

```

C*****
C
C          PRODUCTION SYSTEM
C*****
C
C          DIMENSION STATEMENTS
C*****
C
C          REAL IFBMRF, IOBFRF, LSBMRF, LSBFRF, LIMCRF, MGOBFA, IROBFA, LSTBFA,
$          LMMSRF, MOBFRF, MMPORF, MMPSRF, MFBMRF, MWBMRF, IFNBMA, MTWBMA, LSTBMA,
$          IRFINI, IFNORI, IRNORI, INMIXR, INSHBR, MMSPSR, MMPPRF, MNFBMA, ISINTF,
$          LMPRI, MMSPUP, IRNOI, MNFORI, MTWORI, MNGORI, MNFINI, MNFORF, INBSTI,
$          METWAI, MIXING, MFINOR, MNFBMR, ISINTR, IFNORF, MTWORF, INBMXR, LIMCOA,
$          IFINOR, IFNBMR, METWOR, MTWBMR, IRONOR, IROBFR, IOFGNR, LSTBFR, LSTBMR,
$          MGOBFR, LIBYPR, LIMCOR, LMMPSR, MNFNIP, MMSPOR, MNOORF, MANGOI, MSMPRF,
$          IRFNIP, METWIP, IRNOIP, MNGOIP, MNFNAR, IRFNAR, METWAR, IRNOAR, MMSPIN,
$          INMXRI, INSHBI, MANGOR, MNGOAR, INPBFR, INCONR
C
C          COMMON MMMMM, NNNNN, DT, LU
C
C*****
C          CONTROL STATEMENTS
C*****
C
C          DT=0.125
C          INTER=2.
C          PLTPR=1.
C          PRTPR=0.25
C          LENGTH=19
C          NNNNN=(LENGTH/DT)+1
C
C*****
C          CREATING INPUT AND OUTPUT FILES
C*****
C          OPEN(51,FILE='PROD1.OUT')
C          OPEN(52,FILE='PROD2.OUT')
C          OPEN(53,FILE='PROD3.OUT')
C          OPEN(91,FILE='STEEL.IN')
C*****
C          VALUES FOR CONSTANTS
C*****
777 NN=0
NN=NN+1
AT=1.0
BLMSRF=0.11
BBLTRF=0.7
BLOMRF=0.95
BLTSRF=0.07
BILTOF=0.97
BMLPRF=0.98
BMPPRF=0.4
BFDLOF=1.0
BLSORF=1.0
CKBBRF=0.06
CIBFRF=0.4952
COLORF=2.0
COLORF=1.05
DBLSRF=0.8868
DLBMRF=0.107

```

DOLCRF=0.06
DOLORF=1.0
FALORF=1.0
FALCRF=0.014664
HMCNRF=0.90
HMOPRF=0.95
HMOPRF=1.0
HMREQF=1.25
IFBMRF=0.707
IOBFRF=0.2756
ISINTF=1.25
IFNORF=1.3
LSBMRF=0.06663
LIMCRF=0.09528
LMMSRF=1.0
LSBFRF=0.01408
MMPORF=0.94
MMPPRF=0.3
MMPSRF=1.0
MFBMRF=0.00537
MWBMRF=0.03
MOBFRF=0.0136
MNOORF=1.0
MNFORF=1.0
MTWORF=1.0
QRBFRF=0.00152
QRTORF=1.0
RMCONF=1.0
SQBMRF=0.024
SCRCRF=0.1824
SLGSRF=0.2
SLSCRF=0.5322
SRCCRF=0.5794
STLPRF=0.95
STLBPF=0.05
STBPRF=0.13
STSCRF=0.355
STBFRF=1.2
STLPPF=1.0
SINTOF=0.8
SLSORF=1.0
SCRPOF=1.0
SQFORF=1.0
WIRPRF=1.0
WIRSRF=1.0
WRPPRF=0.57
TCLDEL=1.0
TMFDEL=1.0
TIFDEL=1.0
TMWDEL=1.0
TSQDEL=1.0
TBFDEL=1.0
TIODEL=1.0
TLSDEL=1.0
TQRDEL=1.0
TMODEL=1.0
TSLDEL=1.0

TSCDEL=1.0
TDODEL=1.0
TFRDEL=1.0
TPGDEL=1.0
TRMDEL=1.0
TINDEL=1.0
TAHMTP=1.0
TSHDEL=1.0
TASLAG=1.0
TASTEL=1.0
TASTLP=1.0
TAIFNB=1.0
TAIROB=1.0
TASINT=1.0
TABLOM=1.0
TACLCO=1.0
TAMMSP=1.0
TABMLP=1.0
TAWIRP=1.0
TBILTO=1.0
TASTBY=1.0
TAINPB=1.0
TSLSCL=1.0
TLINCO=1.0
COKORF=0.8
BFCKRF=0.88
SNTPRF=1.0
TASINT=1.0
BFBYPF=0.2039
SINTBF=0.1
HMTOPF=0.5
BFBPRF=0.36
SLGPRF=0.15
STLORF=0.92
RMINPF=1.7
BLOPRF=1.0
MSMPRF=1.0
BLTPRF=1.0
BMPRDF=1.0
WRPRDF=1.0

C

C*****

C INITIAL VALUES FOR LEVELS

C*****

C

BILTUP=1200000.0
BILTIN=30000.0
BMLPUP=450000.0
BFDOLI=220000.0
BFGLSI=350000.0
BSEMIX=4000000.0
BLOMUP=1800000.0
BLOOMI=36000.0
BFDORI=150000.0
BFLORI=1000000.0
COALIP=1000000.0
CLORRI=2500000.0

CALIME=250000.0
COKPRO=2000000.0
DEFBLI=25000.0
FERALI=65000.0
FALORI=4000.0
GSLAGI=1000000.0
HMETUP=5200000.0
IRNORI=800000.0
IRFINI=1800000.0
IRONOI=1500000.0
IFNORI=3000000.0
INSHBI=5200000.0
INBSTI=5000000.0
LMPRI=120000.0
MMSPUP=550000.0
MMSPIN=16000.0
MNFORI=60000.0
MTWORI=100000.0
MNGORI=15000.0
MNFINI=20000.0
MANGOI=10000.0
METWAI=900000.0
MIXING=2500000.0
PIGCAI=50000.0
PPBINS=5000000.0
QRTORI=1500.0
QRTZIN=10000.0
SQRTFI=26000.0
SINTUP=5000000.0
SINTRI=85000.0
SCRAPI=550000.0
SDLORI=7500.0
SMDOLI=250000.0
STELUP=2000000.0
SQFORI=100000.0
SLSORI=300000.0
SCRORI=50000.0
SHBINS=5200000.0
WIREUP=530000.0
WIRODI=400000.0
ASTELO=1600000.0
AINPBF=5200000.0
AIFNBM=1700000.0
AIROBF=280000.0
AHMTPR=2600000.0
ABLOMO=1220000.0
ABILTO=1000000.0
AMMSPO=140000.0
ABMLPP=120000.0
ASTBYP=160000.0
AWIRPO=410000.0
SCRPIP=50000.0
BFDOIP=BFDORI
BFLSIP=BFLORI
IRFNIP=IFNORI
IRNOIP=IRNORI
MNFNIP=MNFORI

METWIP=MTWORI
MNGOIP=MNGORI
QRTZIP=QRTORI
SQFIIP=SQFORI
SMLSIP=SLSORI
FERAIP=FALORI
SMDLIP=SDLORI
ASLAGP=800000.0
INMXRI=2500000.0
HMPGRI=120000.0
ASLSCL=300000.0
ALIMCO=100000.0
COALIN=4000000.0
COKSOR=2000000.0
BFCOKI=550000.0
CKBRZI=300000.0
ASINTO=3400000.0
TIME=1994.0

C
C*****

C SIMULATION STARTS

C*****

DO 100 MMMMM=1,NNNNN

C*****

C

C POLICY 1: 20% INCREASE IN THE PRODUCTIVITY OF HOT METAL & LIQUID STEEL: other
C parameters as in the base run:

C

IF((NN.EQ.2).OR.(NN.EQ.4))THEN
IF(TIME.GT.2003.0)THEN
HMOPRF=1.20
STLPPF=1.20
ENDIF
ENDIF

C

C POLICY 2: 25% INCREASE IN THE PRODUCTIVITY OF HOT METAL & LIQUID STEEL: OTHER
C PARAMETERS AS IN THE BASE RUN:

C

IF((NN.EQ.3).OR.(NN.EQ.5).OR.(NN.EQ.6).OR.(NN.EQ.7).OR.
\$ (NN.EQ.8))THEN
IF(TIME.GT.2003.0)THEN
HMOPRF=1.25
STLPPF=1.25
ENDIF
ENDIF

C

C POLICY 3: 20% INCREASE IN ALLOCATION OF BLOOMS TO BILLETS: OTHER PARAMETERS

AS

C

IN THE BASE RUN:

C

IF((NN.EQ.4).AND.(TIME.GT.2003.0))THEN
BBLTRF=0.84
BLMSRF=0.01
MMPPRF=0.15
ENDIF

C

C POLICY 4: 25% INCREASE IN THE PRODUCTIVITY OF HOT METAL & LIQUID STEEL:
CHANGE IN
C THE PRODUCT MIX(20% INCREASE IN WRM: OTHER PARAMETERS AS IN THE BASE
RUN:
C

IF((NN.EQ.5).AND.(TIME.GT.2003.0))THEN
BMPPRF=0.32
WRPPRF=0.636
BLTSRF=0.044
ENDIF

C
C POLICY 5: 25% INCREASE IN THE PRODUCTIVITY OF HOT METAL & LIQUID
C STEEL: CHANGE IN THE PRODUCT MIX: OTHER PARAMETERS AS IN THE BASE RUN:
C

IF((NN.EQ.6).AND.(TIME.GT.2003.0))THEN
BMPPRF=0.40
WRPPRF=0.60
BLTSRF=0.0
ENDIF

C
C POLICY 6: 25% INCREASE IN THE PRODUCTIVITY OF HOT METAL & LIQUID
C STEEL:10%:10%INCREASE IN PRODUCTIVITY OF BILLETS; CHANGE IN THE PRODUCT
MIX
C (20% DECREASE IN BAR MILL PRODUCTS; 20% INCREASE IN WRM):OTHER
PARAMETERS
C AS IN THE BASE RUN:
C

IF((NN.EQ.7).AND.(TIME.GT.2003.0))THEN
BLTPRF=1.1
BMPPRF=0.32
WRPPRF=0.636
BLTSRF=0.044
ENDIF

C
C POLICY 7: 25% INCREASE IN THE PRODUCTIVITY OF HOT METAL & LIQUID
C STEEL:10% INCREASE IN PRODUCTIVITY OF BILLETS; CHANGE IN PRODUCT MIX:
C OTHER PARAMETERS AS IN THE BASE RUN:
C

IF((NN.EQ.8).AND.(TIME.GT.2003.0))THEN
BLTPRF=1.1
BMPPRF=0.30
WRPPRF=0.70
BLTSRF=0.0
ENDIF

C
C*****

C RATE EQUATIONS

C*****
C

HMTOPR=HMETUP*HMTOPF*HMOPRF
C IF(TIME.GE.2003.0)HMTOPF=0.51
HMCONR=AHMTPR*HMCNRF
HMPIGR=AHMTPR-HMCONR
SLAGPR=HMETUP*SLGPRF
BFBYPR=HMETUP*BFBPRF
SLAGSR=GSLAGI*SLGSRF
SNTBFR=(AHMTPR+ASLAGP)*STBFRF

```

SINTOR=SINTUP*SINTOF*SNTPRF
SIBYPR=SINTUP*(1-SINTOF)
SINTRR=SINTOR-SNTBFR
COKEOR=COKPRO*COKORF
BFCKOR=COKSOR*BFCKRF
CKBZOR=COKSOR*(1-BFCKRF)
CKBBMR=ASINTO*CKBBRF
CKBBMA=CKBRZI/AT
IF(CKBBMR.GT.CKBBMA)CKBBMR=CKBBMA
MNFbMR=ASINTO*MFBMRF
MNFbMA=MNFINI/AT
IF(MNFbMR.GT.MNFbMA)MNFbMR=MNFbMA
IFNBMR=ASINTO*IFBMRF
IFNBMA=IRFINI/AT
IF(IFNBMR.GT.IFNBMA)IFNBMR=IFNBMA
MTWBMR=ASINTO*MWBMRF
MTWBMA=METWAI/AT
IF(MTWBMR.GT.MTWBMA)MTWBMR=MTWBMA
SQFBMR=ASINTO*SQBMRF
SQFBMA=SQRTFI/AT
IF(SQFBMR.GT.SQFBMA)SQFBMR=SQFBMA
DOLBMR=ASINTO*DLBMRF
DOLBMA=BFdOLI/AT
IF(DOLBMR.GT.DOLBMA)DOLBMR=DOLBMA
LSTBMR=ASINTO*LSBMRF
LSTBMA=BFGLSI/AT
IF(LSTBMR.GT.LSTBMA)LSTBMR=LSTBMA
STSCRR=AHMTPR*STSCRF
INBMXR=(CKBBMR+MNFbMR+IFNBMR+MTWBMR+SQFBMR+DOLBMR+LSTBMR+
$ SINTRR+STSCRR)
MGOBFR=AHMTPR*MOBFRF
MGOBFA=MANGOI/AT
IF(MGOBFR.GT.MGOBFA)MGOBFR=MGOBFA
IROBFR=AHMTPR*IOBFRF
IROBFA=IRONOI/AT
IF(IROBFR.GT.IROBFA)IROBFR=IROBFA
LSTBFR=AHMTPR*LSBFRF
LSTBFA=BFGLSI/AT-LSTBMR
IF(LSTBFR.GT.LSTBFA)LSTBFR=LSTBFA
COKBFR=AHMTPR*CIBFRF
COKBFA=BFcOKI/AT
IF(COKBFR.GT.COKBFA)COKBFR=COKBFA
CKBYPR=COKPRO/AT-(COKBFR+CKBBMR)
QRTBFR=AHMTPR*QRBFRF
QRTBFA=QRTZIN/AT
IF(QRTBFR.GT.QRTBFA)QRTBFR=QRTBFA
INSHBR=(IROBFR+LSTBFR+QRTBFR+MGOBFR+COKBFR+SNTBFR)
IF(INSHBR.GE.9000000.0)INSHBR=9000000.0
LIMCOR=ASTELO*LIMCRF
LIMCOA=CALIME/AT
IF(LIMCOR.GT.LIMCOA)LIMCOR=LIMCOA
FRALCR=ASTELO*FALCRF
FRALCA=FERALI/AT
IF(FRALCR.GT.FRALCA)FRALCR=FRALCA
SCRPCR=ASTELO*SCRPRF
SCRPCA=SCRAPI/AT
IF(SCRPCR.GT.SCRPCA)SCRPCR=SCRPCA

```

```

SDOLCR=ASTELO*DOLCRF
SDOLCA=SMDOLI/AT
IF ( SDOLCR .GT. SDOLCA ) SDOLCR=SDOLCA
INMIXR=( HMCONR+LIMCOR+FRALCR+SCRPCR+SDOLCR )
STELOR=STELUP*STLORF*STLPPF
STBYPR=STELUP*( 1-STLORF )
BLOMOR=BLOMUP*BLOMRF*BLOPRF
DEFBOR=BLOMUP*( 1-BLOMRF )
DEFBSR=DEFBLI*DBLSRF
BLOMMR=ABLOMO*MMPPRF
BBILTR=ABLOMO*BBLTRF
BLOMSR=ABLOMO*BLMSRF
MMSPOR=MMSPUP*MMPORF*MSMPRF
MMSPSR=AMMSPO*MMPSRF
DEFMPR=MMSPUP*( 1-MMPORF )
BILTOR=BILTUP*BILTOF*BLTPRF
BIBMLR=ABILTO*BMPPRF
BILWMR=ABILTO*WRPPRF
BILTSR=ABILTO*BLTSRF
BMLPPR=BMLPUP*BMLPRF*BMPRDF
DEFBMR=BMLPUP*( 1-BMLPRF )
LMMPSR=ABMLPP*LMMSRF
WIRPOR=WIREUP*WIRPRF*WRPRDF
DWIRPR=WIREUP*( 1-WIRPRF )
WIRDSR=AWIRPO*WIRSRF
DBILOR=BILTUP*( 1-BILTOF )
IOFGNR=IRONOI/AT-IROBFR
COALOR=COKEOR*COLORF
CALL DELAY( CLORRI , COALOR , R1 , R2 , COLCOR , TCLDEL )
MFINOR=MNFBMR*MNFORF
CALL DELAY( MNFORI , MFINOR , R3 , R4 , MNFNAR , TMFDEL )
IFINOR=( IFNBMR-IOFGNR ) *IFNORF
IF( IFINOR.LE.0.0 ) IFINOR=0.0
CALL DELAY( IFNORI , IFINOR , R5 , R6 , IRFNAR , TIFDEL )
METWOR=MTWBMR*MTWORF
CALL DELAY( MTWORI , METWOR , R7 , R8 , METWAR , TMWDEL )
SQFIOR=SQFBMR*SQFORF
CALL DELAY( SQFORI , SQFIOR , R9 , R10 , SQFNAR , TSQDEL )
BFDLOR=DOLBMR*BFDLOF
CALL DELAY( BFDORI , BFDLOR , R11 , R12 , BFDLAR , TBFDEL )
IRONOR=( AIROBF+AIFNBM ) /AT
CALL DELAY( IRNORI , IRONOR , R13 , R14 , IRNOAR , TIODEL )
BFLSOR=( LSTBFR+LSTBMR ) *BLSORF
CALL DELAY( BFLORI , BFLSOR , R15 , R16 , BFLSAR , TLSDEL )
QRTZOR=QRTBFR*QRTORF
CALL DELAY( QRTORI , QRTZOR , R17 , R18 , QRTZAR , TQRDEL )
MANGOR=MGOBFR*MNOORF
CALL DELAY( MNGORI , MANGOR , R19 , R20 , MNGOAR , TMODEL )
LIBYPR=ASLSCL-ALIMCO
SMLSOR=( LIBYPR+LIMCOR ) *SLSORF
CALL DELAY( SLSORI , SMLSOR , R21 , R22 , SLSCLR , TSLDEL )
FRALOR=FRALCR*FALORF
CALL DELAY( FALORI , FRALOR , R23 , R24 , FEALAR , TFRDEL )
SCRPOR=SCRPCR*SCRPOF
CALL DELAY( SCRORI , SCRPOR , R25 , R26 , SCRPAR , TSCDEL )
SDOLOR=SDOLCR*DOLORF
CALL DELAY( SDLORI , SDOLOR , R27 , R28 , SDOLAR , TDODEL )

```

```
BMSNTR=SNTBFR*ISINTF
CALL DELAY( INBSTI , BMSNTR , R29 , R30 , ISINTR , TRMDEL )
CALL DELAY( INSHBI , INSHBR , R31 , R32 , INPBFR , TSHDEL )
CALL DELAY( HMPGRI , HMPIGR , R33 , R34 , PIGCSR , TPGDEL )
IF( TIME .GE. 2003 .0 ) HMPIGR=0 .0
CALL DELAY( INMXRI , INMIXR , R35 , R36 , INCONR , TINDEL )
```

```
C
C*****
```

```
C PRINTING AND PLOTTING
```

```
C*****
```

```
C
```

```
IF( NN .LT. 9 ) THEN
IF( MMMMM .EQ. 1 ) WRITE( 51 , * ) ' TIME , AHMTPR , ASTELO , ABLOMO , POLICY ' , NN
WRITE( 51 , * ) TIME , AHMTPR , ASTELO , ABLOMO
ENDIF
```

```
C
```

```
IF( NN .LT. 9 ) THEN
IF( MMMMM .EQ. 1 ) WRITE( 52 , * ) ' TIME , AMMSPO , ABILTO , ABMLPP , POLICY ' , NN
WRITE( 52 , * ) TIME , AMMSPO , ABILTO , ABMLPP
ENDIF
```

```
C
```

```
IF( NN .LT. 9 ) THEN
IF( MMMMM .EQ. 1 ) WRITE( 53 , * ) ' TIME , AWIRPO , POLICY ' , NN
WRITE( 53 , * ) TIME , AWIRPO
ENDIF
```

```
C
```

```
C*****
```

```
C LEVEL EQUATIONS
```

```
C*****
```

```
C
```

```
COALIP=COALIP+DT*( COALOR-COLCOR )
MNFNIP=MNFNIP+DT*( MFINOR-MNFNAR )
IRFNIP=IRFNIP+DT*( IFINOR-IRFNAR )
METWIP=METWIP+DT*( METWOR-METWAR )
SQFIIP=SQFIIP+DT*( SQFIOR-SQFNAR )
BFDOIP=BFDOIP+DT*( BFDLOR-BFDLAR )
IRNOIP=IRNOIP+DT*( IRONOR-IRNOAR )
BFLSIP=BFLSIP+DT*( BFLSOR-BFLSAR )
QRTZIP=QRTZIP+DT*( QRTZOR-QRTZAR )
MNGOIP=MNGOIP+DT*( MANGOR-MNGOAR )
SMDLIP=SMDLIP+DT*( SDOLOR-SDOLAR )
SCRPIP=SCRPIP+DT*( SCRPOR-SCRPAR )
FERAIP=FERAIP+DT*( FRALOR-FEALAR )
SMLSIP=SMLSIP+DT*( SMLSOR-SLSCLR )
COKPRO=COKPRO+DT*( COLCOR- ( COKEOR+CKBYPR ) )
COKSOR=COKSOR+DT*( COKEOR- ( BFCKOR+CKBZOR ) )
BFCOKI=BFCOKI+DT*( BFCKOR-COKBFR )
CKBRZI=CKBRZI+DT*( CKBZOR-CKBBMR )
IRONOI=IRONOI+DT*( IRNOAR- ( IROBFR+IOFGNR ) )
BFGLSI=BFGLSI+DT*( BFLSAR- ( LSTBFR+LSTBMR ) )
MNFINI=MNFINI+DT*( MNFNAR-MNFBMR )
IRFINI=IRFINI+DT*( ( IRFNAR+IOFGNR ) - IFNBMR )
METWAI=METWAI+DT*( METWAR-MTWBMR )
SQRTFI=SQRTFI+DT*( SQFNAR-SQFBMR )
BFDOLI=BFDOLI+DT*( BFDLAR-DOLBMR )
QRTZIN=QRTZIN+DT*( QRTZAR-QRTBFR )
MANGOI=MANGOI+DT*( MNGOAR-MGOBFR )
```

```

BSEMIX=BSEMIX+DT*( INBMXR-BMSNTR )
PPBINS=PPBINS+DT*( BMSNTR-ISINTR )
SINTUP=SINTUP+DT*( ISINTR-( SINTOR+SIBYPR ) )
SINTRI=SINTRI+DT*( SINTOR-SNTBFR )
SHBINS=SHBINS+DT*( INSHBR-INPBFR )
HMETUP=HMETUP+DT*( INPBFR-( HMTOPR+SLAGPR+BFBYPR ) )
GSLAGI=GSLAGI+DT*( SLAGPR-SLAGSR )
PIGCAI=PIGCAI+DT*( HMPIGR-PIGCSR )
CALIME=CALIME+DT*( SLSCLR-( LIMCOR+LIBYPR ) )
FERALI=FERALI+DT*( FEALAR-FRALCR )
SCRAPI=SCRAPI+DT*( SCRPAR-SCRPCR )
SMDOLI=SMDOLI+DT*( SDOLAR-SDOLCR )
MIXING=MIXING+DT*( INMIXR-INCONR )
STELUP=STELUP+DT*( INCONR-( STELOR+STBYPR ) )
BLOMUP=BLOMUP+DT*( STELOR-( BLOMOR+DEFBOR ) )
DEFBLI=DEFBLI+DT*( DEFBOR-DEFBSR )
BLOOMI=BLOOMI+DT*( BLOMOR-( BLOMMR+BBILTR+BLOMSR ) )
MMSPUP=MMSPUP+DT*( BLOMMR-( MMSPOR+DEFMPR ) )
MMSPIN=MMSPIN+DT*( MMSPOR-MMSPSR )
BILTUP=BILTUP+DT*( BBILTR-( BILTOR+DBILOR ) )
BILTIN=BILTIN+DT*( BILTOR-( BIBMLR+BILWMR+BILTSR ) )
BMLPUP=BMLPUP+DT*( BIBMLR-( DEFBMR+BMLPPR ) )
LMMPRI=LMMPRI+DT*( BMLPPR-LMMPSR )
WIREUP=WIREUP+DT*( BILWMR-( DWIRPR+WIRPOR ) )
WIRODI=WIRODI+DT*( WIRPOR-WIRDSR )
AIFNBM=SMOOTH( IFNBMR, AIFNBM, TAIFNB )
AIROBF=SMOOTH( IROBFR, AIROBF, TAIROB )
AINPBF=SMOOTH( INPBFR, AINPBF, TAINPB )
AHMTPR=SMOOTH( HMTOPR, AHMTPR, TAHMTP )
ASLAGP=SMOOTH( SLAGPR, ASLAGP, TASLAG )
ASTELO=SMOOTH( STELOR, ASTELO, TASTLP )
ABLOMO=SMOOTH( BLOMOR, ABLOMO, TABLOM )
ABILTO=SMOOTH( BILTOR, ABILTO, TBILTO )
AMMSPO=SMOOTH( MMSPOR, AMMSPO, TAMMSP )
ABMLPP=SMOOTH( BMLPPR, ABMLPP, TABMLP )
AWIRPO=SMOOTH( WIRPOR, AWIRPO, TAWIRP )
ASTBYP=SMOOTH( STBYPR, ASTBYP, TASTBY )
ASLSCL=SMOOTH( SLSCLR, ASLSCL, TSLSCL )
ALIMCO=SMOOTH( LIMCOR, ALIMCO, TLINCO )
ASINTO=SMOOTH( SINTOR, ASINTO, TASINT )
TIME=TIME+DT

```

100

```

CONTINUE
IF(NN.LT.9)GO TO 777
CONTINUE
STOP
END

```

```

C *****
C SUBPROGRAM DELAY3 (THIRD ORDER DELAY)
C *****
SUBROUTINE DELAY(AINST, AIN, R1, R2, R, DEL)
COMMON MMMMM, NNNNN, DT, LU
IF (MMMMM.GT.1) GO TO 20
AIN=AINST
R1=AIN
R2=R1
R=R2
RETURN

```

```

20  R1=R1+(3.*DT/DEL)*(AIN-R1)
    R2=R2+(3.*DT/DEL)*(R1-R2)
    R=R+(3.*DT/DEL)*(R2-R)
    RETURN
    END
C
C
C  SUBPROGRAM FOR PIPELINE DELAY
C
    FUNCTION DELAYP(XIN,DEL,X)
    DIMENSION X(100),XTEMP(100)
    COMMON MMMMM,NNNNN,DT,LU
    N=DEL/DT
    IF(N.GT.100) GO TO 40
    IF(MMMMM.GT.N) GO TO 10
    X(MMMMM)=XIN
    DELAYP=X(1)
    RETURN
10  DELAYP=X(1)
    DO 20 IQ=1,N
20  XTEMP(IQ)=X(IQ)
    X(N)=XIN
    NEW=N-1
    DO 30 IP=1,NEW
30  X(IP)=XTEMP(IP+1)
    RETURN
40  WRITE(LU,*) N
C50  FORMAT(10X,'PLEASE INCREASE THE DIMENSION OF THE VARIABLE 'X' IN
C    1THE SUBPROGRAM DELAYP TO',I5//10X,'ALSO INCREASE THE CORRESPONDING
C    2 DIMENSION OF THE VARIABLE IN THE MAIN PROGRAM ACCORDINGLY'//)
    END
C
C  SUBROUTINE FOR SMOOTHING OF INFORMATION
C
    SUBROUTINE DLINF3(XIN,X1,X2,DLINF,DEL)
    COMMON MMMMM,NNNNN,DT,LU
    IF(MMMMM.GT.1) GO TO 10
    X1=XIN
    X2=X1
    DLINF=X2
    RETURN
10  X1=X1+3.*DT/DEL*(XIN-X1)
    X2=X2+3.*DT/DEL*(X1-X2)
    DLINF=DLINF+3.*DT/DEL*(X2-DLINF)
    RETURN
    END
C
C  SUBPROGRAM FOR DELAY OF VARIABLE ORDER UPTO 20 (TWENTY)
C  XIN IS THE INPUT RATE TO THE DELAY
C  DEL IS THE DELAY PERIOD
C  R IS AN ONE-DIMENSIONED DUMMY VARIABLE
C  NORDER IS THE ORDER OF THE DELAY
C  ROUT IS THE OUT RATE FROM THE DELAY
C
    SUBROUTINE DELVOR(XIN,DEL,R,NORDER,ROUT)
    DIMENSION R(20)
    COMMON MMMMM,NNNNN,DT,LU

```

```

        IF(MMMMM.GT.1)GO TO 20
        DO 10 I=1,NORDER
10     R(I)=XIN
        RETURN
20     R(1)=R(1)+NORDER*DT/DEL*(XIN-R(1))
        DO 30 I=2,NORDER
30     R(I)=R(I)+NORDER*DT/DEL*(R(I-1)-R(I))
        ROUT=R(NORDER)
        RETURN
        END
C     *****
C     SUBPROGRAM SMOOTH ( FOR EXPONENTIAL SMOOTHING )
C     *****
        FUNCTION SMOOTH(Y,AY,DS)
        COMMON MMMMM,NNNNN,DT,LU
        SMOOTH=AY+(DT/DS)*(Y-AY)
        RETURN
        END

C
C     *****
C     SUBPROGRAM FTREND ( FOR TREND EXTRAPOLATION )
C     *****
        SUBROUTINE FTREND(RRF,RSF,PRSF,SDF,DPRSF,FTF,FRSF)
        COMMON MMMMM,NNNNN,DT,LU
        RSF=RSF+(DT/SDF)*(RRF-RSF)
        PRSF=PRSF+(DT/DPRSF)*(RSF-PRSF)
        FRSF=RSF+(FTF/DPRSF)*(RSF-PRSF)
        RETURN
        END

C
C     *****
C     SUBPROGRAM MAXMIN (TO DETERMINE THE OPERATING RANGE)
C     *****
        SUBROUTINE MAXMIN(A,AMIN,AMAX,NAME,NUM)
        CHARACTER*1 NAME(13,6)
        COMMON MMMMM,NNNNN,DT,LU
        IF(MMMMM.GT.1) GO TO 10
        AMIN=A
        AMAX=A
10     IF(AMIN.GT.A) AMIN=A
        IF(AMAX.LT.A) AMAX=A
        IF(MMMMM.EQ.NNNNN) GO TO 20
        RETURN
20     WRITE(LU,30)NUM,(NAME(NUM,I),I=1,6),AMIN,AMAX
30     FORMAT(/20X,'STATISTICS FOR TABLE',I3,3X,'(',6A1,')',/20X,
1'-----',/10X,'OPERATING RANGE',10X,
2'MINIMUM=',F9.3,5X,'MAXIMUM=',F9.3)
        RETURN
        END

C     *****
C     SUBPROGRAM RAMP ( FOR RAMP INPUT )
C     *****
        FUNCTION RAMP(XI,GTHR,RST,RFT)
        COMMON MMMMM,NNNNN,DT,LU
        XXX=MMMMM

```

```

      ST=DT*(XXX-1.0)
      IF(ST.LE.RST) GO TO 5
      IF(ST.GE.RFT) GO TO 10
      RAMP=XI*GTHR*(ST-RST)
      GO TO 15
5     RAMP=0.0
      GO TO 15
10    RAMP=XI*GTHR*(RFT-RST)
15    RETURN
      END

C
C     *****
C     SUBPROGRAM SAMPLE ( FOR SAMPLING A CONTINUOUS VALUE )
C     *****
      SUBROUTINE SAMPLE(SMPL,PNEW,POLD,SMPLI)
      COMMON MMMMM,NNNNN,DT,LU
      XXX=MMMMM
      ST=DT*(XXX-1.0)
      IF(AMOD(ST,SMPLI).EQ.0.0) GO TO 5
      SMPL=POLD
      RETURN
5     SMPL=PNEW
      POLD=PNEW
      RETURN
      END

C
C     *****
C     SUBPROGRAM STEP ( TO GIVE STEP INPUT )
C     *****
      FUNCTION STEP(SHT,STT)
      COMMON MMMMM,NNNNN,DT,LU
      XXX=MMMMM
      ST=DT*(XXX-1.0)
      IF(ST.LT.STT) GO TO 5
      STEP=SHT
      GO TO 10
5     STEP=0.0
10    RETURN
      END

```