

**DECNCM - A FORTRAN PACKAGE FOR
LOAD FLOW ANALYSIS WITH THE
AID OF DECOMPOSITION**

J.W. Bandler, M.A. El-Kady and J. Wojciechowski

SOS-83-11-L

August 1983

© J.W. Bandler, M.A. El-Kady and J. Wojciechowski 1983

No part of this document, computer program, source code, compiled code, related documentation and user manuals, magnetic tape, constituent subprograms, test programs, data and data files may be acquired, copied, reproduced, duplicated, executed, lent, disclosed, circulated, translated, transcribed or entered in any form into any machine without written permission. Address enquiries in this regard to Dr. J.W. Bandler. Neither the authors nor any other person, company, agency or institution make any warranty, express or implied, or assume any legal responsibility for the accuracy, completeness or usefulness of the material presented herein, or represent that its use would not infringe upon privately owned rights. This title page and original cover may not be separated from the contents of this document.

DECNCM - A FORTRAN PACKAGE FOR
LOAD FLOW ANALYSIS WITH THE AID OF DECOMPOSITION

J.W. Bandler, M.A. El-Kady and J. Wojciechowski

Abstract

This document contains a listing of the package DECNCM described in [1] for power system load flow analysis. The package implements the Newton complex method and also employs decomposition techniques for solving large systems of sparse linear equations.

The package DECNCM has been developed for the CDC 170/730 system with the NOS 1.4 level 552 operating system and the Fortran Extended (FTN) version 4.8 compiler. The listing contains a total of 862 lines (including 211 comments) constituting thirteen subroutines. The listing does not include the package CSDSLE for solving large systems of decomposed equations.

This work was supported by the Natural Sciences and Engineering Research Council of Canada under Grant G0647.

The authors are with the Simulation Optimization Systems Research Laboratory and the Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada L8S 4L7.

M.A. El-Kady is also with Ontario Hydro, Toronto, Canada.

J. Wojciechowski is on leave from the Institute of Electronics Fundamentals, Technical University of Warsaw, Warsaw, Poland.

I. INTRODUCTION

DECNCM is a package of thirteen Fortran subroutines for power system load flow analysis. The package implements the Newton complex method [2,3] and also employs decomposition techniques for solving large systems of sparse linear equations by an appropriate call to the package CSDSLE [4].

The package DECNCM and documentation have been developed in Fortran IV for use on the CDC 170/730 system with the NOS 1.4 level 552 operating system. The package is available at McMaster University in the form of a library of binary relocatable subroutines. The library is in the group indirect file LIBDNCM accessible under the charge RJWBAND. The package calls subroutine CSDSLE1 of the package CSDSLE for solving large systems of sparse decomposed equations; the package CSDSLE must thus be available when DECNCM is used. The document does not include the package CSDSLE. Information concerning this package is found in [4].

The package DECNCM contains 862 lines of which 211 are comments.

II. REFERENCES

- [1] J.W. Bandler, M.A. El-Kady and J. Wojciechowski, "DECNCM - A Fortran package for load flow analysis with the aid of decomposition", Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, Report SOS-83-11-L, 1983.
- [2] J.W. Bandler and M.A. El-Kady, "A generalized, complex adjoint approach to power network sensitivities", Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, Report SOS-80-17-R, 1980.
- [3] J.W. Bandler and M.A. El-Kady, "Newton's load flow in complex mode", Proc. ECCTD (The Hague, Holland, August 1981), pp. 500-505.
- [4] J.A. Starzyk and J.W. Bandler, "CSDSLE - A Fortran package for the solution of sparse decomposed systems of linear equations", Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, Report SOS-83-2-U, 1983.

III. LISTING OF THE DECNCM PACKAGE

<u>Subroutine</u>	<u>Number of lines</u> (source text)	<u>Number of words</u> (compiled code)	<u>Listing from page</u>
LFNCD	101	1026	4
CCURR	19	50	6
DECAK	95	520	7
DECDS	69	234	9
DGR	34	100	11
FORMK	106	457	12
INTRN	29	53	14
MDDTF	108	315	15
RHSLD	39	327	17
FRPR	82	442	18
FORMU	43	200	20
FORMYT	59	315	21
RDAT	78	720	22

```

C          1
C          2
C          3
C          4
C          5
C          6
C          7
C          8
C          9
C          10
C          11
C          12
C          13
C          14
C          15
C          16
C          17
C          18
C          19
C          20
C          21
C          22
C          23
C          24
C          25
C          26
C          27
C          28
C          29
C          30
C          31
C          32
C          33
C          34
C          35
C          36
C          37
C          38
C          39
C          40
C          41
C          42
C          43
C          44
C          45
C          46
C          47
C          48
C          49
C          50
C          51
C          52
C          53
C          54
C          55
C          56
C          57
C          58
C          59
C          60
C          61
C          62
C          63
C          64
C          65

SUBROUTINE LFNCD (YT,JRYT,ICYT,BCV,V,BNR,BTYP,DEC,IBV,IBL,CMPWA,IN
1TWA,NB,NTL,ITEL,TIMEL,VEPS,INPT,OTPT,IWRITE) A
SUBROUTINE LFNCD IS THE HIGHEST LEVEL SUBROUTINE FOR SOLVING A
LOAD FLOW PROBLEMS USING THE NEWTON COMPLEX METHOD AND A
THE DECOMPOSITION OF SPARSE SYSTEM OF LINEAR EQUATIONS A
LIBRARY : CSDSLE (SEE SOS-83-2) A
INTEGER JRYT(1),ICYT(1),BNR(1),BTYP(1),DEC(1),IBV(1),IBL(1),INTWA( A
11),OTPT A
COMPLEX YT(1),BCV(1),CMPWA(1),V(1),DV A
NSUB=DEC(1) A
NBE=DEC(2) A
JDS=1 A
JAI=JDS+2*NBE-2 A
JSOLR=JAI+NBE-1 A
JAK=JSOLR+4*NBE-4 A
JJRK=JAK+4*NBE+6*NTL A
JIRK=JJRK+NBE/2+1 A
JICK=JIRK+2*NBE+3*NTL A
JDEC=JICK+2*NBE+3*NTL A
JINTN=JDEC+NBE A
JLNCTH=JINTN+NBE A
JMD=JLNCTH+(NSUB+1)/2 A
JIM=JMD+(5*NSUB+2)/2 A
JIN=JIM-3 A
JCM=JIN+NSUB+15+2***NSUB-2 A
N=NBE-1 A
NR=2*N A
ICALL=0 A
CALL DGR (DEC,IBV,IBL,BNR,CMPWA(JDEC),N) A
CALL SECOND (TS) A
TT=TS A
TA=0 A
ICALL=ICALL+1 A
EPS=0. A
CALL RHSLD (V,YT,JRYT,ICYT,BTYP,BNR,BCV,CMPWA(JDS),CMPWA(JAI),N,OT
1PT,IWRITE) A
CALL FORIK (YT,JRYT,ICYT,BTYP,BNR,V,CMPWA(JAI),CMPWA(JAK),CMPWA(JJ
1RKO,CMPWA(JIRKO),CMPWA(JICK),N,NK,OTPT,IWRITE) A
CALL MDDTF (CMPWA(JIRK),CMPWA(JICK),DEC,CMPWA(JIN),CMPWA(JIM),ENR,
1IBL,CMPWA(JINTN),CMPWA(JLNCTH),CMPWA(JCMD),CMPWA(JMD),NB,NK,ICALL) A
DO 20 ISUB=1,NSUB A
CALL DECDS (DEC,IBL,BNR,CMPWA(JMD),CMPWA(JINTN),CMPWA(JDEC),CMPWA(
1JDS),CMPWA(JCMD),ISUB,N,OTPT,IWRITE) A
CALL DECAK (DEC,IBL,BNR,CMPWA(JIM),CMPWA(JINTN),CMPWA(JDEC),CMPWA(
1JAK),CMPWA(JIRKO),CMPWA(JIRKO),CMPWA(JICK),CMPWA(JCMD),ISUB,N,1,OTPT,
2IWRITE) A
CONTINUE A
CALL CSDSLE1 (CMPWA(JIN),INTWA,CMPWA(JCMD),CMPWA(JIM),CMPWA(JSOLR)) A
IAR=IFIX(REAL(CMPWA(JIN+1))) A
IF (IAR.EQ.1) GO TO 90 A
JJ=JSOLR+1 A
DO 30 I=1,N A
DV=CMPWA(JJ) A
R=CABS(DV) A
L=IFIX(REAL(CMPWA(JJ+1))) A
JJ=JJ+4 A
L=(L+1)/2 A
V(L)=V(L)+DV A

```

```

      IF (R.GT.EPS) EPS=RN          A 66
30    CONTINUE                     A 67
      IF (IWRITE.LT.2) GO TO 40     A 68
      WRITE (OTPT,120) (L,V(BNR(L)),L=1,N) A 69
      CALL SECOND (TIME)          A 70
      TT=TIME-TS                  A 71
      TI=TT-TA                   A 72
      TA=TT                      A 73
      IF (IWRITE.GE.1) WRITE (OTPT,110) ICALL,EPS,TI,TT A 74
      IF (EPS.LE.VEPS) GO TO 70    A 75
      IF (ICALL.GE.1TEL) GO TO 50   A 76
      IF (TT.GE.TIMEL) GO TO 60    A 77
      GO TO 10                     A 78
      A 79
50    IFLAG=1                     A 80
      GO TO 70                     A 81
60    IFLAG=2                     A 82
70    ITEL=ICALL                 A 83
      VEPS=EPS                    A 84
      TIMEL=TT                     A 85
      IF (IWRITE.LT.1) GO TO 30    A 86
      WRITE (OTPT,140) ITEL,VEPS,TIMEL           A 87
      WRITE (OTPT,120)             A 88
      WRITE (OTPT,130) (L,V(BNR(L)),L=1,N) A 89
80    RETURN                      A 90
90    WRITE (OTPT,100) IAR        A 91
      STOP                         A 92
100   FORMAT (///" RETURN FLAG FROM CSDSL: ",I2/) A 93
110   FORMAT (///" IT = ",I2.6X,"EPS =",E12.6,6X,"ITERATION TIME ",F6.3,6X A 94
1, "TOTAL TIME ",F6.3/)         A 95
120   FORMAT (///" VECTOR OF BUS VOLTAGES"/)       A 96
130   FORMAT ((3(3X,I4,1X,2(1X,E13.7))))          A 97
140   FORMAT (///" RESULTS OF ANALYSIS"/" NUMBER OF ITERATIONS:",6X,I3/ A 98
1" ACCURACY OBTAINED:",2X,E10.4/" ANALYSIS TIME:",10X,F6.3," SECOND A 99
2S")                           A 100
      END                          A 101

```

```

C          1
C          2
C          3
C          4
C          5
C          6
C          7
C          8
C          9
C          10
C         11
C         12
C         13
C         14
C         15
C         16
C         17
C         18
C         19
C
C COMPLEX FUNCTION CCURR(V, YT, JRYT, ICYT, M)
C
C THIS FUNCTION SUBPROGRAM CALCULATES THE VALUE OF CURRENT INJECTED
C INTO THE MTH BUS FOR THE GIVEN VECTOR V OF BUS VOLTAGES
C
C INTEGER JRYT(1), ICYT(1)
C COMPLEX V(1), YT(1)
C
C K1=JRYT(M)
C K2=JRYT(M+1)-1
C CCURR=(0.,0.)
C DO 10 I=K1,K2
C CCURR=CCURR+YT(I)*V(ICYT(I))
C 10 CONTINUE
C RETURN
C END

```

```

C          C 1
C          C 2
C          C 3
C          C 4
C          C 5
C          C 6
C          C 7
C          C 8
C          C 9
C          C 10
C          C 11
C          C 12
C          C 13
C          C 14
C          C 15
C          C 16
C          C 17
C          C 18
C          C 19
C          C 20
C          C 21
C          C 22
C          C 23
C          C 24
C          C 25
C          C 26
C          C 27
C          C 28
C          C 29
C          C 30
C          C 31
C          C 32
C          C 33
C          C 34
C          C 35
C          C 36
C          C 37
C          C 38
C          C 39
C          C 40
C          C 41
C          C 42
C          C 43
C          C 44
C          C 45
C          C 46
C          C 47
C          C 48
C          C 49
C          C 50
C          C 51
C          C 52
C          C 53
C          C 54
C          C 55
C          C 56
C          C 57
C          C 58
C          C 59
C          C 60
C          C 61
C          C 62
C          C 63
C          C 64
C          C 65

SUBROUTINE DECAK (DEC, IBL, BNR, MD, INTN, DEG, AK, JRK, IRK, ICK, CMPLXWA, I
1SUB, N, ICALL, OTPT, IWRITE)                                     C 1
SUBROUTINE DECAK INITIALIZES IN THE COMPLEX WORKSPACE VECTOR          C 2
CMPLXWA DATA DESCRIBING THE SUBSYSTEM ISUB (I.E., VECTORS AK,          C 3
NRW, NCOL)                                                       C 4
INTEGER DEC(1), IEL(1), INTN(1), BNR(1), DEG(1), MD(1), IRK(1), ICK(1), JR
1KC(1), OTPT
COMPLEX CMPLXWA(1), AK(1)                                         C 5
NINT IS THE NUMBER OF INTERNAL VERTICES OF A GRAPH REPRESENTING      C 6
SET OF EQUATIONS                                                 C 7
NINT=2*(N-DEC(2))                                                 C 8
CALL INTRN (DEC, IBL, BNR, INTN, 2*N, ISUB)                         C 9
JST (JLT) IS AN INDEX IN VECTOR IBL OF THE FIRST (THE LAST)        C 10
VERTEX OF SUBSYSTEM ISUB                                         C 11
JLT=0
DO 10 I=1,ISUB
JST=JLT+1
JLT=JLT+DEC(I+I+1)
CONTINUE
KK=DEC(ISUB+ISUB+2)                                              C 12
NINTJ IS THE NUMBER OF THE INTERNAL VERTICES OF SUBGRAPH ISUB       C 13
NINTJ=KK+KK
J2=JST+KK-1
ISB=5*ISUB
IA1=MD(ISB-3)-1
IA2=MD(ISB-2)-1
I1=IA1
I2=IA2
IF (IWRITE.LT.3) GO TO 20
WRITE (OTPT,70) ISUB
DO 40 K=JST,J2
TWO ROWS ASSOCIATED WITH AN INTERNAL NODE OF THE SUBSYSTEM       C 14
ISUB ARE NOW initialised                                         C 15
L=BNR(IBL(K))
IST=JRK(L)
ILT=JRK(L+1)-1
IS1=I1
IF (ICALL.EQ.1) IS2=I2
DO 30 J=IST, ILT
I1=I1+1
CMPLXWA(I1)=AK(J)
IF (ICALL.NE.1) GO TO 30
I2=I2+1
CMPLXWA(I2)=CMPLX(FLOAT(INTN(IRK(J))), FLOAT(INTN(ICK(J))))     C 16
CONTINUE
IF (IWRITE.LT.3) GO TO 40
WRITE (OTPT,80) BNR(L)
NE=I1-IS1
WRITE (OTPT,90) (CMPLXWA(IS2+J), CMPLXWA(IS1+J), J=1,NE)           C 17
CONTINUE
J2=J2+1
DO 60 K=J2,JLT

```

```

C          C 66
C          C 67
C          C 68
C          C 69
C          C 70
C          C 71
C          C 72
C          C 73
C          C 74
C          C 75
C          C 76
C          C 77
C          C 78
C          C 79
C          C 80
C          C 81
C          C 82
C          C 83
C          C 84
C          C 85
C          C 86
C          C 87
C          C 88
C          C 89
C          C 90
C          C 91
C          C 92
C          C 93
C          C 94
C          C 95

C TWO ROWS ASSOCIATED WITH A PARTION BUS OF THE SUBSYSTEM ISUB
C ARE NOW INITIALISED.
C
C L=BNR( IBL(K) )          C
C IST=JRK(L)                C
C ILT=JRK(L+1)-1            C
C IS1=I1                     C
C IF ( ICALL.EQ.1) IS2=I2    C
C DO 50 J=IST,ILT           C
C LR= IRK(J)                 C
C LC= ICK(J)                 C
C IF ( INTN(LC).EQ.0) GO TO 50 C
C I1= I1+1                   C
C CMPLXWA( I1)=AK(J)        C
C IF ( LC.GT.NINT) CMPLXWA( I1)=CMPLXWA( I1)/DEG(LR-NINT) C
C IF ( ICALL.NE.1) GO TO 50 C
C I2= I2+1                   C
C CMPLXWA( I2)=CMPLX( FLOAT( INTN(LR) ),FLOAT( INTN(LC) )) C
C
50  CONTINUE                  C
C IF ( IWRITE.LT.3) GO TO 60 C
C WRITE ( OTPT,90) BNR(L)      C
C NE=I1-IS1                  C
C WRITE ( OTPT,90) (CMPLXWA( IS2+J ),CMPLXWA( IS1+J ),J= 1,NE) C
C
60  CONTINUE                  C
C RETURN                      C
C
70  FORMAT (//," MATRIX OF SUBSYSTEM:",I2/)          C
80  FORMAT (" BUS NO.",I3,":",I3)                      C
90  FORMAT ((3(3X,2F5.0,":",2(1X,E11.5))))          C
END

```

```

C          D  1
C          D  2
C          D  3
C          D  4
C          D  5
C          D  6
C          D  7
C          D  8
C          D  9
C          D 10
C          D 11
C          D 12
C          D 13
C          D 14
C          D 15
C          D 16
C          D 17
C          D 18
C          D 19
C          D 20
C          D 21
C          D 22
C          D 23
C          D 24
C          D 25
C          D 26
C          D 27
C          D 28
C          D 29
C          D 30
C          D 31
C          D 32
C          D 33
C          D 34
C          D 35
C          D 36
C          D 37
C          D 38
C          D 39
C          D 40
C          D 41
C          D 42
C          D 43
C          D 44
C          D 45
C          D 46
C          D 47
C          D 48
C          D 49
C          D 50
C          D 51
C          D 52
C          D 53
C          D 54
C          D 55
C          D 56
C          D 57
C          D 58
C          D 59
C          D 60
C          D 61
C          D 62
C          D 63
C          D 64
C          D 65

SUBROUTINE DECDS (DEC, IBL, BNR; MD, INTN, DEG, DS, CMPLXWA, ISUB, N, OTPT, I
1WRITE)
SUBROUTINE DECDS INITIALIZES IN THE COMPLEX WORKSPACE
VECTOR CMPLXWA SUBVECTOR OF THE RHS VECTOR DS ASSOCIATED
WITH THE SUBSYSTEM ISUB
INTEGER DEC(1), IBL(1), BNR(1), DEG(1), MD(1), OTPT
COMPLEX CMPLXWA(1), DS(1)
NINT=2*(N-DEC(2))
JST (JLT) IS THE INDEX IN VECTOR IBL OF THE FIRST (THE LAST)
VERTEX OF SUBSYSTEM ISUB
JLT=0
DO 10 I=1, ISUB
JST=JLT+1
JLT=JST+DEC(2*I+1)-1
CONTINUE
10 J2 IS THE INDEX OF THE LAST INTERNAL BUS OF THE SUBSYSTEM
ISUB APPEARING IN VECTOR IBL
KK=ISUB+ISUB+2
J2=JST+DEC(KK)-1
IA IS THE STARTING POSITION IN VECTOR CMPLXWA OF THE
SUBVECTOR ISUB OF THE RIGHT-HAND SIDE VECTOR DS
IA=MD(5*ISUB-1)-1
IST= IA
ELEMENTS ASSOCIATED WITH INTERNAL BUSES OF THE SUBSYSTEM
ISUB ARE NOW INITIALIZED
DO 20 K=JST, J2
L=BNR(IBL(K))
L=L+L-1
IA= IA+1
CMPLXWA( IA)=DS(L)
IA= IA+1
CMPLXWA( IA)=DS(L+1)
20 CONTINUE
ELEMENTS ASSOCIATED WITH PARTITION BUSES OF THE SUBSYSTEM
ISUB ARE NOW INITIALIZED
J2=J2+1
DO 30 K=J2, JLT
L=BNR(IBL(K))
L=L+L-1
IDG=DEG(L-NINT)
IA= IA+1
CMPLXWA( IA)=DS(L)/IDG
IA= IA+1
CMPLXWA( IA)=DS(L+1)/IDG
30 CONTINUE
IF (IWRITE.LT.3) RETURN
L=JLT-JST+1
JST= JST-1
WRITE (OTPT,40) (ISUB,(IBL(JST+K),CMPLXWA(IST+2*K-1),K=1,L))
40

```

```
40      RETURN          D  66
      FORMAT (//'" RIGHT HAND SIDE VECTOR OF SUBSYSTEM", I2/(1X,4(2X, I3, ":"  D  67
1", 2(1X,E11.5)))  D  68
      END             D  69
```

```

C          1 2 3 4 5 6 7 8 9
C          E E E E E E E E E
C          SUBROUTINE DGR (DEC, IBV, IBL, BNR, DEG, N)      10
C          THIS SUBROUTINE CALCULATES DEGREES OF PARTITION VERTICES
C          OF A GRAPH REPRESENTING THE SET OF EQUATIONS      11
C          INTEGER DEC( 1 ), IBV( 1 ), IBL( 1 ), BNR( 1 ), DEG( 1 )      12
C          NSUB=DEC( 1 )      13
C          NBV=DEC( 2 )      14
C          NINT=2*( N-NBV )      15
C          NINT IS THE NUMBER OF INTERNAL GRAPH VERTICES      16
C          DO 30 I=1,NBV      17
C          J2=0      18
C          IVER=IBV( I )      19
C          IDGR=0      20
C          DO 20 L=1,NSUB      21
C          K=L+L+1      22
C          J1=J2+DEC( K+1 )+1      23
C          J2=DEC( K )+J2      24
C          DO 10 J=J1,J2      25
C          IF ( IBL( J ) .EQ. IVER ) IDGR= IDGR+1      26
10        CONTINUE      27
20        CONTINUE      28
          K=2*BNR( IVER )-NINT      29
          DEC( K ) = IDGR      30
          DEC( K-1 ) = IDGR      31
30        CONTINUE      32
          RETURN      33
          END      34

```

```

C C
C C
      SUBROUTINE FORMK (YT,JRYT,ICYT,BTYP,BNR,V,AI,AK,IRK,ICK,N,NK,O
      1TPT,IWRITE)
C C
      THIS SUBROUTINE FORMULATES MATRIX OF PERTURBED FLOW EQUATIONS
      (SEE EQUATION (16) OF SOS-80-2). VECTORS AK, IRK, ICK STORE THE
      THE PERTURBED MATRIX IN A SPARSE FORM
C C
      INTEGER JRYT(1), ICYT(1), BTYP(1), IRK(1), ICK(1), BNR(1), JRK(1), OTPT
      COMPLEX YT(1), V(1), AK(1), AI(1), CPX, CPY, CCURR, Z, CM
C C
      NK=0
      IF (IWRITE.GE.3) WRITE (OTPT,60)
      DO 50 IR=1,N
C C
      MATRIX YT IS ANALYSED ROW BY ROW IN THIS LOOP
C C
      K1=JRYT(IR)
      K2=JRYT(IR+1)-1
      CCURR=AI(IR)
      Z=CONJG(V(IR))
      JE=NK+1
      JR1=IR+IR
      JR=JR1-1
      IF (BTYP(BNR(IR)).EQ.1) GO TO 26
C C
      SETTING UP ELEMENTS OF A ROW OF MATRIX K CORRESPONDING
      TO LOAD BUS
C C
      DO 10 J=K1,K2
      IC=ICYT(J)
      JC1=IC+IC
      JC=JC1-1
      IF (BTYP(BNR(IC)).EQ.2) GO TO 10
      CPY=YT(J)*Z
      NK=NK+1
      AK(NKO)=CPX
      IRK(NKO)=JR
      ICK(NKO)=JC
      NK=NK+1
      AK(NKO)=CONJG(CPX)
      IRK(NKO)=JR1
      ICK(NKO)=JC1
      CONTINUE
      NK=NK+1
      AK(NKO)=CCURR
      IRK(NKO)=JR
      ICK(NKO)=JR1
      NK=NK+1
      AK(NKO)=CONJG(CCURR)
      IRK(NKO)=JR1
      ICK(NKO)=JR
      GO TO 40
20    DO 30 J=K1,K2
C C
      SETTING UP ELEMENTS OF A ROW OF MATRIX K CORRESPONDING
      TO GENERATOR BUS
C C
      IC=ICYT(J)
      IF (BTYP(BNR(IC)).EQ.2) GO TO 30
      JC1=IC+IC
      JC=JC1-1
      CPX=0.5*YT(J)*Z
      1
      2
      3
      4
      5
      6
      7
      8
      9
      10
      11
      12
      13
      14
      15
      16
      17
      18
      19
      20
      21
      22
      23
      24
      25
      26
      27
      28
      29
      30
      31
      32
      33
      34
      35
      36
      37
      38
      39
      40
      41
      42
      43
      44
      45
      46
      47
      48
      49
      50
      51
      52
      53
      54
      55
      56
      57
      58
      59
      60
      61
      62
      63
      64
      65

```

```

CPY=CONJG(CPX)          F 66
NK=NK+1                 F 67
AK(NK)=CPX               F 68
IRK(NK)=JR                69
ICK(NK)=JC                70
NK=NK+1                 F 71
AK(NK)=CPY               F 72
IRK(NK)=JR                73
ICK(NK)=JC1               74
NK=NK+1                 F 75
AK(NK)=CPX               F 76
IRK(NK)=JR1               77
ICK(NK)=JC                78
NK=NK+1                 F 79
AK(NK)=CPY               F 80
IRK(NK)=JR1               81
ICK(NK)=JC1               82
80 CONTINUE                83
CM=CMLX(0.,1./CABS(Z))   F 84
CPX=(CONJG(CCURR)+CM*Z)/2. F 85
CPY=(CCURR+CM*V(IR))/2.    86
AK(JE)=AK(JE)+CPX         F 87
AK(JE+1)=AK(JE+1)+CPY     F 88
AK(JE+2)=AK(JE+2)+CONJG(CPY) 89
AK(JE+3)=AK(JE+3)+CONJG(CPX) 90
40 JK(IR)=JE                91
IF (IWRITE.LT.3) GO TO 50  F 92
WRITE (OPTP,70) BNR(IR)    F 93
WRITE (OPTP,80) (IRK(J),ICK(J),AK(J),J=JE,NK) 94
50 CONTINUE                95
C ON RETURN, JK(L) POINTS OUT THE LOCATION OF THE FIRST ELEMENT F 96
C OF TWO ROWS OF MATRIX K CORRESPONDING TO THE LTH BUS (L=1,...,N). F 97
C JK(N+1) IS THE FIRST VACANT POSITION IN MATRIX K. F 98
C F 99
C JK(N+1)=NK+1             F 100
C RETURN                   F 101
60 FORMAT (//," MATRIX K OF PERTURBED FLOW EQUATIONS") F 102
70 FORMAT (1X,"BUS NO.",I3)  F 103
80 FORMAT ((3(3X,I3,1X,I3,:",2(1X,E11.5)))) F 104
END                      F 105
                                F 106

```

```

C
C
C      SUBROUTINE INTRN (DEC, IBL, BNR, INTN, NR, ISUB)
C
C      SUBROUTINE INTRN IDENTIFIES AND ORDERS VERTICES OF THE SUBGRAPH
C      ISUB
C
C      INTEGER DEC(1), BNR(1), IBL(1), INTN(1)
C
C      INDX=0
C      DO 10 I=1, NR
C      INTN(I)=0
10    CONTINUE
C      JLT=0
C      DO 20 I=1, ISUB
C      JST=JLT+1
C      JLT=JST+DEC(2*I+1)-1
20    CONTINUE
C      DO 30 I=JST, JLT
C      K=2*BNR(IBL(I))-1
C      INDX=INDX+1
C      INTN(K)=INDX
C      K=K+1
C      INDX=INDX+1
C      INTN(K)=INDX
30    CONTINUE
C      RETURN
C      END

```

C	1
C	2
C	3
C	4
C	5
C	6
C	7
C	8
C	9
C	10
G	11
G	12
G	13
G	14
G	15
G	16
G	17
G	18
G	19
G	20
G	21
G	22
G	23
G	24
G	25
G	26
G	27
G	28
G	29

```

C          H   1
C          H   2
C          H   3
C          H   4
C          H   5
C          H   6
C          H   7
C          H   8
C          H   9
C          H  10
C          H  11
C          H  12
C          H  13
C          H  14
C          H  15
C          H  16
C          H  17
C          H  18
C          H  19
C          H  20
C          H  21
C          H  22
C          H  23
C          H  24
C          H  25
C          H  26
C          H  27
C          H  28
C          H  29
C          H  30
C          H  31
C          H  32
C          H  33
C          H  34
C          H  35
C          H  36
C          H  37
C          H  38
C          H  39
C          H  40
C          H  41
C          H  42
C          H  43
C          H  44
C          H  45
C          H  46
C          H  47
C          H  48
C          H  49
C          H  50
C          H  51
C          H  52
C          H  53
C          H  54
C          H  55
C          H  56
C          H  57
C          H  58
C          H  59
C          H  60
C          H  61
C          H  62
C          H  63
C          H  64
C          H  65

SUBROUTINE MDDTF (IRK, ICK, DEC, IN, IM, BNR, IBL, INTN, LNGTH, CMPLXWA, MD,
1NB, NK, ICALL)                                H   1
C          H   2
C          H   3
C          H   4
C          H   5
C          H   6
C          H   7
C          H   8
C          H   9
C          H  10
C          H  11
C          H  12
C          H  13
C          H  14
C          H  15
C          H  16
C          H  17
C          H  18
C          H  19
C          H  20
C          H  21
C          H  22
C          H  23
C          H  24
C          H  25
C          H  26
C          H  27
C          H  28
C          H  29
C          H  30
C          H  31
C          H  32
C          H  33
C          H  34
C          H  35
C          H  36
C          H  37
C          H  38
C          H  39
C          H  40
C          H  41
C          H  42
C          H  43
C          H  44
C          H  45
C          H  46
C          H  47
C          H  48
C          H  49
C          H  50
C          H  51
C          H  52
C          H  53
C          H  54
C          H  55
C          H  56
C          H  57
C          H  58
C          H  59
C          H  60
C          H  61
C          H  62
C          H  63
C          H  64
C          H  65

SUBROUTINE MDDTF ORGANIZES COMPLEX WORKSPACE VECTOR CMPLXWA FOR
C THE DATA DESCRIBING DECOMPOSED POWER SYSTEM AND INITIALIZES SOME
C ELEMENTS OF CMPLXWA (I.E., N,NINT AND VECTOR NON)                                H   1
C          H   2
C          H   3
C          H   4
C          H   5
C          H   6
C          H   7
C          H   8
C          H   9
C          H  10
C          H  11
C          H  12
C          H  13
C          H  14
C          H  15
C          H  16
C          H  17
C          H  18
C          H  19
C          H  20
C          H  21
C          H  22
C          H  23
C          H  24
C          H  25
C          H  26
C          H  27
C          H  28
C          H  29
C          H  30
C          H  31
C          H  32
C          H  33
C          H  34
C          H  35
C          H  36
C          H  37
C          H  38
C          H  39
C          H  40
C          H  41
C          H  42
C          H  43
C          H  44
C          H  45
C          H  46
C          H  47
C          H  48
C          H  49
C          H  50
C          H  51
C          H  52
C          H  53
C          H  54
C          H  55
C          H  56
C          H  57
C          H  58
C          H  59
C          H  60
C          H  61
C          H  62
C          H  63
C          H  64
C          H  65

INTEGER IRK(1), ICK(1), DEC(1), BNR(1), IBL(1), IN(1), MD(1), IM(1), LNGTH
1(1), INTN(1)
COMPLEX CMPLXWA(1)

C          IF (ICALL.GT.1) IN(8)=7
C          IF (ICALL.GT.1) GO TO 50
C          NR=2*(NB-1)
C          NSUB=DEC(1)
C          NEV=2*DEC(2)
C          IHIGH=NSUB+NSUB-1

C          VECTOR IN IS NOW INITIALIZED

C          IN(4)=NSUB
C          IN(5)=NSUB
C          IN(8)=6
C          IN(9)=IHIGH
C          IN(10)=NSUB
C          IN(15)=1
C          IN(19)=NR-NBV+1
C          IN(20)=1
C          IN(22)=0
C          IN(26)=NBV
C          L=26+IN(9)/2
C          L1=L+NSUB
C          DO 10 I=1,NSUB
C          IN(L+I)=IHIGH
C          IN(L1+I)=IHIGH
C          IHIGH=IHIGH-1
C          CONTINUE
C          CALL INDICAT (IN,IND)
C          IAS=3*IN(16)+4*IN(17)+IN(11)+1
C          IM(1)=0
C          IM(2)=0
C          IM(5)=NR

C          VECTOR MD IDENTIFIES THE LOCATIONS OF DATA SUBVECTORS IN THE
C          WORKSPACE VECTOR CMPLXWA. ENTRIES FROM MD(5*K-4) TO MD(5*K)
C          POINTS OUT THE LOCATIONS OF THE FIRST ELEMENTS OF VECTORS CE,
C          AK, NROW, V, NON, RESPECTIVELY, ASSOCIATED WITH THE KTH SUBSYSTEM

C          NSUB=DEC(1)
C          DO 20 I=1,NSUB
C          LNGTH(I)=0
C          CONTINUE
C          DO 40 ISUB=1,NSUB
C          CALL INTRN (DEC, IBL, BNR, INTN, NR, ISUB)
C          DO 30 I=1,NK
C          IF (INTN(IRK(I)).NE.0 .AND. INTN(ICK(I)).NE.0) LNGTH(ISU
C          1B)+1
C          CONTINUE
C          CONTINUE
C          L=NSUB+NSUB
C          INIT=IAS+L
C          IND=1

```

DO 70 K=1,NSUB	H 66
K1=5*K	H 67
MD(K1-4)=INIT	H 68
MD(K1-3)=INIT+2	H 69
N=DEC(1+2*K)	H 70
CMLXWA(INIT)=CMPLX(FLOAT(2*N),FLOAT(2*DEC(2+2*K)))	H 71
R=FLOAT(LNGTH(K))	H 72
CMLXWA(INIT+1)=CMPLX(R,R)	H 73
 C C N IS THE NUMBER OF BUSES OF SUBSYSTEM. NUMBER OF EQUATIONS IS 2*N	H 74
 MD(K1-2)=MD(K1-3)+LNGTH(K)	H 75
MD(K1-1)=MD(K1-2)+LNGTH(K)	H 76
LL=MD(K1-1)+N+N	H 77
MD(K1)=LL	H 78
INIT=LL+N-1	H 79
DO 60 KK=LL,INIT	H 80
II=ENR(IBL(IND))	H 81
IND=IND+1	H 82
I2=II+II	H 83
CMLXWA(KK)=CMPLX(FLOAT(I2-1),FLOAT(I2))	H 84
60 CONTINUE	H 85
INIT=INIT+1	H 86
70 CONTINUE	H 87
LL=K1+1	H 88
MD(LL)=INIT	H 89
IN(25)=INIT-IAS	H 90
 C C DIRECTORY IS NOW INITIALIZED	H 91
 IHIGH=IN(9)+1	H 92
IDIR=IAS-1	H 93
IA=IDIR+IN(10)	H 94
DO 80 K=1,NSUB	H 95
CMLXWA(IDIR+K)=CMPLX(FLOAT(IHIGH-K),1.)	H 96
K1=5*K	H 97
IS=ID(K1-4)	H 98
LG=MD(K1+1)-IS	H 99
IS=IS-IDIR	H 100
CMLXWA(IA+K)=CMPLX(FLOAT(LG),FLOAT(IS))	H 101
80 CONTINUE	H 102
RETURN	H 103
END	H 104
	H 105
	H 106
	H 107
	H 108

```

C          I   1
C          I   2
C          I   3
C          I   4
C          I   5
C          I   6
C          I   7
C          I   8
C          I   9
C          I  10
C          I  11
C          I  12
C          I  13
C          I  14
C          I  15
C          I  16
C          I  17
C          I  18
C          I  19
C          I  20
C          I  21
C          I  22
C          I  23
C          I  24
C          I  25
C          I  26
C          I  27
C          I  28
C          I  29
C          I  30
C          I  31
C          I  32
C          I  33
C          I  34
C          I  35
C          I  36
C          I  37
C          I  38
C          I  39

SUBROUTINE RHSLD (V, YT, JRYT, ICYT, BTYP, BNR, BCV, DS, AI, N, OTPT, IWRITE)      I
THIS SUBROUTINE CALCULATES VECTOR DS OF PERTURBED LOAD FLOW                      I
EQUATIONS (SEE EQUATION (17) OF SOS-80-2) AND BUS CURRENTS                         I
VECTOR AI                                I
INTEGER JRYT(1), ICYT(1), BTYP(1), BNR(1), OTPT
COMPLEX V(1), YT(1), BCV(1), DS(1), AI(1), CCURR, S
C
N1=N+N
DO 20 I=1,N
J1=I+I
J=J1-1
AI(I)=CCURR(V, YT, JRYT, ICYT, I)
S=V(I)*CONJG(AI(I))
IF (BTYP(BNR(I)).EQ.1) GO TO 10
S=BCV(I)-S
DS(J)=CONJG(S)
DS(J1)=S
GO TO 20
10 S=CMPLX(REAL(S), CAES(V(I)))
S=BCV(I)-S
DS(J)=S
DS(J1)=CONJG(S)
CONTINUE
IF (IWRITE.LT.3) RETURN
WRITE (OTPT,20)
WRITE (OTPT,40) (BNR(I), DS(2*I-1), I=1,N)
WRITE (OTPT,50)
WRITE (OTPT,40) (BNR(I), AI(I), I=1,N)
RETURN
30 FORMAT (///" RIGHT HAND SIDE VECTOR DS OF PERTURBED FLOW EQUATIONS"
1/)
40 FORMAT ((3(3X, I3, ":" , 2(1X, E11.5))))
50 FORMAT (///" VECTOR AI OF BUS CURRENTS"/)
END

```

```

C          J   1
C          J   2
C          J   3
C          J   4
C          J   5
C          J   6
C          J   7
C          J   8
C          J   9
C          J  10
C          J  11
C          J  12
C          J  13
C          J  14
C          J  15
C          J  16
C          J  17
C          J  18
C          J  19
C          J  20
C          J  21
C          J  22
C          J  23
C          J  24
C          J  25
C          J  26
C          J  27
C          J  28
C          J  29
C          J  30
C          J  31
C          J  32
C          J  33
C          J  34
C          J  35
C          J  36
C          J  37
C          J  38
C          J  39
C          J  40
C          J  41
C          J  42
C          J  43
C          J  44
C          J  45
C          J  46
C          J  47
C          J  48
C          J  49
C          J  50
C          J  51
C          J  52
C          J  53
C          J  54
C          J  55
C          J  56
C          J  57
C          J  58
C          J  59
C          J  60
C          J  61
C          J  62
C          J  63
C          J  64
C          J  65

C SUBROUTINE FRPR ( YT, JRYT, ICYT, BCV, V, BNR, BTYP, IBV, WS, LWS, NB, NTL, NBV
1, INPT, OTPT, IFLAG, IWRITE) J
C
C SUBROUTINE FORMPR RENUMBERS BUSES OF THE POWER SYSTEM AND J
C FORMULATES THE LOAD FLOW PROBLEM. ON RETURN, THE LOAD J
C PROBLEM IS GIVEN BY VECTORS YT, JRYT, ICYT, BCV, V INDEXED J
C AS DEFINED BY VECTOR BNR. J
C
C INTEGER BNR( 1 ), IBV( 1 ), JRYT( 1 ), ICYT( 1 ), BTYP( 1 ), OTPT J
C REAL BCV( 1 ), WS( 1 ) J
C COMPLEX YT( 1 ), V( 1 ) J
C
C IFLAG=0 J
C JBINP=1 J
C JEOUT=JBINP+NTL J
C JINPG=JEOOUT+NTL J
C JINPB=JINPG+NTL J
C JLG=JINPB+NTL J
C JLB=JLG+NTL J
C JOUTG=JLB+NTL J
C JOUTB=JOUTG+NTL J
C JTAP=JOUTB+NTL J
C JNR=JTAP+NTL J
C JVM=JNR+NB J
C JVA=JVM+NB J
C JGP=JVA+NB J
C JLP=JGP+NB J
C JLQ=JLP+NB J
C JSTL=JLQ+NB J
C JMK=JSTL+NB J
C IF ( JMK .GE. LWS ) GO TO 50 J
C CALL RDAT ( WS(JBINP), WS(JEOOUT), WS(JINPG), WS(JINPB), WS(JLG), WS(JLB) J
C 1, WS(JOUTG), WS(JOUTB), WS(JTAP), WS(JNR), BTYP, WS(JVM), WS(JVA), WS(JGP) J
C 2, WS(JLP), WS(JLQ), WS(JSTL), JRYT, NB, NTL, NLB, INPT, IWRITE) J
C
C BUSES ARE RENUMBERED TO GIVE THE HIGHEST NUMBERS TO PARTITION J
C BUSES J
C
C DO 10 I=1,NB J
C BNR( I )=I J
C JRYT( I )=JRYT( I+1 )-JRYT( I ) J
C 10 CONTINUE J
C K=NB-1 J
C DO 20 I=1,NBV J
C L=IBV( I ) J
C BNR( L )=K J
C ENR( K )=L J
C JR=JRYT( K ) J
C JRYT( K )=JRYT( L ) J
C JRYT( L )=JR J
C K=K-1 J
C 20 CONTINUE J
C L=1 J
C DO 30 I=1,NB J
C M=JRYT( I ) J
C JRYT( I )=L J
C L=M+L J
C 30 CONTINUE J
C
C JRYT( I ) IS AN INDEX OF THE FIRST ELEMENT OF THE ITH ROW OF THE J
C MODAL ADMITTANCE MATRIX ( I IS A BUS INDEX DEFINED BY VECTOR BNR, J
C I=1,2, ,NB ) J

```

C	CALL FORMYT (WS(JBINP), WS(JBOUT), BNR, IBV, WS(JINPG), WS(JINPB), WS(JL 1G), WS(JLB), WS(JOUTG), WS(JOUTB), WS(JTAP), WS(JSTL), JRYT, ICYT, YT, NB, N 2TL, NYT, OTPT, IWRITE)	J 66
	CALL FORMU (BTYP, BNR, WS(JVM), WS(JVA), WS(JGP), WS(JLP), WS(JLQ), BCV, N 1B, OTPT, IWRITE)	J 67
	K1=JVM-1	J 68
	K2=JVA-1	J 69
	DO 40 I=1,NB	J 70
	R1=WS(K1+I)	J 71
	R2=WS(K2+I)	J 72
	V(ENR(I))=CMPLX(R1*COS(R2), R1*SIN(R2))	J 73
40	CONTINUE	J 74
	RETURN	J 75
50	IFLAG=-1	J 76
	RETURN	J 77
	END	J 78
		J 79
		J 80
		J 81
		J 82

```

C          1
C          2
C          3
C          4
C          5
C          6
C          7
C          8
C          9
C          10
C          11
C          12
C          13
C          14
C          15
C          16
C          17
C          18
C          19
C          20
C          21
C          22
C          23
C          24
C          25
C          26
C          27
C          28
C          29
C          30
C          31
C          32
C          33
C          34
C          35
C          36
C          37
C          38
C          39
C          40
C          41
C          42
C          43

C SUBROUTINE FORMU (BTYP, BNR, BVMOD, BVARG, BGP, BLP, BLQ, BCV, NB, OTPT, IWR
1 ITE)
C
C SUBROUTINE FORMU FORMS THE RIGHT HAND SIDE VECTOR BCV OF THE
C POWER FLOW EQUATIONS
C
C INTEGER BTYP( 1 ), BNR( 1 ), OTPT
C REAL BVMOD( 1 ), BVARG( 1 ), BGP( 1 ), BLP( 1 ), BLQ( 1 ), BCV( 1 )
C
C DO 40 I=1,NB
C J=2*BNR( I )-1
C IF ( BTYP( I )-1 ) 10,20,30
C
C      SETTING UP ELEMENTS OF BCV CORRESPONDING TO LOAD BUS
C
C 10 BCV( J )=BGP( I )-BLP( I )
C J=J+1
C BCV( J )=-BLQ( I )
C GO TO 40
C
C      SETTING UP ELEMENTS OF BCV CORRESPONDING TO GENERATOR BUS
C
C 20 BCV( J )=BGP( I )-BLP( I )
C J=J+1
C BCV( J )=BVMOD( I )
C GO TO 40
C
C      SETTING UP ELEMENTS OF BCV CORRESPONDING TO SLACK BUS
C
C 30 BCV( J )=BVMOD( I )
C J=J+1
C BCV( J )=BVARG( I )
C
C CONTINUE
C IF ( IWRITE .LT. 3 ) GO TO 50
C WRITE ( OTPT, 60 )
C WRITE ( OTPT, 70 ) ( BNR( I ), BCV( 2*I-1 ), BCV( 2*I ), I=1,NB )
C
C 50 RETURN
C
C 60 FORMAT ( // " RHS VECTOR BCV OF POWER FLOW EQUATIONS " / )
C
C 70 FORMAT ( 3( 3X, I3, ":" , 2( 1X, E13.7 ) ) )
C
C END

```

```

C
C
C      SUBROUTINE FORMYT (LBINP,LBOUT,BNR,IBV,LINFG,LINPB,LG,LB,LOUTG,LOU
1TE,LTAP,BSTL,JRYT,ICYT,YT,NB,NTL,NYT,OTPT,IWRITE)
C
C      SUBROUTINE FORMYT FORMS THE NODAL ADMITTANCE MATRIX OF POWER
C      SYSTEM AND STORES IT IN A SPARSE FORM BY VECTORS YT,JRYT,ICYT,
C      USING BUS INDICES DEFINED BY VECTOR BNR.
C
C      INTEGER LBINP(1),LBOUT(1),JRYT(1),ICYT(1),BNR(1),IBV(1),OTPT
C      REAL LINPG(1),LINPB(1),LG(1),LB(1),LOUTG(1),LOUTB(1),LTAP(1),BSTL(
11)
C      COMPLEX YT(1),Y
C
C      NYT=NB+2*NTL
C
C      ON ENTRY JRYT MUST BE INDEXED AS DEFINED BY VECTOR BNR
C
C      DO 10 I=1,NB
C      J=JRYT(I)
C      YT(J)=CMPLX(0.,BSTL(BNR(I)))
C      ICYT(NYT+I)=J
10    CONTINUE
C      DO 20 I=1,NTL
C      IB1=BNR(LBINP(I))
C      IB2=BNR(LBOUT(I))
C      Y=CMPLX(LG(I),LB(I))
C      L1=JRYT(IB1)
C      L2=JRYT(IB2)
C      YT(L1)=YT(L1)+CMPLX(LINPG(I),LINPB(I))+Y/(LTAP(I)**2)
C      YT(L2)=YT(L2)+CMPLX(LOUTG(I),LOUTB(I))+Y
C      ICYT(L1)=IB1
C      ICYT(L2)=IB2
C      K1=NYT+IB1
C      K2=NYT+IB2
C      ICYT(K1)=ICYT(K1)+1
C      ICYT(K2)=ICYT(K2)+1
C      L1=ICYT(K1)
C      L2=ICYT(K2)
C      Y=-Y/LTAP(I)
C      YT(L1)=Y
C      YT(L2)=Y
C      ICYT(L1)=IB2
C      ICYT(L2)=IB1
20    CONTINUE
C      IF (IWRITE.LT.3) GO TO 40
C      WRITE (OTPT,50)
C      DO 30 I=1,NB
C      K1=JRYT(I)
C      K2=JRYT(I+1)-1
C      WRITE (OTPT,60) BNR(ICYT(K1))
C      WRITE (OTPT,70) (BNR(ICYT(J)),YT(J),J=K1,K2)
30    CONTINUE
40    RETURN
50    FORMAT (// " BUS ADMITTANCE MATRIX YT")
51    FORMAT (" BUS NO.",I3)
52    FORMAT (3(GX,I3,":",2(1X,E13.7)))
53    END

```

```

C          M   1
C          M   2
C          M   3
C          M   4
C          M   5
C          M   6
C          M   7
C          M   8
C          M   9
C          M  10
C          M  11
C          M  12
C          M  13
C          M  14
C          M  15
C          M  16
C          M  17
C          M  18
C          M  19
C          M  20
C          M  21
C          M  22
C          M  23
C          M  24
C          M  25
C          M  26
C          M  27
C          M  28
C          M  29
C          M  30
C          M  31
C          M  32
C          M  33
C          M  34
C          M  35
C          M  36
C          M  37
C          M  38
C          M  39
C          M  40
C          M  41
C          M  42
C          M  43
C          M  44
C          M  45
C          M  46
C          M  47
C          M  48
C          M  49
C          M  50
C          M  51
C          M  52
C          M  53
C          M  54
C          M  55
C          M  56
C          M  57
C          M  58
C          M  59
C          M  60
C          M  61
C          M  62
C          M  63
C          M  64
C          M  65

SUBROUTINE RDAT (LBINP,LBOUT,LINPG,LINPB,LG,LB,LOUTG,LOUTB,LTAP,BN
1R,BTYP,BVMOD,BVARG,BGP,BLP,BLQ,BSTL,JRYT,NB,NTL,NLB,INPT,IWRITE)
SUBROUTINE RDAT READS INPUT DATA DESCRIBING POWER SYSTEM
INTEGER LBINP(1),LBOUT(1),BNR(1),BTYP(1),JRYT(1)
REAL LINPG(1),LINPB(1),LG(1),LB(1),LOUTG(1),LOUTB(1),LTAP(1),BVMOD
1(B1),BVARG(1),BGP(1),BLP(1),BLQ(1),BSTL(1),HDLN(12)
COMPLEX Z
READ (INPT,110) (HDLN(I),I=1,12)
READ (INPT,130) X,NB,X,X,NTL
DO 10 I=1,NB
JRYT(I)=1
CONTINUE
IF (IWRITE.LT.1) GO TO 20
WRITE (6,110) (HDLN(I),I=1,12)
WRITE (6,150) NB,NTL
READ (INPT,110) (HDLN(I),I=1,12)
IF (IWRITE.LT.2) GO TO 30
WRITE (6,110) (HDLN(I),I=1,12)
READ (INPT,120) (HDLN(I),I=1,12)
DO 40 I=1,NTL
READ (INPT,140) LBINP(I),LBOUT(I),LINPG(I),LINPB(I),LG(I),LB(I),LO
1UTG(I),LOUTB(I),LTAP(I)
DEGREES OF BOTH BUSES INCIDENT TO THE LINE ARE INCREASED
BY ONE
J=LBINP(I)
JRYT(J)=JRYT(J)+1
J=LBOUT(I)
JRYT(J)=JRYT(J)+1
CONTINUE
IF (IWRITE.LT.2) GO TO 60
WRITE (6,120) (HDLN(I),I=1,12)
DO 50 I=1,NTL
WRITE (6,140) LBINP(I),LBOUT(I),LINPG(I),LINPB(I),LG(I),LB(I),LO
1UTG(I),LOUTB(I),LTAP(I)
CONTINUE
DO 70 I=1,NTL
Z=1./CMPLX(LG(I),LB(I))
LG(I)=REAL(Z)
LB(I)=AIHAG(Z)
CONTINUE
READ (INPT,110) (HDLN(I),I=1,12)
IF (IWRITE.GT.1) WRITE (6,110) (HDLN(I),I=1,12)
READ (INPT,120) (HDLN(I),I=1,12)
NLB=0
IX=JRYT(1)
JRYT(1)=1
JRYT(NB+1)=0
DO 80 I=1,NB
READ (INPT,140) L,BTYP(L),BVMOD(L),BVARG(L),BGP(L),BLP(L),BLQ(L),B
1STL(L)
BNR(L)=I
IF (BTYP(L).EQ.0) NLB=NLB+1
J=I+1
IY=JRYT(J)
JRYT(J)=JRYT(I)+IX
IX=IY
CONTINUE

```

IF (IWRITE.LT.2) GO TO 100	M 66
WRITE (6,120) (HDLN(I), I=1,12)	M 67
DO 90 I=1,NB	M 68
WRITE (6,140) BNR(I), BTYP(I), BVMOD(I), BVARG(I), BGP(I), BLP(I), BLQ(I)	M 69
1), BSTL(I)	M 70
90 CONTINUE	M 71
100 RETURN	M 72
110 FORMAT (//12A10/)	M 73
120 FORMAT (12A10)	M 74
130 FORMAT (1X,A5,I3,2A5,I4)	M 75
140 FORMAT (2(1X,15),2X,7(2X,E13.7))	M 76
150 FORMAT (" NB =",I3.3," ","NTL =",I4.3)	M 77
END	M 78