

# **"SCpower" – Power Network Short Circuit Program**

by Carlos Mauricio Beltran, BS Electrical Engineering 1991, Tulsa University, OK, USA.

This program is freeware; you may use it and share it freely. You may not modify, copy or use any portion of the program or in its entirety in any platform for the purpose of selling it. I reserve those rights; I am the sole intellectual proprietor of the source code. If you find the program useful please e-mail me at: [carlosmbm@msn.com](mailto:carlosmbm@msn.com) or at [carlosmbm@cre.com.bo](mailto:carlosmbm@cre.com.bo)

Just load the **SCpower** text file (ASCII, 9600 baud, parity = none 0, cksum 3, translate 1) and save it in your HP. The program's subroutines are ready to run, though in a certain order. Read the step by step example.

I do not guarantee the results of the program. You assume all risks of using this program or any of its subroutines. I recommend you install the six line screen viewer for the HP48 **STK6 / IStk by Raymond Hellstern** that you may download at <http://www.hpcalc.org>; my program does not use, nor does it requires that source code, but the results look nicer and "fit" better in a six line screen and smaller font.

The **SCpower** program calculates *three phase* and one *phase to ground (1LG)* short circuit currents at faulted buses as well as fault currents that circulate through the power network. It uses 100 MVA as base power. It does not yet calculate LL or 2LG faults, nor does it take into account fault resistance, although I may include those features in future versions.

Three winding transformers should be modeled using the classical "T" model. All impedances must be introduced in per unit. We will start with an example; see the one-line representations of the power network shown in pages 2, 3 and 4. At the end of this document you will find an explanation of the code of the program's subroutines.

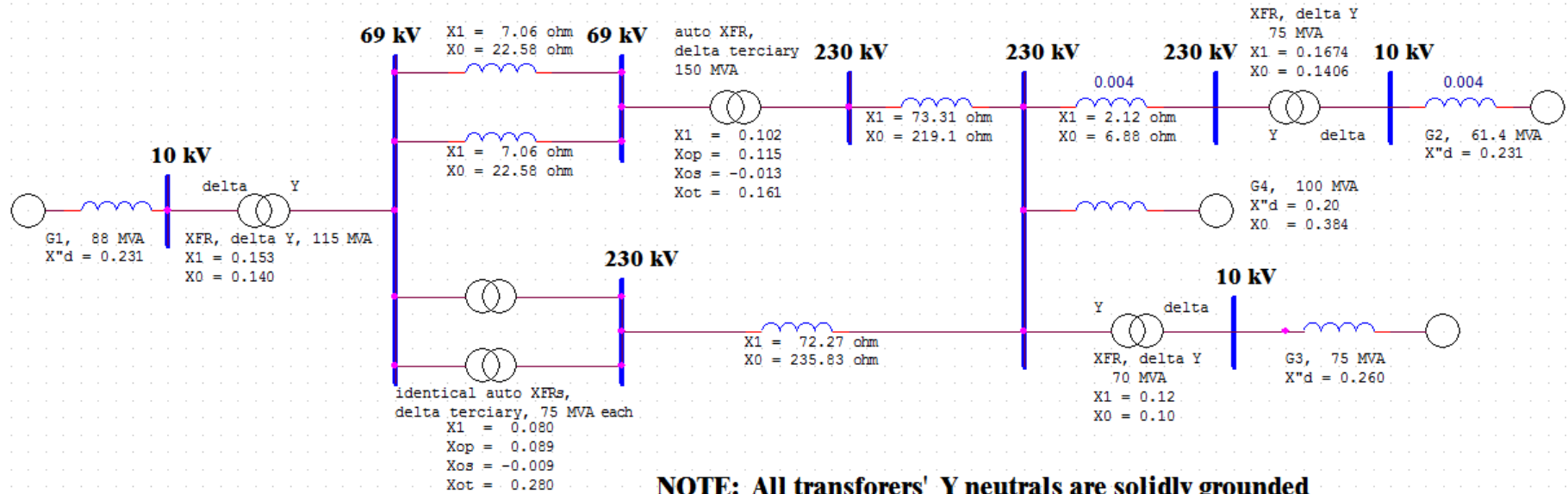
Have fun!

Sincerely,

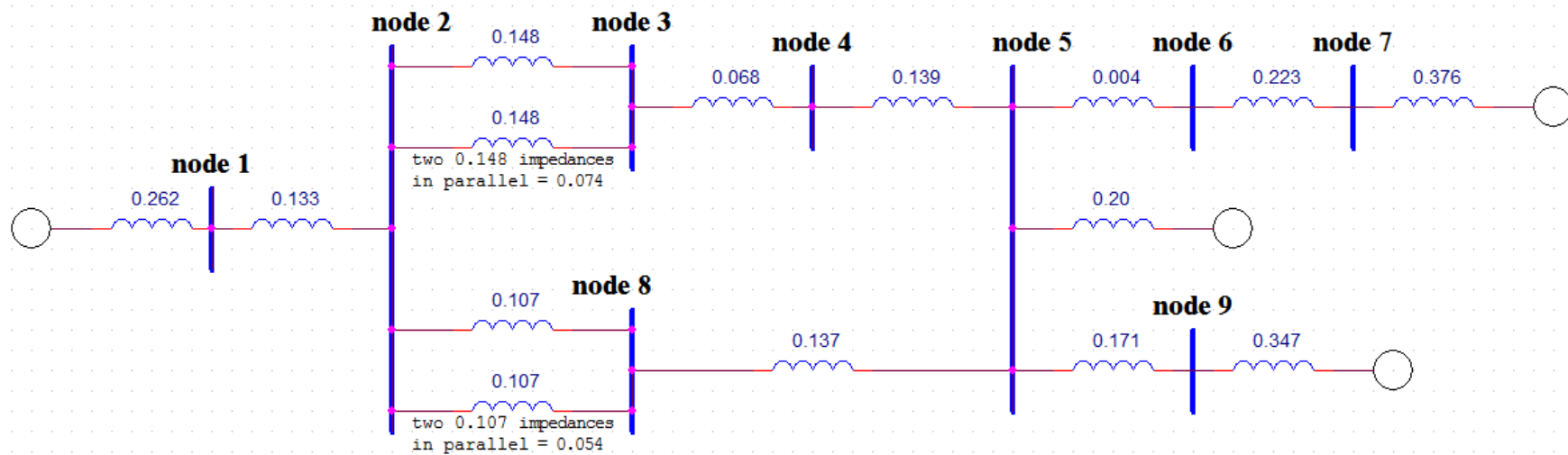
Carlos Mauricio Beltran  
Electrical Engineer  
All rights reserved  
September 2011

Example:

## Power System Description



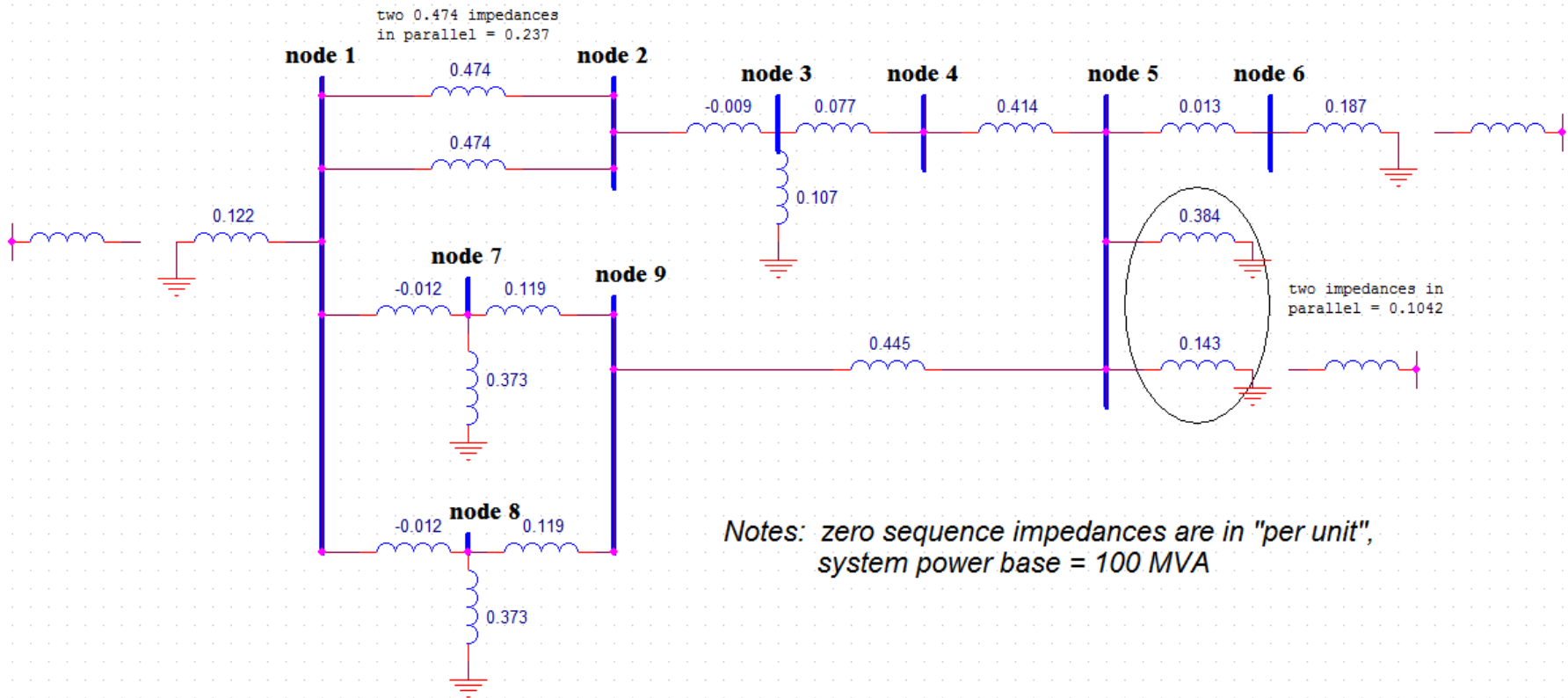
## Positive Sequence Circuit



Notes: impedances are in "per unit", system power base = 100 MVA

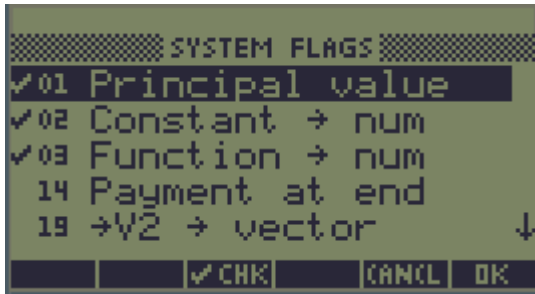
Negative sequence impedances are the same as the positive sequence impedances.

## Zero Sequence Circuit



## Executing the Program

1. Make sure the first three system flags in your calculator are as shown:



2. Activate Subroutine **MATX1**

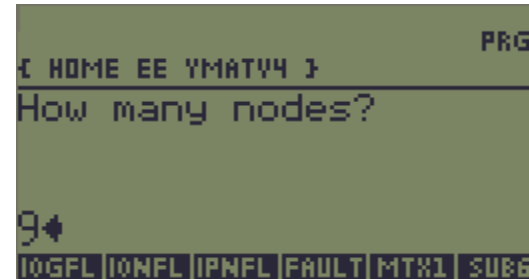
It is used to create the positive and zero sequence impedance matrices.



After the introductory screen, press the **left shift** key and the **ON** key to continue.

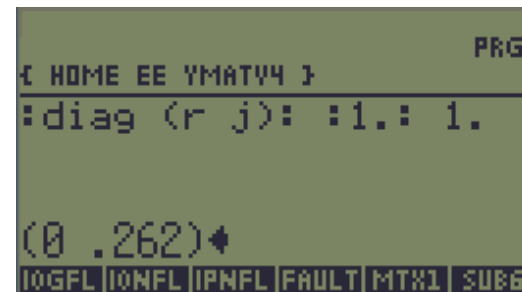
We will first create the positive sequence matrix, so refer to the one-line diagram.

3. The next screen asks



input **9** and **ENTER**.

4. The next screen prompts to enter the *diagonal* impedances. Diagonal impedances are those that are directly connected to a source. Otherwise, enter (0 0).



press the **0** key to enter the corresponding complex impedances each time. Enter the following for each of the nine diagonals.

(0 .262) **ENTER** key for diag (r j): 1.:1  
 (0 0) **ENTER** key for diag (r j): 2.:2  
 (0 0) **ENTER** key for diag (r j): 3.:3

```

(0 0)      ENTER key for diag (r j): 4.:4
(0 .2)     ENTER key for diag (r j): 5.:5
(0 0)      ENTER key for diag (r j): 6.:6
(0 .376)   ENTER key for diag (r j): 7.:7
(0 0)      ENTER key for diag (r j): 8.:8
(0 .347)   ENTER key for diag (r j): 9.:9

```

Instead of (0 0) you may enter just 0.

5. In the next screens enter the prompted impedances between nodes:



```

(0 .133)   ENTER key for :segment : :1.:2.
0          ENTER key for :segment : :1.:3.
0          ENTER key for :segment : :1.:4.
0          ENTER key for :segment : :1.:5.
0          ENTER key for :segment : :1.:6.
0          ENTER key for :segment : :1.:7.
0          ENTER key for :segment : :1.:8.
0          ENTER key for :segment : :1.:9.

(0 .074)   ENTER key for :segment : :2.:3.
0          ENTER key for :segment : :2.:4.
0          ENTER key for :segment : :2.:5.
0          ENTER key for :segment : :2.:6.
0          ENTER key for :segment : :2.:7.
(0 .054)   ENTER key for :segment : :2.:8.
0          ENTER key for :segment : :2.:9.

(0 .068)   ENTER key for :segment : :3.:4.
0          ENTER key for :segment : :3.:5.

```

```

0          ENTER key for :segment : :3.:6.
0          ENTER key for :segment : :3.:7.
0          ENTER key for :segment : :3.:8.
0          ENTER key for :segment : :3.:9.

(0 .139)   ENTER key for :segment : :4.:5.
0          ENTER key for :segment : :4.:6.
0          ENTER key for :segment : :4.:7.
0          ENTER key for :segment : :4.:8.
0          ENTER key for :segment : :4.:9.

(0 .004)   ENTER key for :segment : :5.:6.
0          ENTER key for :segment : :5.:7.
(0 .137)   ENTER key for :segment : :5.:8.
(0 .171)   ENTER key for :segment : :5.:9.

(0 .223)   ENTER key for :segment : :6.:7.
0          ENTER key for :segment : :6.:8.
0          ENTER key for :segment : :6.:9.

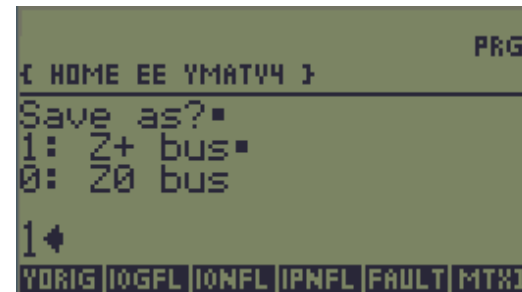
0          ENTER key for :segment : :7.:8.
0          ENTER key for :segment : :7.:9.

0          ENTER key for :segment : :8.:9.

```

Now wait a few seconds...

6. The next screen prompts you to save the data as either the positive or the zero sequence bus impedance.



press **0** **ENTER**

7. Run Subroutine **MATX10** again, this time to create the zero sequence bus impedance matrix.

Refer to the zero sequence one-line diagram. It also has nine numbered nodes. The diagonals of the zero sequence circuit are those impedances that are directly connected to ground.

When prompted to enter the diagonals, enter the following:

```
(0 .122)  ENTER key for diag (r j): 1.:1
0         ENTER key for diag (r j): 2.:2
(0 .107)  ENTER key for diag (r j): 3.:3
0         ENTER key for diag (r j): 4.:4
(0 .1042) ENTER key for diag (r j): 5.:5
(0 .187)  ENTER key for diag (r j): 6.:6
(0 .373)  ENTER key for diag (r j): 7.:7
(0 .373)  ENTER key for diag (r j): 8.:8
0         ENTER key for diag (r j): 9.:9
```

Instead of (0 0) you may enter just 0.

8. In the next screens enter the prompted impedances:

```
(0 .237)  ENTER key for :segment : :1.:2.
0         ENTER key for :segment : :1.:3.
0         ENTER key for :segment : :1.:4.
0         ENTER key for :segment : :1.:5.
0         ENTER key for :segment : :1.:6.
(0 -.012) ENTER key for :segment : :1.:7.
(0 -.012) ENTER key for :segment : :1.:8.
0         ENTER key for :segment : :1.:9.

(0 -.009) ENTER key for :segment : :2.:3.
0         ENTER key for :segment : :2.:4.
0         ENTER key for :segment : :2.:5.
0         ENTER key for :segment : :2.:6.
0         ENTER key for :segment : :2.:7.
```

```
0         ENTER key for :segment : :2.:8.
0         ENTER key for :segment : :2.:9.

(0 .077)  ENTER key for :segment : :3.:4.
0         ENTER key for :segment : :3.:5.
0         ENTER key for :segment : :3.:6.
0         ENTER key for :segment : :3.:7.
0         ENTER key for :segment : :3.:8.
0         ENTER key for :segment : :3.:9.

(0 .414)  ENTER key for :segment : :4.:5.
0         ENTER key for :segment : :4.:6.
0         ENTER key for :segment : :4.:7.
0         ENTER key for :segment : :4.:8.
0         ENTER key for :segment : :4.:9.

(0 .013)  ENTER key for :segment : :5.:6.
0         ENTER key for :segment : :5.:7.
0         ENTER key for :segment : :5.:8.
(0 .445)  ENTER key for :segment : :5.:9.

0         ENTER key for :segment : :6.:7.
0         ENTER key for :segment : :6.:8.
0         ENTER key for :segment : :6.:9.

0         ENTER key for :segment : :7.:8.
(0 .119)  ENTER key for :segment : :7.:9.

(0 .119)  ENTER key for :segment : :8.:9.
```

9. The next screen prompts you to save the data as either as the positive or zero sequence bus impedance.

Save as?  
1: Z+ bus  
2: Z0 bus

press **0** **ENTER**

Now you have created both positive and zero sequence bus impedance and admittance matrices, which are stored in the following variables:

Y0 Z0 Y1 Z1.

10. Activate Subroutine **FAULTS** to calculate a three phase and a 1LG faults at any given node.

Note that the node under fault in the positive and zero sequence circuits have been numbered as defined in the hand drawn one-line diagrams. The node under fault has a certain number associated with it in the positive sequence circuit, and may have another in the zero sequence circuit. **KEEP THIS IN MIND.**

Now let's calculate a three phase fault in node 3 (of the positive sequence circuit); The program will also calculate a 1LG fault in node this node which corresponds to node 2 in the zero sequence circuit.

11. At the next screen enter 3.

```
      R42                PRG
{ HOME EE YMATV4 }
What is + seq faulted=
bus #?

3
YORIG IOGFL IOINFL IPNFL FAULT MTX1
```

12. At the next screen enter 69.

```
      R42                PRG
{ HOME EE YMATV4 }
Whats the prefault kV=
at the faulted bus?

69
YORIG IOGFL IOINFL IPNFL FAULT MTX1
```

*(the prefault voltage of this node is 69 kV phase to phase)*

13. At the next screen enter 2.

```
      R42                PRG
{ HOME EE YMATV4 }
What # is the 0 seq=
faulted bus? Dble chk!

2
YORIG IOGFL IOINFL IPNFL FAULT MTX1
```

14. The results are shown in six levels.

```
074082 R42                <1>
{ HOME EE YMATV4 }
6: MVAsc3Ø: 593.339
5: Isc3Ø: 4964.704
4: puZt: 0.169, 290.0...
3: MVAsc1Ø: 761.686
2: Isc1Ø: 6373.326
1: puZØt: 0.057, 290.0...
YORIG IOGFL IOINFL IPNFL FAULT MTX1
```

This indicates that for a three phase fault at node 3 (whose prefault voltage is 69 kV):

- \* the 3 phase short circuit power is 593.339 MVA.
- \* the 3 phase short circuit current is 4964.704 A.
- \* the positive seq. Thev. pu impd. is 0.169 @90° Ω.

And if a 1LG fault did occur at this node:

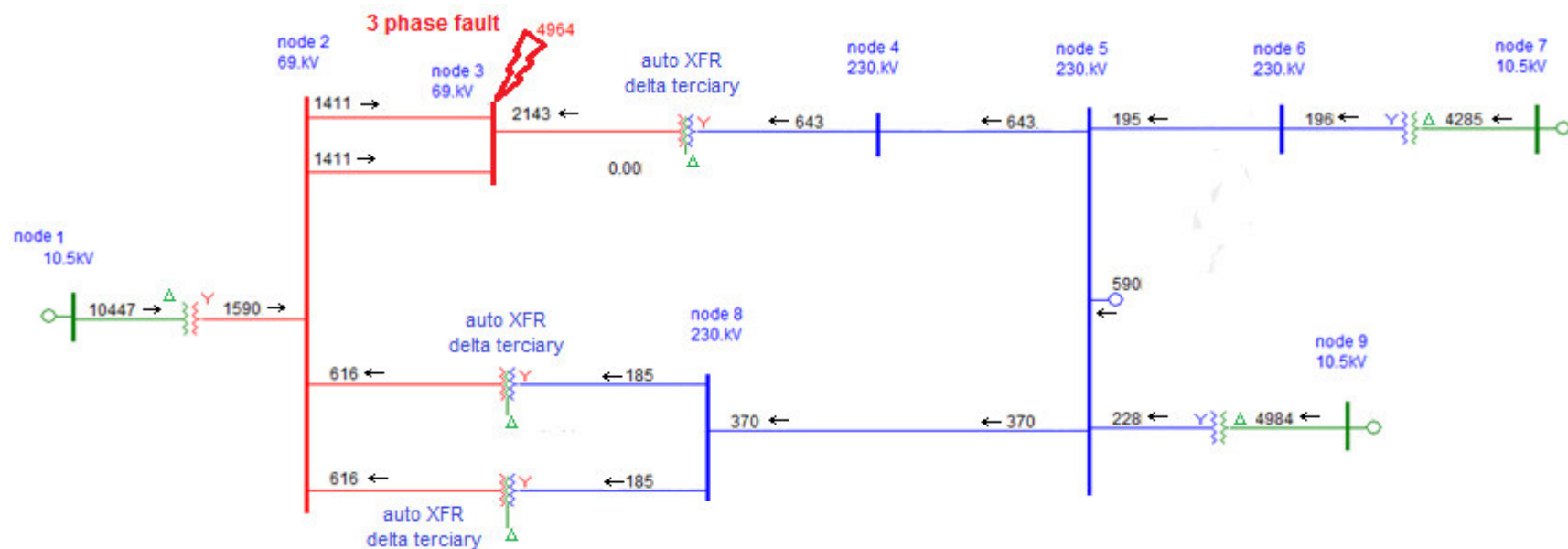
- \* the short circuit power would be 761.686 MVA.

- \* the short circuit current would be 761.686 A.

- \* the zero seq. Thevenin pu imped would be 0.057 @90° Ω.

15. The IPNFL subroutine calculates the positive sequence fault that flows between any 2 adjacent nodes for a three phase fault. Likewise, the IONFL calculates the 3I0 current that flows between any 2 adjacent nodes.

The power network has been modeled in a short circuit program and the results for the 3 phase fault and 1LG fault are as shown in the next page. My HP 48 subroutines come up with the same short circuit values.



16. Activate Subroutine **IPNFL** to calculate the positive sequence current that flows between any two adjacent nodes; between nodes 2 and 3 for example:

```

PRG
[ HOME EE YMATV4 ]
Enter FROM & TO nodes

2 3
YORIG IOGFL IONFL IPNFL FAULT MTX1

```

```

PRG
[ HOME EE YMATV4 ]
What is the voltage=
kV of this segment?

69
YORIG IOGFL IONFL IPNFL FAULT MTX1

```

The result is:

```

1: Iflow: (2821., 490.)
YORIG IOGFL IONFL IPNFL FAULT MTX1

```

This is twice 1411 A. (and in accordance with our positive sequence impedance circuit (0.074 pu) since both lines between nodes 2 and 3 are in parallel.

Current flow from node 4 to node 3, coming from the 230 kV side:

```

PRG
[ HOME EE YMATV4 ]
Enter FROM & TO nodes

4 3
YORIG IOGFL IONFL IPNFL FAULT MTX1

```

Current flow from node 4 to node 3  
(coming from 230 kV side)

```

PRG
[ HOME EE YMATV4 ]
What is the voltage=
kV of this segment?

230
YORIG IOGFL IONFL IPNFL FAULT MTX1

1: Iflow: (643., 490.)
YORIG IOGFL IONFL IPNFL FAULT MTX1

```

Current flow from node 4 to node 3  
arriving at node 3, (69 kV side).

```
R42 PRG
[ HOME EE YMATV4 ]
Enter FROM & TO nodes

4 3
YORIG IOGFL IOHFL IPNFL FAULT MTX1
```

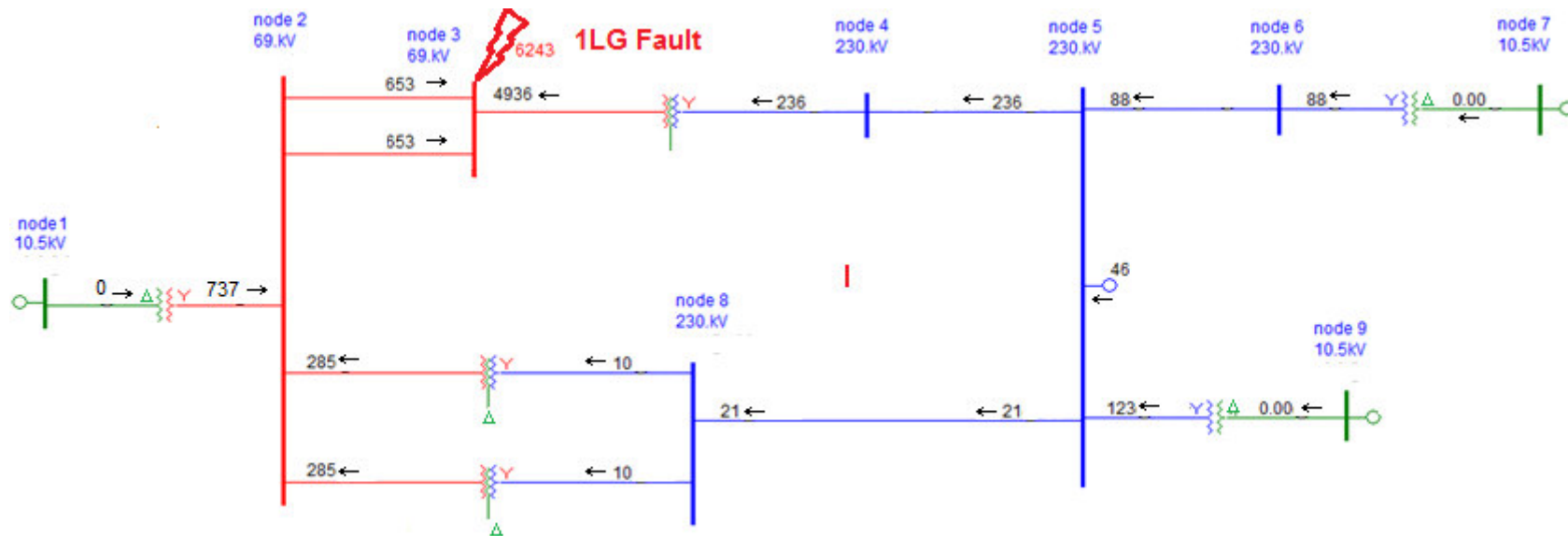
```
R42 PRG
[ HOME EE YMATV4 ]
What is the voltage
kV of this segment?

69
YORIG IOGFL IOHFL IPNFL FAULT MTX1
```

The result is:

```
1: Iflow: (2144., 290.)
YORIG IOGFL IOHFL IPNFL FAULT MTX1
```

As you can see, the results of my program are nearly identical to those calculated by the graphical software.



17. Activate Subroutine **IONFL** to calculate the 3I0 (zero sequence) current that flows between any two adjacent nodes; between nodes 2 and 3 for example, which correspond to nodes 1 and 2 in the hand drawn zero sequence circuit:

```

R42                                PRG
[ HOME EE YMATV4 ]
Enter FROM and TO nod...
of Iflow of 0 seq ckt

1 2
YDRIG I0GFL I0NFL IPNFL FAULT MTX1

```

```

R42                                PRG
[ HOME EE YMATV4 ]
Whats the kV of this
segment?

69
YDRIG I0GFL I0NFL IPNFL FAULT MTX1

```

The result is:

```

2: pu 3I0+: (2., 290.)
1: 3I0+: (1334., 290.)
I0GFL I0NFL IPNFL FAULT MTX1 SUB6

```

Which is 667 A in each line, and close to the result of the graphical software (653 A). All other 3I0 current flows values are also very near the graphical software results.

18. Say you now want to know the 3I0 current that flows through the neutral of the Y side of the transformer between nodes 1 and 2. Run the **IOGFL** subroutine.

```

      R42                                PRG
[ HOME EE YMATV4 ]
gnd I→ into wch node?

1♦
YORIG IOGFL IONFL IPNFL FAULT MTX1

```

```

      R42                                PRG
[ HOME EE YMATV4 ]
gnd impd pu?

(0 .122)♦
YORIG IOGFL IONFL IPNFL FAULT MTX1

```

```

      R42                                PRG
[ HOME EE YMATV4 ]
Whats the kV @ node?

69♦
YORIG IOGFL IONFL IPNFL FAULT MTX1

```

The result is:

```

2: pu3I0→: (1.,2-90.)
1: 3I0→: (753.,2-90.)
IOGFL IONFL IPNFL FAULT MTX1 SUB6

```

19. To find the current through the grounded neutral of transformer between nodes 9 and 5 you must do yourself a simple current division of the program's result because in our HP zero sequence model we defined the grounded impedance at node 5 as a parallel of  $j0.384$  and  $j0.143$ .

## SOURCE CODE EXPLANATION

Read the program comments from the bottom going up (that is, start with subroutine SUB0, and lastly read subroutine IOGFL).

```
%%HP: T(3)A(D)F(.);
DIR
IOGFL @ calculates current flowing from "grounded" impedances @
  \<< V0matrx
    "gnd I\-> into wch node?"
    { "" { 3 3 } } INPUT OBJ\-> GET
    "\191gnd imped pu?" {"" { 3 3 } } INPUT
    OBJ\-> / 3 * 'pu\179I0\->'
    \->TAG DUP 100000 *
    "Whats the kV @ node?"
    { "" { 3 3 } } INPUT OBJ\-> 3 \v/ * /
    '\179I0\->' \->TAG
  \>>
IONFL @calcultes 3Io current flow from any 2 nodes of zero seq. circuit due to fault at FB0 node@
  \<< Z0
    { N0 1 } (0,0) CON @creates N0 x 1 matrix filled with zeros@
    Z0FB Z1FB Z1FB + + INV @ I0 pu current@
    FB0 OBJ\-> DROP DROP SWAP PUT

    @puts I0 as sole source current in N0 X 1 matrix at the position of the faulted node@
    * @ I0 * Z0 = V0@
    'V0matrx' STO @contains the zero seq. fault voltages due the fault on the FB0 node@
    "Enter FROM and TO nodes of Iflow of 0 seq ckt"
    { "" { 3 3 } } INPUT OBJ\-> 'to0' STO 'from0' STO
    V0matrx from0 GET
    V0matrx to0 GET -
    Y0 from0 to0 2 \->LIST GET
    * 3 * NEG
    '\179I0\->' \->TAG DUP
    100000 *
    "Whats the kV of this segment?"
    { "" { 3 3 } } INPUT OBJ\-> / 3 \v/ /
```

```

    '\179Io\>' \>TAG @special character that looks like 3 to form "3I0"@
\>>
IPNFL @calculates 3 phase fault current flow between any 2 nodes@
\<< 0 FIX Z1
    "Enter FROM & TO nodes"
    { "" { 3 3 } } INPUT OBJ\> 'to1' STO 'from1' STO

    "What is the voltage kV of this line?"
    { "" { 3 3 } } INPUT OBJ\> 'KVL' STO
    FB1 OBJ\> DROP DROP from1 SWAP 2 \>LIST
    GET Z1FB / NEG 1 + 'VFRM1' STO

    Z1 FB1 OBJ\> DROP DROP to1 SWAP 2 \>LIST
    GET Z1FB / NEG 1 + 'VTO1' STO

    Y1 from1 to1 2 \>LIST GET
    VFRM1 VTO1 - * 100000 * 3 \v/ KVL * /

    'Iflow' \>TAG
\>>
FAULTS @once Y1 and Y0 have been created, this subroutine @
    @ calls SUB5 & SUB6 to calculate and display 3phase & 1LG fault currents at the faulted node@
    @as well as the thevenin + & 0 pu impedances at faulted node@
\<< SUB5 SUB6 CONT \>>

MTX10 @run this subroutine first to create Y1 matrix & then again to create Y0 matrix@
\<< READ SUB0 SUB2 SUB4
    'Y' RCL INV
    "Save as?
    1: Z+ bus
    0: Z0 bus"
    { "" { 3 3 } } INPUT OBJ\>
    \> Q
    \<< IF 'Q==1' THEN 'Z1' STO
        ELSE 'Z0' STO N 'N0' STO
    END
    YORIG
    IF 'Q==1' THEN 'Y1' STO
        ELSE 'Y0' STO
    END { YORIG Z } PURGE
\>>

```

```

\>>
SUB6 @calculates 1LG fault current at faulted bus@
\<< Z0
    "What # is the 0 seq faulted bus? Dble chk!"
    { "" { 3 3 } } INPUT OBJ\-> DUP 2
    \->LIST DUP 'FB0' STO
    GET DUP 'Z0FB' STO
    'puZ0t' \->TAG DUP
    Z1FB 2 * + INV 100 * 3 * ABS @1LG fault MVA = [3 / (2*Z1 + Z0)] * Sbase@
    'MVAsc1\O/' \->TAG
    DUP KVPFB 3 \v/ * / 1000 *
    'Isc1\O/' \->TAG 3 ROLL CONT    \>>

SUB5 @calculates 3phase fault current at faulted bus@
\<< 3 FIX Z1
    "What is + seq faulted bus #?"
    { "" { 3 3 } } INPUT OBJ\-> DUP
    2 \->LIST DUP 'FB1' STO GET
    DUP 'Z1FB' STO 'puZt' \->TAG DUP INV
    100 * ABS 'MVAsc3\O/' \->TAG DUP

    "Whats the prefault kV at the faulted bus?"
    { "" { 3 3 } } INPUT OBJ\-> DUP
    'KVPFB' STO 3 \v/ * / 1000 * 'Isc3\O/' \->TAG
    3 ROLL

\>>
READ
\<< "Program calculates
    Bus Z+ or Z0 &
    fault I, V. It uses
    100 MVA as Sbase"
    CLLCD 1 DISP 3
    FREEZE HALT \>>

SUB1
\<< 'A' STO 'A' RCL
    IF 'A\=/0'
    THEN INV NEG
    END

\>>
SUB4
\<< N SQ 'C' STO

```

```

N 1 + 'D' STO
N 1 FOR X
  SUB3
    'Y' RCL SWAP 'C'
    RCL SWAP PUT 'Y'
    STO 'C' RCL 'D' -
    'C' STO -1
  STEP
\>>
SUB3
\<< L2A
\<< + \>> STREAM NEG
\>>

SUB2
\<< Y \->ROW DROP \>>

L2A
\<< IF DUP TYPE 5 ==
  THEN LIST\-> \->ARRAY
  ELSE ARRAY\-> 1 GET \->LIST
  END
\>>
SUB0 @subroutine to enter nodes & impedances to create admittance matrix@
\<< "How many nodes?"
"" INPUT OBJ\-> 'N'
STO 'N' RCL IDN @creates an identity matrix N x N@
(1,0) * @makes identity matrix ready for complex numbers@
1 N FOR E
  E E \->TAG "diag (r j)" \->TAG \->STR "" @displays the node diagonal location @
  INPUT OBJ\-> SUB1
  E 1 - N * E + SWAP PUT
NEXT
'DIAG' STO
'DIAG' RCL
1 N 1 - FOR J
  1 J + 'M' STO
  M N FOR K
    K J \->TAG "Node" \->TAG
    \->STR "" INPUT OBJ\->
  SUB1

```

```

      IF 'J==1' THEN
        K SWAP PUT
      ELSE
        IF 'M==N' THEN
          N SQ N - SWAP PUT
          ELSE J 1 - N * K + SWAP PUT
        END
      END
    END
  END

```

NEXT

NEXT

```

DIAG - DUP TRN NEG + DIAG + DUP 'YORIG' STO 'Y' STO
\>>

```

```

VTO1
(.249457276935,\<)0)
VFRM1
(.328887827596,\<)0)
KVL 230
KVPFB 69
from1 8
to1 2
from0 1
to0 2

```

V0matrx @ voltages at nodes of zero seq. circuit. V0=I0 \* Z0; I0 matrix is created in IONFL subroutine@

```

[[ (3.41103640786E-2,0) ]
[ (.159444378468,0) ]
[ (.177127917746,0) ]
[ (.153301299779,0) ]
[ (2.51945486349E-2,0) ]
[ (2.35569029737E-2,0) ]
[ (.035605818212,0) ]
[ (.035605818212,0) ]
[ (3.21353950197E-2,0) ]]

```

Z0FB @ thevenin zero seq. impedance pu at faulted node @

```

(0,6.39395644892E-2)

```

FB1 { 3 3 } @ positive seq. faulted bus @

FB0 { 2 2 } @ zero seq. faulted bus, which must match the + seq. ckt in the diagrams @

KVPFB 69 @ real pre-fault voltage at the faulted bus; use to calculate real 3phase & 1LG fault currents @

Z1FB @ thevenin zero seq. impedance pu at faulted node @

```

(0,.168537645811)

```

Y0

```

[[ (0,8.19672131148) (0,4.2194092827) (0,0) (0,0) (0,0) (0,0) (0,-83.3333333333) (0,-83.3333333333) (0,0) ]
[ (0,4.2194092827) (0,0) (0,-111.111111111) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) ]
[ (0,0) (0,-111.111111111) (0,9.34579439252) (0,12.987012987) (0,0) (0,0) (0,0) (0,0) (0,0) ]
[ (0,0) (0,0) (0,12.987012987) (0,0) (0,2.4154589372) (0,0) (0,0) (0,0) (0,0) ]
[ (0,0) (0,0) (0,0) (0,2.4154589372) (0,9.59692898273) (0,76.9230769231) (0,0) (0,0) (0,8.40336134454) ]

```

```
[ (0,0) (0,0) (0,0) (0,0) (0,76.9230769231) (0,5.34759358289) (0,0) (0,0) (0,0) ]
[ (0,-83.3333333333) (0,0) (0,0) (0,0) (0,0) (0,2.68096514745) (0,0) (0,8.40336134454) ]
[ (0,-83.3333333333) (0,0) (0,0) (0,0) (0,0) (0,0) (0,2.68096514745) (0,8.40336134454) ]
[ (0,0) (0,0) (0,0) (0,0) (0,8.40336134454) (0,0) (0,8.40336134454) (0,8.40336134454) (0,0) ]]

x0
[[ (0,4.72670861329E-2) (0,1.36787627429E-2) (0,1.49542686944E-2) (0,1.48030897447E-2) (0,.01399025747) (0,1.30808907344E-2) (0,5.00921380384E-2) (0,5.00921380384E-2)
(0,3.80581778489E-2) ]
[ (0,1.36787627429E-2) (0,6.39395644892E-2) (0,7.10309264481E-2) (0,6.14760986726E-2) (0,1.01033882956E-2) (0,9.44666805642E-3) (0,.014278462067) (0,.014278462067)
(0,1.28867708099E-2) ]
[ (0,1.49542686944E-2) (0,7.10309264481E-2) (0,6.89014331155E-2) (0,5.97037059284E-2) (0,1.02509909225E-2) (0,9.58467651252E-3) (0,1.56384750785E-2) (0,1.56384750785E-2)
(0,1.38426470265E-2) ]
[ (0,1.48030897447E-2) (0,6.14760986726E-2) (0,5.97037059284E-2) (0,.117831859646) (0,1.63650497668E-2) (0,1.53013215319E-2) (0,1.52566641003E-2) (0,1.52566641003E-2)
(0,1.56261259892E-2) ]
[ (0,.01399025747) (0,1.01033882956E-2) (0,1.02509909225E-2) (0,1.63650497668E-2) (0,.049238041475) (0,4.60375687791E-2) (0,1.32038102694E-2) (0,1.32038102694E-2)
(0,2.52152206713E-2) ]
[ (0,1.30808907344E-2) (0,9.44666805642E-3) (0,9.58467651252E-3) (0,1.53013215319E-2) (0,4.60375687791E-2) (0,5.52001268085E-2) (0,1.23455626018E-2) (0,1.23455626018E-2)
(0,2.35762313276E-2) ]
[ (0,5.00921380384E-2) (0,.014278462067) (0,1.56384750785E-2) (0,1.52566641003E-2) (0,1.32038102694E-2) (0,1.23455626018E-2) (0,3.98014045696E-2) (0,5.36424252801E-2)
(0,.035549213373) ]
[ (0,5.00921380384E-2) (0,.014278462067) (0,1.56384750785E-2) (0,1.52566641003E-2) (0,1.32038102694E-2) (0,1.23455626018E-2) (0,5.36424252801E-2) (0,3.98014045696E-2)
(0,.035549213373) ]
[ (0,3.80581778489E-2) (0,1.28867708099E-2) (0,1.38426470265E-2) (0,1.56261259892E-2) (0,2.52152206713E-2) (0,2.35762313276E-2) (0,.035549213373) (0,.035549213373)
(0,7.17712158058E-2) ]]

y1
[[ (0,3.81679389313) (0,7.51879699248) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) ]
[ (0,7.51879699248) (0,0) (0,13.5135135135) (0,0) (0,0) (0,0) (0,0) (0,18.5185185185) (0,0) ]
[ (0,0) (0,13.5135135135) (0,0) (0,14.7058823529) (0,0) (0,0) (0,0) (0,0) (0,0) ]
[ (0,0) (0,0) (0,14.7058823529) (0,0) (0,7.19424460432) (0,0) (0,0) (0,0) (0,0) ]
[ (0,0) (0,0) (0,0) (0,7.19424460432) (0,5) (0,250) (0,0) (0,7.29927007299) (0,5.84795321637) ]
[ (0,0) (0,0) (0,0) (0,0) (0,250) (0,0) (0,4.48430493274) (0,0) (0,0) ]
[ (0,0) (0,0) (0,0) (0,0) (0,0) (0,4.48430493274) (0,2.65957446809) (0,0) (0,0) ]
[ (0,0) (0,18.5185185185) (0,0) (0,0) (0,7.29927007299) (0,0) (0,0) (0,0) (0,0) ]
[ (0,0) (0,0) (0,0) (0,0) (0,5.84795321637) (0,0) (0,0) (0,0) (0,2.8818443804) ]]

z1
[[ (0,.152194092191) (0,9.64529252503E-2) (0,8.39028160757E-2) (0,7.23702833207E-2) (0,4.87964296008E-2) (0,4.84727385255E-2) (0,3.04269610778E-2) (0,8.29793610352E-2)
(0,3.26879557365E-2) ]
[ (0,9.64529252503E-2) (0,.145415669747) (0,.126494703626) (0,.109107869892) (0,7.35671362302E-2) (0,7.30791286929E-2) (0,.045872708495) (0,.12510247179)
(0,4.92814599843E-2) ]
[ (0,8.39028160757E-2) (0,.126494703626) (0,.168537645811) (0,.139171700792) (0,7.91442543566E-2) (0,7.86192510109E-2) (0,.049350314491) (0,.113107665612)
(0,5.30174831308E-2) ]
[ (0,7.23702833207E-2) (0,.109107869892) (0,.139171700792) (0,.166797923782) (0,8.42691737161E-2) (0,.083710174222) (0,5.25459524332E-2) (0,.102085411287)
(0,5.64505854816E-2) ]
[ (0,4.87964296008E-2) (0,7.35671362302E-2) (0,7.91442543566E-2) (0,8.42691737161E-2) (0,9.47451118182E-2) (0,9.41166201974E-2) (0,5.90782123443E-2) (0,7.95546267105E-2)
(0,6.34682505809E-2) ]
[ (0,4.84727385255E-2) (0,7.30791286929E-2) (0,7.86192510109E-2) (0,.083710174222) (0,9.41166201974E-2) (0,9.74657636786E-2) (0,6.11805127598E-2) (0,7.90269011601E-2)
(0,6.30472339933E-2) ]
[ (0,3.04269610778E-2) (0,.045872708495) (0,.049350314491) (0,5.25459524332E-2) (0,5.90782123443E-2) (0,6.11805127598E-2) (0,.178383760931) (0,4.96062017299E-2)
(0,3.95755592345E-2) ]
[ (0,8.29793610352E-2) (0,.12510247179) (0,.113107665612) (0,.102085411287) (0,7.95546267105E-2) (0,7.90269011601E-2) (0,4.96062017299E-2) (0,.150958054856)
(0,5.32923850744E-2) ]
[ (0,3.26879557365E-2) (0,4.92814599843E-2) (0,5.30174831308E-2) (0,5.64505854816E-2) (0,6.34682505809E-2) (0,6.30472339933E-2) (0,3.95755592345E-2) (0,5.32923850744E-2)
(0,.157066569405) ]]

x0
[[ (0,154.250536072) (0,4.2194092827) (0,0) (0,0) (0,0) (0,0) (0,-83.3333333333) (0,-83.3333333333) (0,0) ]
[ (0,4.2194092827) (0,106.891701828) (0,-111.111111111) (0,0) (0,0) (0,0) (0,0) (0,0) ]
[ (0,0) (0,-111.111111111) (0,88.778303731) (0,12.987012987) (0,0) (0,0) (0,0) (0,0) ]
[ (0,0) (0,0) (0,12.987012987) (0,-15.4024719242) (0,2.4154589372) (0,0) (0,0) (0,0) ]
[ (0,0) (0,0) (0,0) (0,2.4154589372) (0,-97.3388261875) (0,76.9230769231) (0,0) (0,8.40336134454) ]
[ (0,0) (0,0) (0,0) (0,0) (0,76.9230769231) (0,-82.270670506) (0,0) (0,0) ]
[ (0,-83.3333333333) (0,0) (0,0) (0,0) (0,0) (0,0) (0,72.2490068413) (0,0,8.40336134454) ]
[ (0,-83.3333333333) (0,0) (0,0) (0,0) (0,0) (0,0) (0,72.2490068413) (0,8.40336134454) ]]
```

```

[ (0,0) (0,0) (0,0) (0,0) (0,8.40336134454) (0,0) (0,8.40336134454) (0,8.40336134454) (0,-25.2100840336) ] ]
D 10
C -9
M 9
A (0,.119)
N 9
NO 9
DIAG
[[ (0,8.19672131148) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) ]
[ (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) ]
[ (0,0) (0,0) (0,9.34579439252) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) ]
[ (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) ]
[ (0,0) (0,0) (0,0) (0,0) (0,9.59692898273) (0,0) (0,0) (0,0) (0,0) ]
[ (0,0) (0,0) (0,0) (0,0) (0,0) (0,5.34759358289) (0,0) (0,0) (0,0) ]
[ (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,2.68096514745) (0,0) (0,0) ]
[ (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,2.68096514745) (0,0) ]
[ (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) ] ]
END

```