Computer Program for Diffracted Multiple Reflections

by Don C. Riley

The following computer program represents the 1-D and 2-D forward and inverse algorithms as described in the March 1974 report (p. 83 - 135). Also included is a sample job stream (for IBM 360) that was used to compute the figures on p. 133-135 of that report.

```
//CRUNCH EXEC PGM=NOAH
//*
//*     PRINT DATA SET FOLLOWS
//*             FORTRAN LOGICAL UNIT 06
//FT06F001 DD SYSOUT=A
//*
//*     PRINTER-PLOT DATA SET FOLLOWS
//*             FORTRAN LOGICAL UNIT 66
//FT66F001 DD SYSOUT=A,DCB=(BLKSIZE=133,RECFM=UA)
//*
//*     PRIMARY TRACE SEQUENTIAL OUTPUT DATA SET FOLLOWS
//*             FORTRAN LOGICAL UNIT 09
//FT09F001 DD DSN=S091.PLOT,VOL=SER=SYS23,UNIT=2314,
// DISP=(NEW,KEEP),DCB=(RECFM=FB,LRECL=560,BLKSIZE=6720),
// SPACE=(TRK,(18,1),RLSE)
//*
//*     PRIMARY TRACE SEQUENTIAL INPUT DATA SET FOLLOWS
//*             FORTRAN LOGICAL UNIT 11
//FT11F001 DD DUMMY
//*
//*     SCRATCH DATA SET FOLLOWS
//*             FORTRAN LOGICAL UNIT 13
//FT13F001 DD DSN=&&SCR,DISP=(NEW,PASS),UNIT=SYSDA,
//  DCB=(RECFM=VSB,LRECL=284,BLKSIZE=7104),SPACE=(CYL,(3,2))
//*
//*     CONTROL CARD DATA SET FOLLOWS
//*             FORTRAN LOGICAL UNIT 05
//FT05F001 DD *
 &PHASE JOB=1,NX=70,NT=280,VEL=5000.,SAMPRT=.004,DELX=50.0,
        LINE='TESPLT',SHOT=0.0,SMAG=1.0,LS=15,PW=10.,
        XDIP=30.,PLOT=+1,KNORM=1,NORM=1024,&END
 &MODEL GEOL='CIRP',COEFF=-0.25,DEPTH=36.,RELIEF=4.0,
        CYCLES=1.5,NFAULT=0,DIP=0.0,&END
 &MODEL GEOL='LAYR',COEFF=0.05,DEPTH=141.,DIP=-1.5,
        NFAULT=1,THROW=10.,&END
 &MODEL COEFF=999.,&END
 &PHASE JOB=4,IDELZ=10,N1S=25,N2S=60,N1L=25,N2L=134,&END
 &PHASE JOB=5,LU=15,NEST=1,NZAP=2,N1E=30,N2E=52,N3E=65,N4E=88,&END
/*
```

```
C*****************************************************************************
C*************************** N  C  A  H  ****************************
C************************** E X P L I C I T ***************************
C*****************************************************************************
C.....
C.....   FOR EACH PHASE OF THIS PROGRAM ONE JOB IS PERFORMED.
C.....   MULTIPLE PHASES MAY BE EXECUTED IN A SINGLE COMPUTER RUN.
C.....   THE VARIOUS JOBS ARE AS FOLLOWS :
C.....
C.....   JOB=1 -MODEL BUILDING- A REFLECTION COEFFICIENT MODEL IS
C.....                          CONSTRUCTED ACCORDING TO THE &MODEL
C.....                          ELEMENT DESCRIPTION NAMELISTS WHICH
C.....                          MUST FOLLOW A PHASE WITH JOB=1.
C.....                          SEE SUBROUTINE "EARTH" FOR MODEL
C.....                          PARAMETERS.
C.....   JOB=2 -ONE DIMENSIONAL REFLECTION SEISMOGRAM SYNTHESIS-
C.....   JOB=3 -ONE DIMENSIONAL INVERSION OF REFLECTION SEISMOGRAM-
C.....   JOB=4 -TWO DIMENSIONAL REFLECTION SEISMOGRAM SYNTHESIS-
C.....   JOB=5 -TWO DIMENSIONAL INVERSION OF REFLECTION SEISMOGRAM-
C.....   JOB=6 -EXTERNAL REFLECTION SEISMOGRAM INPUT- READS FROM
C.....                          LOGICAL UNIT (LU) 11 DATA IN TRACE
C.....                          SEQUENTIAL, 16 BIT TWO'S COMPLIMENT
C.....                          INTEGER FORMAT.  ASSUMED ARE THE
C.....                          FOLLOWING: NORMALIZATION TO "NORM",
C.....                          EXPONENTIAL GAIN OF 'DB' DB/SEC.
C.....   JOB=7 -SAME AS JOB=6   EXCEPT EXTERNAL REFLECTION COEFFICIENT
C.....                          SECTION IS READ INTO THE C ARRAY.
C.....   JOB=8 -NULL PHASE-     MODIFY ANY PARAMETER
C.....                          ALSO REWINDS UNIT 11
C.....   1>JOB>8 -NORMAL PROGRAM EXIT-
C.....
C.....   NAMELIST PARAMETERS ARE AS FOLLOWS. * INDICATES WHICH OF
C.....   THE PARAMETERS NEED BE SUPPLIED FOR A PARTICULAR JOB.
C.....   NOTE: PARAMETERS FROM ONE PHASE REMAIN IN FORCE FOR SUCCEEDING
C.....   PHASES UNLESS EXPLICITLY RE-ENTERED.
C.....   THE PARAMETERS FOR JOB=7 ARE THE SAME AS FOR JOB=6.
C.....
C  ____JOB____
C /1 2 3 4 5 6/TYPE/ NAME     /    DESCRIPTION
C /-----------/----/---------/-----------------------------------------/
C /*          */ I  / NX      /NUMBER OF TRACES IN SECTION
C /*          */ I  / NT      /NUMBER OF SAMPLES/TRACE
C /  * * * *  / I  / N1S,N2S /SHORT PATH MULTIPLE GATE IN SAMPLES
C /  * * * *  / I  / N1L,N2L /LONG PATH MULTIPLE GATE IN SAMPLES
C /      * *  / R  / SHOT    /INITIAL SHOT PARAMETER, SHOT=L.W WHERE
C /           /    /         / L=GEOPHONE LOC. OF CENTER OF SHOT PATTERN
C /           /    /         / W=WIDTH OF PATTERN. SHOT=0.0 = PLANE WAVE
C/       * *  / R  / SMAG    /SHOT MAGNITUDE
C /* * *  *  */ I  / LS      /LENGTH OF SHOT WAVEFORM IN SAMPLES
C /    *   *  / I  / LU      /LENGTH OF SHOT WAVEFORM INVERSE IN SAMPLE
C /      * *  / I  / IDELZ   /RATIO OF DELTA Z TO DELTA T
C /*          */ R  / VEL     /FRAME VELOCITY FT./SEC.
C /*          */ R  / DELX    /MID-POINT SPACING FT.
C /*          */ R  / SAMPRT  /SAMPLING RATE SEC.
```

```
C.....                          I/O  DATA  SETS
C.....     FORTRAN UNIT         USE          RECORD LEN      RECORD FORMAT
C.....
C.....          6               PRINT            133              UA
C.....         66             PRINTER PLOT       133              UA
C.....          9            SEQ. OUTPUT         2*NT             U OR F
C.....         11            SEQ. INPUT          2*NT             U OR F
C.....         13              SCRATCH         4*(&NX+1)           VSB
C*************************************************************************
C.....
      REAL*4 S(&LS),GAIN(&NT)
      INTEGER IHW*2(&NT),PLOT,T
      COMPLEX*16 LINE
C.....
C.....       PROBLEM DATA COMMON AREAS FOLLOW
      COMMON /GRID$1/ C(&NX,&NT)
      COMMON /GRID$2/ R(&NX,&NT)
      COMMON /GRID$4/ GRID4(&NX,&NZ)
      COMMON /BLOCK1/ V1(&NX),V2(&NX)
      COMMON /BLOCK2/ V3(&NX),V4(&NX),V5(&NX)
      COMMON /GLOBAL/ RHO,NX,NT,NZ,SHOT,SMAG,LS,NZAP
C.....
      NAMELIST /PHASE/ JOB,NX,NT,LS,LU,IDELZ,VEL,DELX,LINE,
     @                 SAMPRT,KNORM,DB,PLOT,SIGN,PW,SHOT,SMAG,
     @                 N2S,N2L,N1S,N1L,NEST,NZAP,NORM,XDIP,
     @                 N1E,N2E,N3E,N4E
      RICKY(T)=(1.-2.*ARR*T*T)*EXP(-ARR*T*T)
      DATA LINE/'                '/
C.....
C.....                BEGIN EXECUTABLE STATEMENTS
C.....
C.....  TURN OFF ERROR MESSAGES FOR FLOAT UNDERFLOWS
      CALL ERRSET(208,0,-1,1)
C.....
      NSET=0
      NPHASE=0
      LARRAY=&NX*&NT
C.....    SET DEFAULT VALUES
      JOB=9999
      LU=13
      IDELZ=25
      VEL=5000.
      SAMPRT=.004
      KNORM=1
      DB=0.0
      PLOT=1
      PW=8.0
      DELX=50.0
      NEST=4
      NORM=1024
      XDIP=30.0
      AKO=3.14159/4.
      RHO=1.0
      A=0.0
```

```
      SHOT=0.0
      SMAG=1.0
      LS=13
      NZ=00
      NZAP=1
C.....      TOP OF PHASE LOOP
  100 READ(5,PHASE,END=555,ERR=888)
      IF(JOB.EQ.1.OR.JOB.EQ.6.OR.JOB.EQ.7) WRITE(6,904) LINE
      WRITE(6,905)
      NPHASE=NPHASE+1
      IF(JOB.LT.1.OR.JOB.GT.8) GO TO 555
      IF(NX*NT.GT.LARRAY) GO TO 666
      IF(NX.GT.&NX) GO TO 897
      IF(NSET.NE.0) GO TO 102
      NSET=1
      ARR=8./PW/PW
      MIDDLE=LS/2+1
      DO 101 K=1,LS
      T=K-MIDDLE
  101 S(K)=RICKY(T)
C.....    BRANCH TO A PARTICULAR JOB
  102 GO TO (1000,2000,2000,3000,4000,5000,5000,5000),JOB
C.....
 1000 VXCALC=2.*DELX/VEL/SAMPRT
      DBIN=DB
      CALL EARTH (C,NX,NT,S,LS,VXCALC,SAMPRT,DB,PW,R)
      IF(DBIN.NE.0.) DB=DBIN
      CALL SETEX (SAMPRT,DB,GAIN,NT,KNORM,NX,IHW)
      IF(PLOT.NE.0) CALL WAVEX (R,NX,NT,IHW,PLOT,NORM)
      WRITE(6,900) JOB,NX,NT,NZ,IDELZ,VEL,DELX,SAMPRT,NORM,KNORM,
     @             DB,PLOT,LS,PW,XDIP,A,RHO,VXCALC,SHOT,SMAG
      WRITE(6,908) NPHASE
      GO TO 100
C.....
 2000 CONTINUE
      SIGN=1.0
      IF(JOB.EQ.3) SIGN=-1.0
      CALL NOAH1D(C,R,N1S,N1L,N2S,N2L,S,LU,SIGN,
     @            NEST,N1E,N2E,N3E,N4E)
      IF(PLOT.NE.0.AND.SIGN.GT.0.) CALL WAVEX (R,NX,NT,IHW,PLOT,NORM)
      IF(PLOT.NE.0.AND.SIGN.LT.0.) CALL WAVEX (C,NX,NT,IHW,PLOT,NORM)
      WRITE(6,900) JOB,NX,NT,NZ,IDELZ,VEL,DELX,SAMPRT,NORM,KNORM,
     @             DB,PLOT,LS,PW,XDIP,A,RHO,VXCALC,SHOT,SMAG
      WRITE(6,901) N1S,N2S,N1L,N2L
      IF(SIGN.LT.0.) WRITE(6,902) N1E,N2E,N3E,N4E,NEST,LU,NZAP
      WRITE(6,908) NPHASE
      GO TO 100
C.....
 3000 CONTINUE
      NZ=1+NT/IDELZ
      IF(NZ.GT.&NZ) GO TO 899
      A=IDELZ/(2.*VXCALC)**2
      RHO=EXP(-AKO*VEL*SAMPRT/(4.*3.14159*DELX*SIN(XDIP/57.296)))
      IF(A.GE.0.166666) GO TO 898
```

```
      CALL SPONGE(C,R,N1S,N1L,N2S,N2L,S,ICELZ,A,R,GRID4)
      IF(PLOT.NE.0) CALL WAVEX (R,NX,NT,IHW,PLOT,NORM)
      WRITE(6,900) JOB,NX,NT,NZ,ICELZ,VEL,DELX,SAMPRT,NORM,KNORM,
     @             DB,PLOT,LS,PW,XDIP,A,RHO,VXCALC,SHOT,SMAG
      WRITE(6,901) N1S,N2S,N1L,N2L
      WRITE(6,908) NPHASE
      GO TO 100
C.....
 4000 CONTINUE
      NZ=1+NT/IDELZ
      IF(NZ.GT.&NZ) GO TO 899
      A=IDELZ/(2.*VXCALC)**2
      RHO=EXP(-AKO*VEL*SAMPRT/(4.*3.14159*DELX*SIN(XDIP/57.296)))
      IF(A.GE.0.166666) GO TO 898
      CALL FLOOD(C,R,N1S,N1L,N2S,N2L,LU,ICELZ,A,
     @           C,NEST,N1E,N2E,N3E,N4E)
      IF(PLOT.NE.0) CALL WAVEX (C,NX,NT,IHW,PLOT,NORM)
      WRITE(6,900) JOB,NX,NT,NZ,ICELZ,VEL,DELX,SAMPRT,NORM,KNORM,
     @             DB,PLOT,LS,PW,XDIP,A,RHO,VXCALC,SHOT,SMAG
      WRITE(6,901) N1S,N2S,N1L,N2L
      WRITE(6,902) N1E,N2E,N3E,N4E,NEST,LU,NZAP
      WRITE(6,908) NPHASE
      GO TO 100
C.....
 5000 VXCALC=2.*DELX/VEL/SAMPRT
      CALL SETEX(SAMPRT,DB,GAIN,NT,KNORM,NX,IHW)
      IF(JOB.EQ.6) CALL WAVEIN(R,NX,NT,IHW,NORM)
      IF(JOB.EQ.7) CALL WAVEIN(C,NX,NT,IHW,NORM)
      IF(JOB.EQ.8) REWIND 11
      WRITE(6,900) JOB,NX,NT,NZ,ICELZ,VEL,DELX,SAMPRT,NORM,KNORM,
     @             DB,PLOT,LS,PW,XDIP,A,RHO,VXCALC,SHOT,SMAG
      WRITE(6,908) NPHASE
      GO TO 100
  555 WRITE(6,906)
      END FILE 9
      STOP
C.....   ERROR CONDITIONS
  666 WRITE(6,907)
      STOP
  888 WRITE(6,909)
      STOP
  897 WRITE(6,912) NX
      STOP
  898 WRITE(6,910) A
      STOP
  899 WRITE(6,911) NZ
      STOP
  900 FORMAT(' &PARAMS JOB=',I1,',NX=',I3,',NT=',I4,',NZ=',I2,
     @ ',IDELZ=',I3,',VEL=',F7.0,',DELX=',F5.1,',SAMPRT=',F6.4,
     @ ',NORM=',I5,',KNORM=',I1,',DB=',F6.1,',PLOT=',I2,',LS=',
     @ I2,',','/1H ,10X,'PW=',F3.0,',XDIP=',F4.1,',A=',F7.6,',RHO=',
     @ F5.3,',VXCALC=',F6.2,',SHOT=',F6.2,',SMAG=',F5.1,',&END')
  901 FORMAT(' &MGATES N1S=',I3,',N2S=',I3,',N1L=',I3,',N2L=',
     @ I3,',&END')
```

```
  902 FORMAT('  &EGATES  N1E=',I3,',N2E=',I3,',N3E=',I3,',N4E=',
     @ I3,',NEST=',I2,',LU=',I2,',NZAP=',I2,',&END')
  904 FORMAT(' ***BEGIN JOB***',33X,'LINE IDENT= ',2A8,41X,
     @'***BEGIN JOB***')
  905 FORMAT(1H0,131(1H-))
  906 FORMAT('0***END JOB***',105X,'***END JOB***')
  907 FORMAT('0--MAIN--MEMORY REQUEST TOO LARGE--')
  908 FORMAT(1H ,59(1H-),' END PHASE ',I1,1X,59(1H-)/)
  909 FORMAT('0--MAIN--ERR. ON NAMELIST READ--')
  910 FORMAT('0--MAIN--UNSTABLE SCHEME, ABORT.  A=',E14.6)
  911 FORMAT('0--MAIN-- NZ =',I6,'  TOO BIG')
  912 FORMAT('0--MAIN-- NX =',I6,'  TOO BIG')
      END
C.....
C*************************************************************************
C************************** S P O N G E **********************************
C*************************************************************************
C.....
      SUBROUTINE SPONGE(C,R,N1S,N1L,N2S,N2L,SS,IDELZ,
     @                  A,DOWN,SIGMAU)
C.....................NOAH FORWARD VERSION 10.5....6/19/74..............
C.....   GRIDS  R AND DOWN ARE OVERLAID
      REAL*4 C(NX,NT),R(NX,NT),SIGMAU(NX,NZ),DOWN(NX,NT),SS(LSS)
      COMMON /BLOCK1/ UPSUM(&NX)
      COMMON /GLOBAL/ RHO,NX,NT,NZ,SHOT,SMAG,LSS
      INTEGER X,Z,T
C.....
      IDZ=IDELZ/2
C.....
      REWIND 13
C.....
      DO 1001 X=1,NX
 1001 UPSUM(X)=0.0
      DO 1011 T=1,N1S
 1011 WRITE(13) UPSUM
C.....
C.....  GENERATE INITIAL SHOT PATTERN
      CALL VSHOT(DOWN,SS)
C.....
      CALL SCANZ(C,DOWN,SIGMAU,IDELZ,A)
      M1L=N1L
      IF(M1L.LE.LSS) M1L=0
      LMIN=N1S+1
      LMAX=N2S+LSS+1
      IF(M1L.EQ.0) LMAX=MIN0(N2S+N2L,NT)
      DO 3031 T=LMIN,LMAX
      CALL SCAN1(N1S,T,T)
      CALL SCAN0(1,N1S-1,T)
      WRITE(13) UPSUM
      DO 3021 X=1,NX
      DOWN(X,T)=DOWN(X,T)+UPSUM(X)
 3021 UPSUM(X)=0.0
      IF(MOD(T+IDZ,IDELZ).EQ.0) CALL FAST1S(DOWN,NX,MIN0(T+IDZ,NT),
     @                                      A,+1,T)
```

```
 3031 CONTINUE
C.....
      IF(M1L.EQ.0) GO TO 3200
      LMIN=MINO(LMAX+1,NT)
      LMAX=MINO(N2S+N2L,NT)
      DO 3131 T=LMIN,LMAX
      MAX=MAXO(N2S+1,T-M1L+1)
      CALL SCAN1(T-LSS,T,T)
      CALL SCAN0(MAX,T-LSS-1,T)
      CALL SCAN1(N1S,MAX-1,T)
      CALL SCAN0(1,N1S-1,T)
      WRITE(13) UPSUM
      DO 3121 X=1,NX
      DOWN(X,T)=DOWN(X,T)+UPSUM(X)
 3121 UPSUM(X)=0.0
      IF(MOD(T+IDZ,IDELZ).EQ.0) CALL FAST15(DOWN,NX,MINO(T+IDZ,NT),
     @                                      A,+1,T)
 3131 CONTINUE
C.....
 3200 LMIN=MINO(LMAX+1,NT)
      DO 3231 T=LMIN,NT
      IF(M1L.EQ.0) GO TO 3201
      CALL SCAN1(T-LSS,T,T)
      CALL SCAN0(T-M1L+1,T-LSS-1,T)
 3201 CALL SCAN1(T-N2L+1,T-M1L,T)
      CALL SCAN0(N2S+1,T-N2L,T)
      CALL SCAN1(N1S,N2S,T)
      CALL SCAN0(1,N1S-1,T)
      WRITE(13) UPSUM
      DO 3221 X=1,NX
      DOWN(X,T)=DOWN(X,T)+UPSUM(X)
 3221 UPSUM(X)=0.0
      IF(MOD(T+IDZ,IDELZ).EQ.0) CALL FAST15(DOWN,NX,MINO(T+IDZ,NT),
     @                                      A,+1,T)
 3231 CONTINUE
C.....
      REWIND 13
      DO 4001 T=1,NT
      READ(13) UPSUM
      DO 4001 X=1,NX
 4001 R(X,T)=UPSUM(X)
C.....
      RETURN
      END
C.....
C*******************************************************************************
C********************************* V S H O T ***********************************
C*******************************************************************************
C.....
      SUBROUTINE VSHOT(DOWN,S)
C..................... SHOT PATTERN ..................................
      COMMON /GLOBAL/ RHO,NX,NT,NZ,SHOT,SMAG,LS
      REAL*4 DOWN(NX,NT),S(LS)
      INTEGER X,T
```

```
C.....
      LSHOT=SHOT
      WSHOT=(SHOT-LSHOT)*10.
C.....
      DO 1002 T=1,NT
      DO 1002 X=1,NX
 1002 DOWN(X,T)=0.0
C.....
      L1=MAX0(LSHOT-IFIX(WSHOT/2.+.4999),1)
      L2=MIN0(LSHOT+IFIX(WSHOT/2.+.4999),NX)
      IF(LSHOT.LE.0) L1=1
      IF(LSHOT.LE.0) L2=NX
C.....
      DO 1102 T=1,LS
      STEMP1=S(T)*SMAG
      DO 1102 X=L1,L2
 1102 DOWN(X,T)=STEMP1
      IF(LSHOT.LE.0) RETURN
C.....
      LS2=LS/2
      DO 1202 T=1,LS
      STEMP1=DOWN(L1,T)/S(LS2+1)
      STEMP2=DOWN(L2,T)/S(LS2+1)
      DO 1202 K=1,LS2
      X=L1-K
      IF(X.GE.1) DOWN(X,T)=STEMP1*S(LS2-K+1)
      X=L2+K
      IF(X.LE.NX) DOWN(X,T)=STEMP2*S(LS2-K+1)
 1202 CONTINUE
C.....
      RETURN
      END
C.....
C*************************************************************************
C************************** S C A N Z ***********************************
C*************************************************************************
C.....
C.....          2-D FORWARD UPCOMING WAVE PROPAGATOR (EXPLICIT)
C.....
      SUBROUTINE SCANZ (C,DOWN,SIGMAU,IDELZ,A)
      REAL*4 C(NX,NT),DOWN(NX,NT),SIGMAU(NX,NZ)
      COMMON /BLOCK1/ UPSUM(&NX),SOURCE(&NX)
      COMMON /BLOCK2/ TEMPT(&NX),TEMPU(&NX),TEMPS(&NX)
      COMMON /GLOBAL/ RHO,NX,NT,NZ
      INTEGER X,Z,T
C.....
C.....  ENTRY 'SCANZ' CLEARS THE UP-COMING WAVE INTEGRATOR, SETS SOME
C.....   CONSTANTS AND ESTABLISHES ADDRESSABILITY FOR PASSED ARRAYS.
C.....
      IDZ=IDELZ/2
      NXM1=NX-1
      A2=2.*A
      AH=A/2.
C.....
```

```
      DO 1012 Z=1,NZ
      DO 1012 X=1,NX
 1012 SIGMAU(X,Z)=0.0
      DO 1021 X=1,NX
      SOURCE(X)=0.0
 1021 UPSUM(X)=0.0
      RETURN
C.....
      ENTRY SCAN1 (MIN,MAX,T)
C.....
C.....  ENTRY 'SCAN1' INTEGRATES  D  UP=-(V/2)*D  UP-D SOURCE  FROM
C.....   Z='MAX' TO Z='MIN'          ZT          XX     T
C.....   UPON ENTRY 'SOURCE' CONTAINS THE SOURCE TERM AT Z='MAX'+1 AND
C.....   'UP' THE UP-COMING WAVEFIELD AT Z='MAX'+1 .
C.....
      DO 2001 X=1,NX
 2001 SOURCE(X)=0.0
      MAD=MIN+MAX
      DO 2072 IZ=MIN,MAX
      Z=MAD-IZ
      KZ=1+Z/IDELZ
      DO 2011 X=1,NX
      TEMPS(X)=SOURCE(X)
 2011 SOURCE(X)=C(X,Z)*DOWN(X,T-Z+1)
      IF(MOD(Z+IDZ,IDELZ).NE.0) GO TO 2040
      DO 2021 X=1,NX
      TEMPU(X)=UPSUM(X)
 2021 TEMPT(X)=A2*(UPSUM(X)+SIGMAU(X,KZ))+AH*(SOURCE(X)+TEMPS(X))
      DO 2031 X=2,NXM1
      UPSUM(X)=UPSUM(X)+0.5*(SOURCE(X)+TEMPS(X))+
     @          TEMPT(X-1)+TEMPT(X+1)-TEMPT(X)-TEMPT(X)
 2031 SIGMAU(X,KZ)=RHO*SIGMAU(X,KZ)+UPSUM(X)+TEMPU(X)
      UPSUM(1)=UPSUM(2)
      SIGMAU(1,KZ)=RHO*SIGMAU(1,KZ)+UPSUM(1)+TEMPU(1)
      UPSUM(NX)=UPSUM(NX-1)
      SIGMAU(NX,KZ)=RHO*SIGMAU(NX,KZ)+UPSUM(NX)+TEMPU(NX)
      GO TO 2072
 2040 DO 2051 X=1,NX
 2051 UPSUM(X)=UPSUM(X)+0.5*(SOURCE(X)+TEMPS(X))
 2072 CONTINUE
      RETURN
C.....
      ENTRY SCAN0 (MIN,MAX,T)
C.....  ENTRY 'SCAN0' INTEGRATES C  UP=-(V/2)*D  UP  FROM Z='MAX'
C.....   TO Z='MIN'                  ZT          XX
C.....
      MAD=MAX+MIN
      DO 3042 IZ=MIN,MAX
      Z=MAD-IZ
      KZ=1+Z/IDELZ
      IF(MOD(Z+IDZ,IDELZ).NE.0) GO TO 3042
      DO 3021 X=1,NX
      TEMPU(X)=UPSUM(X)
 3021 TEMPT(X)=A2*(UPSUM(X)+SIGMAU(X,KZ))
```

```
      DO 3031 X=2,NXM1
      UPSUM(X)=UPSUM(X)+TEMPT(X-1)+TEMPT(X+1)-TEMPT(X)-TEMPT(X)
 3031 SIGMAU(X,KZ)=RHO*SIGMAU(X,KZ)+UPSUM(X)+TEMPU(X)
C.....
      UPSUM(1)=UPSUM(2)
      SIGMAU(1,KZ)=RHO*SIGMAU(1,KZ)+UPSUM(1)+TEMPU(1)
      UPSUM(NX)=UPSUM(NX-1)
      SIGMAU(NX,KZ)=RHO*SIGMAU(NX,KZ)+UPSUM(NX)+TEMPU(NX)
 3042 CONTINUE
      RETURN
C.....
      END
C.....
C*************************************************************************
C***************************** F L O O D *********************************
C*************************************************************************
C.....
      SUBROUTINE FLOOD(C,R,N1S,N1L,N2S,N2L,LU,IDELZ,
     @                 A,UP,NEST,N1E,N2E,N3E,N4E)
C.....
C.....................NCAH INVERSE VERSION 7.0.....6/23/74..........
C.....     GRIDS  UP AND C ARE OVERLAID
C.....           ALSO BOTH OVERLAID ONTO COMMON BLOCK $GRID1 (CAREFUL)
C.....
      REAL*4 C(NX,NT),R(NX,NT),UP(NX,NT)
      COMMON /BLOCK1/ SIGMAL(&NX)
      COMMON /GRID$1/ AA(&ME,&LU),B(&ME),Y(&NT),TEMP(&NT)
      COMMON /BLOCK2/ U(&LU)
      COMMON /GLOBAL/ RHO,NX,NT,NZ,SHOT,SMAG,LS,NZAP
      INTEGER X,Z,T
C.....
      LSHOT=SHOT
      NEINIT=MAX0(1,LSHOT)
      NESTOP=NEINIT+NEST-1
      IDZ=(IDELZ/2)-1
      N1SP1=N1S+1
      N2SP1=N2S+1
      LUM1=LU-1
      M=N4E-N3E+1
      M2=2*M+LUM1
      REWIND 13
      WRITE(13) R
C.....
C.....   MIGRATE PRIOR TO ESTIMATING SOURCE WAVEFORM INVERSE
      NZE=MIN0(N4E+NZAP*IDELZ,NT)
      NTZ=NZE-N1S+1
      NZE=N1S/IDELZ
      DO 4021 Z=1,NZE
 4021 CALL FAST15(R(1,N1S),NX,NTZ,A,-1,Z)
C.....
C.....   ESTIMATE SOURCE WAVEFORM INVERSE
      DO 2021 X=NEINIT,NESTOP
      DO 2020 T=1,NT
 2020 TEMP(T)=R(X,T)
```

```
 2021 CALL SQUASH(TEMP,NT,L,LL,N1E,N2E,N3E,N4E,AA,B,M2,Y)
C.....  RECAL ORIGINAL DATA
      REWIND 13
      READ(13) R
      DO 2024 X=1,NX
      DO 2023 T=N1S,NT
      UP(X,T)=R(X,T)
      KMAX=MINO(NT-T+1,LU)
      SUM=0.0
      DO 2022 K=1,KMAX
 2022 SUM=SUM+U(K)*R(X,T+K-1)
 2023 R(X,T)=SUM
 2024 CONTINUE
C.....
      DO 2041 T=1,N1S
      DO 2041 X=1,NX
      R(X,T)=0.0
 2041 C(X,T)=0.0
      DO 2131 X=1,NX
 2131 R(X,1)=1.0
C.....
      CALL SWEEPT(C,R,UP,A)
      DO 1022 Z=2,N1S
      IF(MOD(Z+IDZ,IDELZ).NE.0) GO TO 1022
      CALL FAST15(R,NX,NT-Z+1,A,+1,-Z)
      DO 1011 X=1,NX
 1011 SIGMAU(X)=0.0
      CALL SWEEP0(N1S,NT,Z)
 1022 CONTINUE
C.....
      DO 3032 Z=N1SP1,N2SP1
      IF(MOD(Z+IDZ,IDELZ).NE.0) GO TO 3021
      CALL FAST15(R,NX,NT-Z+1,A,+1,-Z)
      CALL ZAP(UP,R,C(1,Z-1),C(1,Z),A,IDELZ,Z)
      DO 3121 X=1,NX
 3121 SIGMAU(X)=0.0
      CALL SWEEP1 (Z+1,NT,Z)
      GO TO 3032
 3021 CONTINUE
      CALL SWEEP2(Z+1,NT,Z)
 3032 CONTINUE
C.....
      M1L=N1L
      IF(M1L.LE.4) M1L=0
C.....
      LMIN=N2S+2
      NTM1=NT-1
      DO 5051 Z=LMIN,NTM1
      IF(MOD(Z+IDZ,IDELZ).NE.0) GO TO 5021
      CALL FAST15(R,NX,NT-Z+1,A,+1,-Z)
      CALL ZAP(UP,R,C(1,Z-1),C(1,Z),A,IDELZ,Z)
      DO 5121 X=1,NX
 5121 SIGMAU(X)=0.0
      CALL SWEEP0(Z+N2L,NT,Z)
```

```
      CALL SWEEP1(Z+M1L+1,Z+N2L-1,Z)
      IF(M1L.EQ.0) GO TO 5051
      CALL SWEEP0(Z+1,Z+M1L,Z)
      GO TO 5051
 5021 CONTINUE
      CALL SWEEP2(Z+M1L+1,Z+N2L-1,Z)
 5051 CONTINUE
C.....
      RETURN
      END
C.....
C*************************************************************************
C************************** S W E E P T **********************************
C*************************************************************************
C.....
C.....              2-D INVERSE UPCOMING WAVE PROPAGATOR
C.....
      SUBROUTINE SWEEPT (C,R,UP,A)
      REAL*4 C(NX,NT),R(NX,NT),UP(NX,NT)
      COMMON /BLOCK1/ SIGMAU(&NX),SOURCE(&NX)
      COMMON /BLOCK2/ F(&NX),OLDU(&NX)
      COMMON /GLOBAL/ RHO,NX,NT
      INTEGER X,Z,T
C.....
      A2=2.*A
      AH=A/2.
      NXM1=NX-1
      RETURN
C.....
      ENTRY SWEEP0 (MIN,MAX,Z)
C.....  MIGRATES FROM T='MAX' TO T='MIN' WITH DIFFRACTIONS+NO SOURCE
C.....
      IF(MIN.GE.NT) RETURN
      MAXMIN=MIN0(MAX,NT)
      MAD=MIN+MAXMIN
      DO 2041 IT=MIN,MAXMIN
      T=MAD-IT
      DO 2011 X=1,NX
      OLDU(X)=UP(X,T)
 2011 F(X)=A2*(OLDU(X)+SIGMAU(X))
      DO 2021 X=2,NXM1
      UP(X,T)=OLDU(X)+F(X-1)+F(X+1)-F(X)-F(X)
 2021 SIGMAU(X)=RHO*SIGMAU(X)+UP(X,T)+OLDU(X)
      UP(1,T)=UP(2,T)
      SIGMAU(1)=RHO*SIGMAU(1)+UP(1,T)+OLDU(1)
      UP(NX,T)=UP(NX-1,T)
      SIGMAU(NX)=RHO*SIGMAU(NX)+UP(NX,T)+OLDU(NX)
 2041 CONTINUE
      RETURN
C.....
      ENTRY SWEEP1 (MIN,MAX,Z)
C.....  MIGRATES FROM MAX TO MIN WITH DIFFRACTIONS, WITH SOURCES
C.....
      IF(MIN.GE.NT) RETURN
```

```
      MAXMIN=MIN0(MAX,NT)
      MAD=MIN+MAXMIN
      DO 3041 IT=MIN,MAXMIN
      T=MAD-IT
      DO 3011 X=1,NX
      OLDU(X)=UP(X,T)
      SOURCE(X)=C(X,Z)*R(X,T-Z+1)+C(X,Z-1)*R(X,T-Z+2)
 3011 F(X)=A2*(OLDU(X)+SIGMAU(X))-AH*SOURCE(X)
      DO 3021 X=2,NXM1
      UP(X,T)=OLDU(X)-0.5*SOURCE(X)+F(X+1)+F(X-1)-F(X)-F(X)
 3021 SIGMAU(X)=RHO*SIGMAU(X)+UP(X,T)+OLDU(X)
      UP(1,T)=UP(2,T)
      SIGMAU(1)=RHO*SIGMAU(1)+UP(1,T)+OLDU(1)
      UP(NX,T)=UP(NX-1,T)
      SIGMAU(NX)=RHO*SIGMAU(NX)+UP(NX,T)+OLDU(NX)
 3041 CONTINUE
      RETURN
C.....
      ENTRY SWEEP2 (MIN,MAX,Z)
C.....  MIGRATES FROM MAX TO MIN WITHOUT DIFFRACTIONS, WITH SOURCES
C.....
      IF(MIN.GE.NT) RETURN
      MAXMIN=MIN0(MAX,NT)
      MAD=MIN+MAXMIN
      DO 4011 IT=MIN,MAXMIN
      T=MAD-IT
      DO 4021 X=1,NX
 4021 UP(X,T)=UP(X,T)-0.5*(C(X,Z)*R(X,T-Z+1)+C(X,Z-1)*R(X,T-Z+2))
 4011 CONTINUE
      RETURN
      END
C.....
C*****************************************************************
C*************************   Z  A  P   ***************************
C*****************************************************************
C.....
C.....                  INVERSE REFLECTION COEFFICIENT ESTIMATOR
C.....
      SUBROUTINE ZAP(UP,R,COLD,CNEW,A,IDELZ,Z)
      REAL*4 UP(NX,NT),R(NX,NT),COLD(NX),CNEW(NX)
      INTEGER X,Z,T
      COMMON /BLOCK1/ SOURCE(&NX),F(&NX)
      COMMON /BLOCK2/ EPLUS(&NX),EMINUS(&NX),SIGMAU(&NX)
      COMMON /GLOBAL/ RHO,NX,NT,NZ,SHOT,SMAG,LS,NZAP
C.....
      A2=2.*A
      AH=A/2.
      NXM1=NX-1
      MAXMIN=MIN0(Z+NZAP*IDELZ,NT)
      MAD=Z+MAXMIN
C.....
      DO 1011 X=1,NX
 1011 SIGMAU(X)=0.0
      DO 1041 IT=Z,MAXMIN
```

```
      T=MAD-IT
      DO 1021 X=1,NX
      SOURCE(X)=COLD(X)*R(X,T-Z+2)+R(X,T-Z+1)
 1021 F(X)=A2*(UP(X,T)+SIGMAU(X))-AH*SOURCE(X)
      DO 1031 X=2,NXM1
      EPLUS(X)=UP(X,T)-0.5*SOURCE(X)+F(X+1)+F(X-1)-F(X)-F(X)
 1031 SIGMAU(X)=RHO*SIGMAU(X)+EPLUS(X)+UP(X,T)
      EPLUS(1)=EPLUS(2)
      SIGMAU(1)=RHO*SIGMAU(1)+EPLUS(1)+UP(1,T)
      EPLUS(NX)=EPLUS(NX-1)
 1041 SIGMAU(NX)=RHO*SIGMAU(NX)+EPLUS(NX)+UP(NX,T)
C.....
      DO 2011 X=1,NX
 2011 SIGMAU(X)=0.0
      DO 2041 IT=Z,MAXMIN
      T=MAD-IT
      DO 2021 X=1,NX
      SOURCE(X)=COLD(X)*R(X,T-Z+2)-R(X,T-Z+1)
 2021 F(X)=A2*(UP(X,T)+SIGMAU(X))-AH*SOURCE(X)
      DO 2031 X=2,NXM1
      EMINUS(X)=UP(X,T)-0.5*SOURCE(X)+F(X+1)+F(X-1)-F(X)-F(X)
 2031 SIGMAU(X)=RHO*SIGMAU(X)+EMINUS(X)+UP(X,T)
      EMINUS(1)=EMINUS(2)
      SIGMAU(1)=RHO*SIGMAU(1)+EMINUS(1)+UP(1,T)
      EMINUS(NX)=EMINUS(NX-1)
 2041 SIGMAU(NX)=RHO*SIGMAU(NX)+EMINUS(NX)+UP(NX,T)
C.....
      DO 3011 X=1,NX
 3011 CNEW(X)=(EMINUS(X)+EPLUS(X))/(R(X,1)+EMINUS(X)-EPLUS(X))
C.....
      RETURN
      END
C.....
C*************************************************************************
C************************** F A S T 1 5 ******************************
C*************************************************************************
C.....
      SUBROUTINE FAST15(WAVE,NX,NT,A,MODE,Z)
      REAL*4 WAVE(NX,NT)
      INTEGER X,T,Z
      COMMON /BLOCK2/ SIGMA(&NX),TEMPU(&NX),TEMPT(&NX)
      COMMON /GLOBAL/ RHO
      CONST=2.*A
      DO 1010 X=1,NX
 1010 SIGMA(X)=0.
      NX1=NX-1
      DO 1050 JT=1,NT
      T=JT
      IF(MODE.EQ.-1) T=NT+1-JT
      DO 1020 X=1,NX
      TEMPU(X)=WAVE(X,T)
 1020 TEMPT(X)=CONST*(TEMPU(X)+SIGMA(X))
      DO 1030 X=2,NX1
      WAVE(X,T)=TEMPU(X)+TEMPT(X-1)+TEMPT(X+1)-TEMPT(X)-TEMPT(X)
```

```
 1030 SIGMA(X)=RHO*SIGMA(X)+WAVE(X,T)+TEMPU(X)
      WAVE(1,T)=WAVE(2,T)
      SIGMA(1)=RHO*SIGMA(1)+WAVE(1,T)+TEMPU(1)
      WAVE(NX,T)=WAVE(NX-1,T)
      SIGMA(NX)=RHO*SIGMA(NX)+WAVE(NX,T)+TEMPU(NX)
 1050 CONTINUE
      RETURN
      END
C.....
C*************************************************************************
C************************** N O A H 1 D **********************************
C*************************************************************************
C.....
C.....            1-D FORWARD AND INVERSE ALGORITHM
C.....                          (2-D GRID IS NOT REQUIRED)
      SUBROUTINE NOAH1D(C,R,N1S,N1L,N2S,N2L,S,LU,
     @                  SIGN,NEST,N1E,N2E,N3E,N4E)
C.....
      REAL*4 C(NX,NT),R(NX,NT),S(LS)
      INTEGER X,T
C.....
      COMMON /GRID$1/ AA(&ME,&LU),B(&ME),Y(&NT),TEMP(&NT)
      COMMON /BLOCK2/ U(&LU)
      COMMON /GLOBAL/ RHO,NX,NT,NZ,SHCT,SMAG,LS
C.....
      IF(SIGN) 2001,1001,1001
C.....
C.....              R <- 0   0=T<=N1S     ARRIVAL FREE REGION
 1001 DO 1012 T=1,N1S
      DO 1012 X=1,NX
 1012 R(X,T)=0.0
C.....
C.....
      LMIN=N1S+1
      LMAX=N2S+N1L
      DO 1022 T=LMIN,LMAX
      KMAX=MIN0(T-1,N2S)
      DO 1022 X=1,NX
      SUM=C(X,T)
      DO 1021 K=N1S,KMAX
 1021 SUM=SUM+C(X,K)*R(X,T-K+1)
 1022 R(X,T)=SUM
C.....
C.....                          T-N1L
C.....      R(T) <- C(T)   +    SUM   C(K) * R(T-K+1)   N2S<T<=N2S+N2L
C.....                         K=N1S
C.....                              SHORT PATH MULTIPLE REGION
      LMIN=LMAX+1
      LMAX=MIN0(N2S+N2L,NT)
      DO 1042 T=LMIN,LMAX
      KMAX=T-N1L
      DO 1042 X=1,NX
      SUM=C(X,T)
      DO 1031 K=N1S,KMAX
```

```
 1031 SUM=SUM+C(X,K)*R(X,T-K+1)
 1042 R(X,T)=SUM
C.....
C.....                              N2S                              N2L
C.....          R(T) <- C(T) + SUM  C(K)*R(T-K+1)  + SUM R(K)*C(T-K+1)
C.....                             K=N1S                            K=N1L
C.....                  FOR    N2S+N2L < T      SHORT AND LONG PATH
      LMIN=MIN0(LMAX+1,NT)
      DO 1072 T=LMIN,NT
      DO 1072 X=1,NX
      SUM=C(X,T)
      DO 1051 K=N1S,N2S
 1051 SUM=SUM+C(X,K)*R(X,T-K+1)
      DO 1061 K=N1L,N2L
 1061 SUM=SUM+R(X,K)*C(X,T-K+1)
 1072 R(X,T)=SUM
C.....
      DO 1092 IT=1,NT
      T=NT-IT+1
      KMAX=MIN0(LS,T)
      DO 1092 X=1,NX
      SUM=0.0
      DO 1081 K=1,KMAX
 1081 SUM=SUM+S(K)*R(X,T-K+1)
 1092 R(X,T)=SUM
C.....
      RETURN
C.....
C.....    INVERSE ALGORITHM FOLLOWS
C.....
 2001 M=N4E-N3E+1
      LUM1=LU-1
      M2=2*M+LUM1
      NEINIT=MAX0(1,IFIX(SHCT))
      NESTOP=NEINIT+NEST-1
      DO 2111 X=NEINIT,NESTOP
      DO 2011 T=1,NT
 2011 TEMP(T)=R(X,T)
 2111 CALL SQUASH(TEMP,NT,U,LU,N1E,N2E,N3E,N4E,AA,B,M2,Y)
      DO 2132 X=1,NX
      DO 2032 T=N1S,NT
      C(X,T)=R(X,T)
      KMAX=MIN0(NT-T+1,LU)
      SUM=0.0
      DO 2021 K=1,KMAX
 2021 SUM=SUM+U(K)*R(X,T+K-1)
 2032 R(X,T)=SUM
 2132 CONTINUE
C.....
      DO 2042 T=1,N1S
      DO 2042 X=1,NX
      R(X,T)=0.0
 2042 C(X,T)=0.0
C.....
```

```
      LMIN=N1S+1
      LMAX=N2S+N1L
      DO 2052 T=LMIN,LMAX
      KMAX=MIN0(T-1,N2S)
      DO 2052 X=1,NX
      SUM=0.
      DO 2051 K=N1S,KMAX
 2051 SUM=SUM-C(X,K)*R(X,T-K+1)
 2052 C(X,T)=C(X,T)+SUM
C.....
      LMIN=LMAX+1
      LMAX=MIN0(NT,N2S+N2L)
      DO 2072 T=LMIN,LMAX
      KMAX=T-N1L
      DO 2072 X=1,NX
      SUM=0.
      DO 2061 K=N1S,KMAX
 2061 SUM=SUM-C(X,K)*R(X,T-K+1)
 2072 C(X,T)=C(X,T)+SUM
C.....
      LMIN=MIN0(LMAX+1,NT)
      DO 2101 T=LMIN,NT
      DO 2101 X=1,NX
      SUM=0.
      DO 2081 K=N1S,N2S
 2081 SUM=SUM-C(X,K)*R(X,T-K+1)
      DO 2091 K=N1L,N2L
 2091 SUM=SUM-R(X,K)*C(X,T-K+1)
 2101 C(X,T)=C(X,T)+SUM
C.....
      RETURN
C.....
      END
C.....
C.....
C*****************************************************************************
C************************** S Q U A S H ***************************************
C*****************************************************************************
C.....
C.....        INVERSE SOURCE WAVEFORM ESTIMATOR
C.....
      SUBROUTINE SQUASH (R,LR,U,LU,N1,N2,N3,N4,A,B,M2,Y)
C.....
C.....          N4                  N2                  2
C.....    MIN    SUM   ( R(T) - U* SUM R(T-K+1) R(K) )
C.....     U    T=N3                K=N1
C.....
C.....    M2=2*(N4E-N3E+1)+LU-1 = NO. EQUATIONS
C.....    N=LU = LENGTH OF FILTER U
C.....
      REAL*4 R(LR),U(LU),A(M2,LU),B(M2),Y(1)
      INTEGER T,FLAG/0/
C.....
      IF(FLAG.NE.0) GO TO 1001
```

```
      DO 1040 T=1,M2
      B(T)=0.0
      DO 1040 K=1,LU
 1040 A(T,K)=0.0
      FLAG=1
 1001 CONTINUE
C.....
      I1=N1-1
      DO 4001 T=1,I1
 4001 R(T)=0.
      I1=N2+1
      I2=N3-1
      DO 4002 T=I1,I2
 4002 R(T)=0.
      I1=N4+1
      DO 4003 T=I1,LR
 4003 R(T)=0.
C.....
      LIMIT=N4+LU-1
      DO 1020 T=N3,LIMIT
      SUM=0.0
      DO 1010 K=N1,N2
 1010 SUM=SUM+R(T-K+1)*R(K)
 1020 Y(T-N3+1)=SUM
C.....
      M=N4-N3+1
      DO 2020 T=1,M
      B(T+M)=R(T+N3-1)
      DO 2010 K=1,LU
 2010 A(T+M,K)=Y(T+K-1)
 2020 CONTINUE
C.....
C.....      APPEND SMOOTHING MATRIX OF (1 -1) ON DIAGONAL
C.....       TO LOW PASS THE ESTIMATOR
      BIGB=0.0
      MZ=2*M
      DO 3010 T=1,M
      ABB=ABS(Y(T))
 3010 IF(ABB.GT.BIGB) BIGB=ABB
      DIF=BIGB/25.
      LUM1=LU-1
      DO 3022 T=1,LUM1
      B(T+MZ)=0.0
      DO 3020 K=1,LU
 3020 A(T+MZ,K)=0.0
C.....
      A(T+MZ,T)=DIF
 3022 A(T+MZ,T+1)=-DIF
C.....
      CALL GOLUB (A,U,B,M2,LU,Y)
C.....
      RETURN
      END
C.....
```

```
C*********************************************************************
C************************** G O L U B ********************************
C*********************************************************************
C.....
      SUBROUTINE GOLUB (A,X,B,M,N,U)
C.....
C.....A(M,N) ; B(M)  GIVEN WITH M>N   SOLVES FOR X(N) SUCH THAT
C.....            || B - AX || = MINIMUM
C.....METHOD OF G.GOLUB, NUMERISCHE MATHEMATIK 7,206-216 (1965)
C.....
      IMPLICIT REAL*8 (D)
      REAL*4 A(M,N),X(N),B(M),U(M)
C.....
C.....  PERFORM N ORTHOGONAL TRANSFORMATIONS TO A(.,.) TO
C.....  UPPER TRIANGULARIZE THE MATRIX
C.....
C.....
C.....
      DO 3010 K=1,N
C.....
      DSUM=0.0D0
      DO 1010 I=K,M
      DAJ=A(I,K)
 1010 DSUM=DSUM+DAJ**2
      IF(DSUM.GT.1.0D-10) GO TO 1015
      WRITE(6,900) DSUM,K
  900 FORMAT('0SINGULARITY IN GOLUB DSUM=',D15.6,5X,I4)
      GO TO 3010
 1015 CONTINUE
      DAI=A(K,K)
      DSIGMA=DSIGN(DSQRT(DSUM),DAI)
      DBI=DSQRT(1.0D0+DAI/DSIGMA)
      DFACT=1.0D0/(DSIGMA*DBI)
      U(K)=DBI
      FACT=DFACT
      KPLUS=K+1
      DO 1020 I=KPLUS,M
 1020 U(I)=FACT*A(I,K)
C.....
C.....  I - UU' IS A SYMMETRIC, ORTHOGONAL MATRIX WHICH WHEN APPLIED
C.....  TO A(.,.) WILL ANNIHILATE THE ELEMENTS BELOW THE PIVOT
C.....  DIAGONAL K
C.....
      DO 2030 J=K,N
C.....
C.....  APPLY THE ORTHOGONAL TRANSFORMATION
C.....
      FACT=0.0
      DO 2010 I=K,M
 2010 FACT=FACT+U(I)*A(I,J)
C.....
      DO 2020 I=K,M
 2020 A(I,J)=A(I,J)-FACT*U(I)
 2030 CONTINUE
```

```
C.....
      FACT=0.0
      DO 2040 I=K,M
 2040 FACT=FACT+U(I)*B(I)
C.....
      DO 2050 I=K,M
 2050 B(I)=B(I)-FACT*U(I)
C.....
 3010 CONTINUE
C.....
C.....  BACK SUBSTITUTE TO RECURSIVELY YIELD X(.)
C.....
      X(N)=B(N)/A(N,N)
      LIM=N-1
C.....
      DO 4020 I=1,LIM
      IROW=N-I
      SUM=0.0
C.....
      DO 4010 J=1,I
 4010 SUM=SUM+X(N-J+1)*A(IROW,N-J+1)
C.....
 4020 X(IROW)=(B(IROW)-SUM)/A(IROW,IROW)
C.....
      RETURN
      END
C.....
C*******************************************************************************
C************************** E A R T H ******************************************
C*******************************************************************************
C.....
      SUBROUTINE EARTH(C,NX,NT,SOURCE,LS,VXCALC,SAMPRT,
     @                 DB,PPW,DISP)
C.....................MODEL BUILDER.....................................
C.....
C.....   ENTRY POINT ARGUMENTS:
C.....     C(NX,NT) = MODEL GRID
C.....     SOURCE(LS) = SOURCE WAVEFORM
C.....     VXCALC = VERTICAL EXAGGERATION IN THE CALCULATION
C.....     SAMPRT = SAMPLING RATE IN SECONDS.
C.....     DB = MINIMUM EXP. GAIN REQUIRED IN DB/SEC. (RETURNED)
C.....
C.....  NAMELIST CARD PARAMETERS - ONE NAMELIST STRING/STRUCTURE ELEMENT
C.....     GEOL= 'SINE'   FOR SINUSOIDAL INTERFACE
C.....           'DOME'   FOR CENTERED DOME OF WIDTH 'FOCUS' GRID POINTS
C.....           'LAYR'   FOR FLAT LAYER
C.....           'DISH'   FOR CENTERD DISH OF WIDTH 'FOCUS'
C.....           'CIRP'   FOR CHIRP
C.....           'MONO'   FOR MONOCLINE WITH FLEXURE 'FOCUS'
C.....           'BLOB'   FOR BLOB OF WIDTH 'FOCUS' AT X='NLOC'
C.....           'HOLE'   FOR SPECIAL STRUCTURE #1 WITH BOTTOM
C.....                    'DEPTH' AND RELIEF 'RELIEF'
C.....     COEFF = REFLECTION COEFFICIENT OF INTERFACE
C.....     DIP = DIP OF INTERFACE IN DEGREES
```

```
C.....      DEPTH = DEPTH IN GRID POINTS TO RH SIDE OR TOP OF STRUCTURE
C.....      NFAULT= 0  NO FAULTS
C.....              1  ONE DOWNTHROWN FAULT WITH THROW 'THROW'
C.....              2  DOWNTHROWN GRABEN WITH THROW 'THROW'
C.....
C.....    THE INPUT FILE IS READ UNTIL  | COEFF | > 1.
C.....
      REAL*4 C(NX,NT),DISP(NX,NT),SOURCE(LS)
      INTEGER GEOL,X,KTYPE(8)
      NAMELIST /MODEL/COEFF,DIP,DEPTH,RELIEF,GEOL,CYCLES,FOCUS,NFAULT,
     @                THROW,NLOC
      RICKY(T)=(1.-2.*ARR*T*T)*EXP(-ARR*T*T)
      FAULT(X)=THROW*MOD(((X-1)*(NFAULT+1))/NX),2)
      DATA KTYPE/'SINE','DOME','LAYR','DISH','CIRP','MONO','BLOB',
     @           'HOLE'/
      ARR=8./PPW/PPW
      DO 1022 JT=1,NT
      DO 1022 X=1,NX
      DISP(X,JT)=0.0
 1022 C(X,JT)=0.0
      MIDDLE=NX/2+1
      FLS2=(LS-1)/2.
      DB=200.
      NSTP=0
 1031 DIP=0.0
      FOCUS=0.0
      CYCLES=1.0
      NFAULT=0
      RELIEF=0.0
      NLOC=NX/2
      CFACT=1.0
      THROW=0.0
      READ(5,MODEL,END=9999,ERR=8888)
      IF(ABS(COEFF).GT.1.0) GO TO 9999
      DO 1041 K=1,8
      IF(GEOL.NE.KTYPE(K)) GO TO 1041
      KPICK=K
      GO TO 1051
 1041 CONTINUE
      WRITE(6,901) GEOL
      STOP
 1051 SLOPE=TAN(DIP*0.01745)*VXCALC
      IF(KPICK.NE.8) GO TO 1101
      XH1=29.*NX/53.
      AH1=DEPTH-RELIEF
      BH1=(DEPTH**2-AH1**2)/XH1**2
      AH1=AH1**2
      XH2=12.*NX/53.
      AH2=DEPTH-0.43*RELIEF
      BH2=(DEPTH**2-AH2**2)/XH2**2
      AH2=AH2**2
      IXH1=30.*NX/53.+.4999
      IXH2=43.*NX/53.+.4999
 1101 CONTINUE
```

```
      NSTR=NSTR+1
      DBEST=-20.*ALOG10(ABS(COEFF))/(DEPTH*SAMPRT)
      DB=AMIN1(DB,DBEST)
      WRITE(6,902) NSTR,GFCL,COEFF,DIP,DEPTH,RELIEF,NFAULT,THROW,
     @  FOCUS,CYCLES,DBEST
      DO 1081 X=1,NX
      BURIAL=(LS/2)+DEPTH+FAULT(X)+SLOPE*(X-1)
      GO TO (2001,2002,2003,2004,2005,2006,2007,2008),KPICK
2001  YD=RELIEF*0.5*(1.-COS((X-1)*CYCLES*6.283/(NX-1)))+BURIAL
      GO TO 2000
2002  YD=AMIN1(((X-MIDDLE)**2)*4.*RELIEF/FOCUS/FOCUS,RELIEF)+BURIAL
      GO TO 2000
2003  YD=BURIAL
      GO TO 2000
2004  YD=AMAX1(RELIEF-((X-MIDDLE)**2)*4.*RELIEF/FOCUS/FOCUS,0.)+BURIAL
      GO TO 2000
2005  YD=RELIEF*.5*(1.-COS((1.+2.*(X-1.)/(NX-1))*
     @  6.283*CYCLES*(X-1.)/(NX-1)))+BURIAL
      GO TO 2000
2006  YD=RELIEF*(.5+.31831*ATAN((X-NX/2)*2./FOCUS))+BURIAL
      GO TO 2000
2007  CONTINUE
      IF(IABS(X-NLOC).GT.IFIX(FOCUS)) GO TO 1081
      YD=BURIAL
      CFACT=COS(1.570796*(X-NLOC)/(FOCUS+.5))
      GO TO 2000
2008  CONTINUE
      IF(X.LE.IXH1) YD=(LS/2)+SQRT(AH1+BH1*(X-1)**2)
      IF(X.EQ.IXH1+1) SLOPE=(SQRT(AH2+BH2*(NX-IXH2)**2)
     @                       -DEPTH)/FLOAT(IXH2-IXH1)
      IF(X.GT.IXH1.AND.X.LT.IXH2) YD=(LS/2)+DEPTH+SLOPE*(X-IXH1)
      IF(X.GE.IXH2) YD=(LS/2)+SQRT(AH2+BH2*(NX-X)**2)
2000  ISTART=YD-FLS2
      IEND=YD+FLS2+.5
      IF(ISTART.LT.1.OR.IEND.GT.NT) GO TO 1081
      DIST=YD-FLS2-ISTART
      C(X,ISTART)=C(X,ISTART)+CCEFF*(1.-DIST)*CFACT
      C(X,ISTART+1)=C(X,ISTART+1)+CCEFF*DIST*CFACT
      DO 1061 JT=ISTART,IEND
      T=YD-JT
1061  DISP(X,JT)=DISP(X,JT)+CCEFF*RICKY(T)
1081  CONTINUE
      GO TO 1031
8888  WRITE(6,903)
      STOP
9999  IF(NSTR.EQ.0) GO TO 1091
      WRITE(6,905) NSTR
      RETURN
1091  WRITE(6,906)
      STOP
 901  FORMAT('0--INVALID STRUCTURE TYPE:',A4)
 902  FORMAT('  &MODEL ',I2,2X,A4,'  COEFF=',F6.3,' DIP=',F6.2,
     @  ' DEPTH=',F6.1,' RELIEF=',F5.1,' NFAULT=',I1,' THROW=',F5.1,
     @  ' FOCUS=',F5.1,' CYCLES=',F4.1,' DB=',F4.0,',&END')
```

```
  903 FORMAT('0--NAMELIST READ ERROR')
  905 FORMAT('  &MODEL -',I2,'- STRUCTURE ELEMENT(S) INSERTED - GC')
  906 FORMAT('  && NO STRUCTURE ELEMENTS FOUND--ABORT')
      END
C.....
C*************************************************************************
C************************** W A V E X ************************************
C*************************************************************************
C.....
      SUBROUTINE WAVEX(A,NX,NT,IOUT,PLOT,NORM)
C.......................... I/O PACKAGE ....................................
      REAL*4 A(NX,NT),GAIN(NT)
      INTEGER*2 IOUT(NT),PLOT*4
      PEAK=0.0
      FNORM=NORM
C.....
      GO TO (50,100,300),KNORM
   50 PEAK=1.0
      GO TO 500
  100 DO 200 J=1,NX
      DO 200 K=1,NT
      X=ABS(A(J,K))*GAIN(K)
      IF(X.GT.PEAK) PEAK=X
  200 CONTINUE
      GO TO 500
  300 DO 400 J=1,NX
      DO 400 K=1,NT
  400 PEAK=PEAK+ABS(A(J,K))*GAIN(K)
      PEAK=PEAK/(NX*NT)
  500 IF(PEAK.LT.1.E-10) PEAK=1.0
C.....
      SCALE=FNORM/PEAK
      DO 700 J=1,NX
      DO 600 K=1,NT
      IOUT(K)=SCALE*GAIN(K)*A(J,K)
  600 CONTINUE
      IF(PLOT.NE.0) WRITE(9,1000) IOUT
 1000 FORMAT(12(255A2))
  700 CONTINUE
C.....
      WRITE(6,902) NX,NT,NORM
  902 FORMAT('  &OUTPUT  ',I3,2H X,I4,' FRAME WRITTEN WITH NORM OF ',I5)
      IF(PLOT.EQ.-1) CALL CLT(NX,NT,A,PEAK)
C.....
      RETURN
C.....
      ENTRY WAVEIN(A,NX,NT,IOUT,NORM)
      FNORM=NORM
      SCALE=1./FNORM
      DO 701 J=1,NX
      READ(11,1000) IOUT
      DO 601 K=1,NT
  601 A(J,K)=IOUT(K)*SCALE/GAIN(K)
  701 CONTINUE
```

```
      WRITE(6,901) NX,NT,NCRM
  901 FORMAT('  &INPUT  ',I3,2H X,I4,' FRAME READ WITH NORM OF ',I5)
      IF(PLOT.EQ.-1) CALL OUT(NX,NT,A,FNORM)
      RETURN
C.....
C.....
      ENTRY SETEX(SAMPRT,DB,GAIN,NT,KNORM,NX,IOUT)
      SCALE=0.1155245*DB*SAMPRT
      DO 800 K=1,NT
  800 GAIN(K)=EXP(K*SCALE)
      RETURN
      END
C.....
C*************************************************************************
C*************************** O U T ***************************************
C*************************************************************************
C.....
      SUBROUTINE OUT(NX,NZ,P,FNORM)
C.......................... PRINTER PLOT ..................................
      REAL*4 P(NX,NZ)
      DIMENSION ICHAR(21),LINE(120)
      DATA ICHAR/'HHHH','GGGG','FFFF','EEEE','DDDD','CCCC','BBBB',
     1  'AAAA','    ','    ','1111','2222','3333','4444','5555',
     2  '6666','7777','8888','9999','0000','****'/
      VR=1./.6
      WRITE(66,900) VR
      NXDONE=0
 1011 NL=MIN0(120,NX-NXDONE)
      DO 1031 ID=1,NZ
      DO 1021 IL=1,NL
      IVAL=10.+(P(IL+NXDONE,ID)*12.)/FNORM
 1021 LINE(IL)=ICHAR(MIN0(21,MAX0(1,IVAL)))
 1031 WRITE(66,901) ID,(LINE(IL),IL=1,NL)
      WRITE(66,902)
      NXDONE=NXDONE+120
      IF(NXDONE.LT.NX) GO TO 1011
      RETURN
  900 FORMAT('  VERTICAL EXAGGERATION OF PRINTER PLOT IS',F5.2,' * THE V
     @ERTICAL EXAGGERATION IN THE CALCULATION')
  901 FORMAT(I4,120A1)
  902 FORMAT('1')
      END
```