รายการอ้างอิง

1. William D. Stevenson, Jr. , "Elements of Power System Analysis", McGraw-Hill Series in Electrical Engineering, pp. 95-110, 254-258.
2. M.A. PAI, "Computer Techniques in Power System Analysis", McGraw-Hill Series in Electrical Engineering, pp. 521-586.
3. Merisa L. Crow and Bernard C, Lesieulre, "Voltage Collapse An Engineering Challenge", IEEE Potential 02/94.
4. IEEE Power System Engineering Committee, System Dynamic Performance Subcommittee, "Voltage Stability of Power System : Concepts Analitical Tools, and Industry experience", IEEE Publication No. 900TH0358-2 PWR.
5. Carson W Taylor, "Power System Voltage Stability", pp. 6-61, Mc.Graw-Hill Inc,1994.
6. Venkataramana Ajjarapu, Ping Lin Lau, Srinivasu Bahula, "An Optimal Reactive Power Planning Strategy Against Voltage Callapse", IEEE trans on Power System Vol. 9, No. 2, May 1992.
7. Toran Gonen, "Electrc Power Distribution System Engineering", McGraw-Hill Series in Electrical Engineering, pp. 318-378.
8. T. Van Cutsen, "A Method to Compute Reactive Power Margins with respect to Voltage Collapse", IEEE Trans on Power System Vol. 6, No. 1, February 1991, pp. 145-153.
9. M. El-Marsafawy, "Application of series-capacitor and shunt-reactor compensation to an existing practical AC transmission line", IEEE PROCEDINGS-C Vol. 138, No. 4. ,July 1991, pp. 330-336.
10. R.R. Austria, X.Y. Chao, N.D. Reppen , and D.E. Welsh, "Integrated Approach to Transfer Limit Calculations", IEEE Cmputer Applications in Power 1992, pp. 48-22.
11. Venkataramana Ajjarapu, Colin Christy, "The Continuation Power Flow: A Tool for Steady State Voltage Stability Analysis", IEEE Transon Power System.
12. Glen W. Stagg, Ahmed H Ei-abiad, "Computer Methods in Power System Analysis", McGraw-Hill International Editions 1968, pp. 17-117, 257-324.

13. B. M. Weedy ,"Electric Power Systems", John Wiley and Sons 1988 Third Edition, pp. 210-246.

14. Jasbir S. Arora, "Introduction to Optimum Design", McGraw-Hill International Editions 1989,pp. 178-168.

15. Glover Sarma, "Power System Analysis & Design with personal computer applications", PWS Publishing Company, pp. 253-285.

16. A.C. Zambroni de Souza, V.H. Quintana, "Identification of Voltage Collapse Margins in Power Systems" CCECE/CCGEI 93, pp. 933-937.

ภาคผนวก

ภาคผนวก ก.

โปรแกรมที่ใช้ในการวิเคราะห์หาจุดวิกฤติในระบบไฟฟ้ากำลัง

```
REM xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
REM
REM       CPF- Voltage Collapse porgram with compensate element
REM
REM xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
DIM tap(40, 40), b(40, 40), a(40, 40), m(40, 40), front(40, 40), back(40, 40), inv(40, 40), i(40,
40), r(40, 40), x(40, 40), xy(40, 40)
DIM pf(40), pg(40), pl(40), yb(40, 40), ybb(40, 40), th(40), la(40), value(40), y(40, 40), yy
(40, 40), yyy(40, 40), v(40), ql(40), qg(40)
DIM yyo(40), yo(40), yc(40)
DEFSNG A
maxbus = 30: basemva = 100
est = .005: lp = 0: weak = 0: it = 0: stp = 8: freq = 60: sbase = 1
shunt = 0: kl = .5: kg = .5: ltc = 0: set = 1
TIMER ON
stimes = TIMER
CLS
REM=====================================================================
REM ============       Initial Value of each bus       ==================
REM=====================================================================
REM *****        Load Flow Data - Base on 100 MVA power        *****
REM *****   Change the value here for difference test system   ******
th(1) = 0: th(2) = -2.7339: th(3) = -4.6815: th(4) = -5.6077: th(5) = -8.993
th(6) = -6.4547: th(7) = -8.0244: th(8) = -6.4733: th(9) = -8.03: th(10) = -9.9268
th(11) = -6.1345: th(12) = -9.4036: th(13) = -8.2049: th(14) = -10.3086: th(15) = -10.36
th(16) = -9.9028: th(17) = -10.1356: th(18) = -10.9253: th(19) = -11.0615: th(20) = -10.831
th(21) = -10.4047: th(22) = -10.3936: th(23) = -10.7221: th(24) = -10.8465: th(25) = -10.9074
th(26) = -11.3345: th(27) = -10.6624: th(28) = -6.8671: th(29) = -11.8893: th(30) = -12.7699

v(1) = 1.05: v(2) = 1.0338: v(3) = 1.0313: v(4) = 1.0263: v(5) = 1.0058
v(6) = 1.0208: v(7) = 1.0069: v(8) = 1.023: v(9) = 1.0332: v(10) = 1.0183
v(11) = 1.0913: v(12) = 1.0399: v(13) = 1.0883: v(14) = 1.0236: v(15) = 1.0179
v(16) = 1.0235: v(17) = 1.0144: v(18) = 1.0057: v(19) = 1.0017: v(20) = 1.0051
v(21) = 1.0061: v(22) = 1.0069: v(23) = 1.0053: v(24) = .9971: v(25) = 1.0086
```

```
VERSION 4.00
Begin VB.Form Form1
   AutoRedraw      =   -1  'True
   Caption         =   "VOLTAGE COLLAPSING POINT ANALYSIS PROGRAM"
   ClientHeight    =   8595
   ClientLeft      =   1095
   ClientTop       =   1515
   ClientWidth     =   9885
   Height          =   9270
   Left            =   900
   LinkTopic       =   "Form1"
   ScaleHeight     =   35.813
   ScaleMode       =   4  'Character
   ScaleWidth      =   82.375
   Top             =   1035
   Width           =   10275
   Begin VB.Label Label5
      Height          =   255
      Left            =   6120
      TabIndex        =   4
      Top             =   360
      Width           =   855
   End
   Begin VB.Label Label4
      Height          =   255
      Left            =   5040
      TabIndex        =   3
      Top             =   360
      Width           =   495
   End
   Begin VB.Label Label3
      Caption         =   "seconds"
      Height          =   255
```

```
    Left            =   6960
    TabIndex        =   2
    Top             =   360
    Width           =   735
  End
  Begin VB.Label Label2
    Caption         =   "minutes"
    Height          =   255
    Left            =   5520
    TabIndex        =   1
    Top             =   360
    Width           =   615
  End
  Begin VB.Label Label1
    Caption         =   "Spent times"
    Height          =   255
    Left            =   4080
    TabIndex        =   0
    Top             =   360
    Width           =   975
  End
End
Attribute VB_Name = "Form1"
Attribute VB_Creatable = False
Attribute VB_Exposed = False
DefSng A
Private Sub Form_DblClick()
Rem xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Rem
Rem       CPF- Voltage Collapse porgram with compensate element
Rem
Rem xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Dim tap(40, 40) ', b(40, 40),
```

```
Dim a(40, 40), m(40, 40), front(40, 40), back(40, 40), inv(40, 40), i(40, 40), r(40, 40), x(40, 40), xy(40, 40)
Dim pf(40), pg(40), pl(40), yb(40, 40), ybb(40, 40), th(40), la(40), value(40), y(40, 40), yy(40, 40), yyy(40, 40), v
(40), ql(40), qg(40)
Dim yyo(40), yo(40), yc(40)

maxbus = 30: basemva = 100
est = 0.001: lp = 0: weak = 0: it = 0: stp = 8: freq = 60: sbase = 1
shunt = 0: kl = 0.5: kg = 0.5: ltc = 0: setx = 1
' TIMER ON
stimes = Timer
Cls

Rem==========================================================================
Rem ============        Initial Value of each bus       ================
Rem==========================================================================
Rem *****         Load Flow Data - Base on 100 MVA power         *****
Rem *****    Change the value here for difference test system    ******

th(1) = 0: th(2) = -2.7339: th(3) = -4.6815: th(4) = -5.6077: th(5) = -8.993
th(6) = -6.4547: th(7) = -8.0244: th(8) = -6.4733: th(9) = -8.03: th(10) = -9.9268
th(11) = -6.1345: th(12) = -9.4036: th(13) = -8.2049: th(14) = -10.3086: th(15) = -10.36
th(16) = -9.9028: th(17) = -10.1356: th(18) = -10.9253: th(19) = -11.0615: th(20) = -10.831
th(21) = -10.4047: th(22) = -10.3936: th(23) = -10.7221: th(24) = -10.8465: th(25) = -10.9074
th(26) = -11.3345: th(27) = -10.6624: th(28) = -6.8671: th(29) = -11.8893: th(30) = -12.7699

v(1) = 1.05: v(2) = 1.0338: v(3) = 1.0313: v(4) = 1.0263: v(5) = 1.0058
v(6) = 1.0208: v(7) = 1.0069: v(8) = 1.023: v(9) = 1.0332: v(10) = 1.0183
v(11) = 1.0913: v(12) = 1.0399: v(13) = 1.0883: v(14) = 1.0236: v(15) = 1.0179
v(16) = 1.0235: v(17) = 1.0144: v(18) = 1.0057: v(19) = 1.0017: v(20) = 1.0051
v(21) = 1.0061: v(22) = 1.0069: v(23) = 1.0053: v(24) = 0.9971: v(25) = 1.0086
v(26) = 0.9908: v(27) = 1.0245: v(28) = 1.0156: v(29) = 1.0047: v(30) = 0.9932
```

la(1) = 0: la(2) = 0: la(3) = 0: la(4) = 0: la(5) = 0

la(6) = 0: la(7) = 0: la(8) = 0: la(9) = 0: la(10) = 0

la(11) = 0: la(12) = 0: la(13) = 0: la(14) = 0: la(15) = 0

la(16) = 0: la(17) = 0: la(18) = 0: la(19) = 0: la(20) = 0

la(21) = 0: la(22) = 0: la(23) = 0: la(24) = 0: la(25) = 0

la(26) = 0: la(27) = 0: la(28) = 0: la(29) = 0: la(30) = 0

pg(1) = 138.48: pg(2) = 57.56: pg(3) = 0: pg(4) = 0: pg(5) = 24.56

pg(6) = 0: pg(7) = 0: pg(8) = 35: pg(9) = 0: pg(10) = 0

pg(11) = 17.93: pg(12) = 0: pg(13) = 16.91: pg(14) = 0: pg(15) = 0

pg(16) = 0: pg(17) = 0: pg(18) = 0: pg(19) = 0: pg(20) = 0

pg(21) = 0: pg(22) = 0: pg(23) = 0: pg(24) = 0: pg(25) = 0

pg(26) = 0: pg(27) = 0: pg(28) = 0: pg(29) = 0: pg(30) = 0

qg(1) = -2.79: qg(2) = 2.47: qg(3) = 0: qg(4) = 0: qg(5) = 22.57

qg(6) = 0: qg(7) = 0: qg(8) = 34.84: qg(9) = 0: qg(10) = 0

qg(11) = 30.78: qg(12) = 0: qg(13) = 37.83: qg(14) = 0: qg(15) = 0

qg(16) = 0: qg(17) = 0: qg(18) = 0: qg(19) = 0: qg(20) = 0

qg(21) = 0: qg(22) = 0: qg(23) = 0: qg(24) = 0: qg(25) = 0

qg(26) = 0: qg(27) = 0: qg(28) = 0: qg(29) = 0: qg(30) = 0

pl(1) = 0: pl(2) = 21.7: pl(3) = 2.4: pl(4) = 7.6: pl(5) = 94.2

pl(6) = 0: pl(7) = 22.8: pl(8) = 30: pl(9) = 0: pl(10) = 5.8

pl(11) = 0: pl(12) = 11.2: pl(13) = 0: pl(14) = 6.2: pl(15) = 8.2

pl(16) = 3.5: pl(17) = 9: pl(18) = 3.2: pl(19) = 9.5: pl(20) = 2.2

pl(21) = 17.5: pl(22) = 0: pl(23) = 3.2: pl(24) = 8.7: pl(25) = 0

pl(26) = 3.5: pl(27) = 0: pl(28) = 0: pl(29) = 2.4: pl(30) = 10.6

ql(1) = 0: ql(2) = 12.7: ql(3) = 1.2: ql(4) = 1.6: ql(5) = 19

ql(6) = 0: ql(7) = 10.9: ql(8) = 30: ql(9) = 0: ql(10) = 2

ql(11) = 0: ql(12) = 7.5: ql(13) = 0: ql(14) = 1.6: ql(15) = 2.5

ql(16) = 1.8: ql(17) = 5.8: ql(18) = 0.9: ql(19) = 3.4: ql(20) = 0.7

ql(21) = 11.2: ql(22) = 0: ql(23) = 1.6: ql(24) = 6.7: ql(25) = 0

```
ql(26) = 2.3: ql(27) = 0: ql(28) = 0: ql(29) = 0.9: ql(30) = 1.9

For j = 1 To maxbus
pl(j) = pl(j) / basemva
ql(j) = ql(j) / basemva
pg(j) = pg(j) / basemva
qg(j) = qg(j) / basemva
Next j

pf(1) = 0: pf(2) = 30.33: pf(3) = 26.56: pf(4) = 11.88: pf(5) = 11.4
pf(6) = 0: pf(7) = 25.55: pf(8) = 45: pf(9) = 0: pf(10) = 19.02
pf(11) = 0: pf(12) = 33.8: pf(13) = 0: pf(14) = 14.47: pf(15) = 16.95
pf(16) = 27.21: pf(17) = 32.79: pf(18) = 15.7: pf(19) = 19.69: pf(20) = 17.65
pf(21) = 32.62: pf(22) = 0: pf(23) = 26.56: pf(24) = 37.6: pf(25) = 0
pf(26) = 33.31: pf(27) = 0: pf(28) = 0: pf(29) = 20.55: pf(30) = 10.16

Rem ********************  Ybus Data  ************************
Rem *******  r is the value of Resistance at each line  ********
Rem *******    X isthe value ofimpedance at each line   ********
Rem ************************************************************
Rem *****   Change the value here for difference test system   ******
r(1, 2) = 0.0192: x(1, 2) = 0.0575
r(2, 1) = r(1, 2): x(2, 1) = x(1, 2)
r(1, 3) = 0.0452: x(1, 3) = 0.1852
r(3, 1) = r(1, 3): x(3, 1) = x(1, 3)
r(2, 4) = 0.057: x(2, 4) = 0.1737
r(4, 2) = r(2, 4): x(4, 2) = x(2, 4)
r(3, 4) = 0.0132: x(3, 4) = 0.0379
r(4, 3) = r(3, 4): x(4, 3) = x(3, 4)
r(2, 5) = 0.0472: x(2, 5) = 0.1983
r(5, 2) = r(2, 5): x(5, 2) = x(2, 5)
r(2, 6) = 0.0581: x(2, 6) = 0.1763
r(6, 2) = r(2, 6): x(6, 2) = x(2, 6)
```

r(4, 6) = 0.0119: x(4, 6) = 0.0414

r(6, 4) = r(4, 6): x(6, 4) = x(4, 6)

r(5, 7) = 0.046: x(5, 7) = 0.116

r(7, 5) = r(5, 7): x(7, 5) = x(5, 7)

r(6, 7) = 0.0267: x(6, 7) = 0.082

r(7, 6) = r(6, 7): x(7, 6) = x(6, 7)

r(6, 8) = 0.012: x(6, 8) = 0.042

r(8, 6) = r(6, 8): x(8, 6) = x(6, 8)

r(6, 9) = 0: x(6, 9) = 0.208

r(9, 6) = r(6, 9): x(9, 6) = x(6, 9)

r(6, 10) = 0: x(6, 10) = 0.556

r(10, 6) = r(6, 10): x(10, 6) = x(6, 10)

r(9, 11) = 0: x(9, 11) = 0.208

r(11, 9) = r(9, 11): x(11, 9) = x(9, 11)

r(9, 10) = 0: x(9, 10) = 0.11

r(10, 9) = r(9, 10): x(10, 9) = x(9, 10)

r(4, 12) = 0: x(4, 12) = 0.256

r(12, 4) = r(4, 12): x(12, 4) = x(4, 12)

r(12, 13) = 0: x(12, 13) = 0.14

r(13, 12) = r(12, 13): x(13, 12) = x(12, 13)

r(12, 14) = 0.1231: x(12, 14) = 0.2559

r(14, 12) = r(12, 14): x(14, 12) = x(12, 14)

r(12, 15) = 0.0662: x(12, 15) = 0.1304

r(15, 12) = r(12, 15): x(15, 12) = x(12, 15)

r(12, 16) = 0.0945: x(12, 16) = 0.1987

r(16, 12) = r(12, 16): x(16, 12) = x(12, 16)

r(14, 15) = 0.221: x(14, 15) = 0.197

r(15, 14) = r(14, 15): x(15, 14) = x(14, 15)

r(16, 17) = 0.0824: x(16, 17) = 0.1932

r(17, 16) = r(16, 17): x(17, 16) = x(16, 17)

r(15, 18) = 0.107: x(15, 18) = 0.2185

r(18, 15) = r(15, 18): x(18, 15) = x(15, 18)

r(18, 19) = 0.0639: x(18, 19) = 0.1292

r(19, 18) = r(18, 19): x(19, 18) = x(18, 19)

r(19, 20) = 0.034: x(19, 20) = 0.068

r(20, 19) = r(19, 20): x(20, 19) = x(19, 20)

r(10, 20) = 0.0936: x(10, 20) = 0.209

r(20, 10) = r(10, 20): x(20, 10) = x(10, 20)

r(10, 17) = 0.0324: x(10, 17) = 0.0845

r(17, 10) = r(10, 17): x(17, 10) = x(10, 17)

r(10, 21) = 0.0348: x(10, 21) = 0.0749

r(21, 10) = r(10, 21): x(21, 10) = x(10, 21)

r(10, 22) = 0.0727: x(10, 22) = 0.1499

r(22, 10) = r(10, 22): x(22, 10) = x(10, 22)

r(21, 22) = 0.0116: x(21, 22) = 0.0236

r(22, 21) = r(21, 22): x(22, 21) = x(21, 22)

r(15, 23) = 0.1: x(15, 23) = 0.202

r(23, 15) = r(15, 23): x(23, 15) = x(15, 23)

r(22, 24) = 0.115: x(22, 24) = 0.179

r(24, 22) = r(22, 24): x(24, 22) = x(22, 24)

r(23, 24) = 0.132: x(23, 24) = 0.27

r(24, 23) = r(23, 24): x(24, 23) = x(23, 24)

r(24, 25) = 0.1885: x(24, 25) = 0.3292

r(25, 24) = r(24, 25): x(25, 24) = x(24, 25)

r(25, 26) = 0.2544: x(25, 26) = 0.38

r(26, 25) = r(25, 26): x(26, 25) = x(25, 26)

r(25, 27) = 0.1093: x(25, 27) = 0.2087

r(27, 25) = r(25, 27): x(27, 25) = x(25, 27)

r(26, 27) = 0: x(26, 27) = 0.396

r(27, 26) = r(26, 27): x(27, 26) = x(26, 27)

r(27, 29) = 0.2198: x(27, 29) = 0.4153

r(29, 27) = r(27, 29): x(29, 27) = x(27, 29)

r(27, 30) = 0.3202: x(27, 30) = 0.6027

r(30, 27) = r(27, 30): x(30, 27) = x(27, 30)

r(29, 30) = 0.2399: x(29, 30) = 0.4533

r(30, 29) = r(29, 30): x(30, 29) = x(29, 30)

```
r(8, 28) = 0.0636: x(8, 28) = 0.2
r(28, 8) = r(8, 28): x(28, 8) = x(8, 28)
r(6, 28) = 0.0169: x(6, 28) = 0.0599
r(28, 6) = r(6, 28): x(28, 6) = x(6, 28)


Rem ================================================================
Rem                    Transformer Data
Rem ================================================================
For k = 1 To maxbus
For j = 1 To maxbus
tap(k, j) = 1
Next j
Next k
tap(6, 9) = 1.0155
tap(6, 10) = 0.9629
tap(4, 12) = 1.0129
tap(28, 27) = 0.9581


Rem================================================================
Rem =======          Select for  of compensate element          =======
Rem================================================================
start:
Form1.ScaleMode = 4
'Print "Please input bus number if you want to specify bus "
'INPUT "Or select 0 to automatic diagnose"; selt: CLS
selt = Val(InputBox("Please input bus number if you want to specify bus " & Chr(13) & Chr(10) & "Or select 0 to
automatic diagnose"))
Cls

'INPUT "Please select (1) for compensate system "; cps
```

```
cps = Val(InputBox("Please select (1) for compensate system "))

Rem ==========        If select wrong number        ==========
Rem *****  Change the value here for difference test system  ******
If selt > maxbus Or selt < 0 Then GoTo start
Cls

If cps <> 1 Then GoTo cps

inputitem:
'INPUT "Select 1 for Series C, 2 for Shunt C, 3 for LTC"; item
Item = Val(InputBox("Select 1 for Series C, 2 for Shunt C, 3 for LTC"))
      If Item = 1 Then GoSub series
      If Item = 2 Then
      shunt = 1
      'INPUT "Which Bus this Shunt C be added"; addbus
      addbus = Val(InputBox("Which Bus this Shunt C be added"))
      'INPUT "Please Input the Mvar for this Shunt-C"; xs
      xs = Val(InputBox("Please Input the Mvar for this Shunt-C"))
      GoSub shunt
      Cls
      End If
      If Item = 3 Then GoSub ltc

Rem  *********  In case of never select any item  *****************
If Item > 4 Or Item < 1 Then GoTo inputitem
'INPUT more$
more$ = MsgBox(" MORE ? ", vbYesNo)
If more$ = vbYes Then GoTo inputitem
Cls

Rem ***************************************************************************
Rem *******        If the compensated system was selected        *******
```

```
Rem *******  The value of r,x will changed and will effected here   *******
Rem *******            yb(i,j)+ybb(i,j)j =-1/(r+xj)                *******
Rem ************************************************************************
cps:
For k = 1 To maxbus
For j = 1 To maxbus
  If (r(k, j) * r(k, j)) + (x(k, j) * x(k, j)) = 0 Then GoTo cont
  yb(k, j) = (-r(k, j) / ((r(k, j) * r(k, j)) + (x(k, j) * x(k, j)))) / tap(k, j)
  ybb(k, j) = (x(k, j) / (r(k, j) ^ 2 + x(k, j) ^ 2)) / tap(k, j)
cont:
Next j
Next k
Cls

Rem=========================================================================
Rem                  Transformer  Admittance  Matrix
Rem=========================================================================
yo(6) = yb(6, 9) * (1 - (tap(6, 9)) / (tap(6, 9) ^ 2)) + yb(6, 10) * (1 - (tap(6, 10)) / (tap(6, 10) ^ 2))
yyo(6) = ybb(6, 9) * (1 - (tap(6, 9)) / (tap(6, 9) ^ 2)) + ybb(6, 10) * (1 - (tap(6, 10)) / (tap(6, 10) ^ 2))
yo(9) = yb(9, 6) * ((tap(6, 9) - 1) / tap(6, 9))
yyo(9) = ybb(9, 6) * ((tap(6, 9) - 1) / tap(6, 9))
yo(10) = yb(10, 6) * ((tap(6, 10) - 1) / tap(6, 10))
yyo(10) = ybb(10, 6) * ((tap(6, 10) - 1) / tap(6, 10))
yo(4) = yb(4, 12) * (1 - (tap(4, 12)) / (tap(4, 12) ^ 2))
yyo(4) = ybb(4, 12) * (1 - (tap(4, 12)) / (tap(4, 12) ^ 2))
yo(12) = yb(4, 12) * ((tap(4, 12) - 1) / tap(4, 12))
yyo(12) = ybb(4, 12) * ((tap(4, 12) - 1) / tap(4, 12))
yo(28) = yb(28, 27) * (1 - (tap(28, 27)) / (tap(28, 27) ^ 2))
yyo(28) = ybb(28, 27) * (1 - (tap(28, 27)) / (tap(28, 27) ^ 2))
yo(27) = yb(28, 27) * ((tap(28, 27) - 1) / tap(28, 27))
yyo(27) = ybb(28, 27) * ((tap(28, 27) - 1) / tap(28, 27))

Rem=========================================================================
```

```
Rem                    Half Line Charging Admittance
Rem=========================================================================
yc(1) = 0.0468
yc(2) = 0.0844
yc(3) = 0.0246
yc(4) = 0.0271
yc(5) = 0.0311
yc(6) = 0.0427
yc(7) = 0.0187
yc(8) = 0.0259
yc(28) = 0.0279

Rem ***********************************************************************
Rem ***          Input the value of each element by                 ***
Rem ***       Yij(i<>j)=Yij  and Yii=Sigma Yij(j=1 to n)            ***
Rem ***********************************************************************

Rem *****  Change the value here for difference test system  ******
Rem ========================================================================
Rem                          Form Ybus
Rem ========================================================================
Print "Ybus Forming"

For k = 1 To maxbus
For j = 1 To maxbus
xy(k, j) = 0
yyy(k, j) = 0
If k <> j Then
    xy(k, j) = yb(k, j): yyy(k, j) = ybb(k, j)
End If

If k = j Then
  For u = 1 To maxbus
```

```
            xy(k, k) = xy(k, k) - yb(k, u)
            yyy(k, k) = yyy(k, k) - ybb(k, u)
        Next u
        xy(k, j) = xy(k, j) + yo(k)
        yyy(k, j) = yyy(k, j) + yyo(k) + yc(k)
End If

Next j
Next k

For k = 1 To maxbus
For j = 1 To maxbus
  y(k, j) = Sqr(xy(k, j) ^ 2 + yyy(k, j) ^ 2)
            If xy(k, j) = 0 Then
                If yyy(k, j) < 0 Then yy(k, j) = -90
                If yyy(k, j) > 0 Then yy(k, j) = 90
            End If
        If xy(k, j) <> 0 Then
        res = Abs(yyy(k, j) / xy(k, j))
        'For u = 0 To 90 Step 0.01
        'If Abs(res - (Tan(u * (22 / 7) / 180))) < 0.01 Then
        u = Atn(res) * 180 / 22 * 7
            If xy(k, j) > 0 Then
                If yyy(k, j) > 0 Then yy(k, j) = u
                If yyy(k, j) < 0 Then yy(k, j) = -1 * u
            End If
            If xy(k, j) < 0 Then
                If yyy(k, j) > 0 Then yy(k, j) = 180 - u
                If yyy(k, j) < 0 Then yy(k, j) = 180 + u
            End If
        'End If
        'Next u
        End If
```

```
Next j
Next k
Cls

Rem==========================================================================
Rem ============== Identify that which bus is the weakest ================
Rem==========================================================================
Cls
If selt <> 0 Then
        GoTo busselected
End If

weakest:
weak = 0
c = 0.5 * (3.57888)
Print "Finding Weakest Bus"

For jj = 2 To maxbus
bus = jj

theta = th(bus): lamda = 0
GoSub jacobian
GoSub inverse
        dtheta = inv(1, 3)
        dv = inv(2, 3)
        dlamda = inv(3, 3)
value(jj) = dv / (c)
Next jj
Cls

Max = 2
For jj = 3 To maxbus
If Abs(value(jj)) > Abs(value(Max)) Then
```

```
Max = jj
End If
Next jj

weak = 1
Print "Weakest bus is bus number"; Max
bus = Max
GoTo cpf

Rem=====================================================================
Rem ========  Run CPF to find the critical point of this system  =========
Rem=====================================================================
busselected:
Max = selt
bus = Max
Print "Bus selected is bus number"; Max
weak = 1
GoTo cpf
End


Rem=====================================================================
Rem ============          Continuation Power Flow          ================
Rem=====================================================================
cpf:

Rem ********   vo(bus)=Voltage at each bus (by initial value)    **********
Rem ********        but v is the calculating value               **********

theta = th(bus): lamda = 0

predictor:
Rem xxxxxxxxxxxxxxxxxxxxxxxxx   PREDICTOR   xxxxxxxxxxxxxxxxxxxxxxxxx
```

```
Rem *********  If LTC added this must effect the system    **********
Rem xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Rem ***************************************************************
Rem ******    Increase step size after critical point passed    ******
Rem ***************************************************************

If setx = -1 Then
stp = stp * 2
End If

GoSub jacobian
GoSub inverse
        dtheta = inv(1, 3)
        dv = inv(2, 3)
        dlamda = inv(3, 3)

Rem ************************************************************************
Rem **** Remember the old value to re-use when critical point reached ****
Rem ************************************************************************
oldtheta = th(bus)
oldv = v(bus)
oldlamda = lamda
GoSub update

corrector:
Rem xxxxxxxxxxxxxxxxxxxx   CORRECTOR   xxxxxxxxxxxxxxxxxxxx
Rem IF shunt = 1 THEN GOSUB shunt
GoSub functio
GoSub jacobian

For k = 1 To 3
For j = 1 To 3
```

```
a(k, j) = -a(k, j)
Next j
Next k

GoSub inverse

For k = 1 To 3
For j = 1 To 3
        front(k, j) = inv(k, j)
        back(k, j) = 0
Next j
Next k

back(1, 1) = p
back(2, 1) = q

GoSub mult33
        dtheta = m(1, 1)
        dv = m(2, 1)
        dlamda = m(3, 1)


If (det3 > 0) Then
GoTo critical
End If


If lamda < 0.0001 Then
'LOCATE 5, 5: Print "Lamda < 0.001 End": End
MsgBox ("Program Terminate")
End
End If
Rem ********************************************************************
```

```
Rem *********    Variable to check the display statement        *********
Rem ************************************************************************
If Abs(det3) < 0.01 Then
min1 = Int((Timer - stimes) / 60)
'LOCATE 2, 10:
Form1.CurrentX = 10: form1currenty = 2
Form1.Print "det="; det3; "Lamda ="; lamda; " Collapse times ="; min1; "Mins"; (Timer - stimes) - (mins * 60); "
Secs"
setx = -1: stp = 0.25
th(bus) = oldtheta
v(bus) = oldv
lamda = oldlamda
GoTo predictor
End If

GoSub cupdate

Rem  ========================================================================
Rem  ========      Checking point for critical point        ==========
Rem **********  Dicide if the result is correct enought?  xxxxxxxxx

If (Abs(dtheta) > est) Or (Abs(dlamda) > est) Or (Abs(dv) > est) Then
        it = it + 1
        GoTo corrector
Else
If weak = 1 Then
        lp = lp + 1
        GoSub result
        GoTo predictor
End If

        Return
End If
```

```
End

critical:
th(bus) = oldtheta
v(bus) = oldv
lamda = oldlamda
stp = stp * 0.89
GoTo predictor


Rem ----------------------------------------------------------------
Rem xxxxxxxxxxxxxxxx  Subroutine  xxxxxxxxxxxxxxxx
Rem  ========== Jacobian Matrix ==========
jacobian:
a(1, 1) = 0: a(1, 2) = 0: a(1, 3) = 0
a(2, 1) = 0: a(2, 2) = 0: a(2, 3) = 0
a(3, 1) = 0: a(3, 2) = 0: a(3, 3) = 0
For j = 1 To maxbus
If j = bus Then j = j + 1
a(1, 1) = a(1, 1) + (v(bus) * v(j) * y(bus, j) * Sin(((th(bus) - th(j) - yy(bus, j)) * (22 / 7)) / 180))
a(1, 2) = a(1, 2) - (v(j) * y(bus, j) * Cos(((th(bus) - th(j) - yy(bus, j)) * (22 / 7)) / 180))
a(1, 3) = (pg(bus) * kg) - (kl * sbase * Cos(pf(bus) * (22 / 7) / 180))
a(2, 1) = a(2, 1) - (v(bus) * v(j) * y(bus, j) * Cos(((th(bus) - th(j) - yy(bus, j)) * (22 / 7)) / 180))
a(2, 2) = a(2, 2) - (v(j) * y(bus, j) * Sin(((th(bus) - th(j) - yy(bus, j)) * (22 / 7)) / 180))
a(2, 3) = -(kl * sbase * Sin(pf(bus) * (22 / 7) / 180))
a(3, 1) = 0
a(3, 2) = 0
a(3, 3) = 1
If setx = -1 Then a(3, 3) = -1
Next j


a(1, 2) = a(1, 2) - (2 * v(bus) * y(bus, bus) * Cos((-yy(bus, bus) * (22 / 7) / 180)))
a(2, 2) = a(2, 2) - (2 * v(bus) * y(bus, bus) * Sin((-yy(bus, bus) * (22 / 7) / 180)))
```

```
Return

Rem ============== Inverse Matrix ==================
inverse:
GoSub adjoint
GoSub det3

For k = 1 To 3
For j = 1 To 3
inv(k, j) = (i(k, j) / det3)
Next j
Next k
Return

Rem ============= Find the Predicted Value ==============
update:
th(bus) = th(bus) + (stp * dtheta)
v(bus) = v(bus) + (stp * dv)
lamda = lamda + (stp * dlamda)
Return

cupdate:
th(bus) = th(bus) + (dtheta)
v(bus) = v(bus) + (dv)
lamda = lamda + (dlamda)
Return
```

```
Rem ============ Function Value ================

functio:
p = 0: q = 0: pgx = 0: qgx = 0
For j = 1 To maxbus
If j = bus Then j = j + 1
p = p - (v(bus) * v(j) * y(bus, j) * Cos(((th(bus) - th(j) - yy(bus, j)) * (22 / 7)) / 180))
q = q - (v(bus) * v(j) * y(bus, j) * Sin(((th(bus) - th(j) - yy(bus, j)) * (22 / 7)) / 180))
pgx = pgx - (v(bus) * v(j) * y(bus, j) * Cos(((th(bus) - th(j) - yy(bus, j)) * (22 / 7)) / 180))
qgx = qgx - (v(bus) * v(j) * y(bus, j) * Sin(((th(bus) - th(j) - yy(bus, j)) * (22 / 7)) / 180))
Next j
p = p + pg(bus) * (1 + (lamda * kg)) - pl(bus) - lamda * (kl * sbase * Cos(pf(bus) * (22 / 7) / 180)) - (v(bus) * v
(bus) * y(bus, bus) * Cos(-yy(bus, bus) * (22 / 7) / 180))
q = q + qc + qg(bus) - ql(bus) - lamda * (kl * sbase * Sin(pf(bus) * (22 / 7) / 180)) - (v(bus) * v(bus) * y(bus, bus)
* Sin(-yy(bus, bus) * (22 / 7) / 180))
pgx = pgx - (v(bus) * v(bus) * y(bus, bus) * Cos(-yy(bus, bus) * (22 / 7) / 180))
qgx = qgx - (v(bus) * v(bus) * y(bus, bus) * Sin(-yy(bus, bus) * (22 / 7) / 180))
Return

Rem ***************************
Rem Adjoint - Transpose matrix
Rem ***************************
adjoint:

Rem ************** Element # (1,1) ******************
store1 = a(2, 2): store2 = a(2, 3)
store3 = a(3, 2): store4 = a(3, 3)
GoSub det2
i(1, 1) = det2

Rem ************** Element # (1,2) ****************
store1 = a(2, 1): store2 = a(2, 3)
store3 = a(3, 1): store4 = a(3, 3)
```

```
GoSub det2
i(2, 1) = -det2

Rem ************** Element # (1,3) ****************
store1 = a(2, 1): store2 = a(2, 2)
store3 = a(3, 1): store4 = a(3, 2)
GoSub det2
i(3, 1) = det2

Rem ************** Element # (2,1) ****************
store1 = a(1, 2): store2 = a(1, 3)
store3 = a(3, 2): store4 = a(3, 3)
GoSub det2
i(1, 2) = -det2

Rem ************** Element # (2,2) ****************
store1 = a(1, 1): store2 = a(1, 3)
store3 = a(3, 1): store4 = a(3, 3)
GoSub det2
i(2, 2) = det2

Rem ************** Element # (2,3) ****************
store1 = a(1, 1): store2 = a(1, 2)
store3 = a(3, 1): store4 = a(3, 2)
GoSub det2
i(3, 2) = -det2

Rem ************** Element # (3,1) ****************
store1 = a(1, 2): store2 = a(1, 3)
store3 = a(2, 2): store4 = a(2, 3)
GoSub det2
i(1, 3) = det2
```

```
Rem ************** Element # (3,2) ****************
store1 = a(1, 1): store2 = a(1, 3)
store3 = a(2, 1): store4 = a(2, 3)
GoSub det2
i(2, 3) = -det2

Rem ************** Element # (3,3) ****************
store1 = a(1, 1): store2 = a(1, 2)
store3 = a(2, 1): store4 = a(2, 2)
GoSub det2
i(3, 3) = det2
Return


Rem ***********************************************
Rem Find the Determinant of (3x3) Jacobian Matrix
Rem         The result is var. DET3
Rem ***********************************************
det3:
det3 = -((a(3, 1) * a(2, 2) * a(1, 3)) + (a(3, 2) * a(2, 3) * a(1, 1)) + (a(3, 3) * a(2, 1) * a(1, 2))) + ((a(1, 1) * a(2,
2) * a(3, 3)) + (a(1, 2) * a(2, 3) * a(3, 1)) + (a(1, 3) * a(2, 1) * a(3, 2)))
Return


Rem *********************************
Rem  Determinat of 2x2 matrix
Rem Pls. keep data in form |s1    s2|
Rem                       |s3     s4|
Rem *********************************
det2:
det2 = -(store3 * store2) + (store1 * store4)
Return


Rem ***********************************************
```

```
Rem Subroutine for muktiply two 3x3 matrix [u]x[w]
Rem             m(i,j) is the result !!!
Rem ********************************************
mult33:

Rem ********** Element (1,1) **********
m(1, 1) = (front(1, 1) * back(1, 1)) + (front(1, 2) * back(2, 1)) + (front(1, 3) * back(3, 1))
m(1, 2) = (front(1, 1) * back(1, 2)) + (front(1, 2) * back(2, 2)) + (front(1, 3) * back(3, 2))
m(1, 3) = (front(1, 1) * back(1, 3)) + (front(1, 2) * back(2, 3)) + (front(1, 3) * back(3, 3))
m(2, 1) = (front(2, 1) * back(1, 1)) + (front(2, 2) * back(2, 1)) + (front(2, 3) * back(3, 1))
m(2, 2) = (front(2, 1) * back(1, 2)) + (front(2, 2) * back(2, 2)) + (front(2, 3) * back(3, 2))
m(2, 3) = (front(2, 1) * back(1, 3)) + (front(2, 2) * back(2, 3)) + (front(2, 3) * back(3, 3))
m(3, 1) = (front(3, 1) * back(1, 1)) + (front(3, 2) * back(2, 1)) + (front(3, 3) * back(3, 1))
m(3, 2) = (front(3, 1) * back(1, 2)) + (front(3, 2) * back(2, 2)) + (front(3, 3) * back(3, 2))
m(3, 3) = (front(3, 1) * back(1, 3)) + (front(3, 2) * back(2, 3)) + (front(3, 3) * back(3, 3))
Return
Return:

result:
If cps = 0 Then
'LOCATE 3, 12:
Form1.CurrentX = 12: Form1.CurrentY = 3
Form1.Print "Result from Uncompensated system at bus"; bus
End If

If Item = 1 Then
'LOCATE 3, 1:
Form1.CurrentX = 1: Form1.CurrentY = 3
Form1.Print "System compensated with Series C ="; xv; "PU. between line"; bb; " and "; b
End If

If Item = 2 Then
'LOCATE 3, 1:
```

```
Form1.CurrentX = 1: Form1.CurrentY = 3
Form1.Print "System compensated with Shunt C ="; xs; "MVAR at bus"; addbus
End If


If Item = 3 Then
'LOCATE 3, 1:
Form1.CurrentX = 1: Form1.CurrentY = 3
Form1.Print "System compensated wih LTC between line"; llb; "and"; lb
End If

If lamda < 0.5 Then End


mins = Int((Timer - stimes) / 60)
'LOCATE 4, 3:
Form1.CurrentX = 3: Form1.CurrentY = 4
Form1.Print "Voltage"
': LOCATE 4, 11:
Form1.CurrentX = 20: Form1.CurrentY = 4
Form1.Print "Angle"
': LOCATE 4, 18:
Form1.CurrentX = 31: Form1.CurrentY = 4
Form1.Print "Lamda"
': LOCATE 4, 26:
Form1.CurrentX = 37: Form1.CurrentY = 4
Form1.Print "Process in (it)"
': LOCATE 4, 33:
Form1.CurrentX = 47: Form1.CurrentY = 4
Form1.Print "Det"
': LOCATE 4, 46:
Form1.CurrentX = 58: Form1.CurrentY = 4
Form1.Print "Real Power"
': LOCATE 4, 60:
```

```
Form1.CurrentX = 70: Form1.CurrentY = 4
Form1.Print "Reactive Power"
'LOCATE (5 + lp), 1:
Form1.CurrentX = 1: Form1.CurrentY = (5 + lp)
Form1.Print v(bus)
': LOCATE (5 + lp), 9:
Form1.CurrentX = 14: Form1.CurrentY = (5 + lp)
Form1.Print th(bus)
': LOCATE (5 + lp), 17:
Form1.CurrentX = 26: Form1.CurrentY = (5 + lp)
Form1.Print lamda
': LOCATE (5 + lp), 27:
Form1.CurrentX = 39: Form1.CurrentY = (5 + lp)
Form1.Print it
': LOCATE (5 + lp), 34:
Form1.CurrentX = 45: Form1.CurrentY = (5 + lp)
Form1.Print det3
': LOCATE (5 + lp), 45:
Form1.CurrentX = 52: Form1.CurrentY = (5 + lp)
Form1.Print p
': LOCATE (5 + lp), 61:
Form1.CurrentX = 67: Form1.CurrentY = (5 + lp)
Form1.Print q
'LOCATE 1, 40: Print "Spent times = "; mins ; " Mins " ; (Timer - stimes) - (mins * 60) ; " Secs"
Label4.Caption = mins: Label5.Caption = (Timer - stimes) - (mins * 60)
If (5 + lp) = 22 Then
        'INPUT s$
        s$ = InputBox("s$")
        Cls
        lp = 0
End If
Return
```

```
Rem    ================================================================
Rem    ===  Compansated system with Series-C at Transmission Line  ==
Rem    ================================================================

series:
Cls
Print "Which line Series-C be added"
'PRINT "from bus _ to bus _": INPUT bb, b: CLS
bb = Val(InputBox("from bus _ "))
b = Val(InputBox("to bus _"))
Cls
Rem ***************     If select wrong line      ******************
If x(bb, b) = 0 Then
        'LOCATE 10, 12:
        Form1.CurrentX = 12: Form1.CurrentY = 10
        Form1.Print "No Transmission line connected between this two bus"
        For k = 1 To 100000: Next
Cls
GoTo series
End If


'INPUT "Please Input the Per Unit value of this Series-C"; xv
xv = Val(InputBox("Please Input the Per Unit value of this Series-C"))
Cls

Rem ============  PU value Base =100  ====================

x(bb, b) = x(bb, b) - xv
x(b, bb) = x(bb, b)
Return
```

```
Rem    ================================================================
Rem    ======  Compansated system with Shunt-C at Bus i       ======
Rem    ================================================================

shunt:
qc = (xs / basemva)
Rem ysc = -(basemva / (2 * 3.14 * freq * xs))
If addbus <> selt Then
ql(addbus) = ql(addbus) - qc
Rem yc(addbus) = yc(addbus) + ysc
qc = 0
Rem yc(selt) = yc(selt) + ysc
End If

Return

Rem    ================================================================
Rem    ===   Compansated system with LTC-C at Transmission Line   ===
Rem    ================================================================

ltc:
Print "Which line LTC be added"
'PRINT "from bus _ to bus _"; : INPUT llb, lb: CLS
llb = Val(InputBox("from bus _ "))
lb = Val(InputBox("to bus _ "))
Cls

ltc = 1
Rem ***************     If select wrong line      ********************
If x(llb, lb) = 0 Then
```

```
        'LOCATE 10, 12:
        Form1.CurrentX = 12: Form1.CurrentY = 10
        Form1.Print "No Transmission line connected between this two bus"
        For k = 1 To 100000: Next
Cls
GoTo ltc
End If

Rem **************  Input Transformer data  ***********************
'INPUT "Please input  Turn Ratio ( Nsec/Npri ) of this Transformer"; n
n = Val(InputBox("Please input  Turn Ratio ( Nsec/Npri ) of this Transformer"))
If tap(llb, lb) = 1 Then
tap(llb, lb) = n
tap(lb, llb) = tap(llb, lb)
Else
tap(llb, lb) = tap(llb, lb) + n
tap(lb, llb) = tap(llb, lb)
End If

yo(llb) = yo(llb) + yb(llb, lb) * (1 - (tap(llb, lb)) / (tap(llb, lb) ^ 2))
yyo(llb) = yyo(llb) + ybb(llb, lb) * (1 - (tap(llb, lb)) / (tap(llb, lb) ^ 2))
yo(lb) = yo(lb) + yb(lb, llb) * ((tap(llb, lb) - 1) / tap(llb, lb))
yyo(lb) = yyo(lb) + ybb(lb, llb) * ((tap(llb, lb) - 1) / tap(llb, lb))

Cls
Rem ============  PU value Base =100  ==================
Return
Return
End


End Sub
```

ภาคผนวก ข.

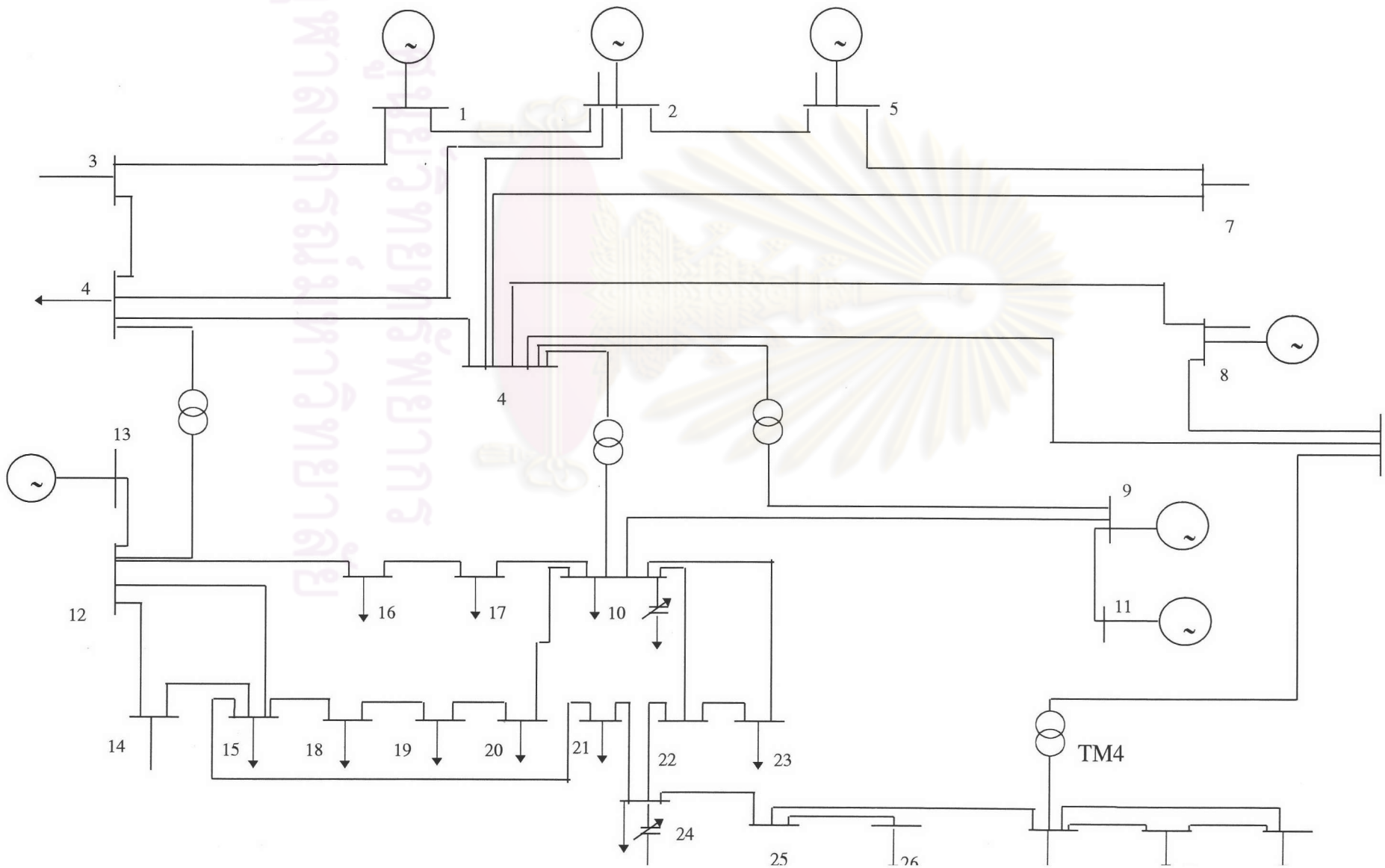
โปรแกรมที่ใช้ในการวิเคราะห์หาความกว้างของช่วงที่เหมาะสมในการคำนวณหาจุดวิกฤติ

```
REM xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
REM
REM       CPF- Voltage Collapse porgram with compensate element
REM
REM xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
DIM i(200), min(200), miss(200)
DEFSNG M, X
CLS
i = 0
begin:

FOR x = .01 TO .99 STEP .01
i = i + 1
uu = uu + 1
TIMER ON
stimes = TIMER
div = 0

FOR u = 1 TO 100
start = 8
var = RND / 2

continue:
start = start * x
IF (start - var) > .05 THEN
GOTO continue

ELSE
```

```
IF (start - var) < 0 THEN div = div + 1
min(i) = min(i) + (TIMER - stimes)
miss(i) = miss(i) + ABS(start - var)
END IF
NEXT u
IF div > 5 THEN
a$ = "Diverge"
ELSE
a$ = ""
END IF
LOCATE uu, 1
PRINT x
LOCATE uu, 15
PRINT min(i)
LOCATE uu, 30
PRINT miss(i) / 100
LOCATE uu, 50
PRINT a$; div
IF uu > 20 THEN
INPUT s$: CLS
uu = 0
END IF
NEXT x
END
```

ภาคผนวก ค.

ระบบทดสอบมาตรฐานของ IEEE ขนาด 30 บัส



ประวัติผู้เขียน

นาย พรประนด ดิษยบุตร์ เกิดเมื่อวันที่ 13 พฤศจิกายน 2514 ที่อำเภอ เมือง จังหวัด ชลบุรี สำเร็จการศึกษาปริญญาวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมไฟฟ้า จากจุฬาลงกรณ์มหาวิทยาลัย เมื่อปี 2536 แล้วได้ศึกษาต่อปริญญาโทในภาควิชาวิศวกรรมไฟฟ้า สาขาพลังงาน ที่จุฬาลงกรณ์มหาวิทยาลัย ภาคปีการศึกษา 2537