL-D-XBAR)*WLLD-(XBAR-D)*WD)+CT*(B-XBAR)*WB
      C(14)=C2*(LL-2.*XBAR)+CT*(B-XBAR)
      C(15)=C2*((XBAR-D)**2+(LL-D-XBAR)**2)+CT*((B-XBAR)**2)
      C(19)=-C2*WD
      C(20)=-C2
      C(21)=C2*(XBAR-D)
      C(22)=C2
      C(25)=-C2*WLLD
      C(26)=-C2
      C(27)=-C2*(LL-D-XBAR)
      C(29)=C2
      C(31)=-CT*WB
      C(32)=-CT
      C(33)=-CT*(B-XBAR)
      C(36)=CT
      C(2)=C(7)
      C(3)=C(13)
      C(4)=C(19)
      C(5)=C(25)
      C(6)=C(31)
      C(9)=C(14)
      C(10)=C(20)
      C(11)=C(26)
      C(12)=C(32)
      C(16)=C(21)
      C(17)=C(27)
      C(18)=C(33)
      K(1)=K11+K2*(WD**2+WLLD**2)+KT*(WB**2)
      K(7)=K2*(WD+WLLD)+KT*WB
      K(8)=2.*K2+KT
      K(13)=K2*((LL-D-XBAR)*WLLD-(XBAR-D)*WD)+KT*(B-XBAR)*WB
      K(14)=K2*(LL-2.*XBAR)+KT*(B-XBAR)
      K(15)=K2*((XBAR-D)**2+(LL-D-XBAR)**2)+KT*((B-XBAR)**2)
      K(19)=-K2*WD
      K(20)=-K2
      K(21)=K2*(XBAR-D)
      K(22)=K1+K2
      K(25)=-K2*WLLD
      K(26)=-K2
      K(27)=-K2*(LL-D-XBAR)
      K(29)=K1+K2
      K(31)=-KT*WB
      K(32)=-KT
      K(33)=-KT*(B-XBAR)
      K(36)=KT
      K(2)=K(7)
      K(3)=K(13)
```

```
C********************************
C** FLEX (7-29-75)
C********************************
      K(4)=K(19)
      K(5)=K(25)
      K(6)=K(31)
      K(9)=K(14)
      K(10)=K(20)
      K(11)=K(26)
      K(12)=K(32)
      K(16)=K(21)
      K(17)=K(27)
      K(18)=K(33)
C
C     WRITE INTERMEDIATE OUTPUT IF REQUESTED
C
      IF(PRINT .EQ. 1)WRITE(OUT,320) (J,M(J),J,C(J),J,K(J),J=1,36)
320   FORMAT(///36(10X,'M(',I2,')=',E15.6,'      C(',I2,
     X')=',E15.6,'      K(',I2,')=',E15.6/))
C
C     COMPUTE CONSTANTS FOR MATRIX OPERATIONS
C
      TWOPI=6.28318
      CONST1=K1/2.
      CONST2=TWOPI*L
      CONST3=2.*TWOPI*A
      CONST4=TWOPI*(L+2.*A)
      CONST5=TWOPI**2/G
      Q(1)=(0.0,0.0)
      Q(2)=(0.0,0.0)
      Q(3)=(0.0,0.0)
      Q(6)=(0.0,0.0)
      LAMBDA=VORLAM
C
C     MATRIX CALCULATIONS ARE PERFORMED FOR EACH
C     POINT IN THE FREQUENCY RANGE
C
C     BEGIN FREQUENCY LOOP
C
      DO 500 NVAL=1,NFREQ
      F=CF(NVAL)
      IF(VLTEST .NE. 'LAMBD') LAMBDA=VORLAM/F
C
C     COMPLETE FORCING VECTOR Q
C     ADD ECCENTRICITY CONTRIBUTION AT F = FW
C
      IF(ECCEN .EQ. 0 .OR. ABS(F-FW) .GT. 0.0001) GO TO 360
      Q(4)=CONST1*((1.0,0.0)+EPS1*CEXP(CMPLX(0.0,THETA1))+
     X((1.0,0.0)+EPS2*CEXP(CMPLX(0.0,THETA2)))*
     XCEXP(CMPLX(0.0,CONST2/LAMBDA)))
```

```
C********************************
C** FLEX (7-29-75)
C********************************
      Q(5)=CONST1*(CEXP(CMPLX(0.0,CONST3/LAMBDA))*((1.0,0.0)+
     XEPS3*CEXP(CMPLX(0.0,THETA3)))+((1.0,0.0)+EPS4*CEXP(CMPLX
     X(0.0,THETA4)))*CEXP(CMPLX(0.0,CONST4/LAMBDA)))
      GO TO 365
360   Q(4)=CONST1*((1.0,0.0)+CEXP(CMPLX(0.0,CONST2/LAMBDA)))
      Q(5)=CONST1*(CEXP(CMPLX(0.0,CONST3/LAMBDA))+
     XCEXP(CMPLX(0.0,CONST4/LAMBDA)))
C
C     FORM COMPLEX COEFFICIENT MATRIX FROM M, C, AND K
C
365   DO 370 J=1,MATSIZ
      MAT(J)=CMPLX(K(J)-((TWOPI*F)**2*M(J)),TWOPI*F*C(J))
370   WORK(J)=MAT(J)
C
C     INVERT COEFFICIENT MATRIX WITH SUBROUTINE MINV
C
      CALL MINV(WORK,NVAR,DETER,WORKL,WORKM)
      IF(DETER .EQ. (0.0,0.0)) GO TO 4010
C
C     WORK2 IS FOR ERROR VECTOR CALCULATION
C
      CALL GMPRD(MAT,WORK,WORK2,NVAR,NVAR,NVAR)
C
C     MULTIPLY  INVERSE WITH FORCING VECTOR Q TO
C     GET COMPLEX SOLUTION VECTOR QVAR
C
      CALL GMPRD(WORK,Q,QVAR,NVAR,NVAR,1)
C
C     SAVE SOLUTION VALUES AND FREQUENCY POINTS
C
      CE(NVAL)=(QVAR(1))
      CZ2(NVAL)=(QVAR(2))
      CPHI(NVAL)=(QVAR(3))
      CY1(NVAL)=(QVAR(4))
      CY2(NVAL)=(QVAR(5))
      CZ1(NVAL)=(QVAR(6))
      PLTF(NVAL)=F
C
C     COMPUTE DISPLACEMENT RESPONSES AT
C     THREE POINTS ON CAR
C
      PLTCG(NVAL)=CABS(QVAR(2)+WXBAR*QVAR(1))
      PLTTC(NVAL)=CABS(QVAR(2)+WLLD*QVAR(1)
     X+LLDXBA*QVAR(3))
      PLTMZ1(NVAL)=CABS(QVAR(6))
      IF(ACC .EQ.0) GO TO 372
C
```

```
C     COMPUTE ACCELERATION RESPONSES
C
      ACCG(NVAL)=PLTCG(NVAL)*CONST5*F**2
      ACTC(NVAL)=PLTTC(NVAL)*CONST5*F**2
      ACZ1(NVAL)=PLTMZ1(NVAL)*CONST5*F**2
372   IF(SPEC .EQ. 0) GO TO 373
C
C     COMPUTE ACCELERATION SPECTRA
C
      SPCG(NVAL)=PLTCG(NVAL)*LAMBDA*CONST5*F**2
      SPTC(NVAL)=PLTTC(NVAL)*LAMBDA*CONST5*F**2
      SPZ1(NVAL)=PLTMZ1(NVAL)*LAMBDA*CONST5*F**2
373   CONTINUE
C
C     CALCULATE ERROR VECTOR
C
      DO 375 J=1,MATSIZ
375   WORK2(J)=WORK2(J)-IDEN(J)
      CALL GMPRD(WORK2,QVAR,WORKN,NVAR,NVAR,1)
      DO 380 J=1,NVAR
380   ERR(J,NVAL)=WORKN(J)/QVAR(J)
500   CONTINUE
      NVAL=NFREQ
C
C     PRINT INTERMEDIATE RESULTS IF REQUESTED
C
      IF(PRINT .EQ. 0) GO TO 641
      WRITE(OUT,605)
605   FORMAT(1H1,55X,'QVAR VALUES'//)
      WRITE(OUT,610)
610   FORMAT('         FREQUENCY',10X,'REAL(E)     IMAGINARY(E)',9X,
     X'REAL(Z2)    IMAGINARY(Z2)',8X,'REAL(PHI)   IMAGINARY(PHI)'//)
      WRITE(OUT,620) (CF(J),CE(J),CZ2(J),CPHI(J),J=1,NVAL)
620   FORMAT(1X,E17.5,6G17.6)
      WRITE(OUT,630)
630   FORMAT(/////8X,'FREQUENCY',9X,'REAL(Y1)    IMAGINARY(Y1)',9X,
     X'REAL(Y2)    IMAGINARY(Y2)',9X,'REAL(Z1)    IMAGINARY(Z1)'//)
      WRITE(OUT,620) (CF(J),CY1(J),CY2(J),CZ1(J),J=1,NVAL)
      WRITE(OUT,640)
640   FORMAT(1H1,55X,'ERROR VALUES'//)
      WRITE(OUT,610)
      WRITE(OUT,620) (CF(J),ERR(1,J),ERR(2,J),ERR(3,J),J=1,NVAL)
      WRITE(OUT,630)
      WRITE(OUT,620) (CF(J),ERR(4,J),ERR(5,J),ERR(6,J),J=1,NVAL)
C
C     PRINT DISPLACEMENT RESPONSES, ACCLERATION RESPONSES,
C     AND ACCELERATION SPECTRA
```

```
******************************************************************************
*                              PROGRAM FLEX                                  *
******************************************************************************
C
641    WRITE(OUT,642)
642    FORMAT(1H1,36X,'DISPLACEMENT RESPONSES (IN/IN)'//)
       WRITE(OUT,645)
645    FORMAT(19X,'FREQUENCY',9X,'CTR.OF GRAVITY',7X,
      X'TRUCK CENTER',8X,'HANGING MASS')
       WRITE(OUT,647)
647    FORMAT(22X,'(HZ)',16X,'CG/V0',15X,'TC/V0',15X,'Z1/V0'/)
       WRITE(OUT,650) (PLTF(J),PLTCG(J),PLTTC(J),
      XPLTMZ1(J),J=1,NVAL)
700    IF(ACC .EQ. 0) GO TO 725
       WRITE(OUT,710)
710    FORMAT(1H1,36X,31HACCELERATION RESPONSES (G'S/IN)/)
       WRITE(OUT,645)
       WRITE(OUT,660)
660    FORMAT(22X,'(HZ)',15X,'ACCG/V0',13X,'ACTC/V0',
      X13X,'ACZ1/V0'/)
       WRITE(OUT,650) (PLTF(J),ACCG(J),ACTC(J),ACZ1(J),J=1,NVAL)
725    IF(SPEC .EQ. 0) GO TO 800
       WRITE(OUT,740)
740    FORMAT(1H1,37X,32HACCELERATION SPECTRA (G'S/CYCLE)/)
       WRITE(OUT,645)
       WRITE(OUT,670)
670    FORMAT(22X,'(HZ)',15X,'SPCG/A1',13X,'SPTC/A1',
      X13X,'SPZ1/A1'/)
       WRITE(OUT,650) (PLTF(J),SPCG(J),SPTC(J),SPZ1(J),J=1,NVAL)
650    FORMAT(10X,4E20.6)
800    CONTINUE
C
C      TEST FOR PSD PLOTS
C
       IPSD=0
       READ(IN,810,END=815) PSD
810    FORMAT(A5)
       IF(PSD .EQ. 'RAILP') IPSD=1
815    CONTINUE
C
C      PLOTTING ROUTINE
C
C      SUBROUTINE PLOTIT IS CALLED ONCE FOR EACH PLOT DESIRED
C
       IF(DISP+ACC+SPEC+IPSD .EQ.0) STOP
       CALL INIPLT(1,0)
       CALL LOGSCA(PLTF,NVAL,XAXIS,FKVAL,FCYCLE)
       IF(DISP .EQ. 0) GO TO 820
       CALL PLOTIT(PLTF,PLTCG,29HDISP.RESPONSE-CTR. OF GRAVITY,29,
      X13HCG/V0 (IN/IN),13)
```

```
***********************************************************************
*                          PROGRAM FLEX                               *
***********************************************************************
      CALL PLOTIT(PLTF,PLTTC,26HDISP.RESPONSE-TRUCK CENTER,26,
     X13HTC/VO (IN/IN),13)
      CALL PLOTIT(PLTF,PLTMZ1,26HDISP.RESPONSE-HANGING MASS,26,
     X13HZ1/VO (IN/IN),13)
820   IF(ACC .EQ. 0) GO TO 840
      CALL PLOTIT(PLTF,ACCG,29HACCEL.RESPONSE-CTR.OF GRAVITY,29,
     X16HACCG/VO (G'S/IN),16)
      CALL PLOTIT(PLTF,ACTC,27HACCEL.RESPONSE-TRUCK CENTER,27,
     X16HACTC/VO (G'S/IN),16)
      CALL PLOTIT(PLTF,ACZ1,27HACCEL.RESPONSE-HANGING MASS,27,
     X16HACZ1/VO (G'S/IN),16)
840   IF(SPEC .EQ. 0) GO TO 860
      CALL PLOTIT(PLTF,SPCG,28HACCEL.SPECTRA-CTR.OF GRAVITY,28,
     X19HSPCG/A1 (G'S/CYCLE),19)
      CALL PLOTIT(PLTF,SPTC,26HACCEL.SPECTRA-TRUCK CENTER,26,
     X19HSPTC/A1 (G'S/CYCLE),19)
      CALL PLOTIT(PLTF,SPZ1,26HACCEL.SPECTRA-HANGING MASS,26,
     X19HSPZ1/A1 (G'S/CYCLE),19)
860   IF(IPSD .EQ. 0 .OR. VLTEST .EQ. 2) GO TO 870
      CALL RAILPL(CF,VORLAM,NVAL,6,IITYPE,SAVEM,200,IN,OUT)
870   CALL FINPLT
      STOP
C
C     DETECTED ERRORS
C
4000  WRITE(OUT,4001) INTER
4001  FORMAT(////' ****** INTEGRATION UNSUCCESSFUL ******'/
     X10X,'INTER =',I3)
      STOP
4010  WRITE(OUT,4011) MAT,M,C,K
4011  FORMAT(////' ****** MATRIX INVERSION UNSUCCESSFUL ******'
     X//10X,'MAT,M,C,K:'//36(10X,G10.5,G10.5/)/
     X3(//36(10X,E11.5/)))
      STOP
      END
```

```
C********************************
C** FLEX (7-29-75)
C********************************
      SUBROUTINE PLOTIT(PLTF,YRAY,TITLE,NCHAR,LABEL,KAR)
C
C     SUBROUTINE PLOTIT PRODUCES ONE PLOT FOR EACH
C     TIME IT IS CALLED.
C
C     ARGUMENTS:
C
C     PLTF     X-AXIS ARRAY
C     YRAY     Y-AXIS ARRAY
C     TITLE    HOLLERITH TITLE FOR THE PLOT
C     NCHAR    NUMBER OF CHARACTERS IN TITLE
C     LABEL    Y-AXIS LABEL
C     KAR      NUMBER OF CHARACTERS IN LABEL
C
      REAL YRAY(1),TITLE(1),LABEL(1),YST,YCOORD(300),
     XXAXIS,YAXIS,PLTF(1),FINC,POSX,POSY,HIGH,X,Y
      REAL SX,SY
      REAL FW,VORLAM,W1,W2,WT,K1,K2,KT,FB,BETA2,BETAT
      INTEGER NCHAR,NVAL,FKVAL,FCYCLE,KVAL,NCYCLE,IDUM,INCODE
      INTEGER VLTEST,ECCEN,KAR
      COMMON /COPLOT/FCYCLE,XAXIS,FKVAL,NVAL,YAXIS
     X,VLTEST,VORLAM,W1,W2,WT,K1,K2,KT,FB,BETA2,BETAT,INCODE
     X,ECCEN,FW
      FINC=.16
      YST=1.9
      POSY=YST
      HIGH=.08
      POSX=.15
      DO 5 JJJ=1,NVAL
5     YCOORD(JJJ)=YRAY(JJJ)
      IF(VLTEST .EQ. 1) GO TO 10
      CALL SYMBOL(POSX,POSY,HIGH,8HLAMBDA= ,0,8)
      CALL NUMBER(999.,999.,HIGH,VORLAM,0,1)
      CALL SYMBOL(999.,999.,HIGH,3H IN,0,3)
      GO TO 20
10    CALL SYMBOL(POSX,POSY,HIGH,5HVEL= ,0,5)
      CALL NUMBER(999.,999.,HIGH,VORLAM,0,1)
      CALL SYMBOL(999.,999.,HIGH,4H MPH,0,4)
20    POSY=POSY-FINC
      CALL SYMBOL(POSX,POSY,HIGH,4HW1= ,0,4)
      CALL NUMBER(999.,999.,HIGH,W1,0,0)
      CALL SYMBOL(999.,999.,HIGH,4H LBS,0,4)
      POSY=POSY-FINC
      CALL SYMBOL(POSX,POSY,HIGH,4HW2= ,0,4)
      CALL NUMBER(999.,999.,HIGH,W2,0,0)
      CALL SYMBOL(999.,999.,HIGH,4H LBS,0,4)
      POSY=POSY-FINC
```

```
C*****************************
C** FLEX (7-29-75)
C*****************************
      CALL SYMBOL(POSX,POSY,HIGH,4HWT= ,0,4)
      CALL NUMBER(999.,999.,HIGH,WT,0,0)
      CALL SYMBOL(999.,999.,HIGH,4H LBS,0,4)
      POSY=POSY-FINC
      CALL SYMBOL(POSX,POSY,HIGH,4HK1= ,0,4)
      CALL NUMBER(999.,999.,HIGH,K1,0,0)
      CALL SYMBOL(999.,999.,HIGH,7H LBS/IN,0,7)
      POSY=POSY-FINC
      CALL SYMBOL(POSX,POSY,HIGH,4HK2= ,0,4)
      CALL NUMBER(999.,999.,HIGH,K2,0,0)
      CALL SYMBOL(999.,999.,HIGH,7H LBS/IN,0,7)
      POSY=POSY-FINC
      CALL SYMBOL(POSX,POSY,HIGH,4HKT= ,0,4)
      CALL NUMBER(999.,999.,HIGH,KT,0,0)
      CALL SYMBOL(999.,999.,HIGH,7H LBS/IN,0,7)
      IF(INCODE .GT. 2) GO TO 100
      POSY=POSY-FINC
      CALL SYMBOL(POSX,POSY,HIGH,4HFB= ,0,4)
      CALL NUMBER(999.,999.,HIGH,FB,0,2)
      CALL SYMBOL(999.,999.,HIGH,3H HZ,0,3)
100   POSY=POSY-FINC
      CALL SYMBOL(POSX,POSY,HIGH,7HBETA2= ,0,7)
      CALL NUMBER(999.,999.,HIGH,BETA2,0,3)
      POSY=POSY-FINC
      CALL SYMBOL(POSX,POSY,HIGH,7HBETAT= ,0,7)
      CALL NUMBER(999.,999.,HIGH,BETAT,0,3)
      POSY=POSY-FINC
      IF(ECCEN .EQ. 0) GO TO 150
      CALL SYMBOL(POSX,POSY,HIGH,4HFW= ,0,4)
      CALL NUMBER(999.,999.,HIGH,FW,0,2)
      CALL SYMBOL(999.,999.,HIGH,3H HZ,0,3)
150   CALL LOGSCA(YCOORD,NVAL,YAXIS,KVAL,NCYCLE)
      SX=POSX+20.*HIGH+0.3
      SY=YST+HIGH+0.1+0.3
      CALL LAXIS(0.0,0.0,FCYCLE,14HFREQUENCY (HZ),14,XAXIS,FKVAL,2)
      CALL LAXIS(0.0,0.0,NCYCLE,LABEL,KAR,YAXIS,KVAL,1)
      CALL GRID(0.,0.,XAXIS,YAXIS,FCYCLE,NCYCLE,3,2,0.,SX,0.,SY)
      CALL SYMBOL((XAXIS-NCHAR*.15)*.5,YAXIS+.3,.15,TITLE,0,NCHAR)
      CALL ALINE(PLTF,YCOORD,NVAL,0.0,1.0,0.0,1.0)
      CALL PLOT(12.0,0.0,-3)
      RETURN
      END
```

```
C*******************************
C** FLEX (7-29-75)
C*******************************
      SUBROUTINE DIFF2(W,WDX2,INTDIM,INTDEL)
C
C     DIFF2 COMPUTES THE SECOND DERIVATIVE OF THE
C     FUNCTION W(X)
C
      REAL W,WDX2,INTDEL
      DIMENSION W(1),WDX2(1)
      WDX2(1)=Ø.Ø
      WDX2(INTDIM)=Ø.Ø
      DO 1ØØ J=2,INTDIM-1
1ØØ   WDX2(J)=(W(J-1)-2*W(J)+W(J+1))/INTDEL**2
      RETURN
      END
```

```
      REAL FUNCTION INTERW(X)
C
C     INTERW PERFORMS A LINEAR INTERPOLATION ON FUNCTION W(X)
C
      REAL INTDEL,W(25)
      COMMON/SUB/ INTDEL,W
      IB=(X/INTDEL)+1.
      DELTA=W(IB+1)-W(IB)
      INTERW=W(IB)+(((X-(IB-1)*INTDEL)/INTDEL)*DELTA)
      RETURN
      END
```

```
C*****************************
C** FLEX (7-29-75)
C*****************************
C
C     ..................................................................
C
C        SUBROUTINE QSF
C
C        PURPOSE
C           TO COMPUTE THE VECTOR OF INTEGRAL VALUES FOR A GIVEN
C           EQUIDISTANT TABLE OF FUNCTION VALUES.
C
C        USAGE
C           CALL QSF (H,Y,Z,NDIM)
C
C        DESCRIPTION OF PARAMETERS
C           H      - THE INCREMENT OF ARGUMENT VALUES.
C           Y      - THE INPUT VECTOR OF FUNCTION VALUES.
C           Z      - THE RESULTING VECTOR OF INTEGRAL VALUES. Z MAY BE
C                    IDENTICAL WITH Y.
C           NDIM   - THE DIMENSION OF VECTORS Y AND Z.
C
C        REMARKS
C           NO ACTION IN CASE NDIM LESS THAN 3.
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           NONE
C
C        METHOD
C           BEGINNING WITH Z(1)=0, EVALUATION OF VECTOR Z IS DONE BY
C           MEANS OF SIMPSONS RULE TOGETHER WITH NEWTONS 3/8 RULE OR A
C           COMBINATION OF THESE TWO RULES. TRUNCATION ERROR IS OF
C           ORDER H**5 (I.E. FOURTH ORDER METHOD). ONLY IN CASE NDIM=3
C           TRUNCATION ERROR OF Z(2) IS OF ORDER H**4.
C           FOR REFERENCE, SEE
C           (1) F.B.HILDEBRAND, INTRODUCTION TO NUMERICAL ANALYSIS,
C               MCGRAW-HILL, NEW YORK/TORONTO/LONDON, 1956, PP.71-76.
C           (2) R.ZURMUEHL, PRAKTISCHE MATHEMATIK FUER INGENIEURE UND
C               PHYSIKER, SPRINGER, BERLIN/GOETTINGEN/HEIDELBERG, 1963,
C               PP.214-221.
C
C     ..................................................................
C
      SUBROUTINE QSF(H,Y,Z,NDIM)
C
C
      DIMENSION Y(1),Z(1)
C
      HT=.3333333*H
      IF(NDIM-5)7,8,1
```

```
C*******************************
C** FLEX (7-29-75)
C*******************************
C
C      NDIM IS GREATER THAN 5. PREPARATIONS OF INTEGRATION LOOP
     1 SUM1=Y(2)+Y(2)
       SUM1=SUM1+SUM1
       SUM1=HT*(Y(1)+SUM1+Y(3))
       AUX1=Y(4)+Y(4)
       AUX1=AUX1+AUX1
       AUX1=SUM1+HT*(Y(3)+AUX1+Y(5))
       AUX2=HT*(Y(1)+3.875*(Y(2)+Y(5))+2.625*(Y(3)+Y(4))+Y(6))
       SUM2=Y(5)+Y(5)
       SUM2=SUM2+SUM2
       SUM2=AUX2-HT*(Y(4)+SUM2+Y(6))
       Z(1)=0.
       AUX=Y(3)+Y(3)
       AUX=AUX+AUX
       Z(2)=SUM2-HT*(Y(2)+AUX+Y(4))
       Z(3)=SUM1
       Z(4)=SUM2
       IF(NDIM-6)5,5,2
C
C      INTEGRATION LOOP
     2 DO 4 I=7,NDIM,2
       SUM1=AUX1
       SUM2=AUX2
       AUX1=Y(I-1)+Y(I-1)
       AUX1=AUX1+AUX1
       AUX1=SUM1+HT*(Y(I-2)+AUX1+Y(I))
       Z(I-2)=SUM1
       IF(I-NDIM)3,6,6
     3 AUX2=Y(I)+Y(I)
       AUX2=AUX2+AUX2
       AUX2=SUM2+HT*(Y(I-1)+AUX2+Y(I+1))
     4 Z(I-1)=SUM2
     5 Z(NDIM-1)=AUX1
       Z(NDIM)=AUX2
       RETURN
     6 Z(NDIM-1)=SUM2
       Z(NDIM)=AUX1
       RETURN
C      END OF INTEGRATION LOOP
C
     7 IF(NDIM-3)12,11,8
C
C      NDIM IS EQUAL TO 4 OR 5
     8 SUM2=1.125*HT*(Y(1)+Y(2)+Y(2)+Y(2)+Y(3)+Y(3)+Y(3)+Y(4))
       SUM1=Y(2)+Y(2)
       SUM1=SUM1+SUM1
```

```
C*****************************
C** FLEX (7-29-75)
C*****************************
      SUM1=HT*(Y(1)+SUM1+Y(3))
      Z(1)=0.
      AUX1=Y(3)+Y(3)
      AUX1=AUX1+AUX1
      Z(2)=SUM2-HT*(Y(2)+AUX1+Y(4))
      IF(NDIM-5)10,9,9
    9 AUX1=Y(4)+Y(4)
      AUX1=AUX1+AUX1
      Z(5)=SUM1+HT*(Y(3)+AUX1+Y(5))
   10 Z(3)=SUM1
      Z(4)=SUM2
      RETURN
C
C     NDIM IS EQUAL TO 3
   11 SUM1=HT*(1.25*Y(1)+Y(2)+Y(2)-.25*Y(3))
      SUM2=Y(2)+Y(2)
      SUM2=SUM2+SUM2
      Z(3)=HT*(Y(1)+SUM2+Y(3))
      Z(1)=0.
      Z(2)=SUM1
   12 RETURN
      END
```

```
C******************************
C** FLEX (7-29-75)
C******************************
      SUBROUTINE FTABG(IFREQ,NDF,DF,FL,F,NF)
C
C     THIS SUBROUTINE COMPUTES THE FREQUENCY TABLE
C
C     IFREQ=1 RANGE INPUTS, =2 OCTAVE INPUTS
C     NDF  - NUMBER OF RANGES OR OCTAVES
C     DF   - FREQUENCY POINTS VECTOR
C     FL   - FREQUENCY LOWER LIMITS VECTOR
C     F    - FREQUENCY TABLE RETURNED
C     NF   - NUMBER OF POINTS IN FREQUENCY TABLE
C
      INTEGER DF(7)
      DIMENSION FL(8),F(200)
      IF(IFREQ .EQ. 2) GO TO 500
C
C     RANGE INPUTS
C
      NF=0
      DO 200 I=1,NDF
      C=EXP(ALOG(FL(I+1)/FL(I))/DF(I))
      NF=NF+1
      F(NF)=FL(I)
      DO 100 J=1,DF(I)
      FT=C*F(NF)
      IF(FT .GE. FL(I+1)-0.000001) GO TO 200
      NF=NF+1
      F(NF)=FT
100   CONTINUE
200   CONTINUE
      RETURN
C
C     OCTAVE INPUTS
C
500   C=2**(1./DF(1))
      NF=1
      F(NF)=FL(1)
      DO 700 I=1,NDF
      DO 700 J=1,DF(1)
      NF=NF+1
      F(NF)=F(NF-1)*C
700   CONTINUE
      RETURN
      END
```

APPENDIX C
PROGRAM LATERAL INPUT PARAMETERS AND LISTINGS

The Program LATERAL input deck consists of the following fifteen cards:

Card 1 UNITS: NONE

|16A5|
RUN

This is an 80-character run description which will be printed on the lineprinter output and at the beginning of a sequence of plots. The run description nominally used consists of the Lateral Test Run No., the IOPT No. and the date. Other data may be included as deemed necessary by the user.

Card 2 UNITS: NONE

|15I1|
DTAG(N), N=1, 15

This is a control vector with an entry for each of the fifteen coordinates for which displacement responses are requested. For the Nth coordinate:

DTAG(N) = 0, No Output
1, Print Displacement Response
2, Print and Plot Displacement Response

Card 3 UNITS: NONE

|15I1 |
ATAG(N),N=1,15

This is a control vector with an entry for each of the fifteen coordinates for which acceleration responses are requested. For the Nth coordinate:

ATAG(N) = 0, No Output
1, Print Acceleration Response
2, Print and Plot Acceleration Response

Card 4 UNITS: RADIANS

|F5.2|
ALPHA

ALPHA is the wheel conicity

Card 5 UNITS: LBS-SEC/IN, IN-LBS-SEC/RAD, LBS-SEC/IN, IN-LBS-SEC/RAD, LBS-SEC/IN

F10.4	F10.4	F10.4	F10.4	F10.4
C1	C2	C3	C4	C6

C1 - Axle Primary Suspension Lateral Damping

C2 - Primary Suspension Yaw Damping

C3 - Secondary Suspension Lateral Damping

C4 - Secondary Suspension Yaw Damping

C6 - Secondary Suspension Vertical Damping

Card 6 UNITS: FEET

F10.4	F10.4	F10.4	F10.4	F10.4	F10.4	F10.4
LL	2L	2HL	D	W	E	E2

LL - Car Body Length

2L* - Gage

2HL* - Axle Spacing

D - Distance from Centerplate to end of car

W - Truck (Centerplate) Spacing

E - Car-Body Center of Mass Height above Secondary Suspension Lateral Stiffness Axis

E2 - Secondary Lateral Suspension Height above Truck/Wheelset Center of Mass

*The quantities 2L, 2HL, and 2B correspond to track gage and axle spacing distances, and the lateral spacing between secondary-suspension vertical stiffness elements. These quantities represent the proper input data (as opposed to the corresponding half distances for these quantities).

Card 7

UNITS: IN/SEC^2, FEET, FEET, $IN\text{-}LBS\text{-}SEC^2$

F10.4	F10.4	F10.4	F10.4
G	RO	2B	IZERO

G - Gravity Constant

RO - Wheel Radius

2B* - Lateral Spacing Between Secondary Suspension Vertical Stiffness effements.

IZERO - Wheelset Spin Axis-Mass Moment of Inertia

Card 8

UNITS: LBS/IN, IN-LBS/RAD, LBS/IN, IN-LBS/RAD, LBS/IN

E12.4	E12.4	E12.4	E12.4	E12.4
K1	K2	K3	K4	K6

K1 - Axle Primary Suspension Lateral Stiffness

K2 - Primary Suspension Yaw Stiffness

K3 - Secondary Suspension Lateral Stiffness

K4 - Secondary Suspension Yaw Stiffness

K6 - Secondary Suspension Vertical Stiffness

Card 9

UNITS: $IN\text{-}LBS\text{-}SEC^2$, $IN\text{-}LBS\text{-}SEC^2$, $IN\text{-}LBS\text{-}SEC^2$, $IN\text{-}LBS\text{-}SEC^2$, LBS, LBS

E12.4	E12.4	E12.4	E12.4	E12.4	E12.4
IW	IT	IBY	IBR	FSUBL	FT

IW - Axle Yaw Inertia

IT - Truck Frame Yaw Inertia

IBY - Car-Body Yaw Inertia

IBR - Car-Body Roll Inertia

FSUBL - Creep Coefficient (Lateral)

FT - Creep Coefficient (Longitudinal)

Card 10

UNITS: LBS, LBS, LBS, LBS-IN/RAD, LBS/IN

E12.4	E12.4	E12.4	E12.4	E12.4
MWG	MTG	MBG	KA	KG

MWG - Wheelset Weight

MTG - Truck Frame Weight

MBG - Car-Body Weight

KA - Wheel-Rail Gravitational Yaw Stiffness

KG - Wheel-Rail Gravitational Lateral Stiffness

Card 11 UNITS: NONE

```
| I2 |
 IOPT
```

IOPT is a control variable to select the forcing function applied to the lateral car model.

IOPT = 1, Sinusoidal lateral displacement of the track center-line in the horizontal plane (alignment perturbation).

IOPT = 2, Crosslevel perturbation.

IOPT = 3, Equivalent fixed base inertia force-excited system, identical to the crosslevel (base-excited) perturbation (i.e., a fixed base force-excited system of IOPT=2). The system is identical in the sense that identical responses are generated relative to the track, for options II and III.

IOPT = 4, As presently coded Option 4 applies inertia forces and torques only to car body, with all other forcing functions set to zero. This option is intended to provide the capability of applying inertia forces to a partial set of coordinates. Coding changes will be required to alter that set of inertia forces if forces other than car body forces (lateral, yaw, and roll) are desired.

Card 12

UNITS:		
	ILAMB	NONE
	V	MPH
	LAMBDA	FEET/CYCLE

```
| I1  |F5.2|
 ILAMB V, if ILAMB = 1
       LAMBDA, if ILAMB = 2
```

Set ILAMB = 1 to specify a constant vehicle velocity, and ILAMB = 2 to specify a constant track wavelength.

Cards 13, 14, 15

CARD (13) UNITS: NONE

```
|  I2  | I2|
 IFREQ NDF
```

CARD (14) UNITS: NONE

|7I4|
DF(N),N=1,NDF If IFREQ = 1
DF(1) If IFREQ = 2

CARD (15) UNITS: HERTZ

|8F10.4|
FL(N),N=1,NDF+1 If IFREQ = 1
FL(1) If IFREQ = 2

IFREQ selects one of two methods for specifying the frequency points at which the response will be found. With IFREQ = 1, logarithmically evenly-spaced points are generated for a set of frequency ranges. NDF gives the number of frequency ranges (maximum of seven). DF(N) specifies the number of points for the range FL(N) to FL(N+1). FL(N) gives the lower limit of the Nth frequency range, and FL(N+1) is the upper limit for the Nth range. FL(NDF+1) gives the upper limit for the last frequency range.

With IFREQ = 2, logarithmically evenly-spaced points are generated for the number of octaves. NDF specifies the number of octaves. DF(1) gives the number of points per octave, and FL(1) gives the starting frequency of the first octave. Note that the remainders of the DF and FL fields are left blank.

```
C*******************************
C** LATERAL (7-29-75)
C*******************************
C** FOLLOWING TEXT PRINTED FROM FILE DSKB:LATER.F4  [4351,533] 29-JUL-75
C*******************************
C
C
C        ******  MAIN PROGRAM LATERAL  ******
C
C        THIS PROGRAM COMPUTES THE LATERAL RESPONSE
C        CHARACTERISTICS OF A TRAIN CAR SUBJECTED
C        TO A CROSS-LEVEL OR ALIGNMENT TRACK
C        PERTURBATION.
C
C        SUBROUTINES REQUIRED: LATERP,MINV,GMPRD,INIPLT,
C                              FINPLT,SYMBOL,MAXMIN,
C                              LOGSCA,LAXIS,PLOT,GRID
C
C        INPUT: FOR02.DAT
C
C        OUTPUT: PRINT FILE (FOR21.DAT), PLOT TAPE (UNIT 16)
C
C
C
C        SUBSYSTEM  ITEM               SYMBOL    UNITS      TYPICAL VALUE
C
C
C        AXLE       AXLE WEIGHT             MWG     POUNDS                 5220
C                   LATERAL DAMPING          C1     LBS-SEC/IN              116
C                   LATERAL STIFFNESS        K1     LBS/IN                27860
C                   AXLE SPACING            2HL     FEET                    8.5
C                   AXLE YAW INERTIA         IW     IN-LBS-SEC2            8125
C                   TORSION YAW DAMP         C2     IN-LBS-SEC/RAD            0
C                   TORSION YAW STIF         K2     IN-LBS/RAD             36E7
C        TRUCK      WEIGHT/NO AXLE          MTG     POUNDS                12610
C                   LATERAL DAMPING          C3     LBS-SEC/IN              173
C                   LATERAL STIFF            K3     LBS/IN                 3385
C                   YAW INERTIA              IT     IN-LBS-SEC2           55000
C                   YAW DAMPING              C4     IN-LBS-SEC/RAD            0
C                   YAW STIFF                K4     IN-LBS/RAD             60E6
C        CARBODY    WEIGHT-TOTAL            MBG     POUNDS               127600
C                   YAW INERTIA             IBY     IN-LBS-SEC2          2.87E7
C                   ROLL INERTIA            IBR     IN-LBS-SEC2          655220
C                   SECONDARY STIFF          K6     LBS/IN                 3900
C                   CAR BODY LENGTH          LL     FEET                     85
C                   TRUCK OFFSET              D     FEET                   12.9
C                   WHEEL BASE LENGTH         W     FEET                   59.5
C                   GAGE                     2L     FEET                    4.7
C                   BODY C.G. HEIGHT          E     FEET                    2.0
C                   SECONDARY HEIGHT         E2     FEET                    2.5
```

```
C********************************
C** LATERAL (7-29-75)
C********************************
C                        SECONDARY DAMP         C6      LBS-SEC/IN          57.5
C                        SEC. SPACING           2B      FEET                8.00
C      WHEEL             YAW GRAV.STIFFNESS     KA      IN-LBS/RAD          3750
C                        LAT.GRAV.STIFFNESS     KG      LBS/IN              7917
C                        CREEP COEFFICIENT   FSUBL      LBS                  3E6
C                        CREEP COEFFICIENT      FT      LBS                  3E6
C                        CONICITY            ALPHA      RADIANS              .05
C                        WHEEL RADIUS           RO      FEET                 1.3
C                        WHEELSET MOMENT     IZERO      IN-LBS-SEC2          700
C
C
C
C
C
C     DEGREES OF FREEDOM IN PROGRAM LATERAL
C
C
C     SYMBOL          COORDINATE
C
C      1 X1           LEADING TRUCK FRONT AXLE LATERAL DISPLACEMENT (SWAY)
C      2 PSI1         LEADING TRUCK FRONT AXLE YAW
C      3 X2           LEADING TRUCK BODY LATERAL DISPLACEMENT (SWAY)
C      4 PSI2         LEADING TRUCK BODY YAW
C      5 X3           LEADING TRUCK REAR AXLE LATERAL DISPLACEMENT (SWAY)
C      6 PSI3         LEADING TRUCK REAR AXLE YAW
C      7 X4           CAR BODY LATERAL DISPLACEMENT (SWAY)
C      8 PSI4         CAR BODY YAW
C      9 THETA        CAR BODY ROLL
C     10 X5           TRAILING TRUCK FRONT AXLE LATERAL DISPLACEMENT (SWAY)
C     11 PSI5         TRAILING TRUCK FRONT AXLE YAW
C     12 X6           TRAILING TRUCK LATERAL DISPLACEMENT (SWAY)
C     13 PSI6         TRAILING TRUCK YAW
C     14 X7           TRAILING TRUCK REAR AXLE LATERAL DISPLACEMENT (SWAY)
C     15 PSI7         TRAILING TRUCK REAR AXLE YAW
C
C
C
        DIMENSION IPPR(3)
        DIMENSION ALFA(2),DEG(2),IO(2)
        DIMENSION MUUG(2),MTEEG(2),MBEEG(2),HEL(2)
        DIMENSION IUU(2),IBER(2),IBEY(2),PSPI(3),PPI(2),PSSI(3)
        DIMENSION FLBS(2),PPF(2),FSBEL(2),PSPR(3)
      DIMENSION SAVEM(200,30),IITYPE(30)
      DIMENSION LDUM(15),MDUM(15)
      DIMENSION F(200),FL(8)
      DIMENSION M(225),C(225),K(225)
      INTEGER DTAG(15),ATAG(15),BLANK,BLANKA,DF(7)
```

```
C******************************
C** LATERAL (7-29-75)
C******************************
      DIMENSION DISPL(200,15),ACCEL(200,15),NAME(6,15),INAME(15),
     XRUN(16),MAME(15),JHOLD(15),ARRAA(200),NAME1(6),NAME2(6)
      DIMENSION ILABD(4,4),ILABA(4,4),KLAB(2,4),JLABD(3,4)
      DIMENSION JLABA(3,4),LATANG(15),DNORM(4),ANORM(4)
      COMPLEX SOLD(200,15)
      REAL L,L2,IW,MW,MT,MB
      REAL MWG,K1,IN,K2,MTG,K3,IT,K4,MBG,IBY,IBR,K6,K6X
      REAL M,K,KA,KG,IZERO
      COMPLEX DET,ZDUM
      COMPLEX BMAT(225),CMAT(225),SOLU(15)
      COMPLEX MAT(225),EYE
      COMPLEX X,Y,Z,XA,YA,Q(15),Y1,X1,Z1,Z2
      COMPLEX CEXP1,CEXP2,CEXP3,CEXP4,CTRM1,CTRM2,CTRM3,X12
      COMPLEX Y12,CEXP5,CFAC1,CFAC2,CFAC3,CFAC4,CFAC5,CFAC6,CFAC7
      EQUIVALENCE (SAVEM(1,1),DISPL(1,1)),(SAVEM(1,16),
     XACCEL(1,1))
      DATA IITYPE/1,3,1,3,1,3,1,3,3,1,3,1,3,1,3,2,4,2,4,2,4,2,4,
     X4,2,4,2,4,2,4/
C
C     THE FOLLOWING DATA STATEMENTS DEFINE THE INPUT PRINTING LABELS
C
        DATA ROW/'RO = '/,IPPR/' IN-L','BS/RA','D    '/
        DATA IO/' IOPT',' = '/
        DATA VEE/' V = '/,MPH/' MPH'/
        DATA ALFA/'ALPHA',' = '/,DEG/' RADI','ANS'/
        DATA MUUG/'MWG =',' '/,MTEEG/'MTG =',' '/,MBEEG/'MBG =',' '/,
     2  IBEY/'IBY =',' '/,IBER/'IBR =',' '/,PND/' LBS'/,
     3  PSPI/' LBS-','SEC/I','N'/,PPI/' LBS/','IN'/,
     4  FOOT/' FT'/,PSSI/' IN-LBS-SEC**2 '/,PSPR/' IN-LBS-SEC/RAD'/,
     5  FLBS/' FT-L','BS'/,PPF/' LBS/',' FT'/,CONE/'C1 = '/,
     6  CTOO/'C2 = '/,CTREE/'C3 = '/,CFORE/'C4 = '/,KONE/'K1 = '/,
     7  KTOO/'K2 = '/,KTREE/'K3 = '/,KFORE/'K4 = '/,KFIVE/'K5 = '/,
     8  KSIX/'K6 = '/,KAYE/'KA = '/,KAGE/'KG = '/,ROW/'RO = '/,
     9  FSBEL/'FSUBL',' = '/,FTEE/'FT = '/,HEL/'2HL =',' '/
        DATA IWW/'IW = '/,ITEE/'IT = '/,ELTOO/'2L = '/,
     2  DEE/' D = '/,UU/' W = '/,EEE/' E = '/,BEE/'2B = '/
        DATA EL/'LL = '/
      DATA CSICK/'C6 = '/,EE2/'E2 = '/
C
C     THESE DATA STATEMENTS DEFINE THE LABELS FOR PRINTED OUTPUT VARIABLES
C
      DATA KLAB/10H/ YZERO   ,10H/ ZZERO   ,10H/ YZERO   ,10H/ THETAO   /
      DATA JLABD/15H  (IN/IN)        ,15H  (IN/IN)       ,
     X15H (DEG/IN)       ,15H(DEG/DEG)      /
      DATA JLABA/15H (G'S/IN)       ,15H (G'S/IN)       ,
     X15H(RAD/SEC2/IN)  ,15H(RAD/SEC2/DEG) /
C
```

```
C*******************************
C** LATERAL (7-29-75)
C*******************************
C     LATANG SPECIFIES WHICH VARIABLES ARE LATERAL(0)
C     AND WHICH ARE ANGULAR(2)
C
      DATA LATANG/0,2,0,2,0,2,0,2,2,0,2,0,2,0,2/
C
C     THE FOLLOWING DATA STATEMENTS DEFINE THE LABELS FOR PLOT VARIABLES
C
      DATA ILABD/20H/ YZERO  (IN/IN)    ,20H/ ZZERO  (IN/IN)    ,
     X20H/ YZERO  (DEG/IN)   ,20H/ THETA0  (DEG/DEG) /
      DATA ILABA/20H/ YZERO  (G'S/IN)   ,20H/ ZZERO  (G'S/IN)   ,
     X20H/YZERO (RAD/SEC2/IN),20H/THETA0(RAD/SEC2/DG)/
      DATA NAME/5HLEAD ,5HTRUCK,5H FRON,5HT AXL,5HE LAT,5H DISP,
     2          5HLEADI,5HNG TR,5HUCK F,5HRONT ,5HAXLE ,5HYAW  ,
     3          5HLEADI,5HNG TR,5HUCK B,5HODY L,5HAT.DI,5HSP.  ,
     4          5HLEADI,5HNG TR,5HUCK B,5HODY Y,5HAW   ,5H     ,
     5          5HLEAD ,5HTRUCK,5H REAR,5H AXLE,5H LAT.,5HDISP.,
     6          5HLEADI,5HNG TR,5HUCK R,5HEAR A,5HXLE Y,5HAW   ,
     7          5HCAR B,5HODY L,5HATERA,5HL DIS,5HPLACE,5HMENT ,
     8          5HCAR B,5HODY Y,5HAW   ,5H     ,5H     ,5H     ,
     9          5HCAR B,5HODY R,5HOLL  ,5H     ,5H     ,5H     ,
     X          5HTR. T,5HRUCK ,5HFRONT,5H AXLE,5H LAT.,5HDISP.,
     1          5HTRAIL,5HING T,5HRUCK ,5HFRONT,5H AXLE,5H YAW ,
     2          5HTRAIL,5HING T,5HRUCK ,5HLATER,5HAL DI,5HSP.  ,
     3          5HTRAIL,5HING T,5HRUCK ,5HYAW  ,5H     ,5H     ,
     4          5HTR. T,5HRUCK ,5HREAR ,5HAXLE ,5HLAT.D,5HISP. ,
     5          5HTRAIL,5HING T,5HRUCK ,5HREAR ,5HAXLE ,5HYAW  /
      DATA MAME/5HX1   ,5HPSI1 ,5HX2   ,5HPSI2 ,5HX3   ,5HPSI3 ,
     1          5HX4   ,5HPSI4 ,5HTHETA,5HX5   ,5HPSI5 ,5HX6   ,
     2          5HPSI6 ,5HX7   ,5HPSI7 /
      DATA INAME/30,28,28,22,30,27,29,12,13,30,29,28,18,29,28/
      DATA BLANK,BLANKA/5H     ,5H    A/
C
C     INITIALIZE MATRICES AND DEFINE CONSTANTS
C
      DATA M/225*0.0/,C/225*0.0/,K/225*0.0/
      PIE=3.14159
      EYE=(0,1.)
C
C       READ INPUT VALUES
C
      READ(2,2) RUN
2     FORMAT(16A5)
      READ(2,3) DTAG,ATAG
3     FORMAT(15I1/15I1)
      READ(2,5) ALPHA
    5 FORMAT(F5.2)
      READ(2,10) C1,C2,C3,C4,C6
```

```
C******************************
C** LATERAL (7-29-95)
C******************************
   10 FORMAT(7F10.4)
      READ(2,10) L,L2,HL,D,W,E,E2
      READ(2,10) G,RO,B,IZERO
      READ(2,15) K1,K2,K3,K4,K6
   15 FORMAT(6E12.4)
      READ(2,15) IW,IT,IBY,IBR,FSUBL,FT
      READ(2,15) MWG,MTG,MBG,KA,KG
C
C        OPTION I SELECTS A FORCING FUNCTION BASED ON TRACK ALIGNMENT
C        PERTURBATION.  OPTION II SELECTS A FORCING FUNCTION BASED
C        ON TRACK CROSS LEVEL DEVIATION.
C
      READ(2,18) IOPT
   18 FORMAT(I2)
      READ(2,183) ILAMB,V
  183 FORMAT(I1,F5.2)
C
C        PRINTS INPUT VALUES
C
         WRITE(21,6000)
         WRITE(21,6009) RUN
         WRITE(21,7050) IO,IOPT
         IF(ILAMB .EQ. 1) WRITE(21,7020) VEE,V,MPH
         IF(ILAMB .NE. 1) WRITE(21,7023) V
         WRITE(21,7021) G
         WRITE(21,6002)
         WRITE(21,7010) MUUG,MWG,PND
         WRITE(21,7020) CONE,C1,PSPI
         WRITE(21,7020) CTOO,C2,PSPR
         WRITE(21,7020) KONE,K1,PPI
         WRITE(21,7020) KTOO,K2,IPPR
         WRITE(21,7010) HEL,HL,FOOT
         WRITE(21,7020) IWW,IW,PSSI
         WRITE(21,6005)
         WRITE(21,7010) MTEEG,MTG,PND
         WRITE(21,7020) CTREE,C3,PSPI
         WRITE(21,7020) CFORE,C4,PSPR
         WRITE(21,7020) KTREE,K3,PPI
         WRITE(21,7020) KFORE,K4,IPPR
         WRITE(21,7020) ITEE,IT,PSSI
         WRITE(21,6006)
         WRITE(21,7010) MBEEG,MBG,PND
         WRITE(21,7020) KSIX,K6,PPI
         WRITE(21,7010) IBEY,IBY,PSSI
         WRITE(21,7010) IBER,IBR,PSSI
         WRITE(21,7020) EL,L,FOOT
         WRITE(21,7020) ELTOO,L2,FOOT
```

```
C********************************
C** LATERAL (7-29-75)
C********************************
          WRITE(21,7020) DEE,D,FOOT
          WRITE(21,7020) UU,W,FOOT
          WRITE(21,7020) EEE,E,FOOT
          WRITE(21,7020) EE2,E2,FOOT
          WRITE(21,7020) BEE,B,FOOT
          WRITE(21,7020) CSICK,C6,PSPI
          WRITE(21,6007)
          WRITE(21,7020) KAYE,KA,IPPR
          WRITE(21,7020) KAGE,KG,PPI
          WRITE(21,7040) FSBEL,FSUBL,PND
          WRITE(21,7020) FTEE,FT,PND
          WRITE(21,7040) ALFA,ALPHA,DEG
          WRITE(21,7020) ROW,RO,FOOT
          WRITE(21,7022) IZERO
 6000     FORMAT(1H1,61X,12HINPUT VALUES///)
 6002     FORMAT(///66X,4HAXLE/)
 6005     FORMAT(1H ,///66X,5HTRUCK/)
 6006     FORMAT(1H ,///66X,8HCAR BODY/)
 6007     FORMAT(1H ,///66X,5HWHEEL/)
 6009     FORMAT(28X,16A5/)
 7010     FORMAT(1H ,57X,A5,A1,E12.4,3A5)
 7020     FORMAT(1H ,58X,A5,E12.4,3A5)
7021  FORMAT(59X,5H G = ,E12.4,10H IN/SEC**2)
7022  FORMAT(55X,9H IZERO = ,E12.4,14H IN-LBS-SEC**2)
7023  FORMAT(55X,9HLAMBDA = ,E12.4,5H FEET)
 7040     FORMAT(1H ,55X,A5,A3,E12.4,3A5)
 7050     FORMAT(1H ,55X,2A5,I5)
C
C     THESE THREE INPUTS ARE NEEDED AS ONE-HALF OF THE VALUE ENTERED
C     NOTE THAT 'LL' IN THE PRINTOUT CORRESPONDS TO 'L' IN THE
C     PROGRAM CODE AND '2L' CORRESPONDS TO 'L2'.
C
      B=B/2.
      HL=HL/2.
      L2=L2/2.
C
C     CONVERT  ALL INPUTS TO IN-LBS-SEC  UNITS
C
      HL=HL*12.0
      L=L*12.0
      L2=L2*12.0
      D=D*12.0
      W=W*12.0
      E=E*12.0
      E2=E2*12.0
      B=B*12.0
      RO=RO*12.0
```

```
C******************************
C** LATERAL (7-29-75)
C******************************
      IF(ILAMB .EQ. 1) V=V*17.6
      IF(ILAMB .NE. 1) V=V*12.
      ZEE=L/2.-D
C
C        CONVERT WEIGHTS TO MASSES
C
      MW=MWG/G
      MT=MTG/G
      MB=MBG/G
C
C        CALCULATE THE COMPONENTS OF THE 15X15 MASS(M),
C        DAMPING(C), AND STIFFNESS(K) MATRICES
C
      M(1)=MW
      M(17)=IW
      M(33)=MT
      M(49)=IT
      M(65)=MW
      M(81)=IW
      M(97)=MB
      M(113)=IBY
      M(129)=IBR
      M(145)=M(1)
      M(161)=M(17)
      M(177)=M(33)
      M(193)=M(49)
      M(209)=M(1)
      M(225)=IW
      K(1)=K1+KG
      K(2)=2.*FT*ALPHA*L2/RO
      K(3)=-K1
      K(4)=-K1*HL
      K(16)=-2.*FSUBL
      K(17)=K2-KA
      K(19)=-K2
      K(31)=-K1
      K(33)=2.*K1+K3
      K(35)=-K1
      K(37)=-K3
      K(38)=-K3*ZEE
      K(39)=K3*E
      K(46)=K(4)
      K(47)=-K2
      K(49)=2.*K2+K4+2.*K1*HL**2
      K(50)=K1*HL
      K(51)=-K2
      K(53)=-K4
```

```
C********************************
C** LATERAL (7-29-95)
C********************************
      K(63)=-K1
      K(64)=K1*HL
      K(65)=K1+KG
      K(66)=K(2)
      K(79)=-K2
      K(80)=K(16)
      K(81)=K2-KA
      K(93)=-K3
      K(97)=2.*K3
      K(99)=-2.*K3*E
      K(102)=-K3
      K(108)=K(38)
      K(109)=-K4
      K(113)=2.*K4+2.*K3*ZEE**2
      K(117)=-K(38)
      K(118)=-K4
      K(123)=K(39)
      K(127)=-2.*K3*E
      K(129)=4.*K6*B**2+2.*K3*E**2
      K(132)=K3*E
      K(145)=K1+KG
      K(146)=K(2)
      K(147)=-K1
      K(148)=-K1*HL
      K(160)=K(16)
      K(161)=K2-KA
      K(163)=-K2
      K(172)=-K3
      K(173)=-K(38)
      K(174)=K(39)
      K(175)=-K1
      K(177)=K(33)
      K(179)=-K1
      K(188)=-K4
      K(190)=K(4)
      K(191)=-K2
      K(193)=K(49)
      K(194)=-K(4)
      K(195)=-K2
      K(207)=-K1
      K(208)=-K(4)
      K(209)=K(1)
      K(210)=K(2)
      K(223)=-K2
      K(224)=K(16)
      K(225)=K2-KA
      C(1)=C1+2.*FSUBL/V
```

```
C******************************
C** LATERAL (7-29-75)
C******************************
      C(3)=-C1
      C(4)=-C1*HL
      C(17)=C2+2.*FT*L2**2/V
      C(19)=-C2
      C(31)=-C1
      C(33)=2.*C1+C3
      C(35)=-C1
      C(37)=-C3
      C(38)=-C3*ZEE
      C(39)=C3*E
      C(46)=C(4)
      C(47)=-C2
      C(49)=2.*C2+C4+2.*C1*HL**2
      C(50)=C1*HL
      C(51)=-C2
      C(53)=-C4
      C(63)=-C1
      C(64)=C1*HL
      C(65)=C(1)
      C(79)=-C2
        C(81)=C(17)
      C(93)=-C3
      C(97)=2.*C3
      C(99)=-2.*C3*E
      C(102)=-C3
      C(108)=-C3*ZEE
      C(109)=-C4
      C(113)=2.*C4+2.*C3*ZEE**2
      C(117)=C3*ZEE
      C(118)=-C4
      C(123)=C3*E
      C(127)=-2.*C3*E
      C(129)=4.*C6*B**2+2.*C3*E**2
      C(132)=C3*E
      C(145)=C(1)
      C(147)=-C1
      C(148)=-C1*HL
      C(161)=C(17)
      C(163)=-C2
      C(172)=-C3
      C(173)=C3*ZEE
      C(174)=C3*E
      C(175)=-C1
      C(177)=C(33)
      C(179)=-C1
      C(188)=-C4
      C(190)=-C1*HL
```

```
C*******************************
C** LATERAL (7-29-75)
C*******************************
      C(191)=-C2
      C(193)=C(49)
      C(194)=C1*HL
      C(195)=-C2
      C(207)=-C1
      C(208)=C1*HL
      C(209)=C(1)
      C(223)=-C2
      C(225)=C(17)
C
C     READ AND WRITE FREQUENCY INPUT
C
      READ(2,20) IFREQ,NDF
20    FORMAT(2I2)
      WRITE(21,21) IFREQ,NDF
21    FORMAT(///10X,6HIFREQ=,I2,5X,4HNDF=,I2)
      READ(2,30) DF
   30 FORMAT(7I4)
      WRITE(21,31) DF
   31 FORMAT(///10X,4HDF =,7I4)
      READ(2,40) FL
   40 FORMAT(8F10.4)
      WRITE(21,41) FL
   41 FORMAT(///10X,8HFL(HZ) =,10F10.4)
C
C     LIST REQUESTED OUTPUT
C
      WRITE(21,51)
51    FORMAT(////10X,16HREQUESTED OUTPUT,32X,5HPRINT,5X,4HPLOT/)
      DO 59 JJJ=1,15
      IF(DTAG(JJJ) .EQ. 0) GO TO 54
      WRITE(21,52) (NAME(III,JJJ),III=1,6)
52    FORMAT(/10X,15HDISPLACEMENT = ,6A5,5X,1HX)
      IF(DTAG(JJJ) .EQ. 2) WRITE(21,53)
53    FORMAT(1H+,69X,1HX)
54    CONTINUE
      IF(ATAG(JJJ) .EQ. 0) GO TO 59
      WRITE(21,55) (NAME(III,JJJ),III=1,6)
55    FORMAT(/10X,15HACCELERATION = ,6A5,5X,1HX)
      IF(ATAG(JJJ) .EQ. 2) WRITE(21,53)
59    CONTINUE
C
C     PRINT M,C, AND K MATRICES
C
      NUM=0
      KNT=0
      WRITE(21,9005)
```

```
C********************************
C** LATERAL (7-29-75)
C********************************
 9005 FORMAT(1H1)
 9010 NUM=NUM+1
      IF(KNT.GE.225) GO TO 9040
      WRITE(21,9020) NUM
 9020 FORMAT(//10X,11HCOLUMN NO. ,I2/)
 9025 KNT=KNT+1
      WRITE(21,9030) KNT,M(KNT),KNT,C(KNT),KNT,K(KNT)
 9030 FORMAT(/10X,3HM (,I3,2H)=,E12.4,
     2  5X,3HC (,I3,2H)=,E12.4,
     3  5X,3HK (,I3,2H)=,E12.4)
      IF(MOD(KNT,15).EQ.0) GO TO 9010
      GO TO 9025
 9040 CONTINUE
C
C     GENERATE FREQUENCY TABLE
C     AND STORE VALUES IN ARRAY F
C
      CALL FTABG(IFREQ,NDF,DF,FL,F,NF)
C
C     INITIALIZE MATRICES
C
      DO 80 JJ=1,15
      CMAT(JJ)=0.
      Q(JJ)=0.
   80 CONTINUE
C
C     COMPUTE CONSTANTS FOR FORCING VECTOR.
C     L2 IS HALF THE TRACK GAGE
C
      TWOL=1/(2.*L2)
      PI2=PIE*2.
      ZZ=2.*B*B
      HL2=2.*HL
      XL2D=L-2.*D
      Y1=EYE*XL2D*PI2
      Y12=Y1/2.
      Z1=EYE*(XL2D+HL2)*PI2
      X1=EYE*HL2*PI2
      X12=X1/2.
      K6X=ZZ*K6-E*E2*K3
      C6X=ZZ*C6-E*E2*C3
      ZEE=L/2.-D
C
C     DEFINE SOLUTION VECTOR NORMALIZATION
C
      DNORM(1)=1.0
      DNORM(2)=TWOL
```

```
C*******************************
C** LATERAL (7-29-75)
C*******************************
      DNORM(3)=57.29577
      DNORM(4)=1.0
      ANORM(1)=1.0/G
      ANORM(2)=TWOL/G
      ANORM(3)=1.0
      ANORM(4)=1.0/57.29577
      ICHECK=IOPT
      IOPT=1
      IF(ICHECK .GT. 1) IOPT=2
      XLAM=V
C
C        BEGIN FREQUENCY LOOP
C
   78 DO 1000 I=1,NF
C
C        OMEGA IS ANGULAR FREQUENCY
C
      OMEGA=2*PIE*F(I)
      IF(ILAMB .EQ. 1) XLAM=V/F(I)
      Y=Y1/XLAM
      IF(IOPT.NE.1) GO TO 150
C
C     DEFINE FORCING VECTOR FOR ALIGNMENT PERTURBATION
C
      X=X1/XLAM
      Z=Z1/XLAM
      XY=ALPHA*L2/RO
      Q(1)=KG
      Q(2)=2.*FT*XY
      Q(5)=KG*CEXP(X)
      Q(6)=Q(2)*CEXP(X)
      Q(10)=KG*CEXP(Y)
      Q(11)=Q(2)*CEXP(Y)
      Q(14)=KG*CEXP(Z)
      Q(15)=Q(2)*CEXP(Z)
      GO TO 160
150   CONTINUE
C
C     DEFINE FORCING VECTOR FOR CROSS-LEVEL PERTURBATION
C
      XA=X12/XLAM
      ZDUM=CEXP(Y)
      CEXP1=CEXP(XA)
      CEXP2=1./CEXP1
      CEXP3=ZDUM*CEXP2
      CEXP4=ZDUM*CEXP1
      CFAC2=RO*(EYE*OMEGA*C3+K3)
```

```
C********************************
C** LATERAL (7-29-75)
C********************************
      Q(3)=E2*(K3+EYE*OMEGA*C3)*(-1,0)
      X=1.+ZDUM
      Q(12)=Q(3)*ZDUM
      Q(7)=-Q(3)-Q(12)
      Z2=K6X+EYE*OMEGA*C6X
      Q(9)=Z2*X
      CTRM1=RO*(MW*OMEGA*OMEGA-EYE*C1*OMEGA-K1)
      CTRM2=RO*(K1+EYE*C1*OMEGA)
      CTRM3=IZERO*V/RO*(+EYE*OMEGA)
      Q(1)=CTRM1*CEXP2
      Q(2)=CTRM3*CEXP2
      Q(3)=Q(3)+CTRM2*(CEXP1+CEXP2)
      Q(4)=CTRM2*(CEXP2-CEXP1)*HL
      Q(5)=CTRM1*CEXP1
      Q(6)=CTRM3*CEXP1
      Q(8)=ZEE*E2*CFAC2/RO*(1.-ZDUM)
      Q(10)=CTRM1*CEXP3
      Q(11)=CTRM3*CEXP3
      Q(12)=Q(12)+CTRM2*(CEXP3+CEXP4)
      Q(13)=CTRM2*(CEXP3-CEXP4)*HL
      Q(14)=CTRM1*CEXP4
      Q(15)=CTRM3*CEXP4
      IF(ICHECK .EQ. 2) GO TO 160
C
C     FORCING FUNCTION FOR OPTION 3
C
      YA=Y12/XLAM
      CEXP5=CEXP(YA)
      OMEGA2=OMEGA*OMEGA
      CFAC1=RO*(EYE*OMEGA*C1+K1)
      CFAC3=OMEGA2*MT*RO
      CFAC4=(RO+E2)*CFAC2/RO
      CFAC5=OMEGA2*MB*(RO+E+E2)
      CFAC6=IBR*OMEGA2
      CFAC7=4*B*B*(K6+EYE*OMEGA*C6)
      Q(1)=Q(1)+CFAC1
      Q(3)=Q(3)+CFAC3-2.*CFAC1+CFAC4*CEXP5-CFAC2
      Q(5)=Q(5)+CFAC1
      Q(7)=Q(7)+CFAC5*CEXP5+CFAC2*(1.+ZDUM-2.*CEXP5)
     X-CFAC2*E2*2.*CEXP5/RO
      Q(8)=Q(8)+CFAC2*(1.-ZDUM)*ZEE
      Q(9)=Q(9)+E*E2*CFAC2*(1.+ZDUM)/RO+(CFAC6-CFAC7)
     X*CEXP5+E*(RO+E)*CFAC2*(2.*CEXP5-1.-ZDUM)/RO
      Q(10)=Q(10)+CFAC1*ZDUM
      Q(12)=Q(12)+(CFAC3-2.*CFAC1)*ZDUM+CFAC4
     X*CEXP5-CFAC2*ZDUM
      Q(14)=Q(14)+CFAC1*ZDUM
```

```
C********************************
C** LATERAL (7-29-75)
C********************************
      IF(ICHECK .EQ. 3) GO TO 160
C
C     FORCING FUNCTION FOR OPTION 4
C
      DO 151 LM=1,6
151   Q(LM)=(0.0,0.0)
      DO 155 LM=10,15
155   Q(LM)=(0.0,0.0)
  160 DO 170 LM=1,15
      CMAT(LM)=Q(LM)
  170 CONTINUE
C
C       FORM COMPLEX COEFFICIENT MATRIX
C
      DO 100 MM=1,225
      MAT(MM)=-OMEGA**2*M(MM)+EYE*OMEGA*C(MM)+K(MM)
      BMAT(MM)=MAT(MM)
  100 CONTINUE
C
C       INVERT COEFFICIENT MATRIX
C
      CALL MINV(BMAT,15,DET,LDUM,MDUM)
      IF(CABS(DET)) 200,1100,200
  200 CONTINUE
C
C       MULTIPLY INVERSE OF COEFFICIENT MATRIX BY FORCING
C       FUNCTION TO OBTAIN SOLUTION VECTOR
C
      CALL GMPRD(BMAT,CMAT,SOLU,15,15,1)
C
C     ADJUST SOLUTION VECTOR FOR OPTIONS 3 AND 4
C
      IF(ICHECK .LT. 3) GO TO 210
      SOLU(3)=SOLU(3)+RO
      SOLU(12)=SOLU(12)+RO*(ZDUM)
      SOLU(9)=SOLU(9)+CEXP5
      SOLU(7)=SOLU(7)+CEXP5*(RO+E+E2)
C
C     SAVE MAGNITUDE OF SOLUTION FOR PRINTING AND PLOTTING
C     NORMALIZE OUTPUT VARIABLES
C
210   CONST9=OMEGA*OMEGA
      DO 225 JJJ=1,15
      SOLD(I,JJJ)=SOLU(JJJ)*DNORM(IOPT+LATANG(JJJ))
      DISPL(I,JJJ)=CABS(SOLD(I,JJJ))
      ACCEL(I,JJJ)=CABS(SOLU(JJJ))*CONST9*ANORM(IOPT+LATANG(JJJ))
225   CONTINUE
```

```
C********************************
C** LATERAL (7-29-75)
C********************************
1000  CONTINUE
C
C     PRINT THE MAGNITUDE OF THE COMPLEX SOLUTIONS
C     -GENERALIZED COORDINATES
C
      JNUM=0
      DO 1305 J=1,15
      IF(DTAG(J) .EQ. 0) GO TO 1305
      JNUM=JNUM+1
      JHOLD(JNUM)=J
1305  CONTINUE
      IF(JNUM .EQ. 0) GO TO 1500
      DO 1450 J=1,JNUM,3
      J1=JHOLD(J)
      J2=JHOLD(J+1)
      J3=JHOLD(J+2)
      IF(J+1 .GT. JNUM) J2=0
      IF(J+2 .GT. JNUM) J3=0
      IF(IOPT .EQ.1) WRITE(21,1310)
1310  FORMAT(1H1,13X,66HDISPLACEMENT AND ANGULAR RESPONSES
     X BASED ON ALIGNMENT PERTURBATION//)
      IF(IOPT .EQ. 2) WRITE(21,1320)
1320  FORMAT(1H1,12X,68HDISPLACEMENT AND ANGULAR RESPONSES
     X BASED ON CROSS-LEVEL PERTURBATION//)
      JL1=IOPT+LATANG(J1)
      IF(J2 .NE. 0) JL2=IOPT+LATANG(J2)
      IF(J3 .NE. 0) JL3=IOPT+LATANG(J3)
      WRITE(21,1330) MAME(J1),KLAB(1,JL1),KLAB(2,JL1)
1330  FORMAT(9X,14HFREQUENCY (HZ),7X,3A5)
      IF(J2 .NE. 0) WRITE(21,1340) MAME(J2),KLAB(1,JL2),KLAB(2,JL2)
1340  FORMAT(1H+,49X,3A5)
      IF(J3 .NE. 0) WRITE(21,1350) MAME(J3),KLAB(1,JL3),KLAB(2,JL3)
1350  FORMAT(1H+,69X,3A5)
      WRITE(21,1351) (JLABD(JKL,JL1),JKL=1,3)
1351  FORMAT(30X,3A5)
      IF(J2 .NE. 0) WRITE(21,1340) (JLABD(JKL,JL2),JKL=1,3)
      IF(J3 .NE. 0) WRITE(21,1350) (JLABD(JKL,JL3),JKL=1,3)
      WRITE(21,1355)
1355  FORMAT(5X)
      DO 1400 I=1,NF
      WRITE(21,1360) F(I),DISPL(I,J1)
1360  FORMAT(1X,2E20.5)
      IF(J2 .NE. 0) WRITE(21,1370) DISPL(I,J2)
1370  FORMAT(1H+,40X,E20.5)
      IF(J3 .NE. 0) WRITE(21,1380) DISPL(I,J3)
1380  FORMAT(1H+,60X,E20.5)
1400  CONTINUE
```

```
C********************************
C** LATERAL (7-29-75)
C********************************
1450  CONTINUE
C
C     PRINT THE ACCELERATION RESPONSE
C
1500  JNUM=0
      DO 1505 J=1,15
      IF(ATAG(J) .EQ. 0) GO TO 1505
      JNUM=JNUM+1
      JHOLD(JNUM)=J
1505  CONTINUE
      IF(JNUM .EQ. 0) GO TO 1700
      DO 1650 J=1,JNUM,3
      J1=JHOLD(J)
      J2=JHOLD(J+1)
      J3=JHOLD(J+2)
      IF(J+1 .GT. JNUM) J2=0
      IF(J+2 .GT. JNUM) J3=0
      IF(IOPT .EQ. 1) WRITE(21,1510)
1510  FORMAT(1H1,20X,53HACCELERATION RESPONSE BASED ON
     XALIGNMENT PERTURBATION//)
      IF(IOPT .EQ.2) WRITE(21,1520)
1520  FORMAT(1H1,19X,55HACCELERATION RESPONSE BASED ON
     XCROSS-LEVEL PERTURBATION//)
      JL1=IOPT+LATANG(J1)
      IF(J2 .NE. 0) JL2=IOPT+LATANG(J2)
      IF(J3 .NE. 0) JL3=IOPT+LATANG(J3)
      WRITE(21,1530) MAME(J1),KLAB(1,JL1),KLAB(2,JL1)
1530  FORMAT(9X,14HFREQUENCY (HZ),6X,1HA,3A5)
      IF(J2 .NE. 0) WRITE(21,1540) MAME(J2),KLAB(1,JL2),KLAB(2,JL2)
1540  FORMAT(1H+,48X,1HA,3A5)
      IF(J3 .NE. 0) WRITE(21,1550) MAME(J3),KLAB(1,JL3),KLAB(2,JL3)
1550  FORMAT(1H+,68X,1HA,3A5)
      WRITE(21,1331) (JLABA(JKL,JL1),JKL=1,3)
      IF(J2 .NE. 0) WRITE(21,1340) (JLABA(JKL,JL2),JKL=1,3)
      IF(J3 .NE. 0) WRITE(21,1350) (JLABA(JKL,JL3),JKL=1,3)
      WRITE(21,1355)
      DO 1600 I=1,NF
      WRITE(21,1360) F(I),ACCEL(I,J1)
      IF(J2 .NE. 0) WRITE(21,1370) ACCEL(I,J2)
      IF(J3 .NE. 0) WRITE(21,1380) ACCEL(I,J3)
1600  CONTINUE
1650  CONTINUE
C
C     CALL LATERP FOR REQUESTED PLOTS
C     OF GENERALIZED COORDINATES
C
1700  II=1
```

```
C*********************************
C** LATERAL (7-29-75)
C*********************************
      DO 2100 J=1,15
      IF(DTAG (J) .NE. 2) GO TO 2050
      IF(II .EQ. 0) GO TO 2010
      II=0
      CALL INIPLT(1,1)
2010  DO 2020 N=1,NF
2020  ARRAA(N)=DISPL(N,J)
      DO 2030 N=1,6
2030  NAME1(N)=NAME(N,J)
      NAME2(1)=BLANK
      NAME2(2)=MAME(J)
      NCHOZ=IOPT+LATANG(J)
      DO 2047 N=1,4
2047  NAME2(N+2)=ILABD(N,NCHOZ)
      CALL LATERP(NF,F,ARRAA,NAME1,INAME(J),NAME2,30,1,IOPT)
C
C     CALL LATERP FOR REQUESTED ACCELERATION PLOTS
C
2050  IF(ATAG(J) .NE.2) GO TO 2100
      IF(II .EQ. 0) GO TO 2060
      II=0
      CALL INIPLT(1,1)
2060  DO 2070 N=1,NF
2070  ARRAA(N)=ACCEL(N,J)
      DO 2080 N=1,6
2080  NAME1(N)=NAME(N,J)
      NAME2(1)=BLANKA
      NAME2(2)=MAME(J)
      NCHOZ=IOPT+LATANG(J)
      DO 2097 N=1,4
2097  NAME2(N+2)=ILABA(N,NCHOZ)
      CALL LATERP(NF,F,ARRAA,NAME1,INAME(J),NAME2,30,2,IOPT)
2100  CONTINUE
C
C     TEST FOR PSD PLOTS
C
      IF(ILAMB .NE. 1) GO TO 2190
      IPSD=0
      READ(2,2110,END=2115) PSD
2110  FORMAT(A5)
      IF(PSD .EQ. 'RAILP') IPSD=1
2115  CONTINUE
      IF(IPSD .EQ. 0) GO TO 2190
      IF(II .EQ. 0) GO TO 2130
      II=0
      CALL INIPLT(1,1)
2130  CCCC=57.29577*TWOL
```

```
C*******************************
C** LATERAL (7-29-75)
C*******************************
      IF(IOPT .NE. 2) GO TO 2125
      DO 2121 J=1,15
      IF(LATANG(J) .NE. 2) GO TO 2121
      DO 2120 N=1,NF
      DISPL(N,J)=DISPL(N,J)*CCCC
2120  ACCEL(N,J)=ACCEL(N,J)*CCCC
2121  CONTINUE
2125  CONTINUE
      CALL RAILPL(F,V,NF,30,IITYPE,SAVEM,200,2,21)
2190  IF(II .EQ. 1) GO TO 2200
      CALL SYMBOL(0.0,0.0,0.125,RUN,90.,80)
      CALL FINPLT
2200  STOP
 1100 WRITE(21,1200)
 1200 FORMAT(1H1,24H MATRIX AMAT IS SINGULAR)
      STOP
      END
```

```
C********************************
C** LATERAL (7-29-75)
C********************************
C
C
C
      SUBROUTINE LATERP(NEM,SAV,ARRAY,TITLE,NCT,LABL,NCL,
     XICODE,IOPT)
C
C     PLOTTING ROUTINE FOR PROGRAM LATERAL
C
C     NEM      NUMBER OF POINTS
C     SAV      FREQUENCY ARRAY
C     ARRAY    DEPENDENT VARIABLE ARRAY
C     TITLE    LABEL FOR TOP OF PLOT
C     NCT      NUMBER OF CHARACTERS IN TITLE
C     LABL     LABEL FOR Y-AXIS
C     NCL      NUMBER OF CHARACTERS IN LABL
C     ICODE     =1 DISPLACEMENT PLOT
C               =2 ACCELERATION PLOT
C     IOPT      =1 SURFACE PERTURBATION
C               =2 ALIGNMENT PERTURBATION
C
      DIMENSION ARRAY(200),SAV(200),TITLE(20),LABL(20),
     XFEQ(200),PLTFIL(200)
      XAXIS=6.0
      YAXIS=6.0
C
C     PLOT ONLY THE FIRST 6 SIGNIFICANT DECADES
C
      CALL MAXMIN(ARRAY,NEM,DMAX,DMIN)
      CUT=DMAX*1.0E-05
      DO 10 I=1,NEM
      FEQ(I)=SAV(I)
      PLTFIL(I)=ARRAY(I)
      IF(PLTFIL(I) .LT. CUT) PLTFIL(I)=CUT
      IF(PLTFIL(I) .LT. 1.1E-03) PLTFIL(I)=1.1E-03
      IF(PLTFIL(I) .GT. 9.0) PLTFIL(I)=9.0
10    CONTINUE
      L=NEM
C
C     PLOTTING DONE BELOW
C
      CALL LOGSCA(FEQ,L,XAXIS,IXVAL,IX)
      CALL LOGSCA(PLTFIL,L,YAXIS,IYVAL,IY)
      CALL LAXIS(0.,0.,IY,LABL,NCL,YAXIS,IYVAL,1)
      CALL LAXIS(0.,0.,IX,14HFREQUENCY (HZ),14,XAXIS,IXVAL,2)
      CALL SYMBOL(0.5*(XAXIS-(NCT*0.15)),YAXIS+0.25,.15,TITLE,
     X0.,NCT)
      IF(IOPT .EQ. 1) CALL SYMBOL(0.45,YAXIS+0.55,.15,
```

```
C******************************
C** LATERAL (7-29-75)
C******************************
     X23HALIGNMENT PERTURBATION-,0.,23)
      IF(IOPT .EQ. 2) CALL SYMBOL(0.3,YAXIS+0.55,.15,
     X25HCROSS-LEVEL PERTURBATION-,0.,25)
      IF(ICODE .EQ. 1) CALL SYMBOL(999.,999.,.15,
     X11HRESPONSE OF,0.,11)
      IF(ICODE .EQ. 2) CALL SYMBOL(999.,999.,.15,
     X13HACCEL.RESP.OF,0.,13)
      CALL PLOT(FEQ(1),PLTFIL(1),3)
      DO 30 K=2,L
30    CALL PLOT(FEQ(K),PLTFIL(K),2)
      CALL GRID(0.,0.,XAXIS,YAXIS,IX,IY,1,0,D,D,D,D)
      CALL PLOT(12.,0.,-3)
      RETURN
      END
```

```
C******************************
C** LATERAL (7-29-75)
C******************************
      SUBROUTINE FTABG(IFREQ,NDF,DF,FL,F,NF)
C
C     THIS SUBROUTINE COMPUTES THE FREQUENCY TABLE
C
C     IFREQ=1 RANGE INPUTS, =2 OCTAVE INPUTS
C     NDF   - NUMBER OF RANGES OR OCTAVES
C     DF    - FREQUENCY POINTS VECTOR
C     FL    - FREQUENCY LOWER LIMITS VECTOR
C     F     - FREQUENCY TABLE RETURNED
C     NF    - NUMBER OF POINTS IN FREQUENCY TABLE
C
      INTEGER DF(7)
      DIMENSION FL(8),F(200)
      IF(IFREQ .EQ. 2) GO TO 500
C
C     RANGE INPUTS
C
      NF=0
      DO 200 I=1,NDF
      C=EXP(ALOG(FL(I+1)/FL(I))/DF(I))
      NF=NF+1
      F(NF)=FL(I)
      DO 100 J=1,DF(I)
      FT=C*F(NF)
      IF(FT .GE. FL(I+1)-0.000001) GO TO 200
      NF=NF+1
      F(NF)=FT
100   CONTINUE
200   CONTINUE
      RETURN
C
C     OCTAVE INPUTS
C
500   C=2**(1./DF(1))
      NF=1
      F(NF)=FL(1)
      DO 700 I=1,NDF
      DO 700 J=1,DF(1)
      NF=NF+1
      F(NF)=F(NF-1)*C
700   CONTINUE
      RETURN
      END
```

APPENDIX D
PROGRAM HALF INPUT PARAMETERS AND LISTINGS

The Program HALF input deck consists of the following eighteen cards:

Card 1 — UNITS: LBS/IN^2

$\dfrac{|F10.2|}{k}$

k = Track foundation stiffness per unit track length

Card 2 — UNITS: LBS/IN

$\dfrac{|F10.2|}{K_1}$

K_1 = Secondary suspension vertical spring constant

Card 3 — UNITS: LBS/IN

$\dfrac{|F10.2|}{K_2}$

K_2 = Primary suspension vertical spring constant

Card 4 — UNITS: LBS

$\dfrac{|F10.2|}{W_1}$

W_1 = One-half car body weight

Card 5 — UNITS: LBS

$\dfrac{|F10.2|}{W_2}$

W_2 = Truck frame assembly weight

Card 6 UNITS: LBS

|F10.2|
W

W = Wheelset weight

Card 7 UNITS: LBS/YD

|F10.2|
ρ_t

ρ_t = Single rail weight per unit length

Card 8 UNITS: $INCHES^4$

|F10.2|
I

I = Single rail area moment of inertia

Card 9 UNITS: LBS/IN^2

|F10.7|
E

Young's Modulars for steel rail

Card 10 UNITS: INCHES/CYCLE
IN/SEC

|F10.2|4x|I1|
λ IFV = 1 (Column 15)
V IFV = 2 (Column 15)

IFV specifies that a constant track wavelength (IFV=1) or a constant vehicle velocity (IFV=2) is to be input. Since $V=f*\lambda$, for a fixed velocity, the wavelength must be adjusted in order to sweep frequency. For a fixed wavelength, the velocity must be adjusted.

Card 11 UNITS: INCHES

|F10.2|
ℓ

ℓ = Truck wheel base

Card 12 UNITS: NONE

|F10.2|
β

β = Vehicle (secondary suspension) damping ratio

Card 13 UNITS: NONE

|F10.2|
ξ

ξ = Track structure damping ratio

Card 14 UNITS: LBS

|F10.2|
W_t

W_t = Weight of a single rail tie

Card 15 UNITS: INCHES

|F10.2|
L_t

L_t = Distance between rail ties

Cards 16, 17, 18 (Frequency Control Variables)

(16) UNITS: NONE

| I2 | I2 |
IFREQ NDF

(17) UNITS: NONE

|7I4|
DF(N),N=1,NDF FOR IFREQ = 1
DF(1) FOR IFREQ = 2

(18) UNITS: HERTZ

```
|8F10.4|
 FL(N) N=1, NDF+1    FOR IFREQ = 1
 FL(1)               FOR IFREQ = 2
```

IFREQ selects one of two methods for specifying the frequency points at which the response will be found. With IFREQ = 1, logarithmically evenly-space points are generated for a set of frequency ranges. NDF gives the number of requency ranges (maximum of seven). DF(N) specifies the number of points for the range FL(N) to FL(N+1). FL(N) gives the lower limit of the Nth frequency range, and FL(N+1) is the upper limit for the Nth range. FL(NDF+1) gives the upper limit for the last frequency range.

With IFREQ = 2, logarithmically evenly spaced points are generated for a set of octaves. NDF specifies the number of octaves. DF(1) gives the number of points per octave, and FL(1) gives the starting frequency of the first octave. Note that the remainders of the DF and FL fields are left blank.

```
C*****************************
C**  HALF (7-29-75)
C*****************************
C** FOLLOWING TEXT PRINTED FROM FILE DSKB:HALFPS.F4 [4351,533] 29-JUL-75
C*****************************
      DIMENSION SAVEM(100,6),IITYPE(6)
      DATA IITYPE/6*1/
      COMPLEX BCOMP(64),BBCOMP(64)
      COMPLEX BLI,BLK,BLJ,BLN,BLM
      COMPLEX ACOMP(64),CCOMP(8),SOLU(8)
      DATA ACOMP/(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(1.,Ø.),(Ø.,Ø.)
     1,(Ø.,Ø.),
     2(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(1.,Ø.),
     3(1.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),
     4(Ø.,Ø.),(1.,Ø.) ,(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),
     5(-1.,Ø.),(Ø.,Ø.),(-1.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),
     6(Ø.,Ø.),(-1.,Ø.),(Ø.,Ø.),(-1.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),
     7(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(1.,Ø.),(Ø.,Ø.),( 1.,Ø.),(Ø.,Ø.),
     8(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(Ø.,Ø.),(-1.,Ø.),(Ø.,Ø.),( 1.,Ø.),(Ø.,Ø.)/
      DATA CCOMP /(1.,Ø.),7*(Ø.,Ø.)/
      DIMENSION LDUM(8),MDUM(8)
      COMPLEX DET
      INTEGER DF(7)
      DIMENSION FL(8)
      DIMENSION F(200)
      DIMENSION AF1(100),TF1(100),AF2(100),TF2(100)
      DIMENSION ADEL1(100),TDEL1(100),ADEL2(100),TDEL2(100)
      DIMENSION AY1(100),TY1(100),AY2(100),TY2(100)
      DIMENSION AY3(100),TY3(100),AY4(100),TY4(100)
      DIMENSION AG1(100),TG1(100),AG2(100),TG2(100)
      COMPLEX F1(100),F2(100),DEL1(100),DEL2(100)
      COMPLEX Y1(100),Y2(100),Y3(100),Y4(100)
      COMPLEX G1(100),G2(100)
      COMPLEX B1,B2,B3,B6,B7
      COMPLEX A1,A2,A3,A4
      COMPLEX A5
      COMPLEX X,X1,X2,E1,E2,E11,E22,E12
      COMPLEX GB,GC
      EQUIVALENCE(SAVEM(1,1),AY1),(SAVEM(1,2),AY2),(SAVEM(1,3),AY3),
     X(SAVEM(1,4),AY4),(SAVEM(1,5),ADEL1),(SAVEM(1,6),ADEL2)
        CALL ASSDEV(20,'DSK')
        CALL ASSDEV(16,'DSK')
        CALL ASSDEV(6,'DSK')
      CALL PLOTS(IBUF,360,16)
      CALL PLOT(Ø.,-11.,-3)
      CALL PLOT(Ø.,1.1,-3)
      SQ2=SQRT(2.)
      G=386.
      PI=3.14159265
      EPSW=.Ø1
```

```
C***************************
C**  HALF (7-27-75)
C****************************
C   READ INPUT VARIABLES
      READ (20,9) XK
      READ (20,9) XK1
      READ (20,9) XK2
      READ (20,9) XM1
      READ (20,9) XM2
      READ (20,9) XM
      READ (20,9) WT
      READ (20,9) XI
      READ (20,7) E
      READ (20,11) XX,JA
      READ (20,9) XL
      READ (20,9) BET
      READ(20,9) ZETA
      READ (20,9) WTIE
      READ (20,9) XLTIE
    7 FORMAT (E10.7)
    9 FORMAT (F10.2)
   11 FORMAT (F10.2,4X,I1)
C    WRITE INPUT VARIABLES
      WRITE (6,10)
   10 FORMAT (1H0,15HINPUT VARIABLES)
      WRITE (6,12) XK
   12 FORMAT (1H0,5X,4HK = ,E15.8,10H  LB/IN**2)
      WRITE (6,13) XK1
   13 FORMAT (1H0,4X,5HK1 = ,E15.8, 7H  LB/IN)
      WRITE (6,14) XK2
   14 FORMAT (1H0,4X,5HK2 = ,E15.8, 7H  LB/IN)
      WRITE (6,15) XM1
   15 FORMAT (1H0,4X,5HM1 = ,E15.8, 4H  LB)
      WRITE (6,16) XM2
   16 FORMAT (1H0,4X,5HM2 = ,E15.8, 4H  LB)
      WRITE (6,17) XM
   17 FORMAT (1H0,5X,4HM = ,E15.8, 4H  LB)
      WRITE (6,18) WT
   18 FORMAT (1H0,5X,4HW = ,E15.8, 7H  LB/YD)
      WRITE (6,22) E
   22 FORMAT (1H0 5X,4HE = ,E15.8, 10H  LB/IN**2)
      WRITE (6,23) XI
   23 FORMAT (1H0,5X,4HI = ,E15.8, 7H  IN**4)
      WRITE (6,24) BET
   24 FORMAT (1H0,2X,7HBETA = ,E15.8, 17H  NON-DIMENSIONAL)
      WRITE(6,122) ZETA
  122 FORMAT(1H0,2X,7HZETA = ,E15.8, 17H  NON-DIMENSIONAL)
      WRITE (6,27) XL
   27 FORMAT (1H0,5X,4HL = ,E15.8, 4H  IN)
      WRITE (6,28) WTIE
```

```
C********************************
C**  HALF (7-2?-75)
C********************************
   28 FORMAT (1H0,4X,5HWT = ,E15.8,4H  LB)
      WRITE (6,29) XLTIE
   29 FORMAT (1H0,4X,5HLT = ,E15.8,4H  IN)
      IF (JA.EQ.1) GO TO 73
   73 CONTINUE
      IF(JA.EQ.1) GO TO 31
      IF(JA.EQ.2) GO TO 32
      IF((JA.NE.1).AND.(JA.NE.2))GO TO 6
   32 WRITE (6,21) XX
   21 FORMAT (1H0,5X,4HV = ,E15.8,5H  MPH)
      GO TO 34
   31 WRITE (6,19) XX
   19 FORMAT (1H0,9HLAMBDA = ,E15.8,5H FEET)
      GO TO 34
    6 WRITE (6,8)
    8 FORMAT (2X,32HERROR ON V/LAMBDA IDENTIFICATION)
   34 CONTINUE

C  CONVERSIONS AND COMPUTEDCONSTANTS
      IF (JA.EQ.1) XX=XX*12.
      IF (JA.EQ.2) XX=XX*17.6
      XM2=XM2/386.
      XM1=XM1/386.
      XM=XM/386.
      WT=WT/36.
      WT=2.*WT+WTIE/XLTIE
      XI=2.*XI
      XLL=XL
      WA=SQRT(XK*G/WT)
      XLAMO=2.*PI*(4.*E*XI/XK)**.25
      W1=SQRT(XK1/XM1)

C  GENERATE FREQUENCY TABLE
      READ(20,20) IFREQ,NDF
   20 FORMAT (2I2)
      READ (20,30) DF
   30 FORMAT (7I4)
      READ (20,40) FL
   40 FORMAT (8F10.4)
      CALL FTABG(IFREQ,NDF,DF,FL,F,NF)
C    DO LOOP TO COMPUTE FUNCTIONAL VALUES
 1000 FORMAT (1H0,2(E15.8,10X))
      DO 80 I=1,NF
      KK=I
      IF(JA.EQ.1)XLAM=XX
      IF(JA.EQ.2)XLAM=XX/F(I)
    5 V=XLAM*F(KK)
```

```
C******************************
C**  HALF (7-29-75)
C******************************
      W=2.*PI*F(KK)
      WW=W-WA
      AWW=ABS(WW)
      IF (AWW.LT.EPSW) GO TO 25
      IF(WW.GT.0.0) GO TO 3
      GO TO 2
   25 WRITE (6,26)
   26 FORMAT(3X,5HERROR,5X,4HW=WA)
C     COMPUTE G11, G1I FOR W LESS THAN WA
    2 WB=W/WA
      DA=1.-WB*WB
      DB=2.*ZETA*WB
      PHI=ATAN2 (DB,DA)
      DC=(DA*DA+DB*DB)**0.125
      BETA1=2.*PI*DC/XLAMO
      ZA=PI/(XK*XLAMO*DC**3)
      ZB=COS(3.*PHI/4.)
      ZB=ZB*ZA
      ZC=SIN(3.*PHI/4.  )
      ZC=(-1.)*ZC*ZA
      G1(I)=CMPLX(ZB,ZC)
      G1(I)= -G1(I)
      X=CMPLX(0.,-0.25*PI)
      X1=CMPLX(0.,0.75*PI+0.25*PHI)
      X2=CMPLX(0.,1.25*PI+0.25*PHI)
      EC=SQ2*BETA1*XLL
      E1=CEXP(X1)*EC
      E2=CEXP(X2)*EC
      E11=CEXP(E1)
      E22=CEXP(E2)*(0.,1.)
      E12R=REAL(E11)+REAL(E22)
      E12I=AIMAG(E11)+AIMAG(E22)
      E12=CMPLX(E12R,E12I)
      G2(I)=G1(I)*E12*CEXP(X)/SQ2
      GO TO 4
C     COMPUTE G11, G12 FOR W GREATER THAN WA
    3 WB=W/WA
      DA=WB*WB-1.
      DB=2.*ZETA*WB
      PHI=ATAN2(DB,DA)
      DC=(DA*DA+DB*DB)**0.125
      BETA1=2.*PI*DC/XLAMO
      GA=(-1.)*PI/(SQ2*XK*XLAMO*DC**3)
      PHIN=3.*PHI/4.
      DE=COS(PHIN)
       DFF=SIN(PHIN)
      GB=CMPLX(DE,DFF)
```

```
C*******************************
C**  HALF (7-29-75)
C*******************************
      DFF=-DFF
      GC=CMPLX(DFF,DE)
      G1(I)=GA*(GB+GC)
      G1(I)=-G1(I)
      X1=CMPLX(Ø.,(4.*PI-PHI)/4.)
      X2=CMPLX(Ø.,(6.*PI-PHI)/4.)
      EC=SQ2*BETA1*XLL
      E1=CEXP(X1)*EC
      E2=CEXP(X2)*EC
      E11=CEXP(E1)
      E22=CEXP(E2)*(Ø.,1.)
      E12R=REAL(E11)+REAL(E22)
      E12I=AIMAG(E11)+AIMAG(E22)
      E12=CMPLX(E12R,E12I)
      G2(I)=GA*GB*E12
      G2(I)=-G2(I)
    4 D7=2.*BET*W/W1
      D8=1.-(W/W1)**2
      C1=(D8+D7**2)/(D8**2+D7**2)
      C2=(D7*D8-D7)/(D8**2+D7**2)
      B1=CMPLX(C1,C2)
      CS1=COS(2.*PI*XL/XLAM)
      CS2=SIN(2.*PI*XL/XLAM)
      A5=CMPLX(CS1,CS2)
      XMDUM=XM1*W**2
      ACOMP(7)=CMPLX(XMDUM ,Ø.)
      ACOMP(8)=CMPLX(-XMDUM,Ø.)
      ACOMP(14)=-B1
      XMDUM= XM2*W**2
      ACOMP(15)=CMPLX ( XMDUM,Ø.)
      XMDUM= XK2-XM2*W**2
      ACOMP(16)= CMPLX (XMDUM,Ø.)
      XMDUM=XM*W**2
      ACOMP(23)=CMPLX(XMDUM,Ø.)
      XMDUM=XK2/2.
      ACOMP(24)=CMPLX(-XMDUM,Ø.)
      XMDUM=XM*W**2
      ACOMP(21)=CMPLX (XMDUM,Ø.)
      ACOMP(29)=CMPLX(-XMDUM,Ø.)
      ACOMP(31)=ACOMP(23)
      ACOMP(32)=ACOMP(24)
      ACOMP(51)=G1(I)
      ACOMP(52)=G2(I)
      ACOMP(59)=ACOMP(52)
      ACOMP(60)=ACOMP(51)
      DO 201 MM=1,64
  201 BCOMP(MM)=ACOMP(MM)
```

```
C*****************************
C**  HALF (7-29-75)
C*****************************
      CCOMP(2)=A5
       CALL MINV(BCOMP,8,DET,LDUM,MDUM)
      IF (CABS(DET)) 500,501,500
  500 CONTINUE
      CALL GMPRD(BCOMP,CCOMP,SOLU,8,8,1)
  401 FORMAT (1H0,8E15.8)
      Y1(I)=SOLU(1)
      Y2(I)=SOLU(2)
      Y3(I)=SOLU(3)
      Y4(I)=SOLU(4)
      DEL1(I)=SOLU(5)
      DEL2(I)=SOLU(6)
      F1(I)=SOLU(7)
      F2(I)=SOLU(8)
      BLI=Y3(I)-DEL1(I)-(1.,0.)
      BLK=Y4(I)-DEL2(I)-A5
      BLJ=G1(I)*F1(I)+G2(I)*F2(I)-DEL1(I)
      BLM=G1(I)*F2(I)+G2(I)*F1(I)-DEL2(I)
      BLN=XK2*(Y2(I)-(Y3(I)+Y4(I))/2.)-(XM2*Y2(I)+XM1*Y1(I))*W**2
      IPRINT=0
C     REMOVE \C\ FROM COMMENT COLUMN IN NEXT STATEMENT
C     IF FOLLOWING PRINTOUTS DESIRED.
C     IPRINT=1
      IF (IPRINT.EQ.0) GO TO 782
      WRITE (6,1000) G1(I)
      WRITE (6,1000) G2(I)
      WRITE(6,401) D7
      WRITE(6,401) D8
      WRITE(6,401) C1
      WRITE(6,401) C2
      WRITE(6,401) B1
      WRITE(6,401) CS1
      WRITE(6,401) CS2
      WRITE (6,901) F(I)
  901 FORMAT (1X //55X,5HFREQ=,F7.2/)
      WRITE (6,10000) BLI,BLK,BLJ
      WRITE (6,10000) BLM,BLN
  782 CONTINUE
10000 FORMAT (1H0,10X,2E15.8,10X,2E15.8,10X,2E15.8)
   80 CONTINUE
C    CONVERT COMPLEX NOS, TO MAGNITUDE-PHASE FORM
      DO 150 I=1,NF
      CALL POLAR(F1(I),AF1(I),TF1(I))
      CALL POLAR(F2(I),AF2(I),TF2(I))
      CALL POLAR(DEL1(I),ADEL1(I),TDEL1(I))
      CALL POLAR(DEL2(I),ADEL2(I),TDEL2(I))
      CALL POLAR(Y1(I),AY1(I),TY1(I))
```

```
C*******************************
C**  HALF (7-29-75)
C*******************************
      CALL POLAR(Y2(I),AY2(I),TY2(I))
      CALL POLAR(Y3(I),AY3(I),TY3(I))
      CALL POLAR(Y4(I),AY4(I),TY4(I))
      CALL POLAR(G1(I),AG1(I),TG1(I))
      CALL POLAR(G2(I),AG2(I),TG2(I))
  150 CONTINUE
C     PRINTED OUTPUT
      WRITE (6,102)
  102 FORMAT(1H0,2X,4HFREQ,5X,10HMAGN F1/VO,4X,9HPHS F1/VO,4X,
     110HMAGN F2/VO,4X,9HPHS F2/VO)
  105 FORMAT (1H0,F6.2,3X,2(E13.6,3X,F7.2,5X))
      DO 110 I=1,NF
      WRITE (6,105) F(I),AF1(I),TF1(I),AF2(I),TF2(I)
  110 CONTINUE
      WRITE (6,103)
  103 FORMAT(1H0,2X,4HFREQ,5X,12HMAGN DEL1/VO,2X,11HPHS DEL1/VO,
     12X,12HMAGN DEL2/VO,2X,11HPHS DEL2/VO)
      DO 111 I=1,NF
      WRITE (6,105) F(I),ADEL1(I),TDEL1(I),ADEL2(I),TDEL2(I)
  111 CONTINUE
      WRITE (6,107)
  107 FORMAT(1H0,2X,4HFREQ,5X,10HMAGN Y1/VO,4X,9HPHS Y1/VO,4X,
     110HMAGN Y2/VO,4X,9HPHS Y1/VO)
      DO 112 I=1,NF
      WRITE (6,105) F(I),AY1(I),TY1(I),AY2(I),TY2(I)
  112 CONTINUE
      WRITE (6,108)
  108 FORMAT (1H0,2X,4HFREQ,5X,10HMAGN Y3/VO,4X,9HPHS Y3/V0,4X,
     110HMAGN Y4/VO,4X,9HPHS Y4/VO)
      DO 113 I=1,NF
      WRITE (6,105) F(I),AY3(I),TY3(I),AY4(I),TY4(I)
  113 CONTINUE
      WRITE (6,109)
  109 FORMAT (1H0,2X,4HFREQ,5X,8HMAGN G11,6X,7HPHS G11,6X,8HMAGN G12,6X,
     17HPHS G11)
      DO 114 I=1,NF
      WRITE (6,105) F(I),AG1(I),TG1(I),AG2(I),TG2(I)
  114 CONTINUE
C
C     TEST FOR PSD PLOTS
C
      IPSD=0
      READ(20,810,END=815) PSD
810   FORMAT(A5)
      IF(PSD .EQ. 'RAILP') IPSD=1
815   CONTINUE
      IF(IPSD .EQ. 0) GO TO 820
```

```
C*******************************
C**  HALF (7-29-75)
C*******************************
      CALL RAILPL(F,XX,NF,6,IITYPE,SAVEM,100,20,6)
820   CONTINUE
      CALL PLOT(0.,-.6,-3)
C     SCALING FOR PLOT ROUTINE
      DO 155 I=1,NF
      F(I)=(ALOG10(F(I))+1.)*2.
      AF1(I)=(ALOG10(AF1(I))+1.)
      AF2(I)=(ALOG10(AF2(I))+1.)
      ADEL1(I)=(ALOG10(ADEL1(I))+6.)
      ADEL2(I)=(ALOG10(ADEL2(I))+6.)
      AY1(I)=(ALOG10(AY1(I))+7.)
      AY2(I)=(ALOG10(AY2(I))+7.)
      AY3(I)=(ALOG10(AY3(I))+7.)
      AY4(I)=(ALOG10(AY4(I))+7.)
      AG1(I)=(ALOG10(AG1(I))+6.)*4.
      AG2(I)=(ALOG10(AG2(I))+6.)*4.
      IF(AF1(I).LT.0.)AF1(I)=0.
      IF(AF2(I).LT.0.)AF2(I)=0.
      IF(AY1(I).LT.0.)AY1(I)=0.
      IF(AY2(I).LT.0.)AY2(I)=0.
      IF(AY3(I).LT.0.)AY3(I)=0.
      IF(AY4(I).LT.0.)AY4(I)=0.
      IF(AG1(I).LT.0.)AG1(I)=0.
      IF(AG2(I).LT.0.)AG2(I)=0.
      IF(ADEL1(I).LT.0.)ADEL1(I)=0.
      IF(ADEL2(I).LT.0.)ADEL2(I)=0.
  155 CONTINUE
C-----------PLOT ROUTINE
      IF (JA.EQ.1) XLAMFT=XX/12.
      IF (JA.EQ.2) VFPS=XX*88./60./17.6
      IF(JA .EQ. 2) V=VFPS*60./88.
      CALL LAXIS(0.,0.,3,14HFREQUENCY (HZ),14,6.,-1,2)
      CALL LAXIS(0.,0.,9,23HY1/V0 Y2/V0 Y3/V0 Y4/V0,23,9.,-7,1)
      CALL SYMBOL(0.4,9.9,.2,26HCAR BODY, TRUCK, AND WHEEL,0.,26)
      CALL SYMBOL(0.2,9.55,.2,28HDISPLACEMENT AMPLITUDE RATIO,0.,28)
      CALL PLWAVE(VFPS,9.,XLAMFT,JA)
      CALL GRID(0.,0.,6.,9.,3,9,1,2.0,.3,76,1.,2,26)
      CALL CLINE(F,AY1,NF,0.,1.,0.,1.,1)
      CALL CLINE(F,AY2,NF,0.,1.,0.,1.,3)
      CALL CLINE(F,AY3,NF,0.,1.,0.,1.,5)
      CALL CLINE(F,AY4,NF,0.,1.,0.,1.,2)
      CALL NOMEN(.14,1.98,1,.14,15HCAR BODY(Y1/V0),15)
      CALL NOMEN(.14,1.70,3,.14,12HTRUCK(Y2/V0),12)
      CALL NOMEN(.14,1.42,5,.14,17HREAR WHEEL(Y3/V0),17)
      CALL NOMEN(.14,1.14,2,.14,18HFRONT WHEEL(Y4/V0),18)
      CALL SYMBOL(0.25,7.75,0.14,5HBETA=,0.,5)
      CALL NUMBER(999.,999.,0.14,BET,0.,2)
```

```
C******************************
C** HALF (7-29-75)
C******************************
      IF(JA .EQ. 1) GO TO 160
      CALL SYMBOL(0.25,7.50,0.14,2HV=,0.,2)
      CALL NUMBER(999.,999.,0.14,V,0.,-1)
      CALL SYMBOL(999.,999.,0.14,3HMPH,0.,3)
      GO TO 170
 160  CALL SYMBOL(0.25,7.50,0.14,5HWAVE=,0.,5)
      CALL NUMBER(999.,999.,0.14,XLAMFT,0.,1)
      CALL SYMBOL(999.,999.,0.14,2HFT,0.,2)
 170  CALL PLOT(12.,0.,-3)
      CALL LAXIS(0.,0.,3,14HFREQUENCY (HZ),14,6.,-1,2)
      CALL LAXIS(0.,0.,8,11HF1/V0 F2/V0,11,8.,-1,1)
      CALL SYMBOL(0.1,8.9,.2,29HWHEEL RAIL FORCES PRODUCED BY,0.,29)
      CALL SYMBOL(-0.1,8.55,.2,30HUNIT TRACK IRREGULARITY(LB/IN),0.,30)
      CALL PLWAVE(VFPS,8.,XLAMFT,JA)
      CALL GRID(0.,0.,6.,8.,3,8,1,2,1.,4.76,.5,1.2)
      CALL CLINE(F,AF1,NF,0.,1.,0.,1.,1)
      CALL CLINE(F,AF2,NF,0.,1.,0.,1.,3)
      CALL NOMEN(1.14,.92,1,.14,17HREAR WHEEL(F1/V0),17)
      CALL NOMEN(1.14,.64,3,.14,18HFRONT WHEEL(F2/V0),18)
      CALL SYMBOL(0.25,7.75,0.14,5HBETA=,0.,5)
      CALL NUMBER(999.,999.,0.14,BET,0.,2)
      IF(JA .EQ. 1) GO TO 180
      CALL SYMBOL(0.25,7.50,0.14,2HV=,0.,2)
      CALL NUMBER(999.,999.,0.14,V,0.,-1)
      CALL SYMBOL(999.,999.,0.14,3HMPH,0.,3)
      GO TO 190
 180  CALL SYMBOL(0.25,7.50,0.14,5HWAVE=,0.,5)
      CALL NUMBER(999.,999.,0.14,XLAMFT,0.,1)
      CALL SYMBOL(999.,999.,0.14,2HFT,0.,2)
 190  CALL PLOT(12.,0.,-3)
      CALL LAXIS(0.,0.,3,14HFREQUENCY (HZ),14,6.,-1,2)
      CALL LAXIS(0.,0.,8,15HDEL1/V0 DEL2/V0,15,8.,-6,1)
      CALL SYMBOL(1.4,8.9,.2,16HTRACK DEFLECTION,0.,16)
      CALL SYMBOL(1.5,8.55,.2,15HAMPLITUDE RATIO,0.,15)
      CALL PLWAVE(VFPS,8.,XLAMFT,JA)
      CALL GRID(0.,0.,6.,8.,3,8,1,2,1.,5.08,.5,1.2)
      CALL CLINE(F,ADEL1,NF,0.,1.,0.,1.,1)
      CALL CLINE(F,ADEL2,NF,0.,1.,0.,1.,3)
      CALL NOMEN(1.14,.92,1,.14,19HREAR WHEEL(DEL1/V0),19)
      CALL NOMEN(1.14,.64,3,.14,20HFRONT WHEEL(DEL2/V0),20)
      CALL SYMBOL(0.25,7.75,0.14,5HBETA=,0.,5)
      CALL NUMBER(999.,999.,0.14,BET,0.,2)
      IF(JA .EQ. 1) GO TO 200
      CALL SYMBOL(0.25,7.50,0.14,2HV=,0.,2)
      CALL NUMBER(999.,999.,0.14,V,0.,-1)
      CALL SYMBOL(999.,999.,0.14,3HMPH,0.,3)
      GO TO 210
```

```
C******************************
C**  HALF (7-2?-75)
C******************************
200   CALL SYMBOL(0.25,7.50,0.14,5HWAVE=,0.,5)
      CALL NUMBER(999.,999.,0.14,XLAMFT,0.,1)
      CALL SYMBOL(999.,999.,0.14,2HFT,0.,2)
210   CALL PLOT(12.,0.,-3)
      CALL LAXIS(0.,0.,3,14HFREQUENCY (HZ),14,6.,-1,2)
       CALL LAXIS(0.,0.,2,7HG11 G12,7,8.,-6,1)
      CALL SYMBOL(0.5,8.65,.2,25HTRACK COMPLIANCE FUNCTION,0.,25)
      CALL PLWAVE(VFPS,8.,XLAMFT,JA)
      CALL GRID(0.,0.,6.,8.,3,2,1,2,0.,1.81,6.74,8.)
      CALL CLINE(F,AG1,NF,0.,1.,0.,1.,1)
      CALL CLINE(F,AG2,NF,0.,1.,0.,1.,3)
      CALL NOMEN(0.25,7.16,1,.14,3HG11,3)
      CALL NOMEN(.25,6.88,3,.14,3HG12,3)
      CALL SYMBOL(0.25,7.75,0.14,5HBETA=,0.,5)
      CALL NUMBER(999.,999.,0.14,BET,0.,2)
      IF(JA .EQ. 1) GO TO 220
      CALL SYMBOL(0.25,7.50,0.14,2HV=,0.,2)
      CALL NUMBER(999.,999.,0.14,V,0.,-1)
      CALL SYMBOL(999.,999.,0.14,3HMPH,0.,3)
      GO TO 230
220   CALL SYMBOL(0.25,7.50,0.14,5HWAVE=,0.,5)
      CALL NUMBER(999.,999.,0.14,XLAMFT,0.,1)
      CALL SYMBOL(999.,999.,0.14,2HFT,0.,2)
230   CALL PLOT(12.,0.,-3)
      CALL PLOT(0.,0.,999)
      GO TO 400
  501 WRITE (6,502)
  502 FORMAT( 1H1,24HMATRIX ACOMP IS SINGULAR)
  400 STOP
      END
```

```
C*******************************
C**  HALF (7-29-75)
C*******************************
      SUBROUTINE CLINE(X,Y,N,XM,DELX,YM,DELY,NC)
      DIMENSION X(1),Y(1),IPEN(4)
      REAL L(4,4),LL(4)
      DATA IPEN/2,3,2,3/
      DATA L/.3,.1,.3,.1,.5,3*.05,.3,3*.1,.1,.05,.1,.05/
      IC=NC-1
      XP1=(X(1)-XM)/DELX
      YP1=(Y(1)-YM)/DELY
      CALL PLOT(XP1,YP1,3)
      IF(IC.EQ.0)GO TO 1000
      K=1
      I=2
      XP2=(X(2)-XM)/DELX
      YP2=(Y(2)-YM)/DELY
    1 LL(K)=L(K,IC)
   10 DIFFX=XP2-XP1
      DIFFY=YP2-YP1
      DIS=SQRT(DIFFX*DIFFX+DIFFY*DIFFY)
      IF(DIS.GT.LL(K))GO TO 100
      CALL PLOT(XP2,YP2,IPEN(K))
      XP1=XP2
      YP1=YP2
      I=I+1
      IF(I.GT.N)RETURN
      XP2=(X(I)-XM)/DELX
      YP2=(Y(I)-YM)/DELY
      LL(K)=LL(K)-DIS
      GO TO 10
  100 RATIO=DIS/LL(K)
      XP1=XP1+DIFFX/RATIO
      YP1=YP1+DIFFY/RATIO
      CALL PLOT(XP1,YP1,IPEN(K))
      K=K+1
      IF(K.EQ.5)K=1
      GO TO 1
 1000 DO 50 I=2,N
      XP1=(X(I)-XM)/DELX
      YP1=(Y(I)-YM)/DELY
   50 CALL PLOT(XP1,YP1,2)
      RETURN
      END
```

```
C*******************************
C**  HALF (7-2?-75)
C*******************************
      SUBROUTINE NOMEN(X,Y,NC,HT,BCD,NCHAR)
      DIMENSION BCD(1),XL(2),YL(2)
      XL(1)=X
      XL(2)=X+1.
      YL(1)=Y+HT/2.
      YL(2)=YL(1)
      CALL CLINE(XL,YL,2,Ø.,1.,Ø.,1.,NC)
      CALL SYMBOL(X+1.1,Y,HT,BCD,Ø.,NCHAR)
      RETURN
      END
```

```
      SUBROUTINE PLWAVE(VFPS,Y,XLAMFT,JA)
      DIMENSION WAVE(4)
      IF(JA .EQ. 1) GO TO 12Ø
      DO 1ØØ J=1,4
1ØØ   WAVE(J)=VFPS/(1Ø.**(J-2))
      GO TO 16Ø
12Ø   DO 14Ø J=1,4
14Ø   WAVE(J)=XLAMFT*(1Ø.**(J-2))*6Ø./88.
16Ø   CALL PLOT(6.,Y,3)
      CALL PLOT(Ø.,Y,2)
      DO 2ØØ J=1,4
      X=(J-1)*2
      CALL PLOT(X,Y-.Ø5,3)
      CALL PLOT(X,Y+.Ø5,2)
      CALL NUMBER(X   -.,.Ø7+Y,.1,WAVE(J),Ø.,2)
2ØØ   CONTINUE
      IF(JA .EQ. 2) CALL SYMBOL(1.11,Y+.2,.14,27HIRREGULARITY WAVELENGTH
     X(FT),Ø.,27)
      IF(JA .EQ. 1) CALL SYMBOL(2.11,Y+.2,.14,13HVELOCITY(MPH),Ø.,13)
      RETURN
      END
```

```
C*******************************
C**  HALF (7-2Y-75)
C*******************************
      SUBROUTINE GRID(XØ,YØ,XGRID,YGRID,KX,KY,LAST,IOPN,
     X XLOW,XUP,YLOW,YUP)
C-----------------------------------------------------------------------
C
C     THIS SUBROUTINE DRAWS GRID LINES INSIDE SPECIFIED
C     BOUNDARIES WITH AN OPTIONAL BOX LEFT OPEN FOR LABELS.
C     IT IS MEANT TO BE USED WITH AXIS ROUTINES AND CAN
C     ADD THE OUTER LINES AT THE TOP AND RIGHT OF A GRAPH
C     IF THESE ARE DESIRED.
C
C     ARGUMENTS+
C
C     XØ,YØ - COORDINATES OF LOWER LEFT CORNER OF GRAPH
C     XGRID,YGRID - LENGTH OF X AND Y AXES
C     KX,KY - NUMBER OF CYCLES ON EACH AXIS, ONE LESS LINE
C             WILL BE DRAWN.
C     LAST - Ø NO OUTLINES DRAWN
C            1 RIGHT OUTLINE DRAWN
C            2 UPPER OUTLINE DRAWN
C            3 BOTH OUTLINES DRAWN
C     IOPN - Ø NO BOX LEFT
C            1 BOX WITH NO BORDERS
C            2 BOX WITH BORDERS DRAWN
C     XLOW,XUP, - COORDINATES OF BOX BOUNDARIES RELATIVE
C     YLOW,YUP    TO XØ,YØ, DUMMY VALUES IF IOPN=Ø,
C
C     SUBROUTINES CALLED+  PLOT
C
C-----------------------------------------------------------------------
C
C----------------- CHECK INPUTS ------------------------------------
      IF(KX .LT. 1) GO TO 9ØØ
      IF(KY .LT. 1) GO TO 9ØØ
      IF(LAST .LT. Ø .OR. LAST .GT. 3) GO TO 9ØØ
      IF(IOPN .LT. Ø .OR. IOPN .GT. 2) GO TO 9ØØ
      IF(XGRID .LE. Ø. .OR. YGRID .LE. Ø.) GO TO 9ØØ
C------------------ SETUP ---------------------------------------
      XTOP=XØ+XGRID
      YTOP=YØ+YGRID
      AX=XGRID/FLOAT(KX)
      AY=YGRID/FLOAT(KY)
      NX=KX-1
      NY=KY-1
      IF(IOPN .EQ. Ø) GO TO 5Ø
      XL=XLOW+XØ
      XU=XUP+XØ
      YL=YLOW+YØ
```

```
C********************************
C**  HALF (7-2Y-75)
C********************************
      YU=YUP+YØ
      GO TO 75
50    XL=50000.
      XU=-50000.
      YL=50000.
      YU=-50000.
75    CONTINUE
C----------- DRAW VERTICAL GRID IF NECESSARY ------------------
      IF(NX .EQ. Ø) GO TO 400
      DO 300 N=1,NX
      PX=AX*FLOAT(N)+XØ
      IF(PX .GT. XL .AND. PX .LT. XU) GO TO 200
C------------- FULL LINES ---------------------------------
      IF(MOD(N,2) .EQ. Ø) GO TO 150
      CALL PLOT(PX,YØ,3)
      CALL PLOT(PX,YTOP,2)
      GO TO 300
150   CALL PLOT(PX,YTOP,3)
      CALL PLOT(PX,YØ,2)
      GO TO 300
C------------ BROKEN LINES --------------------------------
200   IF(MOD(N,2) .EQ. Ø) GO TO 250
      CALL PLOT(PX,YØ,3)
      CALL PLOT(PX,YL,2)
      CALL PLOT(PX,YU,3)

      CALL PLOT(PX,YTOP,2)
      GO TO 300
250   CALL PLOT(PX,YTOP,3)
      CALL PLOT(PX,YU,2)
      CALL PLOT(PX,YL,3)
      CALL PLOT(PX,YØ,2)
300   CONTINUE
C------------- DRAW HORIZONTAL GRID IF NECESSARY ---------------

400   IF(NY .EQ. Ø) GO TO 700
      DO 600 N=1,NY
      PY=AY*FLOAT(N)+YØ
      IF(PY .GT. YL .AND. PY .LT. YU) GO TO 500
C------------- FULL LINES ---------------------------------
      IF(MOD(N,2) .EQ. Ø) GO TO 450
      CALL PLOT(XØ,PY,3)
      CALL PLOT(XTOP,PY,2)
      GO TO 600
450   CALL PLOT(XTOP,PY,3)
      CALL PLOT(XØ,PY,2)
      GO TO 600
```

```
C********************************
C**  HALF (7-29-75)
C********************************
C------------- BROKEN LINES ------------------------------------
500   IF(MOD(N,2) .EQ. 0) GO TO 550
      CALL PLOT(X0,PY,3)
      CALL PLOT(XL,PY,2)
      CALL PLOT(XU,PY,3)
      CALL PLOT(XTOP,PY,2)
      GO TO 600
550   CALL PLOT(XTOP,PY,3)
      CALL PLOT(XU,PY,2)
      CALL PLOT(XL,PY,3)
      CALL PLOT(X0,PY,2)
600   CONTINUE
C----------- DRAW END LINES IF WANTED ---------------------
700   IF(LAST .EQ. 0) GO TO 800
      IF(LAST .EQ. 1) GO TO 750
      CALL PLOT(X0,YTOP,3)
      CALL PLOT(XTOP,YTOP,2)
      IF(LAST .EQ. 2) GO TO 800
      GO TO 760
750   CALL PLOT(XTOP,YTOP,3)
760   CALL PLOT(XTOP,Y0,2)
C-------------DRAW BOX IF WANTED ----------------------------
800   IF(IOPN .NE. 2) GO TO 900
      CALL PLOT(XL,YL,3)
      CALL PLOT(XL,YU,2)
      CALL PLOT(XU,YU,2)
      CALL PLOT(XU,YL,2)
      CALL PLOT(XL,YL,2)
900   RETURN
      END


      SUBROUTINE GMPRD (A,B,R,N,M,L)
      DIMENSION A(1),B(1),R(1)
      COMPLEX A,B,R
      IR=0
      IK=-M
      DO 10 K=1,L
      IK=IK+M
      DO 10 J=1,N
      IR=IR+1
      JI=J-N
      IB=IK
      R(IR)=(0.,0.)
      DO 10 I=1,M
      JI=JI+N
      IB=IB+1
   10 R(IR)=R(IR)+A(JI)*B(IB)
      RETURN
      END
```

```
C******************************
C**  HALF (7-29-75)
C******************************
C
C     ..................................................................
C
C        SUBROUTINE MINV
C
C        PURPOSE
C           INVERT A MATRIX
C
C        USAGE
C           CALL MINV(A,N,D,L,M)
C
C        DESCRIPTION OF PARAMETERS
C           A - INPUT MATRIX, DESTROYED IN COMPUTATION AND REPLACED BY
C               RESULTANT INVERSE.
C           N - ORDER OF MATRIX A
C           D - RESULTANT DETERMINANT
C           L - WORK VECTOR OF LENGTH N
C           M - WORK VECTOR OF LENGTH N
C
C        REMARKS
C           MATRIX A MUST BE A GENERAL MATRIX
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           NONE
C
C        METHOD
C           THE STANDARD GAUSS-JORDAN METHOD IS USED. THE DETERMINANT
C           IS ALSO CALCULATED. A DETERMINANT OF ZERO INDICATES THAT
C           THE MATRIX IS SINGULAR.
C
C     ..................................................................
C
      SUBROUTINE MINV(A,N,D,L,M)
      DIMENSION A(1),L(1),M(1)
      COMPLEX A,D,BIGA,HOLD
C
C        ...............................................................
C
C        IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE
C        C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION
C        STATEMENT WHICH FOLLOWS.
C
C     DOUBLE PRECISION A,D,BIGA,HOLD
C
C        THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS
C        APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS
C        ROUTINE.
```

```
C*******************************
C**  HALF (7-22-75)
C*******************************
C                                                                       MINV 470
C        THE DOUBLE PRECISION VERSION OF THIS SUBROUTINE MUST ALSO      MINV 480
C        CONTAIN DOUBLE PRECISION FORTRAN FUNCTIONS.  ABS IN STATEMENT  MINV 490
C        10 MUST BE CHANGED TO DABS.                                    MINV 500
C                                                                       MINV 510
C        ...............................................................MINV 520
C                                                                       MINV 530
C        SEARCH FOR LARGEST ELEMENT                                     MINV 540
C                                                                       MINV 550
      D=(1.0,0.0)
      NK=-N                                                             MINV 570
      DO 80 K=1,N                                                       MINV 580
      NK=NK+N                                                           MINV 590
      L(K)=K                                                            MINV 600
      M(K)=K                                                            MINV 610
      KK=NK+K                                                           MINV 620
      BIGA=A(KK)                                                        MINV 630
      DO 20 J=K,N                                                       MINV 640
      IZ=N*(J-1)                                                        MINV 650
      DO 20 I=K,N                                                       MINV 660
      IJ=IZ+I                                                           MINV 670
   10 IF(CABS(BIGA)-CABS(A(IJ))) 15,20,20
   15 BIGA=A(IJ)                                                        MINV 690
      L(K)=I                                                            MINV 700
      M(K)=J                                                            MINV 710
   20 CONTINUE                                                          MINV 720
C                                                                       MINV 730
C        INTERCHANGE ROWS                                               MINV 740
C                                                                       MINV 750
      J=L(K)                                                            MINV 760
      IF(J-K) 35,35,25                                                  MINV 770
   25 KI=K-N                                                            MINV 780
      DO 30 I=1,N                                                       MINV 790
      KI=KI+N                                                           MINV 800
      HOLD=-A(KI)                                                       MINV 810
      JI=KI-K+J                                                         MINV 820
      A(KI)=A(JI)                                                       MINV 830
   30 A(JI) =HOLD                                                       MINV 840
C                                                                       MINV 850
C        INTERCHANGE COLUMNS                                            MINV 860
C                                                                       MINV 870
   35 I=M(K)                                                            MINV 880
      IF(I-K) 45,45,38                                                  MINV 890
   38 JP=N*(I-1)                                                        MINV 900
      DO 40 J=1,N                                                       MINV 910
      JK=NK+J                                                           MINV 920
      JI=JP+J                                                           MINV 930
```

```
C******************************
C**  HALF (7-2[illegible]-75)
C******************************
      HOLD=-A(JK)                                                      MINV 940
      A(JK)=A(JI)                                                      MINV 950
   40 A(JI) =HOLD                                                      MINV 960
C                                                                      MINV 970
C        DIVIDE COLUMN BY MINUS PIVOT (VALUE OF PIVOT ELEMENT IS       MINV 980
C        CONTAINED IN BIGA)                                            MINV 990
C                                                                      MINV1000
   45 BIGX=CABS(BIGA)
      IF(BIGX) 48,46,48
   46 D=(0.,0.)
      RETURN                                                           MINV1030
   48 DO 55 I=1,N                                                      MINV1040
      IF(I-K) 50,55,50                                                 MINV1050
   50 IK=NK+I                                                          MINV1060
      A(IK)=A(IK)/(-BIGA)                                              MINV1070
   55 CONTINUE                                                         MINV1080
C                                                                      MINV1090
C        REDUCE MATRIX                                                 MINV1100
C                                                                      MINV1110
      DO 65 I=1,N                                                      MINV1120
      IK=NK+I                                                          MINV1130
      HOLD=A(IK)                                                       MINV1140
      IJ=I-N                                                           MINV1150
      DO 65 J=1,N                                                      MINV1160
      IJ=IJ+N                                                          MINV1170
      IF(I-K) 60,65,60                                                 MINV1180
   60 IF(J-K) 62,65,62                                                 MINV1190
   62 KJ=IJ-I+K                                                        MINV1200
      A(IJ)=HOLD*A(KJ)+A(IJ)                                           MINV1210
   65 CONTINUE                                                         MINV1220
C                                                                      MINV1230
C        DIVIDE ROW BY PIVOT                                           MINV1240
C                                                                      MINV1250
      KJ=K-N                                                           MINV1260
      DO 75 J=1,N                                                      MINV1270
      KJ=KJ+N                                                          MINV1280
      IF(J-K) 70,75,70                                                 MINV1290
   70 A(KJ)=A(KJ)/BIGA                                                 MINV1300
   75 CONTINUE                                                         MINV1310
C                                                                      MINV1320
C        PRODUCT OF PIVOTS                                             MINV1330
C                                                                      MINV1340
      D=D*BIGA                                                         MINV1350
C                                                                      MINV1360
C        REPLACE PIVOT BY RECIPROCAL                                   MINV1370
C                                                                      MINV1380
      A(KK)=(1.,0.)/BIGA
```

```
C*******************************
C**  HALF (7-29-75)
C*******************************
   80 CONTINUE
C
C        FINAL ROW AND COLUMN INTERCHANGE
C
      K=N
  100 K=(K-1)
      IF(K) 150,150,105
  105 I=L(K)
      IF(I-K) 120,120,108
  108 JQ=N*(K-1)
      JR=N*(I-1)
      DO 110 J=1,N
      JK=JQ+J
      HOLD=A(JK)
      JI=JR+J
      A(JK)=-A(JI)
  110 A(JI) =HOLD
  120 J=M(K)
      IF(J-K) 100,100,125
  125 KI=K-N
      DO 130 I=1,N
      KI=KI+N
      HOLD=A(KI)
      JI=KI-K+J
      A(KI)=-A(JI)
  130 A(JI) =HOLD
      GO TO 100
  150 RETURN
      END
```

```
C*****************************
C**  HALF (7-29-75)
C*****************************
      SUBROUTINE POLAR(X,ABSX,TNX)
      COMPLEX X
      ABSX=CABS(X)
      X2=REAL(X)
      X3=AIMAG(X)
      TNX=ATAN2(X3,X2)*180./3.14159265
      RETURN
      END
```

```
      SUBROUTINE LLINE(X,Y,N)
      DIMENSION X(1),Y(1)
      CALL PLOT (X(1),Y(1),3)
      DO 10 I=2,N
      CALL PLOT (X(I),Y(I),2)
   10 CONTINUE
      RETURN
      END
```

```
C********************************
C**  HALF (7-29-75)
C********************************
      SUBROUTINE FTABG(IFREQ,NDF,DF,FL,F,NF)
C
C     THIS SUBROUTINE COMPUTES THE FREQUENCY TABLE
C
C     IFREQ=1 RANGE INPUTS, =2 OCTAVE INPUTS
C     NDF   - NUMBER OF RANGES OR OCTAVES
C     DF    - FREQUENCY POINTS VECTOR
C     FL    - FREQUENCY LOWER LIMITS VECTOR
C     F     - FREQUENCY TABLE RETURNED
C     NF    - NUMBER OF POINTS IN FREQUENCY TABLE
C
      INTEGER DF(7)
      DIMENSION FL(8),F(200)
      IF(IFREQ .EQ. 2) GO TO 500
C
C     RANGE INPUTS
C
      NF=0
      DO 200 I=1,NDF
      C=EXP(ALOG(FL(I+1)/FL(I))/DF(I))
      NF=NF+1
      F(NF)=FL(I)
      DO 100 J=1,DF(I)
      FT=C*F(NF)
      IF(FT .GE. FL(I+1)-0.000001) GO TO 200
      NF=NF+1
      F(NF)=FT
100   CONTINUE
200   CONTINUE
      RETURN
C
C     OCTAVE INPUTS
C
500   C=2**(1./DF(1))
      NF=1
      F(NF)=FL(1)
      DO 700 I=1,NDF
      DO 700 J=1,DF(1)
      NF=NF+1
      F(NF)=F(NF-1)*C
700   CONTINUE
      RETURN
      END
```

Frequency Domain Computer Programs for Prediction and Analysis of Rail Vehicle Dynamics, Volume II: Appendixes, 1975
US DOT, FRA, AB Perlman, FP DiMasi

SMEAD CO VP533A