XLF3 - A FORTRAN IMPLEMENTATION OF THE COMPLEX LAGRANGIAN METHOD TO POWER SYSTEM ANALYSIS AND DESIGN

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

SOS-83-18-L

October 1983

XLF3 - A FORTRAN IMPLEMENTATION OF THE COMPLEX LAGRANGIAN METHOD TO POWER SYSTEM ANALYSIS AND DESIGN

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

Abstract

This report contains a listing of the computer package XLF3 described in [1]. The package is capable of handling the nonreciprocal power network elements and has been developed for the CDC 170/815 system with the NOS 2.1-580/577 operating system and the Fortran Extended (FTN) version 4.8 compiler. The listing contains a total of 754 lines (including 209 comments) constituting twelve subroutines. The listing does not include the Harwell package ME28 for solving the sparse linear equations.

This work was supported by the Natural Sciences and Engineering Research Council of Canada under Grants A7239, A1708 and 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

This document contains a Fortran listing of all the subroutines of the XLF3 package [1]. This package is specially designed to handle the complex turns ratio of the phase shifting transformers [2]. The load flow problem is solved by using the complex Newton method [3] and network sensitivities are obtained by implementing the complex Lagrangian approach [4].

The XLF3 package has been developed for the CDC 170/815 system with the NOS 2.1-580/577 operating system and the Fortran Extended (FTN) version 4.8 compiler. 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 LIBXLF3 accessible under the charge RJWBAND. The package calls subroutines ME28A, ME28B and ME28C of the Harwell Subroutine Library (Harwell package ME28) for solving sparse linear equations [5].

The XLF3 package contains 754 lines which includes 209 comment statements. It has been modularized into 12 subroutines listed in Table I.

TABLE I

LIST OF SUBROUTINES OF THE XLF3 PACKAGE

	Subroutine	Number of lines (source text)	Description (page of [1])	Listing (page)
1	CCURR	20	13	17
2	DERIVX	80	15	18
3	FORMDTX	39	20	6
4	FORMK	99	23	15
5	FORMMU	18	26	20
6	FORMPR	55	29	4
7	FORMU	45	33	5
8	FORMYTX	61	36	7
9	LFNCM	167	40	10
10	RDATAX	85	45	8
11	RHSLD	38	49	14
12	STEPNCM	47	51	13

II. REFERENCES

[1] J.W. Bandler, M.A. El-Kady, H.K. Grewal and J. Wojciechowski, "XLF3 - A Fortran implementation of the complex Lagrangian method to power system analysis and design", Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, Report SOS-83-18-U, 1983.

[2] H.K. Grewal, "Sensitivity evaluation and optimization of electrical power systems with emphasis on nonreciprocal elements", Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, Report SOS-83-17-T, 1983.

[3] J.W. Bandler and M.A. El-Kady, "Newton's load flow in complex mode", Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, Report SOS-80-2-R, 1980.

[4] 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.

[5] I.S. Duff, "ME28 - A set of Fortran subroutines for sparse unsymmetric linear equations", Computer Science and Systems Division, AERE Harwell, Oxfordshire, England, Report R.8730, 1980.

IV. FORTRAN LISTING

```
      SUBROUTINE FORMPR (LBINP,LBOUT,BTYP,YT,JRYT,ICYT,BCV,V,WS,LWS,NB,N
     1TL,NLB,IP,INPT,OTPT,IFLAG,IWRITE)
C
C
C     SUBROUTINE  FORMPR FORMULATES THE LOAD FLOW PROBLEM, I.E.,
C     THE SPARSE BUS ADMITTANCE  MATRIX  (VECTORS YT,JRYT,ICYT),
C     THE RIGHT-HAND-SIDE VECTOR BCV OF POWER FLOW EQUATIONS AND
C     VECTOR V OF THE INITIAL BUS VOLTAGES
C
      INTEGER LBINP(1),LBOUT(1),JRYT(1),ICYT(1),BTYP(1),OTPT
      REAL BCV(1),WS(1)
      COMPLEX YT(1),V(1)
      COMMON /MDFRMPR/ JINPG,JINPB,JLG,JLB,JOUTG,JOUTB,JTAP,JNR,JVM,JVA,
     1JGP,JLP,JLQ,JSTL,JMX
      IFLAG=0
      JINPG=1
      JINPB=JINPG+NTL
      JLG=JINPB+NTL
      JLB=JLG+NTL
      JOUTG=JLB+NTL
      JOUTB=JOUTG+NTL
      JTAP=JOUTB+NTL
      JNR=JTAP+NTL+NTL
      JVM=JNR+NB
      JVA=JVM+NB
      JGP=JVA+NB
      JLP=JGP+NB
      JLQ=JLP+NB
      JSTL=JLQ+NB
      JMX=JSTL+NB
      IF (JMX.GE.LWS) GO TO 50
      CALL RDATAX (LBINP,LBOUT,WS(JINPG),WS(JINPB),WS(JLG),WS(JLB),WS(JO
     1UTG),WS(JOUTB),WS(JTAP),WS(JNR),BTYP,WS(JVM),WS(JVA),WS(JGP),WS(JL
     2P),WS(JLQ),WS(JSTL),JRYT,NB,NTL,NLB,INPT,OTPT,IWRITE)
      CALL FORMYTX (LBINP,LBOUT,WS(JINPG),WS(JINPB),WS(JLG),WS(JLB),WS(J
     1OUTG),WS(JOUTB),WS(JTAP),WS(JSTL),JRYT,ICYT,YT,NB,NTL,NYT,OTPT,IWR
     2ITE)
      CALL FORMU (BTYP,WS(JVM),WS(JVA),WS(JGP),WS(JLP),WS(JLQ),BCV,NB,OT
     1PT,IWRITE)
      K1=JVM-1
      K2=JVA-1
      IF (IP.EQ.1) GO TO 20
      DO 10 I=1,NB
      R1=WS(K1+I)
      R2=WS(K2+I)
      V(I)=CMPLX(R1*COS(R2),R1*SIN(R2))
   10 CONTINUE
      GO TO 40
   20 DO 30 I=1,NB
      V(I)=CMPLX(WS(K1+I),WS(K2+I))
   30 CONTINUE
   40 RETURN
   50 IFLAG=-1
      RETURN
      END
```

```
CC
C
C
C
      SUBROUTINE FORMU (BTYP,BVMOD,BVARG,BGP,BLP,BLQ,BCV,NB,OTPT,IWRITE)
C
C
C     SUBROUTINE FORMU FORMS VECTOR BCV OF BUS CONTROL VARIABLES
C     OF THE POWER SYSTEM
C
      INTEGER BTYP(1),OTPT
      REAL BVMOD(1),BVARG(1),BGP(1),BLP(1),BLQ(1),BCV(1)
      DO 40 I=1,NB
      J=2*I-1
      IF (BTYP(I)-1) 10,20,30
C
C     SETTING UP VALUES OF ELEMENTS OF BCV  CORRESPONDING TO A LOAD BUS
C
   10 BCV(J)=BGP(I)-BLP(I)
      J=J+1
      BCV(J)=-BLQ(I)
      GO TO 40
C
C     SETTING UP VALUES OF ELEMENTS OF BCV CORRESPONDING TO A GENERATOR
C     BUS
C
   20 BCV(J)=BGP(I)-BLP(I)
      J=J+1
      BCV(J)=BVMOD(I)
      GO TO 40
C
C     SETTING UP VALUES OF ELEMENTS OF  BCV  CORRESPONDING TO SLACK BUS
C
   30 BCV(J)=BVMOD(I)
      J=J+1
      BCV(J)=BVARG(I)
   40 CONTINUE
      IF (IWRITE.LT.3) GO TO 50
      WRITE (OTPT,60)
      WRITE (OTPT,70) (I,BCV(2*I-1),BCV(2*I),I=1,NB)
   50 RETURN
   60 FORMAT (//" VECTOR BCV OF BUS CONTROL VARIABLES"/1X,35("-")/" VALU
     1E OF AN ELEMENT IS PRECEDED BY THE BUS INDEX"/)
   70 FORMAT (2(3X,I3,":",2(F13.6)))
      END
```

```
CC
C
C
C
      SUBROUTINE FORMDTX (LBINP,LBOUT,LINPG,LINPB,LR,LX,LOUTG,LOUTB,LTAP
     1,BNR,BTYP,BVMOD,BVARG,BGP,BLP,BLQ,BSTL,HDLN,NB,NTL,OTPT)
C
C
C     SUBROUTINE FORMDTX CREATES A FORMATTED DATA FILE DESCRIBING
C     THE POWER SYSTEM
C
      INTEGER LBINP(1),LBOUT(1),BNR(1),BTYP(1),OTPT
      REAL LINPG(1),LINPB(1),LR(1),LX(1),LOUTG(1),LOUTB(1),BVMOD(1),BVAR
     1G(1),BGP(1),BLP(1),BLQ(1),BSTL(1),RL(8),RB(8),HDLN(8)
      COMPLEX LTAP(1)
C
C     MATRIX HDLN KEEPS AN IDENTIFIER OF THE CREATED DATA FILE
C
      DATA RL/"ELEMENT","LINPG","LINPB","LR","LX","LOUTG","LOUTB","LTAP"
     1/,RB/"   BNR","BTYPE","BVMOD","BVARG","BGP","BLP","BLQ","BSTL"/
      WRITE (OTPT,10) NB,(HDLN(I),I=1,8)
      WRITE (OTPT,70) NB,NTL
      WRITE (OTPT,20)
      WRITE (OTPT,30) (RL(I),I=1,8)
      WRITE (OTPT,50) (LBINP(I),LBOUT(I),LINPG(I),LINPB(I),LR(I),LX(I),L
     1OUTG(I),LOUTB(I),LTAP(I),I=1,NTL)
      WRITE (OTPT,40)
      WRITE (OTPT,30) (RB(I),I=1,8)
      WRITE (OTPT,60) (BNR(I),BTYP(I),BVMOD(I),BVARG(I),BGP(I),BLP(I),BL
     1Q(I),BSTL(I),I=1,NB)
      RETURN
   10 FORMAT (//1X,"B",I3.3,8A10/)
   20 FORMAT (//1X,"TRANSMISSION NETWORK DATA",/,1X,25("-"),/)
   30 FORMAT (1X,A7,4X,2(A5,3X),2(2X,A5,1X),1X,2(A5,3X),2X,A5)
   40 FORMAT (//1X,"BUS DATA"/1X,8("-"),/)
   50 FORMAT (1X,I3,",",I3,1X,2(2F8.4,1X),2F8.4,1X,2(1X,F5.2))
   60 FORMAT (1X,I3,8X,I3,1X,2F9.4,3F8.4,F9.4)
   70 FORMAT (1X,"NB = ",I3.3,",       ","NTL =",I4.3)
      END
```

```
CC
C
C
C
      SUBROUTINE FORMYTX (LBINP,LBOUT,LINPG,LINPB,LG,LB,LOUTG,LOUTB,LTAP
     1,BSTL,JRYT,ICYT,YT,NB,NTL,NYT,OTPT,IWRITE)
C
C
C     SUBROUTINE FORMYTX FORMS THE NODAL ADMITTANCE MATRIX OF POWER
C     SYSTEM AND STORES IT IN A SPARSE FORM (VECTORS YT,JRYT,ICYT)
C
      INTEGER LBINP(1),LBOUT(1),JRYT(1),ICYT(1),OTPT
      REAL LINPG(1),LINPB(1),LG(1),LB(1),LOUTG(1),LOUTB(1),BSTL(1)
      COMPLEX YT(1),Y,LTAP(1)
      NYT=NB+2*NTL
C
C     BUS STATIC LOADS ARE PLACED INTO THE NODAL ADMITTANCE MATRIX
C
      DO 10 I=1,NB
      J=JRYT(I)
      YT(J)=CMPLX(0.,BSTL(I))
      ICYT(NYT+I)=J
   10 CONTINUE
C
C     LINE PARAMETERS ARE ADDED TO THE NODAL ADMITTANCE MATRIX
C
      DO 20 I=1,NTL
      IB1=LBINP(I)
      IB2=LBOUT(I)
      Y=CMPLX(LG(I),LB(I))
      L1=JRYT(IB1)
      L2=JRYT(IB2)
      YT(L1)=YT(L1)+CMPLX(LINPG(I),LINPB(I))+Y/(CABS(LTAP(I))**2)
      YT(L2)=YT(L2)+CMPLX(LOUTG(I),LOUTB(I))+Y
      ICYT(L1)=IB1
      ICYT(L2)=IB2
      K1=NYT+IB1
      K2=NYT+IB2
      ICYT(K1)=ICYT(K1)+1
      ICYT(K2)=ICYT(K2)+1
      L1=ICYT(K1)
      L2=ICYT(K2)
      YT(L1)=-Y/CONJG(LTAP(I))
      YT(L2)=-Y/LTAP(I)
      ICYT(L1)=IB2
      ICYT(L2)=IB1
   20 CONTINUE
      IF (IWRITE.LT.3) GO TO 40
      WRITE (OTPT,50)
      DO 30 I=1,NB
      K1=JRYT(I)
      K2=JRYT(I+1)-1
      WRITE (OTPT,60) ICYT(K1)
      WRITE (OTPT,70) (ICYT(J),YT(J),J=K1,K2)
   30 CONTINUE
   40 RETURN
   50 FORMAT (//," BUS ADMITTANCE MATRIX YT"/1X,24("-"),/," THE"1X,"SEQU
     1ENCE IN EACH ROW IS: COLUMN INDEX,REAL(YT),IMAG(YT)")
   60 FORMAT (/" BUS NO.",I3)
   70 FORMAT (2(3X,I3,":",2(F13.6)))
      END
```

```
CC
C
C
C
      SUBROUTINE RDATAX (LBINP,LBOUT,LINPG,LINPB,LG,LB,LOUTG,LOUTB,LTAP,
     1BNR,BTYP,BVMOD,BVARG,BGP,BLP,BLQ,BSTL,JRYT,NB,NTL,NLB,INPT,OTPT,IW
     2RITE)
C
C
C     SUBROUTINE RDATAX READS INPUT DATA DESCRIBING POWER SYSTEM FROM
C     A FILE CREATED BY SUBROUTINE FORMDTX AND PREPROCESS THIS DATA
C
      INTEGER LBINP(1),LBOUT(1),BNR(1),BTYP(1),JRYT(1),OTPT
      REAL LINPG(1),LINPB(1),LG(1),LB(1),LOUTG(1),LOUTB(1),BVMOD(1),BVAR
     1G(1),BGP(1),BLP(1),BLQ(1),BSTL(1),HDLN(12)
      COMPLEX Z,LTAP(1)
      READ (INPT,110) (HDLN(I),I=1,12)
      READ (INPT,130) X,NB,X,X,NTL
C
C     BECAUSE BUS IS INCIDENT TO ITSELF THEN ALL ELEMENTS OF VECTOR
C     JRYT ARE PRESET TO ONE
C
      DO 10 I=1,NB
      JRYT(I)=1
   10 CONTINUE
      IF (IWRITE.LT.1) GO TO 20
      WRITE (OTPT,110) (HDLN(I),I=1,12)
      WRITE (OTPT,150) NB,NTL
   20 READ (INPT,110) (HDLN(I),I=1,12)
      IF (IWRITE.LT.2) GO TO 30
      WRITE (OTPT,110) (HDLN(I),I=1,12)
   30 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)
C
C     DEGREES OF BOTH BUSES INCIDENT TO THE LINE ARE INCREASED
C     BY ONE
C
      J=LBINP(I)
      JRYT(J)=JRYT(J)+1
      J=LBOUT(I)
      JRYT(J)=JRYT(J)+1
   40 CONTINUE
      IF (IWRITE.LT.2) GO TO 60
      WRITE (OTPT,120) (HDLN(I),I=1,12)
      DO 50 I=1,NTL
      WRITE (OTPT,140) LBINP(I),LBOUT(I),LINPG(I),LINPB(I),LG(I),LB(I),L
     1OUTG(I),LOUTB(I),LTAP(I)
   50 CONTINUE
   60 DO 70 I=1,NTL
      Z=1./CMPLX(LG(I),LB(I))
      LG(I)=REAL(Z)
      LB(I)=AIMAG(Z)
   70 CONTINUE
      READ (INPT,110) (HDLN(I),I=1,12)
      IF (IWRITE.GT.1) WRITE (OTPT,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
   80 CONTINUE
      IF (IWRITE.LT.2) GO TO 100
      WRITE (OTPT,120) (HDLN(I),I=1,12)
      DO 90 I=1,NB
      WRITE (OTPT,140) BNR(I),BTYP(I),BVMOD(I),BVARG(I),BGP(I),BLP(I),BL
     1Q(I),BSTL(I)
   90 CONTINUE
  100 RETURN
  110 FORMAT (//12A10/)
  120 FORMAT (12A10)
  130 FORMAT (1X,A5,I3,2A5,I4)
  140 FORMAT (2(1X,I5),2X,8(2X,E13.7))
  150 FORMAT (" NB = ",I3.3,",     ","NTL =",I4.3)
      END
```

```
CC
C
C
C
      SUBROUTINE LFNCM (NB,NLB,NYT,JRYT,ICYT,BTYP,YT,V,BCV,W,LW,DSX,MODE
     1,IFLAG,OTPT,IWRITE)
C
C
C     THIS SUBROUTINE IS THE HIGHEST LEVEL SUBROUTINE TO SOLVE THE LOAD
C     FLOW PROBLEM IN COMPLEX MODE USING A SPARSE MATRIX TECHNIQUE
C     (HARWELL PACKAGE) AND THE NEWTON METHOD
C
C     LIBRARY: LIBCHSM (HARWELL PACKAGE ME28)
C
C     NB     NUMBER OF BUSES
C     NLB    NUMBER OF LOAD BUSES
C     NYT    NUMBER OF NONZEROS IN BUS ADMITTANCE MATRIX
C     JRYT   VECTOR OF LENGTH (NB+1). IT HOLDS ROW INDICES OF SPARSE BUS
C            ADMITTANCE MATRIX
C     ICYT   VECTOR OF LENGTH NYT. IT HOLDS COLUMN INDICES OF SPARSE BUS
C            ADMITTANCE MATRIX
C     BTYP   VECTOR OF BUS TYPES (0  LOAD BUS,  1  GENERATOR BUS)
C     YT     SPARSE BUS ADMITTANCE MATRIX
C     V      COMPLEX BUS VOLTAGES (RECTANGULAR COORDINATES)
C     BCV    COMPLEX VECTOR OF LENGTH (NB-1). IT HOLDS NOMINAL VALUES
C            OF BUS CONTROL VARIABLES
C     W      REAL WORKSPACE
C     LW     LENGTH OF THE WORKSPACE W. IT SHOULD BE DECLARED AT LEAST
C            LW=36*(NB-1)+11*NZ, WHERE NZ=4*NYT-(NYT*NLB)/(NB-1)
C     IADJ    0   WHEN RHS OF THE ADJOINT SYSTEM OF EQUATIONS
C                 HAS BEEN CHANGED
C             1   FRESH CALCULATION OF THE ADJOINT SYSTEM OF EQUATIONS
C                 IS REQUIRED
C     IDER=   0   IF DERIVATIVES OF THE FUNCTION ARE NOT REQUIRED
C         =   M   IF DERIVATIVES OF THE MTH FUNCTION ARE REQUIRED
C     ILOAD=  0   IF LOAD FLOW SOLUTION IS NOT REQUIRED
C          =1     IF LOAD FLOW SOLUTION IS REQUIRED
C     ITEL   LIMIT OF ITERATIONS
C     VEPS   REQUIRED ACCURACY OF THE SOLUTION
C     TIMEL  LIMIT OF ITERATION TIME
C     MODE   MODE OF OPERATION
C             1 - MATRIX OF PERTURBED FLOW EQUATIONS IS FORMED AND
C                 FACTORIZED IN THE FIRST ITERATION OF THE CURRENT CALL
C                 TO THE SUBROUTINE AND IS UPDATED AND FACTORIZED IN
C                 THE SUBSEQUENT ITERATIONS USING THE PIVOTAL STRATEGY
C                 DETERMINED EARLIER
C             2 - MATRIX OF PERTURBED FLOW EQUATIONS IS UPDATED AND
C                 FACTORIZED IN EACH ITERATION USING THE PIVOTAL STRATEGY
C                 DETERMINED IN THE PREVIOUS CALL TO THE SUBROUTINE
C             3 - MATRIX OF PERTURBED FLOW EQUATIONS IS KEPT CONSTANT
C                 FROM THE PREVIOUS CALL TO THE SUBROUTINE
C     IFLAG  RETURN FLAG
C            -2 - INCORRECT MODE
C            -1 - WORKSPACE TO SMALL
C             0 - NORMAL RETURN
C             1 - LIMIT OF ITERATIONS REACHED
C             2 - LIMIT OF TIME REACHED
C     OTPT   NUMBER OF OUTPUT UNIT
C
      INTEGER JRYT(1),ICYT(1),BTYP(1),OTPT
      REAL W(1)
      COMPLEX YT(1),V(1),BCV(1),DS(1),DSX(1),Z,DSL
      COMMON /MDLFNCM/ JAK,JIRK,JICK,JICN,JAI,JDS,JIKEEP,JIW,JMAX/LAK/NZ
     1,LICK,LIRK
      COMMON /XLF3ID/ ITEL,TIMEL,VEPS,IDER,ILOAD,IADJ
```

```
      COMMON /AMUFL/ IMUFX
      DATA ITEL/10/,TIMEL/2/,VEPS/1.0E-06/,IDER/0/,ILOAD/1/,IADJ/1/
      IFLAG=0
      IT=0
      IF (IWRITE.GE.1) WRITE (OTPT,140) NB
      CALL SECOND (TS)
      TT=TS
      TA=0
      IF (MODE.LE.3.OR.MODE.GE.1) GO TO 10
      IFLAG=-2
      RETURN
   10 IF (MODE.GE.2) GO TO 20
      N=NB-1
      NR=N+N
      U=0.1
      NZ=4*NYT-(NYT*NLB)/N
      NI=4*N
      LICK=3*NZ
      LIRK=N+N+NZ
      JAK=1
      JIRK=JAK+LICK+LICK
      JICK=JIRK+LIRK
      JICN=JICK+NZ
      JAI=JICN+LICK
      JDS=JAI+NI
      JIKEEP=JDS+NI
      JIW=JIKEEP+10*N
      JWR=JIW+16*N
      JMAX=JWR+5*N-1
   20 IF (JMAX.LE.LW) GO TO 30
      IFLAG=-1
      RETURN
   30 IF (ILOAD.EQ.0) GO TO 100
      CALL STEPNCM (YT,JRYT,ICYT,V,W(JAI),BTYP,BCV,W(JAK),W(JIRK),W(JICK
     1),W(JICN),W(JDS),W(JIKEEP),W(JIW),W(JWR),NB,MODE,EPS,OTPT,IWRITE)
      IT=IT+1
      IF (MODE.EQ.1) MODE=2
      CALL SECOND (TIME)
      TT=TIME-TS
      TI=TT-TA
      TA=TT
      IF (IWRITE.GE.1) WRITE (OTPT,150) IT,EPS,TI,TT
      IF (EPS.LE.VEPS) GO TO 70
      IF (ITEL.LE.0) GO TO 40
      IF (IT.GE.ITEL) GO TO 50
   40 IF (TIMEL.LE.0) GO TO 30
      IF (TT.GE.TIMEL) GO TO 60
      GO TO 30
   50 IFLAG=1
      GO TO 70
   60 IFLAG=2
   70 VEPS=EPS
      TIMEL=TT
      ITEL=IT
      IF (IWRITE.LT.1) GO TO 80
      WRITE (OTPT,160) ITEL,IFLAG,VEPS,TIMEL
      WRITE (OTPT,170)
   80 J1=JIW-1
      J2=J1+NB
      DO 90 I=1,NB
      Z=V(I)
      J3=J1+I
      J4=J2+I
      W(J3)=CABS(Z)
      W(J4)=ATAN2(AIMAG(Z),REAL(Z))
```

```
      IF (IWRITE.LT.1) GO TO 90
      WRITE (OTPT,180) I,Z,W(J3),W(J4)
   90 CONTINUE
C     DETERMINATION OF THE SOLUTION OF THE ADJOINT SYSTEM OF EQUATIONS
  100 IF (IDER.EQ.0) GO TO 130
      IF (IADJ.EQ.0) GO TO 120
      IF (ILOAD.NE.0) GO TO 110
      CALL RHSLD (V,YT,JRYT,ICYT,BTYP,BCV,DS,W(JAI),N,OTPT,IWRITE)
      CALL FORMK (YT,JRYT,ICYT,BTYP,V,W(JAI),W(JAK),W(JIRK),W(JICK),N,NK
     1,OTPT,IWRITE)
      CALL ME28A (NR,NK,W(JAK),LICK,W(JIRK),LIRK,W(JICN),U,W(JIKEEP),W(J
     1IW),IFLAG)
      GO TO 120
  110 CALL FORMK (YT,JRYT,ICYT,BTYP,V,W(JAI),W(JAK),W(JIRK),W(JICK),N,NK
     1,OTPT,IWRITE)
      CALL ME28B (NR,NK,W(JAK),LICK,W(JIRK),W(JICK),W(JICN),W(JIKEEP),W(
     1JIW),W(JAI),IFLAG)
  120 IMUFX=0
      CALL FORMMU (V,W(JAI),NB,DSX,DSL,IDER,YT,JRYT,ICYT,BTYP,NYT,OTPT)
      IF (IMUFX.EQ.-1) RETURN
      CALL ME28C (NR,W(JAK),LICK,W(JICN),W(JIKEEP),DSX,W(JAI),0)
      DSX(NR+1)=DSL
      DSX(NR+2)=CONJG(DSL)
C
  130 RETURN
  140 FORMAT (///," LOAD FLOW SOLUTION OF ",I3,"-BUS SYSTEM USING THE",1
     1X,"COMPLEX NEWTON METHOD"/1X,68("-")/)
  150 FORMAT (//" IT = ",I2,4X,"EPS =",E12.6,4X,"ITERATION TIME ",F5.3,4
     1X,"TOTAL TIME ",F6.3/)
  160 FORMAT (//" RESULTS OF ANALYSIS",/" NUMBER OF ITERATIONS:",6X,I3,/
     1" RETURN FLAG:",16X,I2,/" ACCURACY OBTAINED:",E12.6,/" ANALYSIS TI
     2ME:",10X,F6.3,"SECONDS")
  170 FORMAT (/" VECTOR OF BUS VOLTAGES",//4X,"BUS",9X,"RECTANGULAR ","C
     1OORDINATES",11X,"POLAR COORDINATES"/,4X,3("-"),9X,22("-"),9X,21("-"
     2),/)
  180 FORMAT (3X,I4,2(3X,2(1X,F13.6)))
      END
```

```
CC
C
C
C
      SUBROUTINE STEPNCM (YT,JRYT,ICYT,V,AI,BTYP,BCV,AK,IRK,ICK,ICN,DS,I
     1KEEP,IW,WR,NB,IH,EPS,OTPT,IWRITE)
C
C     SUBROUTINE STEPNCM PERFORMS ONE ITERATION OF THE SOLUTION OF
C     POWER FLOW EQUATIONS USING THE COMPLEX NEWTON'S METHOD
C
C
C     LIBRARY: HARWELL PACKAGE ME28
C
      INTEGER JRYT(1),ICYT(1),BTYP(1),IRK(1),ICK(1),ICN(1),IKEEP(1),IW(1
     1),OTPT
      COMPLEX BCV(1),AK(1),DS(1),V(1),AI(1),YT(1),WR,Z
      COMMON /LAK/ NZ,LICK,LIRK
      N=NB-1
      NR=N+N
      IF (IWRITE.GE.2) WRITE (OTPT,60) IH
      EPS=0.
      CALL RHSLD (V,YT,JRYT,ICYT,BTYP,BCV,DS,AI,N,OTPT,IWRITE)
      IF (IH.EQ.3) GO TO 30
      CALL FORMK (YT,JRYT,ICYT,BTYP,V,AI,AK,IRK,ICK,N,NK,OTPT,IWRITE)
      IF (IH.EQ.2) GO TO 20
      U=0.1
      DO 10 I=1,NK
      ICN(I)=ICK(I)
   10 CONTINUE
      CALL ME28A (NR,NK,AK,LICK,IRK,LIRK,ICN,U,IKEEP,IW,IFLAG)
      GO TO 30
   20 CALL ME28B (NR,NK,AK,LICK,IRK,ICK,ICN,IKEEP,IW,WR,IFLAG)
   30 CALL ME28C (NR,AK,LICK,ICN,IKEEP,DS,WR,1)
      DO 40 I=1,N
      Z=DS(2*I-1)
      V(I)=V(I)+Z
      R=CABS(Z)
      IF (R.GT.EPS) EPS=R
   40 CONTINUE
      IF (IWRITE.LT.2) GO TO 50
      WRITE (OTPT,70)
      WRITE (OTPT,80) (L,V(L),L=1,N)
   50 RETURN
   60 FORMAT (///" RESULTS OF ITERATION.  MODE =",I2)
   70 FORMAT (//" VECTOR OF BUS VOLTAGES"/)
   80 FORMAT (2(2X,I4,":",2(F13.6)))
      END
```

```
CC
C
C
C
      SUBROUTINE RHSLD (V,YT,JRYT,ICYT,BTYP,BCV,DS,AI,N,OTPT,IWRITE)
C
C     THIS SUBROUTINE CALCULATES VECTOR DS OF PERTURBED LOAD FLOW
C     EQUATIONS /SEE EQUATION (17) SOS-80-2-R/ AND BUS CURRENT
C     VECTOR AI
C
      INTEGER JRYT(1),ICYT(1),BTYP(1),OTPT
      COMPLEX V(1),YT(1),BCV(1),DS(1),AI(1),CCURR,S
      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(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),CABS(V(I)))
      S=BCV(I)-S
      DS(J)=S
      DS(J1)=CONJG(S)
   20 CONTINUE
      IF (IWRITE.LT.3) RETURN
      WRITE (OTPT,30)
      WRITE (OTPT,40) (I,DS(I),I=1,N1)
      WRITE (OTPT,50)
      WRITE (OTPT,40) (I,AI(I),I=1,N)
      RETURN
   30 FORMAT (///" RHS VECTOR DS OF PERTURBED FLOW EQUATIONS"/)
   40 FORMAT ((2(3X,I3,":",2(F13.6))))
   50 FORMAT (///" VECTOR AI OF BUS CURRENTS"/)
      END
```

```
CC
C
C
C
      SUBROUTINE FORMK (YT,JRYT,ICYT,BTYP,V,AI,AK,IRK,ICK,N,NK,OTPT,IWRI
     1TE)
C
C     THIS SUBROUTINE FORMULATES MATRIX OF PERTURBED FLOW EQUATIONS
C     /SEE EQUATION (16) SOS-80-2-R/.  VECTORS AK,IRK,ICK ARE TO STORE
C     THE PERTURBED MATRIX IN A SPARSE FORM
C
      INTEGER JRYT(1),ICYT(1),BTYP(1),IRK(1),ICK(1),OTPT
      COMPLEX YT(1),V(1),AK(1),AI(1),CPX,CPY,CCURR,Z,CM
      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 DO LOOP
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(IR).EQ.1) GO TO 20
C
C     FILLING UP ELEMENTS OF A ROW OF MATRIX K CORRESPONDING
C     TO LOAD BUS
C
      DO 10 J=K1,K2
      IC=ICYT(J)
      JC1=IC+IC
      JC=JC1-1
      IF (BTYP(IC).EQ.2) GO TO 10
      CPX=YT(J)*Z
      NK=NK+1
      AK(NK)=CPX
      IRK(NK)=JR
      ICK(NK)=JC
      NK=NK+1
      AK(NK)=CONJG(CPX)
      IRK(NK)=JR1
      ICK(NK)=JC1
   10 CONTINUE
      NK=NK+1
      AK(NK)=CCURR
      IRK(NK)=JR
      ICK(NK)=JR1
      NK=NK+1
      AK(NK)=CONJG(CCURR)
      IRK(NK)=JR1
      ICK(NK)=JR
      GO TO 40
   20 DO 30 J=K1,K2
C
C     FILLING UP ELEMENTS OF A ROW OF MATRIX K CORRESPONDING
C     TO GENERATOR BUS
C
      IC=ICYT(J)
      IF (BTYP(IC).EQ.2) GO TO 30
      JC1=IC+IC
      JC=JC1-1
      CPX=0.5*YT(J)*Z
```

```
      CPY=CONJG(CPX)
      NK=NK+1
      AK(NK)=CPX
      IRK(NK)=JR
      ICK(NK)=JC
      NK=NK+1
      AK(NK)=CPY
      IRK(NK)=JR
      ICK(NK)=JC1
      NK=NK+1
      AK(NK)=CPX
      IRK(NK)=JR1
      ICK(NK)=JC
      NK=NK+1
      AK(NK)=CPY
      IRK(NK)=JR1
      ICK(NK)=JC1
   30 CONTINUE
      CM=CMPLX(0.,1./CABS(Z))
      CPX=(CONJG(CCURR)+CM*Z)/2.
      CPY=(CCURR+CM*V(IR))/2.
      AK(JE)=AK(JE)+CPX
      AK(JE+1)=AK(JE+1)+CPY
      AK(JE+2)=AK(JE+2)+CONJG(CPY)
      AK(JE+3)=AK(JE+3)+CONJG(CPX)
   40 IF (IWRITE.LT.3) GO TO 50
      WRITE (OTPT,70) IR
      WRITE (OTPT,80) (IRK(J),ICK(J),AK(J),J=JE,NK)
   50 CONTINUE
      RETURN
   60 FORMAT (///" MATRIX K OF PERTURBED FLOW EQUATIONS"/)
   70 FORMAT (" BUS NO.",I3)
   80 FORMAT ((2(3X,I3,1X,I3,":",2(F13.6))))
      END
```

```
CC
C
C
C
      COMPLEX FUNCTIONCCURR(V,YT,JRYT,ICYT,M)
C
C
C     THIS FUNCTION SUBPROGRAM CALCULATES THE VALUE OF CURRENT INJECTED
C     INTO M-TH BUS FOR THE GIVEN VECTOR V OF BUS VOLTAGES
C
      INTEGER JRYT(1),ICYT(1)
      COMPLEX V(1),YT(1)
      K1=JRYT(M)
      K2=JRYT(M+1)-1
      CCURR=(0.,0.)
      DO 10 I=K1,K2
      CCURR=CCURR+YT(I)*V(ICYT(I))
   10 CONTINUE
      RETURN
      END
```

```
CC
C
C
C
      SUBROUTINE DERIVX (LBINP,LBOUT,LG,LB,LTAP,BTYP,V,NB,L1,L2,DSX,NTL,
     1ICODE,DF1,DF2)
C
C
C     THIS SUBROUTINE CALCULATES THE REQUIRED NETWORK SENSITIVITIES
C     USING THE SOLUTION OF AN ADJOINT SYSTEM WHOSE RIGHT-HAND-SIDE
C     VECTOR IS DEFINED BY THE USER IN HIS SUBROUTINE FORMMU IN THE
C     MAIN PROGRAM
C
C     DSX      THE ADJOINT SOLUTION VECTOR FOR THE FUNCTION CONSIDERED
C              IN THE SUBROUTINE FORMMU BY THE USER
C     ICODE    THE CODE FOR VARIOUS CONTROL VARIABLES
C         1 - BUS-TYPE CONTROL VARIABLES(PL,QL,PG,/VG/,ETC.)
C         2 - CONTROL VARIABLES ASSOCIATED WITH SHUNT ADMITTANCE AT A BUS
C         3 - CONTROL VARIABLES ASSOCIATED WITH LINE ADMITTANCE
C         4 - COMPLEX TURNS RATIO OF A PHASE-SHIFTER IN RECTANGULAR MODE
C         5 - COMPLEX TURNS RATIO OF A PHASE-SHIFTER IN POLAR MODE
C         6 - REAL TURNS RATIO OF A TAP-CHANGING-UNDER-LOAD TRANSFORMER
C         7 - CONTROL VARIABLES ASSOCIATED WITH THE INTERNAL IMPEDANCE
C             OF A TRANSFORMER
C
      INTEGER LBINP(1),LBOUT(1),BTYP(1),L1,L2,NTL,ICODE
      REAL LG(1),LB(1)
      COMPLEX V(1),DSX(1),LTAP(1),DFX,FA1,FA2,CA,CA2,YTAP
      IF (ICODE.EQ.1) GO TO 20
      DO 10 I=1,NB
      IF (BTYP(I).EQ.0) GO TO 10
      DSX(I)=REAL(DSX(I))
   10 CONTINUE
      DSX(NB)=(0.0,0.0)
   20 GO TO (30,60,70,80,80,80,80), ICODE
   30 IF (BTYP(L1).NE.0) GO TO 40
      DFX=CONJG(DSX(L1))
      GO TO 50
   40 DFX=DSX(L1)
   50 GO TO 120
   60 ABS=CABS(V(L1))
      ABS=ABS*ABS
      DFX=-ABS*DSX(L1)
      GO TO 120
   70 FA1=DSX(L1)*CONJG(V(L1))
      FA2=DSX(L2)*CONJG(V(L2))
      DFX=(FA2-FA1)*(V(L1)-V(L2))
      GO TO 120
   80 DO 90 I=1,NTL
      IB1=LBINP(I)
      IF (IB1.NE.L1) GO TO 90
      IB2=LBOUT(I)
      IF (IB2.NE.L2) GO TO 90
      CA=LTAP(I)
      YTAP=CMPLX(LG(I),LB(I))
      GO TO 100
   90 CONTINUE
  100 CA2=V(L1)/CA
      FA1=DSX(L1)*YTAP
      FA2=DSX(L2)*YTAP
      IF (ICODE.EQ.7) GO TO 110
      DFX=CA2*(2*REAL(FA1)*CONJG(CA2)-(CONJG(FA1)+FA2)*CONJG(V(L2)))/CA
      GO TO 120
  110 YTAP=YTAP*YTAP
      DFX=YTAP*(CA2-V(L2))*(DSX(L1)*CONJG(CA2)-DSX(L2)*CONJG(V(L2)))
```

```
120 DF1=2*REAL(DFX)
    DF2=-2*AIMAG(DFX)
    IF (ICODE.EQ.6) GO TO 130
    IF (ICODE.NE.5) GO TO 140
    DFR=DF1
    DFI=DF2
    A1=REAL(CA)
    A2=AIMAG(CA)
    ABSCA=CABS(CA)
    DF1=(A1*DFR+A2*DFI)/ABSCA
    DF2=(-A2*DFR+A1*DFI)
    GO TO 140
130 DF2=0.0
140 RETURN
    END
```

```
C
C
C
      SUBROUTINE FORMMU (V,AI,NB,DSX,DSL,IDER,YT,JRYT,ICYT,BTYP,NYT,OTPT
     1)
C
C        THIS SUBROUTINE SHOLUD BE PROVIDED BY THE USER IN HIS
C          MAIN PROGRAM WHEN IDER IS NOT EQUAL TO 0
      INTEGER JRYT(1),ICYT(1),OTPT
      COMPLEX V(1),AI(1),DSX(1),YT(1)
      COMMON /AMUF/ IMUFX
C
      WRITE (OTPT,10)
      IMUFX=-1
      RETURN
C
   10 FORMAT (/,"   ERROR - MISSING SUBROUTINE FORMMU.")
      END
```