To the Graduate Council:

I am submitting herewith a thesis written by Dennis A. Reed entitled "An Interactive Shunt Fault Analysis Package for Small Power Systems." I have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Electrical Engineering.

Jack S. Lawler, Major Professor

We have read this thesis
and recommend its acceptance:

Accepted for the Council:

Vice Chancellor
Graduate Studies and Research

AN INTERACTIVE SHUNT FAULT ANALYSIS PACKAGE FOR SMALL POWER SYSTEMS

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Dennis A. Reed

December 1982

This thesis is dedicated

to my mother

Mildred L. Reed

ACKNOWLEDGEMENTS

The author wishes to acknowledge with deepest appreciation his thesis advisor, Dr. Jack S. Lawler, for his advice and unlimited patience. Appreciation is also extended to committee members Dr. J. M. Bailey and Dr. F. W. Symonds. The author wants to express thanks to Dr. F. W. Symonds for giving assistance in research and for his constructive criticism.

The author wishes to express his appreciation to Dr. T. W. Reddoch for his assistance. Finally, it gives the author special pleasure to express is gratitude to his fiancee, Wanda Mae Grace, for her assistance in the preparation of this thesis.

ABSTRACT

An interactive shunt fault analysis package is developed for small power systems. The fault solution procedure implemented is the classical technique based on the symmetrical component transformation and Thevenin's theorem.

The interactive features of the code are designed to make efficient use of the user's time. A base line system design can be loaded into the program by reading an external data file. The user can then interactively make minor system changes, alter fault type, impedance or location for a given fault and execute a fault study.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Short circuits (or "faults") pose a serious hazard to the components of an electrical power system. If faulted conditions are allowed to persist the currents flowing in affected generators and lines may be of such large magnitude as to result in thermal damage. In addition, the voltage profile may be seriously altered throughout the system such that damage is done to sensitive loads connected at points remote from the location of the fault. It is the responsibility of power system protection engineers to design a protection system that quickly isolates faulted portions of the system while maintaining service to the greatest possible degree.

Faults may occur for numerous reasons such as insulation damage due to lightning, trees falling against conductors, conductor icing, salt sprays and corrosion, small animals short circuiting switch gear, vehicle collisions with transmission line towers or conductors, etc. In a three phase system short circuits are usually categorized into four basic types; single line-to-ground, line-to-line, double line-to-ground or three phase. The time constants involved are such that the effects of faults can be analyzed using steady state AC techniques, that is, phasor analysis can be applied. However, the first three fault types result in unbalanced system operation and cannot be studied using the classical per phase method appropriate for balanced systems.

A primary tool of the protection engineer is a computer code which solves for the system voltages and currents in each phase under faulted conditions. Ideally this code should be interactive so as to optimize the use of engineering time and effort. Such a code will allow large data base representing the nominal system configuration to be stored and recalled on demand. The engineer can then use this code to rapidly assess the effects of any shunt fault by simply specifying the fault type and location. The code could also be used to compare several design alternatives under "what if" situations with respect to the system model. For example, the engineer may want to know if the protection system is sensitive enough to detect and isolate a particular fault "if" some system components are out for scheduled maintenance. The interactive code would allow the engineer to simply delete the appropriate components from the nominal system description and the perform the desired fault study.

An interactive computer code for solving short circuit problems in small power systems is developed in this thesis. The fault solution procedure implemented is based on the symmetrical component transformation and Thevenin's theorem. The next section provides an overview of the analysis of faults.

1.1 Overview of Shunt Fault Analysis

For purposes of discussion an n-bus, three phase power system is postulated having phase sequence a,b,c. The prefault system is assumed to be in a symmetric steady state operating condition. Prefault bus voltages and component currents can be calculated by solving the classical load flow equations on a per phase basis and then invoking three phase symmetry.

A fault is an abnormal condition occuring at one paricular system bus which is designated as the fault point. This thesis is restricted to the analysis of a single shunt fault. A shunt fault involves fault impedance between phase conductors and/or between phase conductors and ground at the fault point. Figure 1.1 shows the details of a general shunt fault occuring at an arbitrary bus k. If the fault impedances $\overline{Z}_a$, $\overline{Z}_b$, $\overline{Z}_c$, in Figure 1.1 are identical then the shunt fault is a symmetrical shunt fault and the faulted power system will remain in a balanced three phase state (but different than the prefault state). If these fault impedances are not equal, then the fault is an unsymmetrical shunt fault and unbalanced voltages and currents will develop throughout the network. The objective of shunt fault analysis is to determine the system bus voltages and component currents in each phase under faulted conditions.

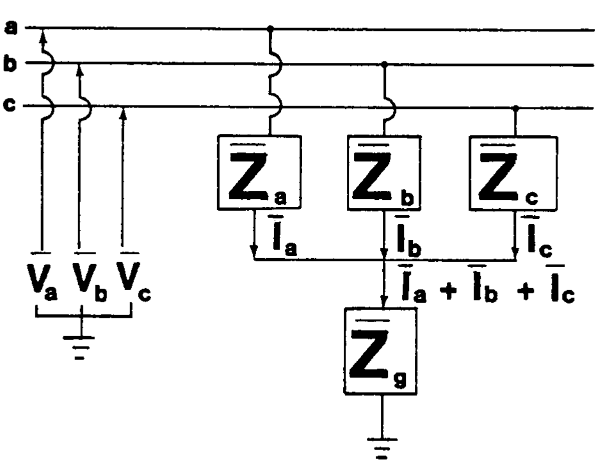


Figure 1.1. Generalized shunt fault diagram

It should be emphasized that if unbalanced voltages and currents occur under faulted conditions they are due solely to the application of an unsymmetrical shunt fault. Looking into the power system from the fault point (bus k in Figure 1.1) one would see an "otherwise balanced" three phase system that would return to prefault conditions if the shunt fault were removed from the system.

The computer code developed in this thesis solves shunt fault problems by applying symmetrical component theory and using Thevenin's theorem. Symmetrical components are used to resolve the "otherwise balanced" three phase power system into three decoupled per phase sequence networks, namely, the positive, negative and zero sequence networks. Voltages and currents in the sequence networks are the symmetrical components of the phase voltages and currents in the power system. The negative and zero sequence networks are strictly passive, and under balanced conditions all voltages and currents in these networks are zero (there are no sources of negative or zero sequence currents in a balanced system). Active sources are present only in the positive sequence network which is the same as the per phase representation of the balanced prefault system.

Under balanced conditions there are no circuit paths between the sequence networks (they are decoupled). A faulted condition causes paths to appear between the networks in the form of simple connections through external impedances. These external connections allow the sources in the positive sequence network to drive currents through all three networks. The voltages and currents developed in the sequence networks under faulted conditions are the symmetrical components of the desired faulted values of the power system phase voltages and component currents.

When the external impedance connections are made between the sequence networks, the sequence voltages and currents change from prefault values to faulted values. The positive, negative and zero sequence network impedance matrices are used in conjunction with Thevenin's theorem for computing the changes in sequence variables due to the fault. Faulted values of the sequence quantities are then determined by superposition of prefault values with the computed changes. The desired faulted values of system phase voltages and currents are found by application of the symmetrical component transformation to the faulted sequence network quantities.

1.2 Thesis Objectives and Outline

The objective of this thesis is the development of an efficient, interactive computer code for analyzing shunt faults in small power systems. The shunt fault analysis package has been integrated with a load flow program which was also developed by the author (Reddoch 1980).

The overview of shunt fault analysis above is expanded in Chapter 2 which is a detailed discussion of the procedure for solving shunt faults. Two algorithms which enhance computational efficiency are presented in Chapter 3. The first algorithm is a triangular factorization technique which computes only those portions of the sequence network bus impedance matrices that are actually needed to solve a given fault problem. A second algorithm is presented for keeping track of phase shift introduced by wye-delta transformers. Interactive features of the code are discussed in Chapter 4. Chapter 5 is a summary and offers suggestions for possible improvements in the computer package.

CHAPTER 2

PRINCIPLES OF SHUNT FAULT ANALYSIS

This chapter outlines the basic principles of shunt fault analysis that are incorporated in the computer code developed in this thesis. The discussion below is a comprehensive review of the well known shunt fault analysis technique based on Fortescue's symmetrical component transformation and Thevenin's theorem. Matrix and vector notation is used so that the equations developed for shunt fault analysis can be easily formulated and solved using a digital computer.

2.1 Notation

The notation used to distinguish phasor, matrix and vector quantites can be a source of confusion in power system analysis problems. This section describes the notational preferences of the author and should make it easier for the reader to follow subsequent developments.

Quantities which can be expressed as a single point in a complex plane, such as a voltage (or current) phasor or complex admittance (or impedance) will be denoted by overscored upper case letters. As an example the instantaneous current

$$i(t) = \sqrt{2}\, I \sin(\omega t + \theta) \tag{2-1a}$$

$$= \sqrt{2}\, I \cos\theta \sin\omega t + \sqrt{2}\, I \sin\theta \cos\omega t \tag{2-1b}$$

will be denoted in the complex plane as the phasor $\bar{I}$. Either polar or rectangular coordinates may be used to specify the point in the complex plane represented by $\bar{I}$. If polar coordinates are used then

$$\bar{I} = I\underline{/\theta} \tag{2-2a}$$

where

$$I = \text{rms magnitude of } i(t) \tag{2-2b}$$

$$\theta = \text{phase angle of } i(t) \text{ with respect to } \sin\omega t \tag{2-2c}$$

When expressed in terms of rectangular coordinates the phasor $\bar{I}$ is represented as

$$\bar{I} = \text{Re}(\bar{I}) + j\ \text{Im}(\bar{I}) \tag{2-3a}$$

where by Euler identity

$$\text{Re}(\bar{I}) = I\cos\theta \tag{2-3b}$$

$$= \text{rms magnitude of the component of } i(t) \text{ in phase with } \sin\omega t$$

$$\text{Im}(\bar{I}) = I\sin\theta \tag{2-3c}$$

$$= \text{rms magnitude of the component of } i(t) \text{ in phase with } \cos\omega t$$

Collections of points in the complex plane will be represented as complex matrices (or vectors) and denoted by an underscore. For example, let

$$\underline{Y} = n \times n \text{ admittance matrix (complex)} \tag{2-4a}$$

where the ij th entry is the complex admittance $\bar{Y}_{ij}$ (a single point in the complex plane), i.e.

$$\{\underline{Y}\}_{ij} = ij \text{ th entry in the complex matrix } \underline{Y} \tag{2-4b}$$

$$= \bar{Y}_{ij} \tag{2-4c}$$

As a further example let the voltage profile in the power system be represented by a bus voltage vector in each phase, $\underline{V}_a$, $\underline{V}_b$, $\underline{V}_c$ defined as

$\underline{V}_p$ = vector of bus voltage phasors in phase p (an n x 1 vector), p = a,b,c (2-5a)

with,

$\{\underline{V}_p\}_i$ = i th entry in the vector $\underline{V}_p$ (i = 1,2,...,n) (2-5b)

= voltage phasor in phase p at bus i

$= \overline{V}_{p-i}$ (2-5c)

Superscripts are used to distinguish prefault and faulted conditions. Prefault conditions are denoted with the superscript "o". For example $\underline{V}_a^o$, $\underline{V}_b^o$, $\underline{V}_c^o$ represent the prefault voltage profile (the load flow solution). The double superscript "fk" is used to denote faulted conditions. The second superscript "k" indicates the fault location (bus k). For example $\underline{V}_a^{f3}$, $\underline{V}_b^{f3}$, $\underline{V}_c^{f3}$ is the system voltage profile under faulted conditions with the fault occuring at bus 3.

It is convenient to define the complex constant $\overline{a}$ as

$$\overline{a} = \text{the "a-operator"} \tag{2-6}$$

$$= 1\underline{/120}$$

$$= -0.5 + j0.866$$

The a-operator has the property that when it multiplies (or operates on) a phasor, it leaves the phasor's magnitude unchanged but advances the phase positively (counterclockwise) by 120 degrees. Note also that $\overline{a}^0$, $\overline{a}^1$ and $\overline{a}^2$ are a symmetrical three phase set:

$$1 + \overline{a} + \overline{a}^2 = 1\underline{/0} + 1\underline{/120} + 1\underline{/240} = 0 \tag{2-7}$$

Finally, the transpose of a matrix (or vector) is denoted by the superscript "T". For example:

$\underline{V}^T$ = transpose of vector $\underline{V}$ (1 x n row vector)

$\underline{Y}^T$ = transpose of matrix $\underline{Y}$

2.2 Symmetrical Components

Symmetrical components and per phase analysis are important concepts in the solution of unbalanced shunt fault problems. Per phase analysis techniques reduce the complexity of analyzing a balanced three phase system to the level of complexity involved in analyzing a single phase system. The symmetrical component transformation is a transformation of variables that allows per phase techniques to be extended to the analysis of unbalanced systems. These important concepts are reviewed below.

2.2.1 Per phase analysis.

Many types of power system components exhibit phase-to-phase coupling. For example Faraday's law indicates that the current flowing in one conductor of a three phase line induces voltages in all three conductors. However, under balanced operating conditions the phase voltage and current relationships of all system components can be developed in such a way that the phases appear to be decoupled. That is to say, the equations of a balanced system are written such that phase p (p = a,b,c) currents cause only phase p voltage drops and vice versa. The apparent phase decoupling under balanced conditions is precisely what makes per phase analysis valid for analyzing symmetrical power systems.

Consider the hypothetical component shown in Figure 2.1 which is connected between buses i and k of some power system.

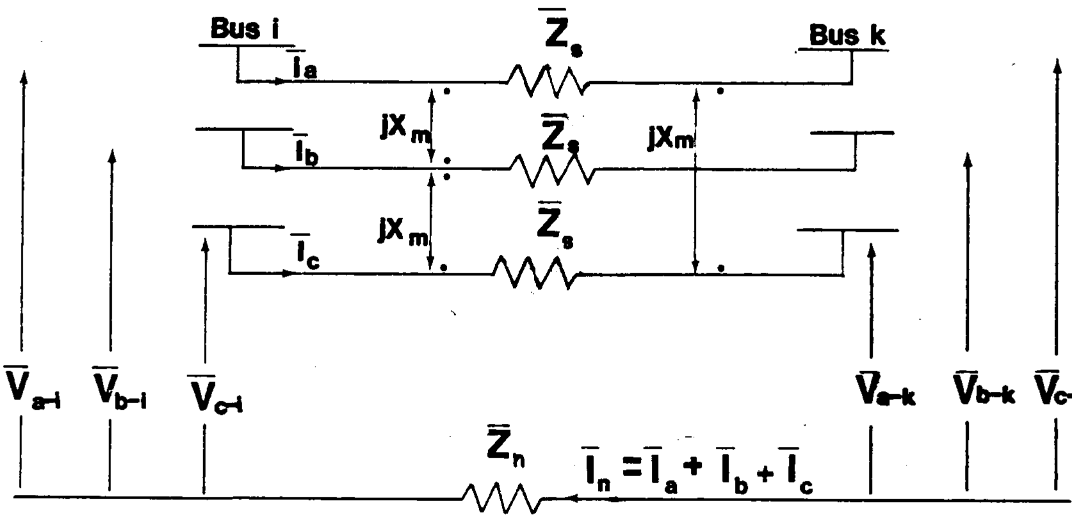


Figure 2.1. Hypothetical component with mutual coupling between phases

The relationship between phase voltages and currents for this component can be written as

$$\begin{vmatrix} \overline{V}_{a-i} \\ \overline{V}_{b-i} \\ \overline{V}_{c-i} \end{vmatrix} \quad \begin{vmatrix} \overline{Z}_s+\overline{Z}_n & jX_m+\overline{Z}_n & jX_m+\overline{Z}_n \\ jX_m+\overline{Z}_n & \overline{Z}_s+\overline{Z}_n & jX_m+\overline{Z}_n \\ jX_m+\overline{Z}_n & jX_m+\overline{Z}_n & \overline{Z}_s+\overline{Z}_n \end{vmatrix} \begin{vmatrix} \overline{I}_a \\ \overline{I}_b \\ \overline{I}_c \end{vmatrix} \quad \begin{vmatrix} \overline{V}_{a-k} \\ \overline{V}_{b-k} \\ \overline{V}_{c-k} \end{vmatrix} \tag{2-8a}$$

or

$$\underline{V}_{abc-i} = \underline{Z}_{abc}\, \underline{I}_{abc} + \underline{V}_{abc-k} \tag{2-8b}$$

Off diagonal terms in $\underline{Z}_{abc}$ represent phase-to-phase coupling and are due to the presence of mutual inductances (represented in Figure 2.1 as the X_m reactances) and the effects on the neutral return impedance $\overline{Z}_n$. Extracting the expression for $\overline{V}_{a-i}$ from (2-8a)

$$\overline{V}_{a-i} = (\overline{Z}_s+\overline{Z}_n)\overline{I}_a + (jX_m+\overline{Z}_n)\overline{I}_b + (jX_m+\overline{Z}_n)\overline{I}_c + \overline{V}_{a-k} \tag{2-9a}$$

$$\overline{V}_{a-i} = (\overline{Z}_s+\overline{Z}_n)\overline{I}_a + (jX_m+\overline{Z}_n)(\overline{I}_b+\overline{I}_c) + \overline{V}_{a-k} \qquad (2\text{-}9b)$$

makes it clear that the phase a voltage drops depend on the currents in all three phases. Equation (2-8) holds under any conditions (balanced or unbalanced).

Under symmetrical operating conditions the neutral return current ($\overline{I}_n$) is zero, i.e.

$$\overline{I}_n = 0 = \overline{I}_a + \overline{I}_b + \overline{I}_c \qquad (2\text{-}10)$$

Equation (2-10) can be used to derive a decoupled form of (2-8). Solving (2-10) for the sum of $\overline{I}_b$ and $\overline{I}_c$

$$\overline{I}_b + \overline{I}_c = -\overline{I}_a \qquad (2\text{-}11)$$

and substituting into (2-9b) the phase a expression for the hypothetical component becomes:

$$\overline{V}_{a-i} = (\overline{Z}_s+\overline{Z}_n)\overline{I}_a + (jX_m+\overline{Z}_n)(-\overline{I}_a) + \overline{V}_{a-k} \qquad (2\text{-}12a)$$

$$= (\overline{Z}_s+jX_m)\overline{I}_a + \overline{V}_{a-k} \qquad (2\text{-}12b)$$

Equation (2-12b) is a per phase equation and shows that under balanced conditions the phase a voltage drops can be written so that they depend only on the phase a current. Expressions similar to (2-12b) could obviously be derived for phases b anc c such that under balanced conditions (2-8) becomes

$$\begin{vmatrix} \overline{V}^o_{a-i} \\ \overline{V}^o_{b-i} \\ \overline{V}^o_{c-i} \end{vmatrix} = \begin{vmatrix} \overline{Z}_s-jX_m & 0 & 0 \\ 0 & \overline{Z}_s-jX_m & 0 \\ 0 & 0 & \overline{Z}_s-jX_m \end{vmatrix} \begin{vmatrix} \overline{I}^o_a \\ \overline{I}^o_b \\ \overline{I}^o_c \end{vmatrix} + \begin{vmatrix} \overline{V}^o_{a-k} \\ \overline{V}^o_{b-k} \\ \overline{V}^o_{c-k} \end{vmatrix} \qquad (2\text{-}13a)$$

or

$$\underline{V}^o_{abc-i} = \underline{Z}^o_{abc}\, \underline{I}^o_{abc} + \underline{V}^o_{abc-k} \qquad (2\text{-}13b)$$

where the superscript "o" denotes symmetrical operation. Equation (2-13) is the desired phase decoupled form of (2-8). The fact that $\underline{Z}^{o}_{abc}$ is diagonal is what makes per phase analysis possible.

It is not necessary to solve the third order decoupled equation (2-13) to analyze a symmetrical system. Since the diagonal entries of $\underline{Z}^{o}_{abc}$ are identical and if the phase voltages are balanced three phase sets it automatically follows that the phase currents will be a balanced three phase set. The per phase equation (2-12b) can be solved for the phase a quantities and then the phase sequence can be used to infer the phase b and c variables. With phase sequence a,b,c the phase a solution is delayed 120 degrees to obtain phase b and then delayed an additional 120 degrees to obtain the phase c solution. Figure 2.2 is the per phase equivalent circuit of the hypothetical component for symmetrical operating conditions.

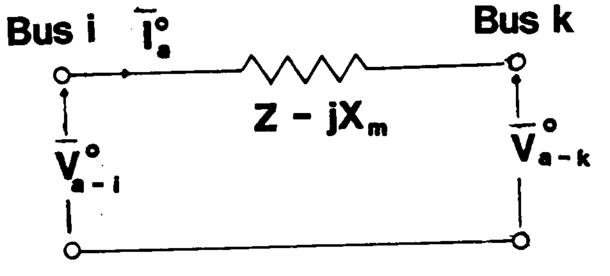


Figure 2.2. Per phase equivalent circuit of the hypothetical component under symmetrical operating conditions

Under symmetrical conditions the per phase solution (2-12b) also satisfies the general equation (2-8). However under unbalanced conditions (2-12b) no longer applies and yet the solution to equations such as (2-8) must still be obtained if shunt fault problems are to be solved. The reader will readily appreciate that the difficulty in solving n coupled linear equations increases at least as n squared (actually as n cubed minus n divided by 3) (Burden, 1978). Thus it appears that the three phase equation (2-8) under unbalanced conditions is roughly nine times more difficult to solve than a per phase analysis. Symmetrical components is a technique for solving unbalanced fault problems that reduces the complexity of the analysis to about three times the difficulty of per phase analysis.

2.2.2 The symmetrical component transformation. Shunt fault analysis can be greatly simplified using a reversable transformation of variables called the symmetrical component transformation (SCT). The technique was first proposed in 1918 be C. L. Fortescue. Fortescue showed that a single n-phase system (balanced or otherwise) can be resolved by the SCT into n n-phase systems called the sequence systems. Each sequence system has a different phase sequence. The procedure for solving shunt faults is to transform the original problem into symmetrical components for solution purposes and then reverse the transformation after the solution is found.

It might seem that the SCT would add rather than reduce the complexity since the original system is transformed to n n-phase systems. However symmetry in the sequence systems allows per phase analysis to be used in each of those systems. Recall that the difficulty in solving the original unbalanced n-phase problem is of the order of n squared where as the solution difficulty of the transformed problem is only of order n due to the use of per phase techniques. Of course some small amount of additional work is involved in transforming the problem and using the reverse transform on the solution. The discussion of the SCT in this section is restricted to the three phase case.

Let $\overline{I}_a$, $\overline{I}_b$, and $\overline{I}_c$ be the phase currents flowing at some point in a three phase power system. According to Fortescue's theorem it is possible to resolve this original system of three phase phasors (which can be balanced or unbalanced) into three sequence systems which are three phase sets having different phase sequences. Mathematically, $\overline{I}_a$, $\overline{I}_b$ and $\overline{I}_c$ are expressed in terms of their symmetrical components as

$$\begin{aligned} \overline{I}_a &= \overline{I}_{a0} + \overline{I}_{a1} + \overline{I}_{a2} \\ \overline{I}_b &= \overline{I}_{b0} + \overline{I}_{b1} + \overline{I}_{b2} \\ \overline{I}_c &= \overline{I}_{c0} + \overline{I}_{c1} + \overline{I}_{c2} \end{aligned} \tag{2-14}$$

The symmetrical components $\overline{I}_{a0}$, $\overline{I}_{b0}$ and $\overline{I}_{c0}$ are called the zero sequence set. These components have identical magnitudes and a phase angle of $0 \times (-2 \times \pi/3) = 0$ radians between successive components a0, b0, c0; i.e. $\overline{I}_{a0}$, $\overline{I}_{b0}$ and $\overline{I}_{c0}$ are in phase. The zero sequence system can properly be viewed as three identical single phase systems and is referred to as zero sequence since phase sequence is meaningless in single phase systems.

The components $\overline{I}_{a1}$, $\overline{I}_{b1}$ and $\overline{I}_{c1}$ are called the positive sequence set. These components have an identical magitude and a phase angle of 1 x (-2 x π/3) radians (-120 degrees) between succesive components a1, b1, c1. The name positive sequence is used to denote the fact that this set of components has the same clockwise phase sequence (abc) as the original three phase network.

Components $\overline{I}_{a2}$, $\overline{I}_{b2}$, $\overline{I}_{c2}$ are called the negative sequence set. These components have identical magnitude and a phase angle of 2 x (-2 x π/3) radians (-240 degrees) between successive components a2, b2, c2 resulting in a phase sequence acb. The term negative sequence is used to denote the fact that the traditional clockwise phase sequence of these components (acb, or equivalently cba) is the same as the counterclockwise phase sequence of the original system phasors.

The left hand side of equation (2-14) has six degrees of freedom since three magnitudes and three angles must be specified to determine the original phase currents $\overline{I}_a$, $\overline{I}_b$ and $\overline{I}_c$. If the SCT is to be an invertable transformation then the right hand side of (2-14) must also have exactly six degrees of freedom. At first glance it may seem that there are 18 degrees of freedom associated with the symmetrical components (9 magniudes and 9 angles). However each sequence system has been defined to have a particular form of three phase symmetry such that per phase knowledge within each sequence is sufficient to uniquely specify all phases of the symmetrical components. More precisely, each of the three sequence systems contribues two degrees of freedom to the right hand side of (2-14) and knowledge of only the phase a symmetrical components ($\overline{I}_{a0}$, $\overline{I}_{a1}$, $\overline{I}_{a2}$) is sufficient to determine all the symmetrical components and the original phase currents.

To demonstrate that per phase analysis within each sequence system is sufficient, assume that the phase a symmetrical components $\overline{I}_{a0}$, $\overline{I}_{a1}$, and $\overline{I}_{a2}$ are known. Then in terms of the phase a components the phase b and c components are by definition

$$\begin{aligned} \overline{I}_{b0} &= \overline{I}_{a0} \\ \overline{I}_{c0} &= \overline{I}_{a0} \end{aligned} \tag{2-15a}$$

$$\begin{aligned} \overline{I}_{b1} &= \overline{a}^2\overline{I}_{a1} \\ \overline{I}_{c1} &= \overline{a}\overline{I}_{a1} \end{aligned} \tag{2-15b}$$

$$\begin{aligned} \overline{I}_{b2} &= \overline{a}\overline{I}_{a2} \\ \overline{I}_{c2} &= \overline{a}^2\overline{I}_{a2} \end{aligned} \tag{2-15c}$$

Substituting (2-15) into (2-14) allows the original phase currents to be expressed solely in terms of their phase a symmetrical components. The resulting equation is called the indirect symmetrical component transformation (ISCT) and can be written as

$$\begin{vmatrix} \overline{I}_a \\ \overline{I}_b \\ \overline{I}_c \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 \\ 1 & \overline{a}^2 & \overline{a} \\ 1 & \overline{a} & \overline{a}^2 \end{vmatrix} \begin{vmatrix} \overline{I}_{a0} \\ \overline{I}_{a1} \\ \overline{I}_{a2} \end{vmatrix} \tag{2-16a}$$

or

$$\underline{I}_{abc} = \underline{A}\underline{I}_{012} \tag{2-16b}$$

Notice that there are six degrees of freedom on both sides of equation (2-16). The ISCT is sometimes called the synthesis equation since it tells how to synthesize the unbalanced phasors from the phase a symmetrical components. An expression analogous to (2-16) would hold if the original unbalanced phasors were phase voltages rather than phase currents.

The transformation matrix $\underline{A}$ is nonsingular and the ISCT (2-16) can be inverted to obtain the direct symmetrical component transformation (DSCT):

$$\begin{vmatrix} \overline{I}_{a0} \\ \overline{I}_{a1} \\ \overline{I}_{a2} \end{vmatrix} = \frac{1}{3} \begin{vmatrix} 1 & 1 & 1 \\ 1 & \overline{a} & \overline{a}^2 \\ 1 & \overline{a}^2 & \overline{a} \end{vmatrix} \begin{vmatrix} \overline{I}_a \\ \overline{I}_b \\ \overline{I}_c \end{vmatrix} \qquad (2\text{-}17a)$$

or

$$\underline{I}_{012} = \underline{A}^{-1}\underline{I}_{abc} \qquad (2\text{-}17b)$$

Equation (2-17) is sometimes called the analysis equation since it tells how to determine the phase a symmetrical components given the unbalanced phase quantities. An expression analogous to (2-17) also holds for voltages.

<u>2.2.3 Properties of the symmetrical component transformation</u>. An important property of the SCT can be demonstrated by applying it to the symmetric prefault voltage profile represented by the bus voltage vectors $\underline{V}_a^o$, $\underline{V}_b^o$, $\underline{V}_c^o$. These vectors are a symmetrical three phase set with phase sequence a,b,c. Thus in terms of the phase a bus voltage vector

$$\underline{V}_b^o = \overline{a}^2\underline{V}_a^o \qquad (2\text{-}18a)$$

$$\underline{V}_c^o = \overline{a}\underline{V}_a^o \qquad (2\text{-}18b)$$

Applying the DSCT,

$$\underline{V}_{a0}^{o} = 1/3(\underline{V}_{a}^{o} + \underline{V}_{b}^{o} + \underline{V}_{c}^{o}) \qquad (2\text{-}19a)$$

$$= 1/3(1 + \bar{a}^2 + \bar{a})\underline{V}_{a}^{o}$$

$$= \underline{0}$$

$$\underline{V}_{a1}^{o} = 1/3(\underline{V}_{a}^{o} + \bar{a}\underline{V}_{b}^{o} + \bar{a}^2\underline{V}_{c}^{o}) \qquad (2\text{-}19b)$$

$$= 1/3(1 + 1 + 1)\underline{V}_{a}^{o}$$

$$= \underline{V}_{a}^{o}$$

$$\underline{V}_{a2}^{o} = 1/3(\underline{V}_{a}^{o} + \bar{a}^2\underline{V}_{b}^{o} + \bar{a}\underline{V}_{c}^{o}) \qquad (2\text{-}19c)$$

$$= 1/3(1 + \bar{a} + \bar{a}^2)\underline{V}_{a}^{o}$$

$$= \underline{0}$$

where $\underline{0}$ is an n-dimensional vector containing only zero.

Equation (2-19) exhibits a very important property of symmetrical components. Under symmetrical conditions the zero sequence system voltages (and currents) are zero while the "a" phase of the positive sequence is the per phase description in phase a of the original balanced system. Nonzero values of zero and negative sequence voltages and currents indicate an unbalanced operating condition.

A further indication of the value of the SCT can be demonstrated by applying it to the analysis of the hypothetical component discussed previously (Figure 2.1 on page 11). Recall that the component is connected between buses i and k and that the phase voltages and currents associated with the device satisfy (2-8). The vectors of phase voltages at buses i and k, $\underline{V}_{abc-i}$ and $\underline{V}_{abc-k}$, and the vector of phase currents, $\underline{I}_{abc}$, defined in (2-7) have the transforms $\underline{V}_{012-i}$, $\underline{V}_{012-k}$, and $\underline{I}_{012}$ respectively. The relationship between phase quantity vectors and sequence quantity vectors can be expressed by either the DSCT or the ISCT. Choosing the later,

$$\underline{V}_{012-i} = \underline{A}\,\underline{V}_{abc-i} = \underline{A} \begin{vmatrix} \overline{V}_{a0-i} \\ \overline{V}_{a1-i} \\ \overline{V}_{a2-i} \end{vmatrix} \qquad (2\text{-}20a)$$

$$\underline{V}_{012-k} = \underline{A}\,\underline{V}_{abc-k} = \underline{A} \begin{vmatrix} \overline{V}_{a0-k} \\ \overline{V}_{a1-k} \\ \overline{V}_{a2-k} \end{vmatrix} \qquad (2\text{-}20b)$$

$$\underline{I}_{012} = \underline{A}\,\underline{I}_{abc} = \underline{A} \begin{vmatrix} \overline{I}_{a0} \\ \overline{I}_{a1} \\ \overline{I}_{a2} \end{vmatrix} \qquad (2\text{-}20c)$$

Using the expressions in (2-20), equation (2-7) can be written as

$$\underline{A}\,\underline{V}_{012-i} = \underline{Z}_{abc}\,\underline{A}\,\underline{I}_{012} + \underline{A}\,\underline{V}_{012-k} \qquad (2\text{-}21a)$$

which after multiplying through by $\underline{A}$ inverse becomes

$$\underline{V}_{012-i} = \underline{A}^{-1}\,\underline{Z}_{abc}\,\underline{A}\,\underline{I}_{012} + \underline{V}_{012-k} \qquad (2\text{-}21b)$$

Define the triple matrix product $\underline{A}^{-1}\underline{Z}_{abc}\underline{A}$ from (2-21b) as

$$\underline{Z}_{012} = \underline{A}^{-1}\,\underline{Z}_{abc}\,\underline{A} \qquad (2\text{-}22)$$

Using $\underline{A}^{-1}$ as defined by (2-17), $\underline{Z}_{abc}$ as in (2-8) and $\underline{A}$ from (2-16), it is easy to show that

$$\underline{Z}_{012} = \begin{vmatrix} \overline{Z}_s+j2X_m+3\overline{Z}_n & 0 & 0 \\ 0 & \overline{Z}_s-jX_m & 0 \\ 0 & 0 & \overline{Z}_s-jX_m \end{vmatrix} \qquad (2\text{-}23a)$$

or,

$$\underline{Z}_{012} = \begin{vmatrix} \overline{Z}_0 & 0 & 0 \\ 0 & \overline{Z}_1 & 0 \\ 0 & 0 & \overline{Z}_2 \end{vmatrix} \quad \text{(2-23b)}$$

where

$$\overline{Z}_0 = \overline{Z}_s + j2X_m + 3\overline{Z}_n \quad \text{(2-24)}$$

= per phase impedance of the component to zero sequence current or "the zero sequence impedance"

$$\overline{Z}_1 = \overline{Z}_s - jX_m \quad \text{(2-25)}$$

= per phase impedance of the component to positive sequence current or the "positive sequence impedance"

$$\overline{Z}_2 = \overline{Z}_s - jX_m \quad \text{(2-26)}$$

= per phase impedance of the component to negative sequence current or the "negative sequence impedance"

Finally substituting (2-23b) into (2-21b) the relationship between the sequence system voltages and the currents can be written as

$$\begin{vmatrix} \overline{V}_{a0-i} \\ \overline{V}_{a1-i} \\ \overline{V}_{a2-i} \end{vmatrix} = \begin{vmatrix} \overline{Z}_0 & 0 & 0 \\ 0 & \overline{Z}_1 & 0 \\ 0 & 0 & \overline{Z}_2 \end{vmatrix} \begin{vmatrix} \overline{I}_{a0} \\ \overline{I}_{a1} \\ \overline{I}_{a2} \end{vmatrix} + \begin{vmatrix} \overline{V}_{a0-k} \\ \overline{V}_{a1-k} \\ \overline{V}_{a2-k} \end{vmatrix} \quad \text{(2-27a)}$$

or

$$\underline{V}_{012-i} = \underline{Z}_{012}\, \underline{I}_{012} + \underline{V}_{012-k} \quad \text{(2-27b)}$$

Equations (2-8) and (2-27) are symmetrical component transform pairs and both equations represent the hypothetical component under either symmetrical or unsymmetrical operating conditions. In the previous section it was noted that the off diagonal entries in $\underline{Z}_{abc}$ imply a phase-to-phase coupling that makes the relationship between phase voltage and currents difficult to solve under unbalanced conditions. It is particularly significant that the transformed impedance matrix $\underline{Z}_{012}$ is diagonal. The decoupled model in (2-27) is much easier to solve than the coupled model in (2-8b).

The diagonal character of $\underline{Z}_{012}$ implies that s-sequence (s=0,1,2) currents result only in (or from) s-sequence voltage drops. This allows the transformed component model to be represented by the decoupled per phase equivalent circuits, one in each sequence system. The appropriate per phase sequence system models for the component under study can be deduced from (2-27) and are shown in Figure 2.3.

Under symmetrical conditions the voltages and currents in the zero and negative sequence models are zero and the positive sequence model fully characterizes the voltage and current relationships of the component. It is obvious that the positive sequence system equation

$$\begin{aligned} \overline{V}_{a1-i} &= \overline{Z}_1 \, \overline{I}_{a1} + \overline{V}_{a1-k} \\ &= (\overline{Z}_s - jX_m) \, \overline{I}_{a1} + \overline{V}_{a1-k} \end{aligned} \qquad (2\text{-}28)$$

has exactly the same form as the decoupled per phase expression derived in (2-12b) for symmetrical operation. Note that the positive sequence equivalent circuit Figure 2.3(b) is the same as the symmetrical per phase model depicted in Figure 2.2 (page 13). Under unsymmetrical conditions voltages and currents will be developed in the zero and negative sequence models (the mechanism is described later) which will properly account for phase-to-phase coupling under unbalanced conditions.

In order to fully exploit the power of the SCT in shunt fault analysis it is necessary to have the per phase zero, positive and negative sequence equivalent circuits of all the components of the otherwise balanced system. These equivalents can be derived using

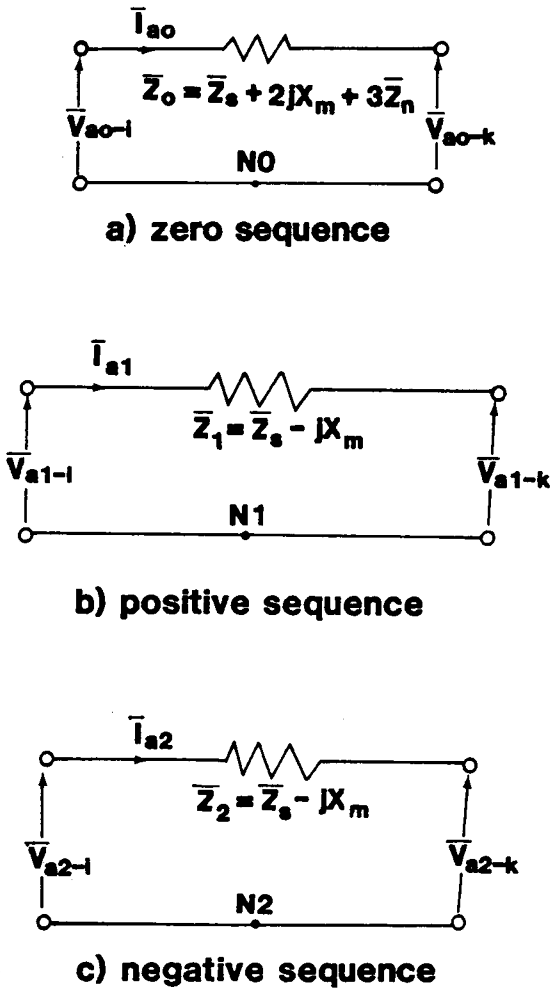


Figure 2.3. Sequence models for the hypothetical component

analysis techniques similar to that applied to the hypothetical component considered in this section. The next section presents sequence system models for common types of power system components.

2.3 Sequence Models of Components

Equivalent circuits in each sequence are presented below for generators, transmission lines, transformers and impedance type loads. Derivations of these models are not given but can be found in (Anderson, Stevenson, Gross). Equations are written expressing the sequence network currents of the various system components as functions of the sequence network bus voltages and the parameters of the equivalent circuits. These equations will be used in a later section to calculate phase currents under faulted conditions.

2.3.1 Sequence models for generators.

A wye grounded generator connected to bus i and operating under balanced conditions has the three phase representation given in Figure 2.4. The per phase positive, negative and zero sequence equivalent circuits are shown in Figure 2.5. In general the sequence impedances of the generator ($\bar{Z}_0$, $\bar{Z}_1$, $\bar{Z}_2$) can have different values. Methods for the experimental evaluation of these parameters are given in (Elgerd, 1982).

If the sequence components of the voltages at bus i are known, then from Figure 2.5 the generator sequence currents can be computed from:

$$\begin{vmatrix} \bar{I}_{a0-i} \\ \bar{I}_{a1-i} \\ \bar{I}_{a2-i} \end{vmatrix} = \begin{vmatrix} 0 \\ \bar{E}_a/\bar{Z}_1 \\ 0 \end{vmatrix} - \begin{vmatrix} 1/(\bar{Z}_0+3\bar{Z}_n) & 0 & 0 \\ 0 & 1/\bar{Z}_1 & 0 \\ 0 & 0 & 1/\bar{Z}_2 \end{vmatrix} \begin{vmatrix} \bar{V}_{a0-i} \\ \bar{V}_{a1-i} \\ \bar{V}_{a2-i} \end{vmatrix} \qquad (2\text{-}29)$$

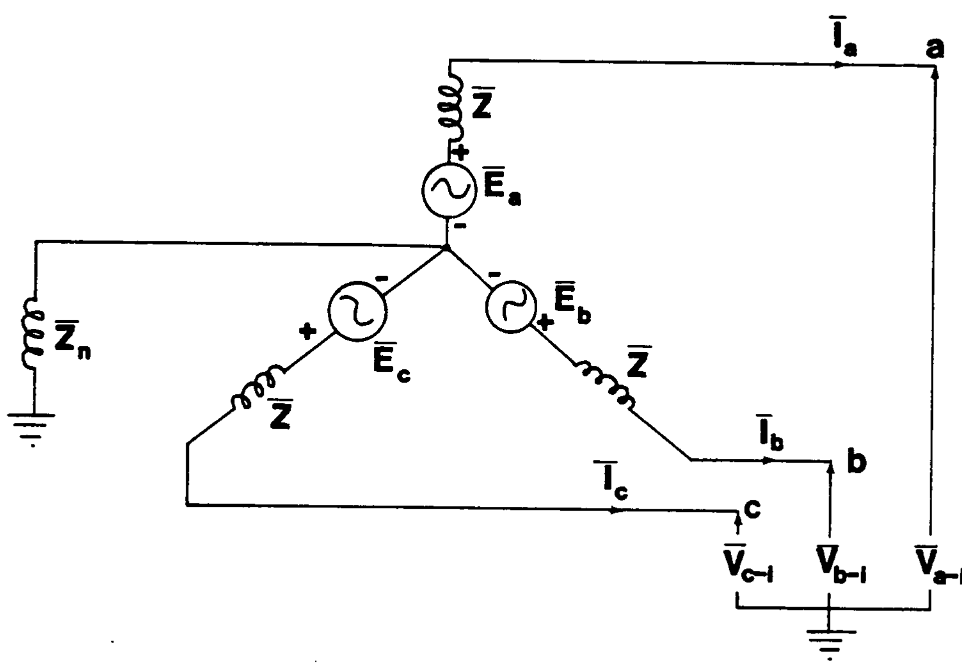


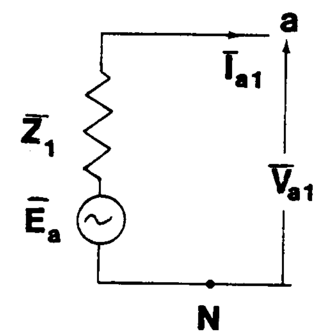
Figure 2.4. Three phase generator representation

2.3.2 Sequence models for a transmission line. Transmission lines are modelled in each sequence by a symmetric-pi equivalent circuit as shown in Figure 2.6. The parameters of the symmetric-pi are the total series impedance $\bar{Z}_{series-s}$ (s = 0,1,2) and the total shunt admittance $\bar{Y}_{shunt-s}$, one half of which is connected at each end of the line. Passive components like lines and transformers have the property that their positive and negative sequence parameters are the same, that is,

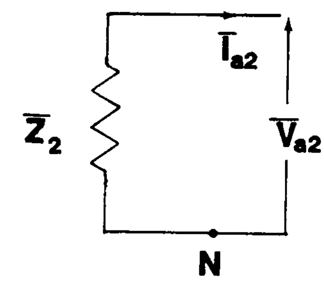
$$\bar{Z}_{series-1} = \bar{Z}_{series-2} \qquad (2\text{-}30)$$

$$\bar{Y}_{shunt-1} = \bar{Y}_{shunt-2}$$

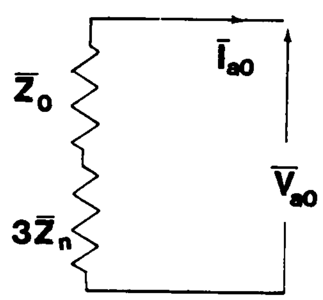
A proof of (2-30) can be found in Gross. The zero sequence model parameters are generally different than those of the positive and negative sequence models and depend on overhead groundwires and the properties of return paths through earth.



a) Positive sequence



b) Negative sequence



c) Zero sequence

Figure 2.5. Sequence representations of a generator

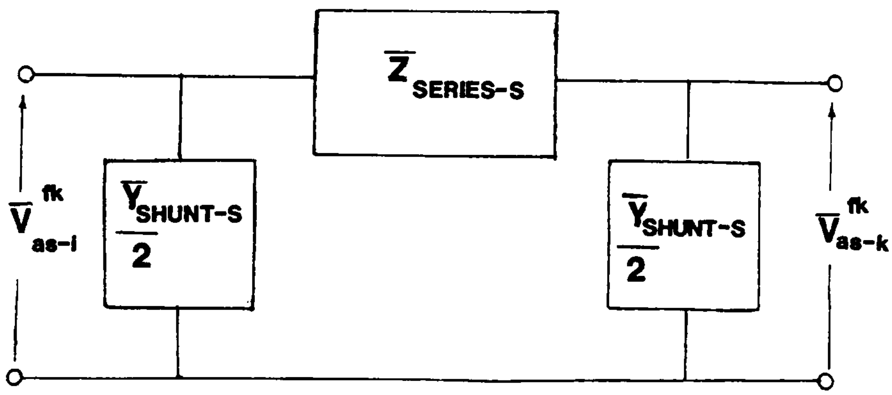


Figure 2.6. Sequence representations of a transmission line

In Figure 2.6 it is assumed that the line is connected between arbitrary buses i and k. The symmetrical components of the line currents flowing from i to k and from k to i can be expressed in terms of the symmetric-pi parameters and bus voltages in each sequence as

$$\begin{vmatrix} \overline{I}_{as-ik} \\ \overline{I}_{as-ki} \end{vmatrix} = \begin{vmatrix} \overline{Y}_{11} & \overline{Y}_{12} \\ \overline{Y}_{21} & \overline{Y}_{22} \end{vmatrix} \begin{vmatrix} \overline{V}_{as-i} \\ \overline{V}_{as-k} \end{vmatrix} \qquad (2\text{-}31a)$$

where (s = 0,1,2) and

$$\overline{Y}_{11} = \overline{Y}_{22} = \overline{Y}_{shunt-s}/2 + 1/\overline{Z}_{series-s} \qquad (2\text{-}31b)$$

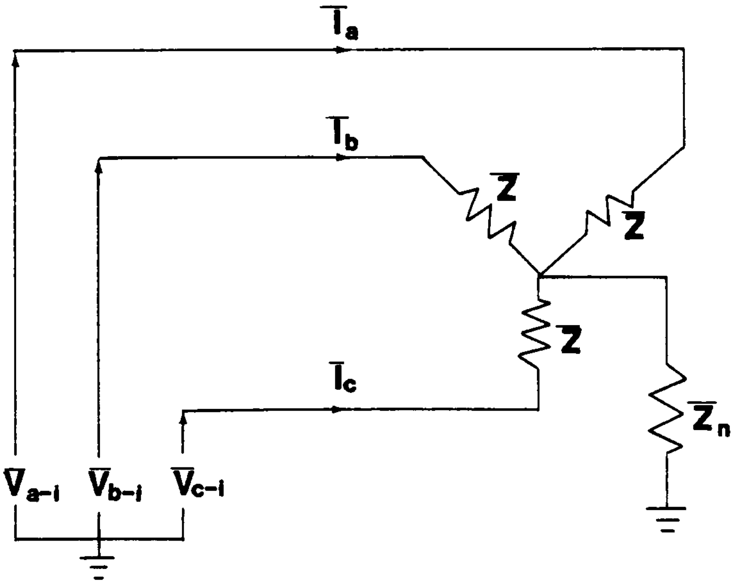
$$\overline{Y}_{21} = \overline{Y}_{12} = -1/\overline{Z}_{series-s}$$

<u>2.3.3 Sequence models of an impedance type load</u>. Three phase models of wye grounded and delta connected impedance type loads are shown in Figure 2.7. Per phase sequence models and current equations for such loads are shown in Figure 2.8.

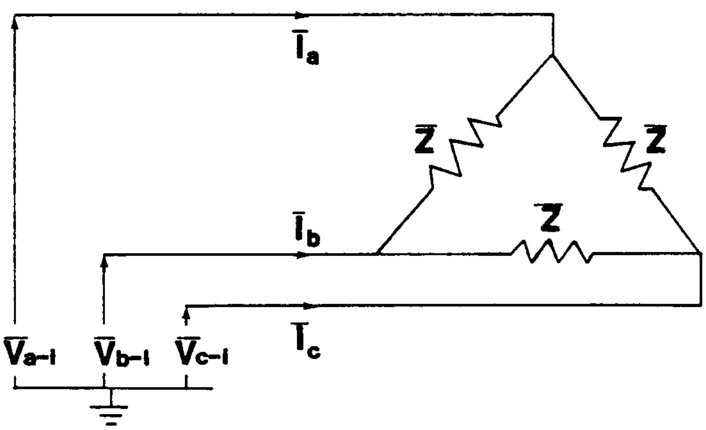
<u>2.3.4 Sequence models of a transformer</u>. Transformers are passive devices and the structure and parameters of the positive and negative sequence models are identical. If the magnetizing branch is neglected the appropriate positive and negative sequence equivalent circuits are as shown in Figure 2.9, where $\overline{Z}_T$ represents the total transformer leakage impedance in per unit.

In Figure 2.9 it is assumed that the transformer is connected between arbitrary buses i and k. The symmetrical components of the line currents flowing from i to k and from k to i can be expressed in terms of the transformer impedance and the bus voltages as

$$\begin{vmatrix} \overline{I}_{as-ik} \\ \overline{I}_{as-ki} \end{vmatrix} = \begin{vmatrix} 1/\overline{Z}_T & -1/\overline{Z}_T \\ -1/\overline{Z}_T & 1/\overline{Z}_T \end{vmatrix} \begin{vmatrix} \overline{V}_{as-i} \\ \overline{V}_{as-k} \end{vmatrix} \text{ for } s=1,2 \qquad (2\text{-}32)$$

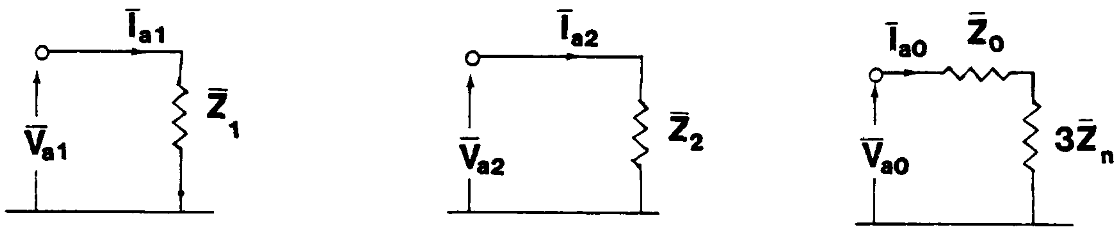


(a) wye connected impedence load



(b) delta connected impedance load

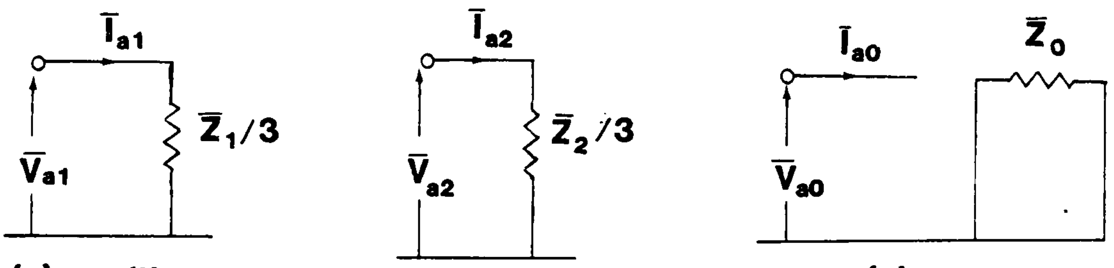
Figure 2.7. Three phase models of impedance type loads



(a) positive (b) negative (c) zero

$$\begin{bmatrix} \bar{I}_{a0} \\ \bar{I}_{a1} \\ \bar{I}_{a2} \end{bmatrix} = \begin{bmatrix} 1/(\bar{Z}_0+3\bar{Z}_n) & 0 & 0 \\ 0 & 1/\bar{Z}_1 & 0 \\ 0 & 0 & 1/\bar{Z}_2 \end{bmatrix} \begin{bmatrix} \bar{V}_{a0} \\ \bar{V}_{a1} \\ \bar{V}_{a2} \end{bmatrix}$$

(a) wye connected



(a) positive (b) negative (c) zero

$$\begin{bmatrix} \bar{I}_{a0} \\ \bar{I}_{a1} \\ \bar{I}_{a2} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 3/\bar{Z}_1 & 0 \\ 0 & 0 & 3/\bar{Z}_2 \end{bmatrix} \begin{bmatrix} \bar{V}_{a0} \\ \bar{V}_{a1} \\ \bar{V}_{a2} \end{bmatrix}$$

(b) delta connected

Figure 2.8. Sequence models of impedance type loads representation

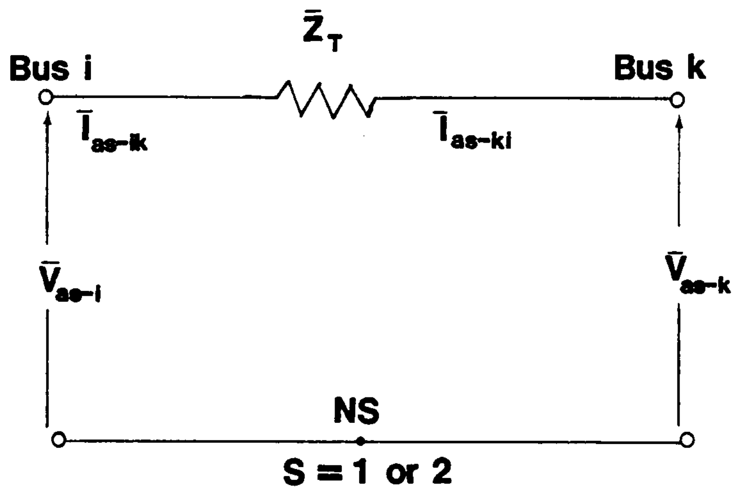


Figure 2.9. Positive and negative sequence representation of a transformer

The positive and negative sequence representations shown in Figure 2.9 fail to take in one very important detail. When wye-wye and delta-delta connections are used, the phases can always be labeled so that there is no phase shift between the two sides of the transformer. For the case of the delta-wye and wye-delta connection it is not possible to arrange the windings or label the phases so that no phase shift results. There is an American standard for the connection of wye-delta transformer windings that minimizes the phase shift to 30 degrees. This standard requires that a positive (negative) sequence voltage on the high voltage side lead (lag) the corresponding positive (negative) sequence voltage on the low voltage side by 30 degrees irrespective of which side of the transformer is delta (or wye) connected. There is no phase shift in the zero sequence for any type of transformer connection. Wye-delta phase shift will not be neglected. It is not represented in the sequence models of transformers so that the sequence networks (described in the next section) will be bilateral and characterized by symmetric admittance and impedance matrices. The phase shift will be included by appropriately correcting sequence network voltages and currents before final application of the ISCT to find faulted phase voltages and currents. An algorithm for correcting the sequence voltages and currents is given in Chapter 3.

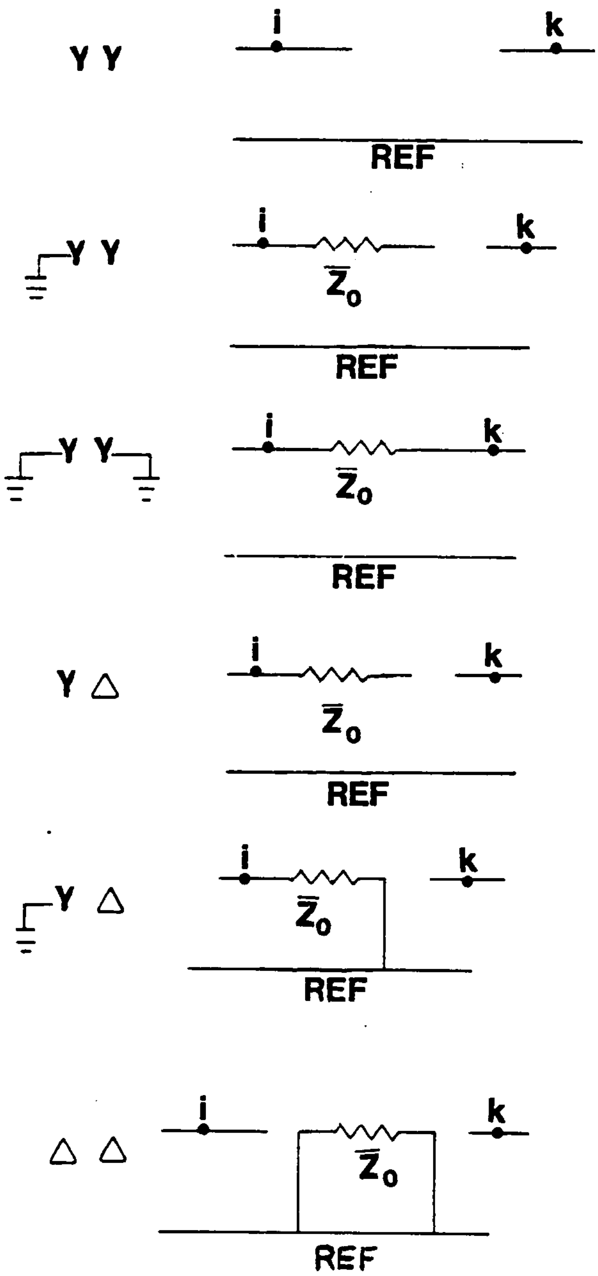
The zero sequence representation of a transformer is dependent on how the transformer windings are connected. There are six different connection possibilities which are shown in Figure 2.10 along with the appropriate zero sequence equivalent circuit and current equation.

2.4 Sequence Network

An important step in shunt fault analysis is the construction and modelling of the per phase zero, positive and negative sequence networks of the balanced prefault system. The s-sequence network (s = zero, positive, negative) is obtained by interconnecting the per phase s-sequence equivalent circuits of the various system components according to the topology specified by the power system single line diagram. The procedure is demonstrated below for an example system.

2.4.1 Construction of the zero, positive and negative sequence networks.

Figure 2.11 is the single line diagram and component data for a simple four bus power system. For simplicity the series resistance and shunt fault admittance associated with the system components



$$\begin{vmatrix} \bar{I}_{a0-ik} \\ \bar{I}_{a0-ki} \end{vmatrix} = \begin{vmatrix} 0 & 0 \\ 0 & 0 \end{vmatrix} \begin{vmatrix} \bar{V}_{a0-i} \\ \bar{V}_{a0-k} \end{vmatrix}$$

$$\begin{vmatrix} \bar{I}_{a0-ik} \\ \bar{I}_{a0-ki} \end{vmatrix} = \begin{vmatrix} 0 & 0 \\ 0 & 0 \end{vmatrix} \begin{vmatrix} \bar{V}_{a0-i} \\ \bar{V}_{a0-k} \end{vmatrix}$$

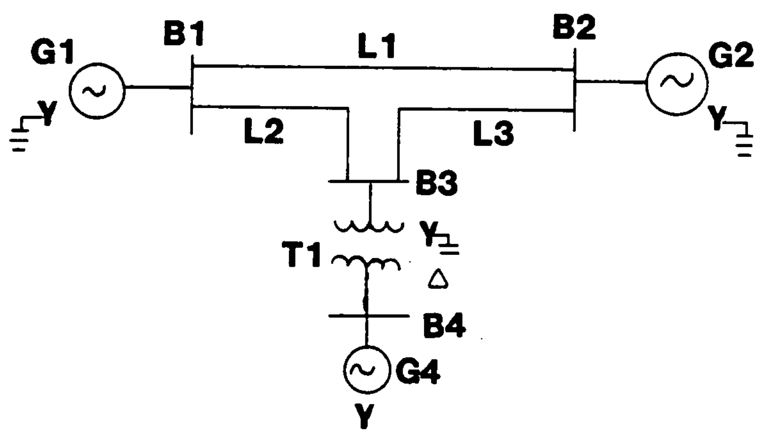
$$\begin{vmatrix} \bar{I}_{a0-ik} \\ \bar{I}_{a0-ki} \end{vmatrix} = \begin{vmatrix} 1/\bar{Z}_0 & -1/\bar{Z}_0 \\ -1/\bar{Z}_0 & 1/\bar{Z}_0 \end{vmatrix} \begin{vmatrix} \bar{V}_{a0-i} \\ \bar{V}_{a0-k} \end{vmatrix}$$

$$\begin{vmatrix} \bar{I}_{a0-ik} \\ \bar{I}_{a0-ki} \end{vmatrix} = \begin{vmatrix} 0 & 0 \\ 0 & 0 \end{vmatrix} \begin{vmatrix} \bar{V}_{a0-i} \\ \bar{V}_{a0-k} \end{vmatrix}$$

$$\begin{vmatrix} \bar{I}_{a0-ik} \\ \bar{I}_{a0-ki} \end{vmatrix} = \begin{vmatrix} 1/\bar{Z}_0 & 0 \\ 0 & 0 \end{vmatrix} \begin{vmatrix} \bar{V}_{a0-i} \\ \bar{V}_{a0-k} \end{vmatrix}$$

$$\begin{vmatrix} \bar{I}_{a0-ik} \\ \bar{I}_{a0-ki} \end{vmatrix} \quad \begin{vmatrix} 0 & 0 \\ 0 & 0 \end{vmatrix} \begin{vmatrix} \bar{V}_{a0-i} \\ \bar{V}_{a0-k} \end{vmatrix}$$

Figure 2.10. Zero sequence equivalents of transformer banks



L1,L2,L3	$\bar{Z}_1 = \bar{Z}_2 = j0.1$
	$\bar{Z} = j0.4$
T1	$\bar{Z}_0 = \bar{Z}_1 = \bar{Z}_2 = j0.05$
G1,G2,G3	$\bar{Z}_1 = \bar{Z}_2 = j0.1$
	$\bar{Z}_0 = j0.02$

Figure 2.11. Single line diagram of example system

have been neglected. These factors can be included in the analysis (the code developed will accomodate them) but they generally have a small impact on fault calculations. Using the per phase sequence equivalent circuits presented in the previous sections, the zero, positive and negative sequence networks can be constructed. Sequence networks for the example are shown in Figure 2.12.

Although the example system is small, it does demonstrate several general properties of sequence networks. In particular:

1. Each sequence network has exactly the same number of buses as in the single line diagram (four in this case). In addition, each sequence network has a well defined neutral denoted as N0, N1, or N2.
2. The sequence networks represent the otherwise balanced system. Since synchronous generators are designed to generate symmetrical three phase emfs, the only sources in the sequence networks appear in the positive sequence. The zero and negative sequence networks are strictly passive.
3. Under prefault conditions the three sequence networks are decoupled. There are no circuit paths between the networks.
4. In the positive or negative sequence network there will be at least one path from every bus to the sequence neutral. However, the presence of the ungrounded wye and/or delta con-connections may create situations in the zero sequence network where there are no paths to N0 from one or more buses (for instance bus 4 in the example system).

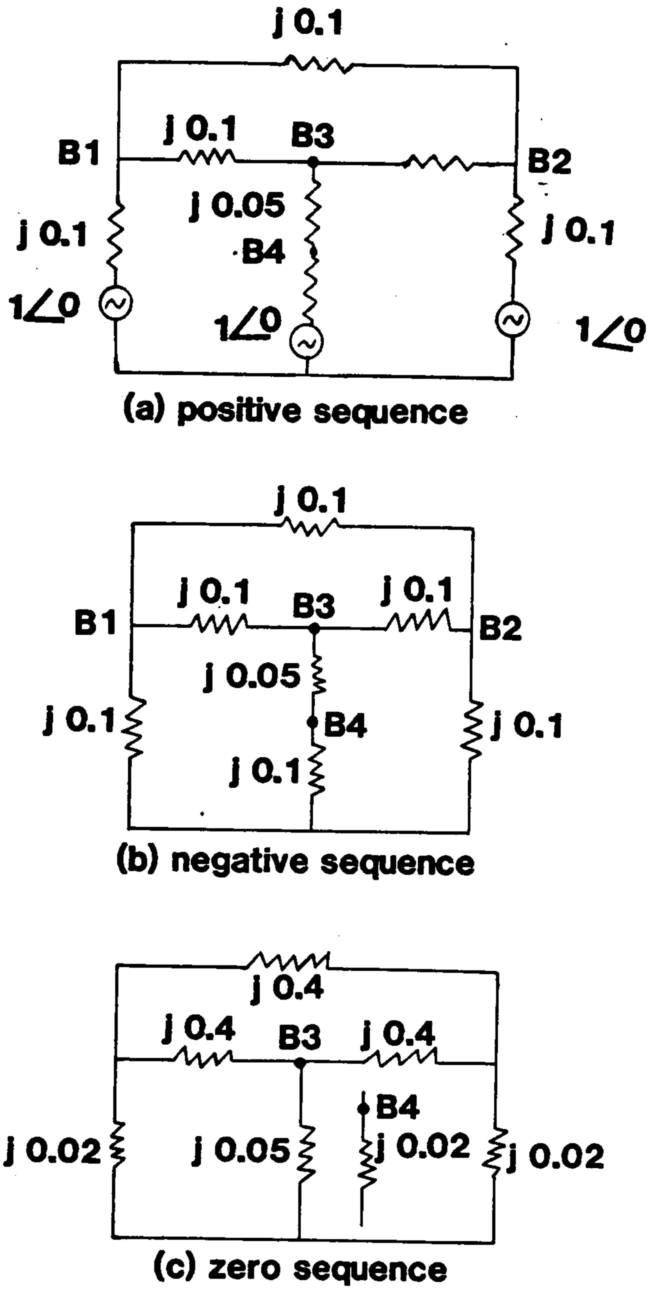


Figure 2.12. Sequence networks for example system

If the emfs in the positive sequence network are short circuited then all three sequence networks are linear bilateral passive networks. Such networks can be modelled by their admittance and impedance matrices which are reviewed in the next section.

2.4.2 The admittance and impedance matrices of the sequence networks. The interaction between a linear bilateral passive network and external sources applied between the buses (or nodes) of the network and the neutral can be described by the bus admittance and impedance matrices. Let the bus voltages in the n-bus sequence networks be denoted by the vectors $\underline{V}_{a0}$, $\underline{V}_{a1}$ and $\underline{V}_{a2}$ which are defined as

$\underline{V}_{as}$ = vector of bus voltage phasors in the s-sequence network (s=0,1,2)

where

$\{\underline{V}_{as}\}_i$ = bus i voltage phasor in the s-sequence network (s=0,1,2) (i=1,...,n)

$= \overline{V}_{as-i}$

In addition let the vectors $\underline{J}_{a0}$, $\underline{J}_{a1}$ and $\underline{J}_{a2}$ denote the vectors of phasor currents injected (by external sources) at the n-buses of the sequence network, i.e.

$\underline{J}_{as}$ = vector of phasor currents injected at the buses of the s-sequence network by external sources (s=0,1,2)

where

$\{\underline{J}_{as}\}_i$ = current injected at bus i of the s-sequence

= network (s=0,1,2) (i=1,2,...,n)

$$\{\underline{J}_{as}\}_i = \bar{J}_{as-i}$$

For each sequence the bus voltages arising due to the currents injected by external agents are deteremined by the equation

$$\underline{J}_{as} = \underline{Y}_s \underline{V}_{as} \tag{2-33}$$

where

$\underline{Y}_s$ = matrix of driving point and transfer admittances of the s-sequence network

= the s-sequence bus admittance matrix

The bus admittance matrix for a particular network can be easily generated from knowledge of the admittances of the various branches in that network. In the absence of parallel circuit, mutually coupled lines, the bus admittance matrix of an n-bus sequence network can be constructed using the simple rules

$$\{\underline{Y}_s\}_{ii} = \bar{Y}_{s-ii} \tag{2-34a}$$

= sum of the admittances of all branches connected to bus i (i=1,2,...,n) in the s-sequence network

$$\{\underline{Y}_s\}_{ij} = \bar{Y}_{s-ij} \tag{2-34b}$$

= negative of the admittance of the branch connecting bus i to bus j (i=1,2,...,n) (j=1,2,...,n) $i \neq j$ in the s-sequence network

The reader will note that equation (2-34b) implies that the bus admittance matrix is symmetric, i.e

$$\bar{Y}_{ij} = \bar{Y}_{ji}$$

Using (2-34) the bus admittance matrix of the zero, positive and negative sequence network ($\underline{Y}_0$, $\underline{Y}_1$ and $\underline{Y}_2$ respectively) of the example in Figure 2.12 are

$$\underline{Y}_0 = j \begin{vmatrix} -55.00 & 2.50 & 2.50 & 0.00 \\ 2.50 & -55.00 & 2.50 & 0.00 \\ 2.50 & 2.50 & -25.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 \end{vmatrix} \qquad (2\text{-}35)$$

$$\underline{Y}_1 = \underline{Y}_2 = j \begin{vmatrix} -30.00 & 10.00 & 10.00 & 0.00 \\ 10.00 & -30.00 & 10.00 & 0.00 \\ 10.00 & 10.00 & -40.00 & 0.00 \\ 0.00 & 0.00 & 20.00 & -30.00 \end{vmatrix} \qquad (2\text{-}36)$$

If there is at least one path to the sequence neutral from every system bus then the bus admittance matrix is nonsingular and equation (2-33) can be inverted to obtain

$$\underline{V}_{as} = \underline{Z}_s \, \underline{J}_{as} \qquad (2\text{-}37)$$

where

$$\underline{Z}_s = \underline{Y}_s^{-1} \qquad (2\text{-}38)$$

= matrix of driving point and transfer impedances of the s-sequence network (s=0,1,2)

= the s-sequence bus impedance matrix

For the example system the positive and negative sequence admittance matrices given by (2-36) are nonsingular and can be inverted to obtain

$$\underline{Z}_1 = \underline{Z}_2 = j \begin{vmatrix} 0.0525 & 0.0275 & 0.0300 & 0.0200 \\ 0.0275 & 0.0525 & 0.0300 & 0.0200 \\ 0.0300 & 0.0300 & 0.0600 & 0.0400 \\ 0.0200 & 0.0200 & 0.0400 & 0.0600 \end{vmatrix} \qquad (2\text{-}39)$$

The zero sequence bus admittance matrix for the example system (2-35) is singular because there is no path from bus 4 to neutral N0. However, by a limiting process it can be shown that the zero sequence bus admittance matrix for the example approaches

$$\underline{Z}_0 = j \begin{vmatrix} 0.01831 & 0.00092 & 0.00192 & 0.00000 \\ 0.00092 & 0.01831 & 0.00192 & 0.00000 \\ 0.00192 & 0.00192 & 0.04038 & 0.00000 \\ 0.00000 & 0.00000 & 0.00000 & \text{infinity} \end{vmatrix} \qquad (2\text{-}40)$$

The bus impedance matrices of the sequence networks will be extremely useful in applying Thevenin's theorem to the analysis of shunt faults. An algorithm for generating $\underline{Z}_0$, $\underline{Z}_1$ and $\underline{Z}_2$ is presented in Chapter 3. For those cases where $\underline{Y}_0$ is singular the algorithm computes an approximation of the limiting value of $\underline{Z}_0$.

The uncoupled sequence networks represent only the balanced prefault system. In the next section, the modelling of the faulted system is reviewed.

2.5 Sequence Model of the Faulted Power System

Previous sections have shown that the SCT can be used to resolve the three phase models of the components in the "otherwise balanced system" (or OBS) into three decoupled sequence networks. A sequence model of the faulted system can be obtained by combining the sequence networks with an appropriate sequence model of the fault. The sequence model of a particular fault will involve connections through external impedances made between sequence networks at the faulted bus. The effect of these connections will be to allow the sources in the positive sequence network to drive currents through all three networks in order to represent the effects of the fault on the OBS. Establishing the details of the sequence network connections for a specific fault is a boundary value problem which is discussed below.

Let the components in the OBS be represented by their three phase models and assume that an arbitrary shunt fault has been applied at bus k. The three phase model of the faulted system is shown in Figure 2.13. In the figure, $\bar{I}_a^{fk}$, $\bar{I}_b^{fk}$, $\bar{I}_c^{fk}$ are the phase currents flowing into the fault and $\bar{V}_{a-k}^{fk}$, $\bar{V}_{b-k}^{fk}$, $\bar{V}_{c-k}^{fk}$ are the phase voltages at the faulted bus under fault conditions. The figure has been drawn to emphasize the fact that there is a well defined boundary between the OBS and the fault.

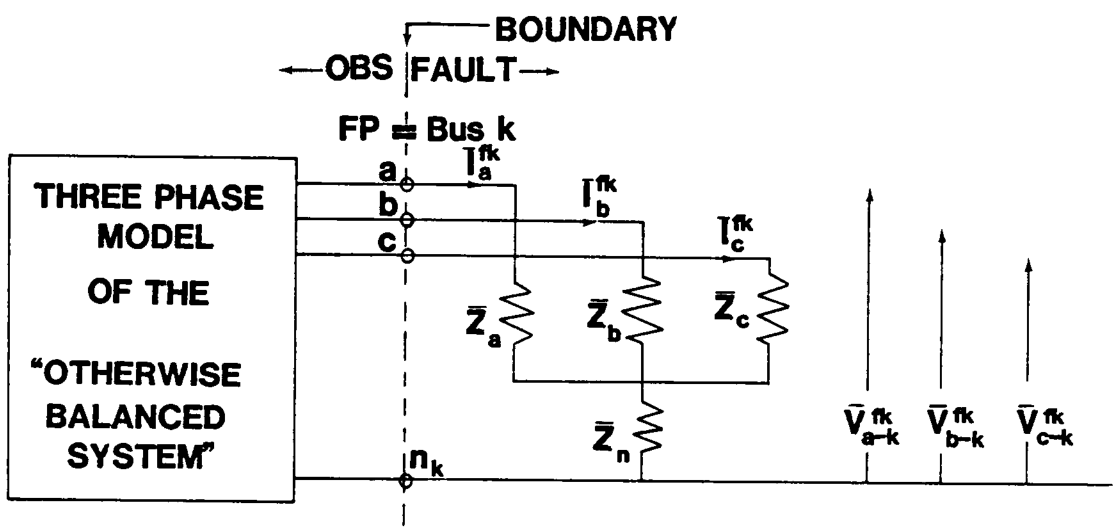


Figure 2.13 Three phase model of the faulted system

A sequence model of the faulted system can be obtained by applying the SCT to the three phase model of Figure 2.13. The result of the transformation is shown in the conceptual diagram of Figure 2.14. Figure 2.14 has also been drawn to emphasize the boundary between the OBS (which is now represented by the sequence networks) and the fault (whose transformed model is still to be determined).

Note that the boundary between the OBS and the fault in the three phase model of Figure 2.13 consists of four nodes; the faulted bus in each phase (labeled a,b,c) and the neutral (n_k). In the sequence model of Figure 2.14 the boundary consists of six nodes; the faulted bus in each sequence network (labeled FP0,FP1, FP2) and the sequence neutrals (N0,N1,N2). If the three phase model and the sequence model of the faulted system are to be symmetrical component transform pairs, then the conditions at the boundary between the OBS and the fault in the two models must agree. More precisely, the details of the sequence model of the fault in Figure 2.14 are the solution to a boundary value problem that requires that the sequence variables at the boundary

$$(\overline{I}_{a0}^{fk}, \overline{I}_{a1}^{fk}, \overline{I}_{a2}^{fk}, \overline{V}_{a0-k}^{fk}, \overline{V}_{a1-k}^{fk}, \overline{V}_{a2-k}^{fk})$$

be symmetrical components of the phase variables at the boundary

$$(\overline{I}_{a}^{fk}, \overline{I}_{b}^{fk}, \overline{I}_{c}^{fk}, \overline{V}_{a-k}^{fk}, \overline{V}_{b-k}^{fk}, \overline{V}_{c-k}^{fk}).$$

The solution to the boundary value problem for a specific shunt fault can be obtained through the following five step procedure (Anderson, 1978).

1. Draw a three phase diagram of the fault labeling all phase currents and phase voltages at the fault point.

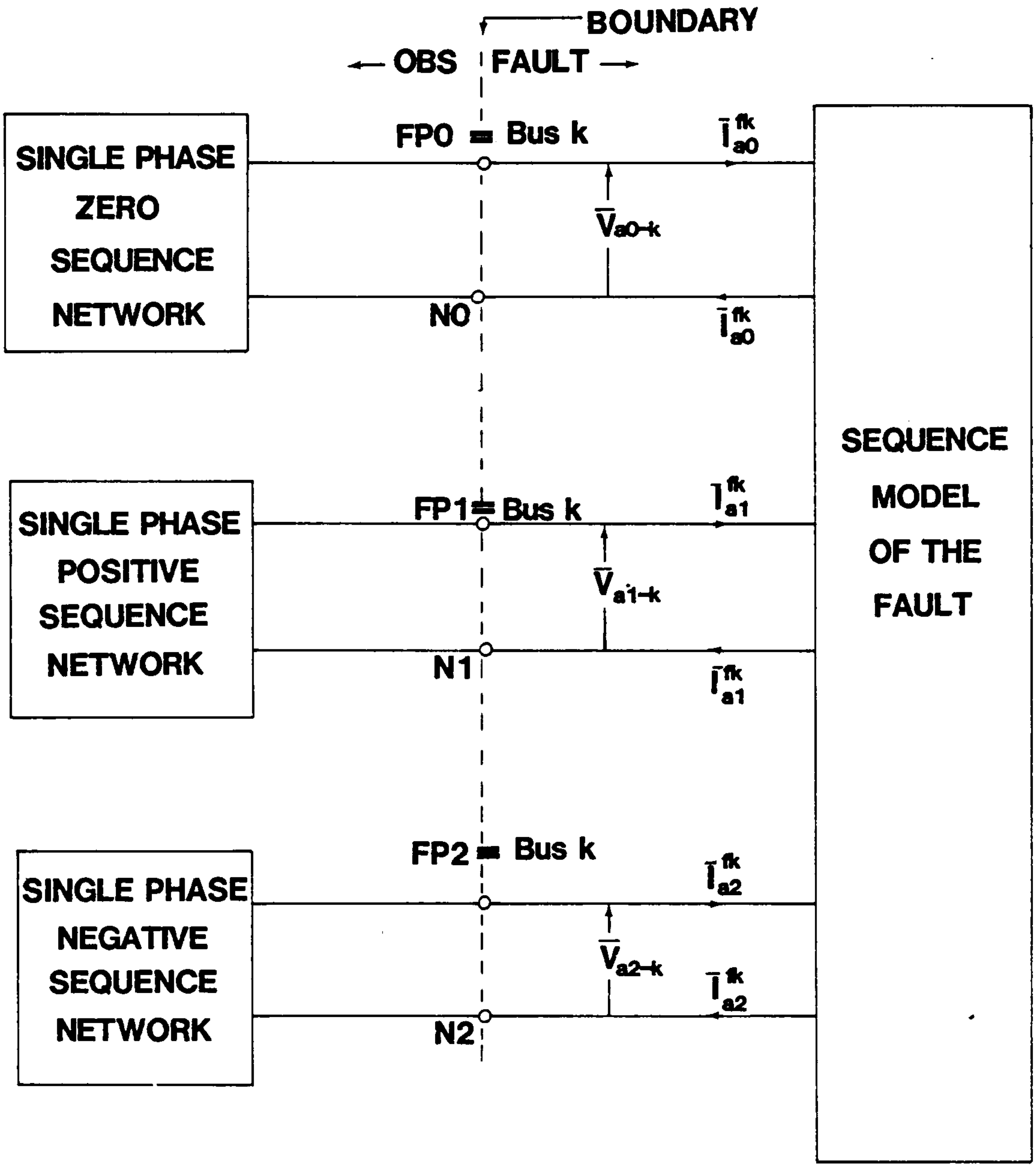


Figure 2.14 Sequence model of the faulted system

2. Write the boundary equations which relate the fault currents ($\overline{I}_a^{fk}$, $\overline{I}_b^{fk}$, $\overline{I}_c^{fk}$) and phase voltages at the fault point ($\overline{V}_{a-k}^{fk}$, $\overline{V}_{b-k}^{fk}$, $\overline{V}_{c-k}^{fk}$).

3. Transform the boundary equations from phase quantities (a,b,c) to sequence quantities (0,1,2).

4. Examine the sequence currents (I_{a0}^{fk}, I_{a1}^{fk}, I_{a2}^{fk}) to determine any appropriate connections to the FPs and Ns (s = 0,1,2) terminals.

5. Examine the sequence voltages ($\overline{V}_{a0-k}^{fk}$, $\overline{V}_{a1-k}^{fk}$, $\overline{V}_{a2-k}^{fk}$) to determine any additional connections to the FPs and Ns terminals. Add any impedances necessary to satisfy the boundary equations in 3.

This procedure is applied below to determine the fault model for five specific types of shunt faults in three phase systems.

<u>2.5.1 The single line-to-ground fault</u>. Sequence network connections for the single line-to-ground (SLG) fault are determined below using Anderson's five step procedure.

Step 1) A fault diagram of the single line to ground fault is shown in Figure 2.15.

Step 2) By inspection of Figure 2.15 the boundary equations of the phase variables for the SLG fault are:

$$\overline{I}_b^{fk} = \overline{I}_c^{fk} = 0 \qquad (2\text{-}41a)$$

$$\overline{V}_{a-k}^{fk} = Z_f \, \overline{I}_a^{fk} \qquad (2\text{-}41b)$$

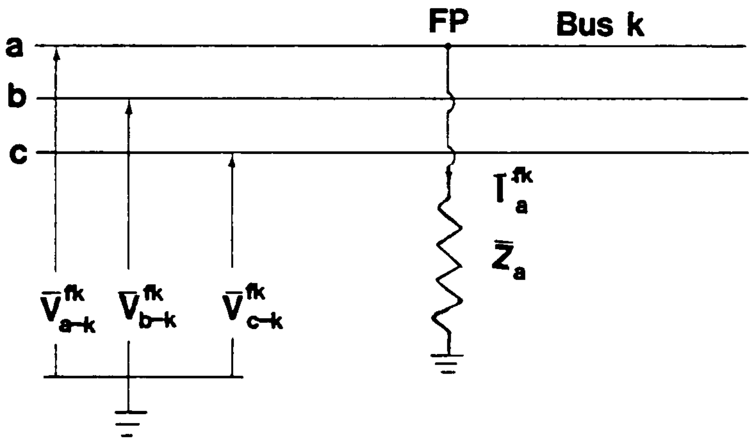


Figure 2.15. Diagram of single line-to-ground fault

Step 3) Applying the SCT to the phase variable boundary equations in (2-41) the appropriate sequence variable boundary equations are:

$$\bar{I}_{a0}^{fk} = \bar{I}_{a1}^{fk} = \bar{I}_{a2}^{fk} = \bar{I}_{a}^{fk}/3 \qquad \text{(2-42a)}$$

$$\bar{V}_{a0-k}^{fk} + \bar{V}_{a1-k}^{fk} + \bar{V}_{a2-k}^{fk} = \bar{Z}_f(\bar{I}_{a0}^{fk} + \bar{I}_{a1}^{fk} + \bar{I}_{a2}^{fk}) \qquad \text{(2-42b)}$$

$$= 3\bar{Z}_f\bar{I}_{a0}^{fk}$$

Step 4) The sequence current equation (2-42a) shows that the sequence components of the fault current are all identical. This implies that the sequence networks are connected in series at the fault point.

Step 5) The sequence voltage equations (2-42b) indicate that the series connected sequence networks are in parallel with an external impedance of $3\bar{Z}_f$. The appropriate sequence network connections for the SLG fault are shown in Figure 2.16.

2.5.2 The almost general fault. Three other interesting shunt faults are the symmetrical three phase fault, the line-to-line fault and the double line-to-ground fault. Each of these faults can be viewed as a special case of an "almost general" (AG) fault. Sequence network connections for the AG fault are derived below.

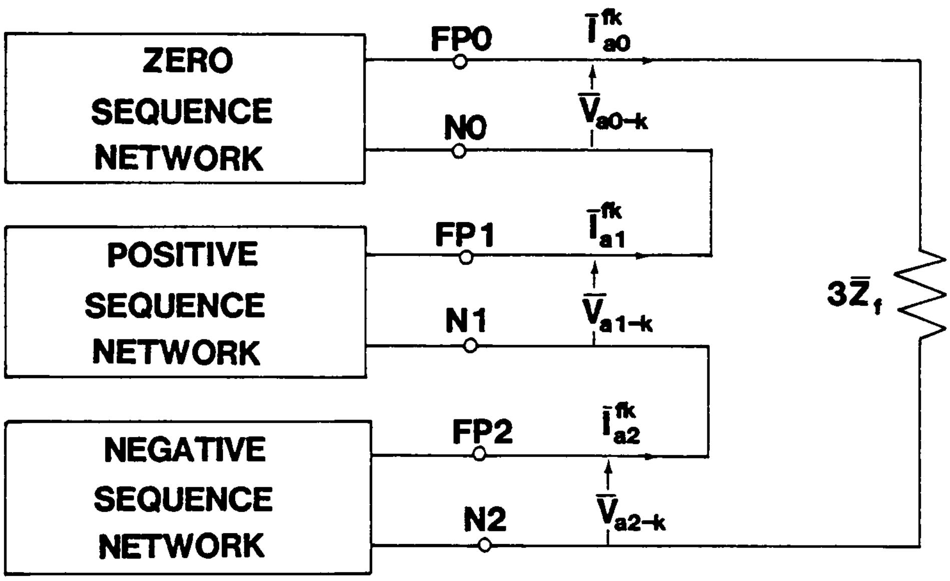


Figure 2.16. Sequence network connections for the single line-to-ground fault

Step 1) A phase diagram of the almost general fault is shown in Figure 2.17.

Step 2) By inspection of Figure 2.17 the boundary equations (a,b,c) for the AG fault are:

$$\begin{vmatrix} \bar{V}^{fk}_{a-k} \\ \bar{V}^{fk}_{b-k} \\ \bar{V}^{fk}_{c-k} \end{vmatrix} = \begin{vmatrix} \bar{Z}_a+\bar{Z}_n & \bar{Z}_n & \bar{Z}_n \\ \bar{Z}_n & \bar{Z}_b+\bar{Z}_n & \bar{Z}_n \\ \bar{Z}_n & \bar{Z}_n & \bar{Z}_b+\bar{Z}_n \end{vmatrix} \begin{vmatrix} \bar{I}^{fk}_{a} \\ \bar{I}^{fk}_{b} \\ \bar{I}^{fk}_{c} \end{vmatrix} \qquad (2\text{-}43)$$

Step 3) Transforming (2-43) the sequence boundary equations are:

$$\begin{vmatrix} \bar{V}^{fk}_{a0-k} \\ \bar{V}^{fk}_{a1-k} \\ \bar{V}^{fk}_{a2-k} \end{vmatrix} = \frac{1}{3} \begin{vmatrix} \bar{Z}_a+2\bar{Z}_b+9\bar{Z}_n & \bar{Z}_a-\bar{Z}_b & \bar{Z}_a-\bar{Z}_b \\ \bar{Z}_a-\bar{Z}_b & \bar{Z}_a+2\bar{Z}_b & \bar{Z}_a-\bar{Z}_b \\ \bar{Z}_a-\bar{Z}_b & \bar{Z}_a-\bar{Z}_b & \bar{Z}_a+2\bar{Z}_b \end{vmatrix} \begin{vmatrix} \bar{I}^{fk}_{a0} \\ \bar{I}^{fk}_{a1} \\ \bar{I}^{fk}_{a2} \end{vmatrix} \qquad (2\text{-}44)$$

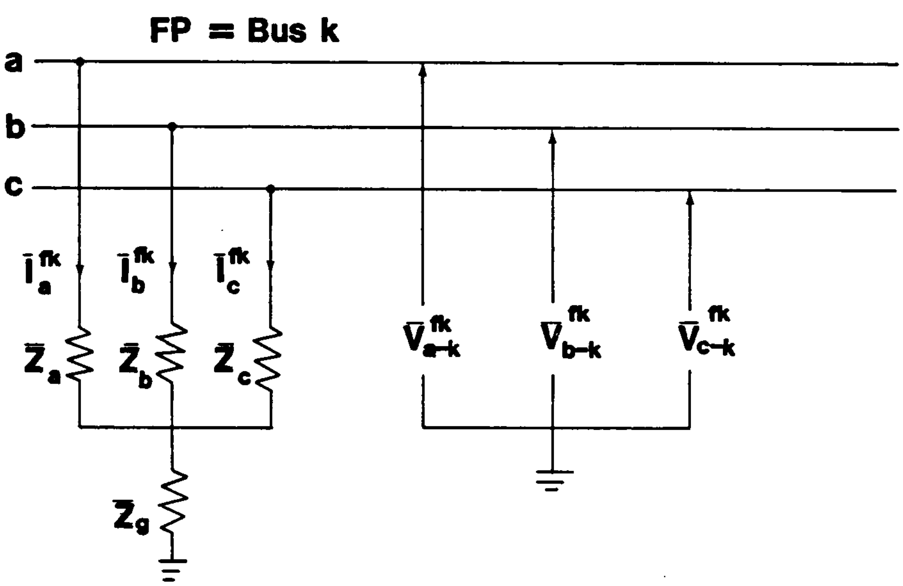


Figure 2.17. Diagram of the almost general fault

Step 4) For this particular fault there are no sequence current equations to consider.

Step 5) No sequence network connections are obvious from the boundary equations in step 3. However, by considering sequence voltage differences the following expressions can be derived from equation (2-44).

$$\overline{V}^{fk}_{a0-k} - \overline{I}^{fk}_{a0}\,(\overline{Z}_b+3\overline{Z}_n) = \overline{V}^{fk}_{a1-k} - \overline{I}^{fk}_{a1}\,\overline{Z}_b \tag{2-45a}$$

$$= \overline{V}^{fk}_{a2-k} - \overline{I}^{fk}_{a2}\,\overline{Z}_b \tag{2-45b}$$

$$= \overline{V}_x \tag{2-45c}$$

where $\overline{V}_x$ is simply the common value of all the voltage drops appearing in the equalities (2-45a) and (2-45b).

Equation (2-45) can be given the network interpretation in Figure 2.18. Note that the model in Figure 2.18 implies that the sequence components of the fault current sum to zero (they all enter a common node denoted as node x in Figure 2.18). In other words, the model thus far assumes that the phase a fault current is zero since from the ISCT,

$$\overline{I}^{fk}_{a} = \overline{I}^{fk}_{a0} + \overline{I}^{fk}_{a1} + \overline{I}^{fk}_{a2} \tag{2-46}$$

Referring to the diagram of the AG fault in Figure 2.17 it is clear that $\overline{I}^{fk}_{a}$ is not zero unless the fault impedance $\overline{Z}_a$ is infinite. Observe that the impedance $\overline{Z}_a$ does not appear in equations (2-45) or the external connections of Figure 2.18. The difficulty is that equations (2-45a) and (2-45b) do not exhaust all the information about sequence network connections contained in the boundary equations of (2-44). By considering the sum of all the sequence voltages in equation (2-44) the following equation can be obtained after some manipulation.

$$\overline{V}_x = 1/3(\overline{I}_{a0} + \overline{I}_{a1} + \overline{I}_{a2})(\overline{Z}_a - \overline{Z}_b) \tag{2-47}$$

Equation (2-47) implies that there is a return path between node x in Figure 2.18 and the common connection of the sequence neutrals through an impedance of $1/3(\overline{Z}_a - \overline{Z}_b)$. The completed sequence network connections for the AG fault are shown in Figure 2.19.

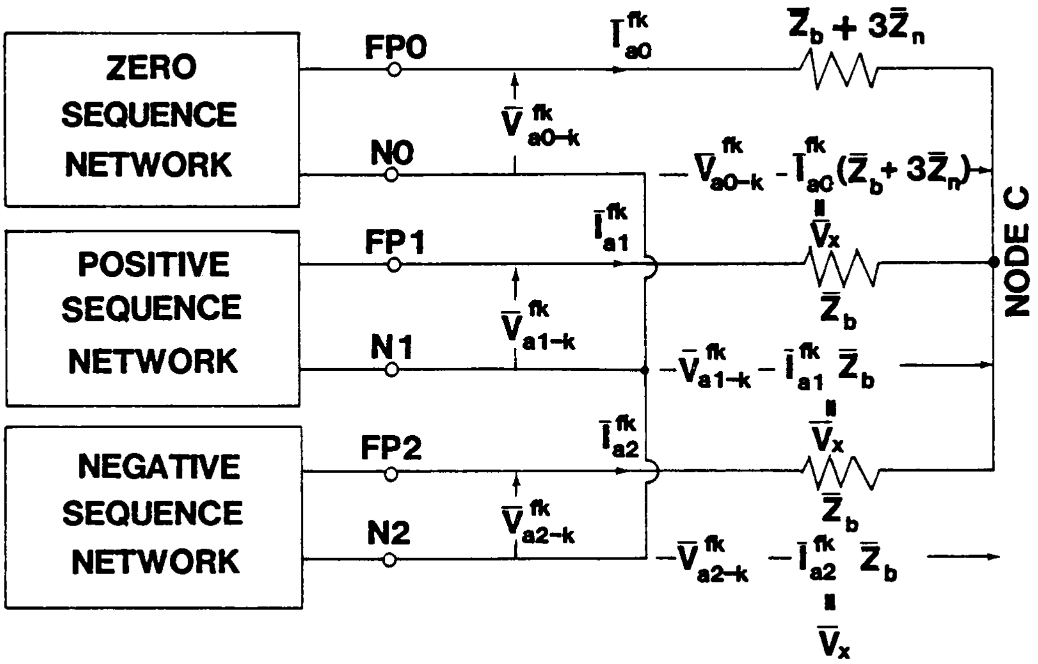


Figure 2.18. Network interpretation of equation (2-45)

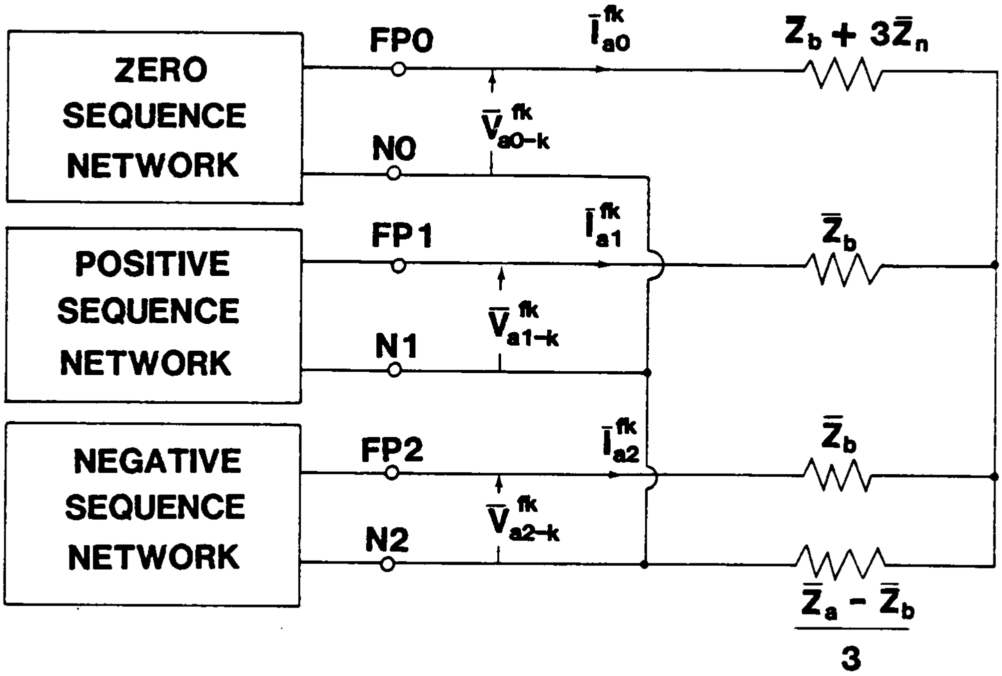


Figure 2.19. Sequence network connections for the almost general fault

2.5.3 The three phase fault. A fault diagram of the symmetrical three phase fault is shown in Figure 2.20. This fault can be viewed as a special case of the AG fault with

$$\bar{Z}_a = \bar{Z}_f \tag{2-48a}$$

$$\bar{Z}_b = \bar{Z}_f \tag{2-48b}$$

$$\bar{Z}_g = \bar{Z}_n \tag{2-48c}$$

Sequence network connections for the three phase fault can be obtained by substituting the impedances of (2-48) into the connections for the AG fault of Figure 2.19. The result is shown in Figure 2.21.

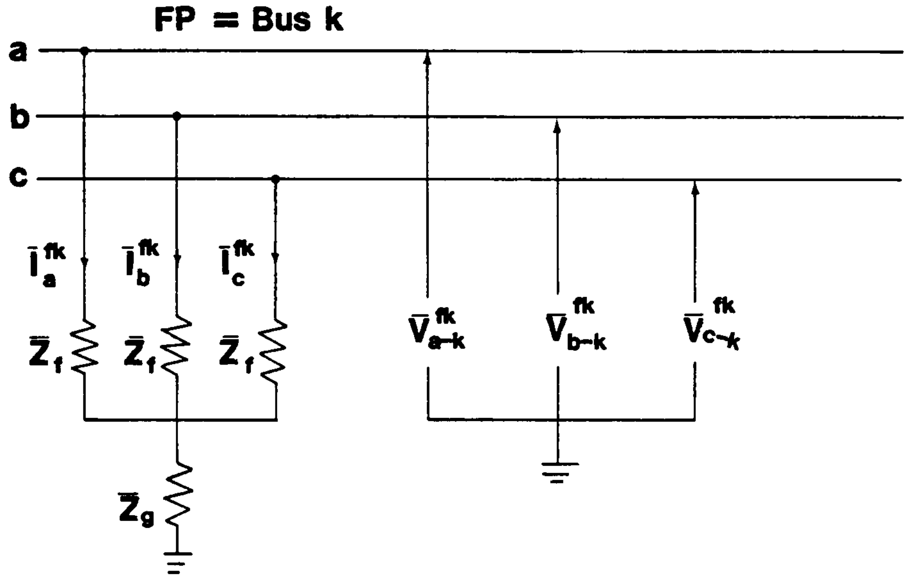


Figure 2.20. Diagram of the symmetrical three phase fault

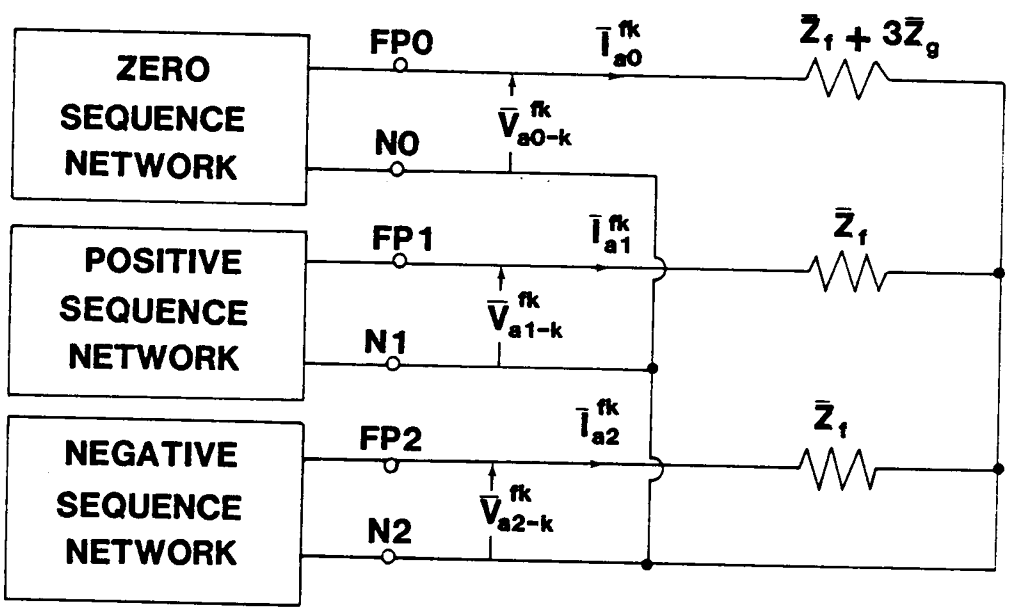


Figure 2.21. Sequence network connections for the symmetrical three phase fault

2.5.4 The double line-to-ground fault. A diagram of the double line-to-ground fault (DLG) is shown in Figure 2.22. This can be viewed as a special case of the AG fault with

$$\overline{Z}_a = \text{infinity} \tag{2-49a}$$

$$\overline{Z}_b = \overline{Z}_f \tag{2-49b}$$

$$\overline{Z}_n = \overline{Z}_g \tag{2-49c}$$

Sequence network connections for the DLG fault can be obtained by substituting the impedances of (2-49) into the connections for the AG fault of Figure 2.19. The result is shown in Figure 2.23.

2.5.5 The line-to-line fault. A diagram of the line-to-line fault (LL) fault is shown in Figure 2.24. This fault can also be viewed as a special case of the AG fault with

$$\overline{Z}_a = \text{infinity} \tag{2-50a}$$

$$\overline{Z}_b = \overline{Z}_f/2 \tag{2-50b}$$

$$\overline{Z}_n = \text{infinity} \tag{2-50c}$$

Sequence network connections for the LL fault can be obtained by substituting the impedance of (2-50) into the connections for the AG fault of Figure 2.19. The result is shown in Figure 2.25.

2.6 Solution of Shunt Faults Using Thevenin's Theorem

Once the external connections between sequence networks are established for a given fault, the three interconnected sequence networks can be viewed as one large single phase system. For example, consider a solid single line-to-ground fault at bus 1 of the 4-bus example system shown previously in Figure 2.11 (page 34). Using the

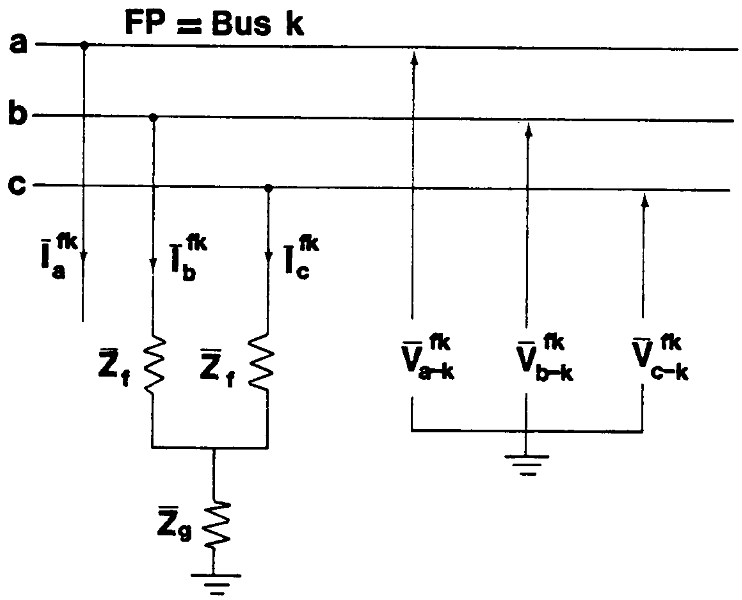


Figure 2.22. Diagram of the double line-to-ground fault

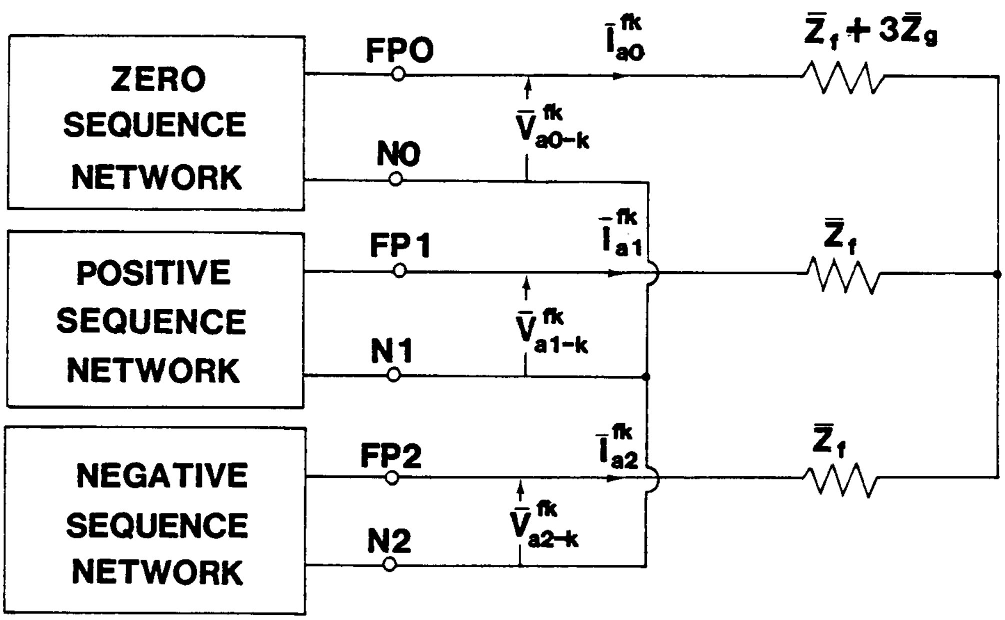


Figure 2.23. Sequence network connections for the double line-to-ground fault

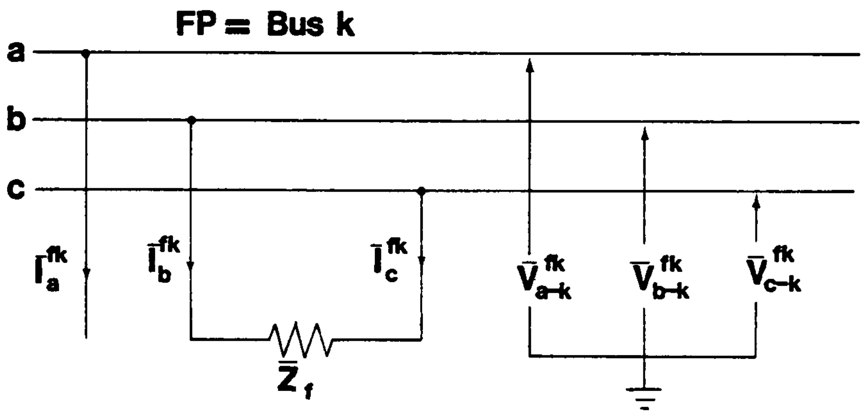


Figure 2.24. Diagram of the line-to-line fault

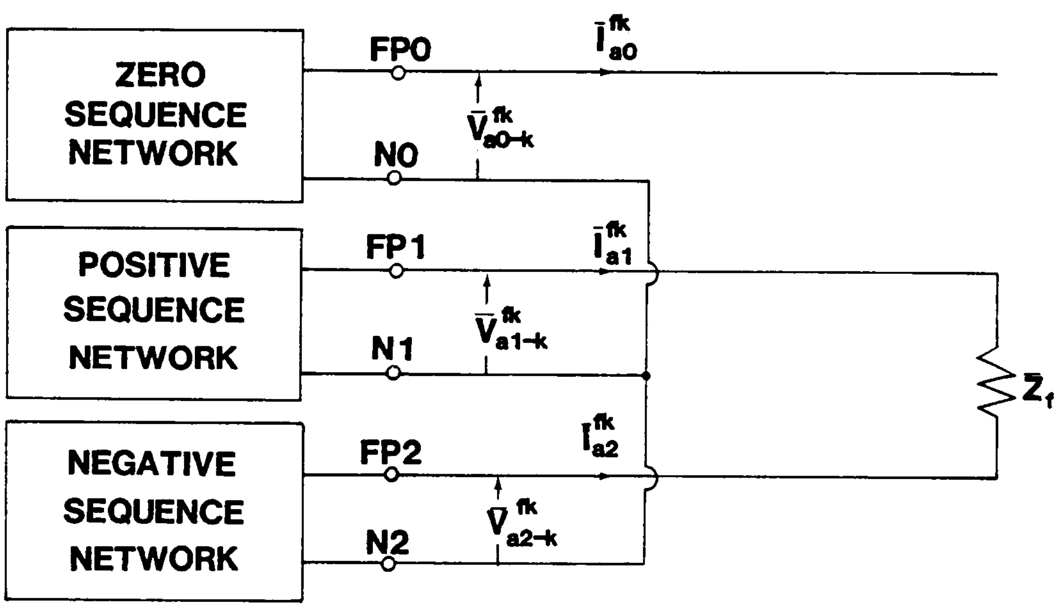


Figure 2.25. Sequence network connections for the line-to-line fault

example system's sequence networks (Figure 2.12 on page 36) and the connections appropriate for a single line-to-ground fault (Figure 2.16 on page 46), the single phase circuit to be solved is that shown in Figure 2.26. A solution for the faulted values of sequence network voltages and currents can be obtained by considering one positive sequence emf at a time (with all others short circuited) and then using superposition. This straight forward technique can be very tedious when a large number of generators is involved. A simpler method based on Thevenin's theorem is described in the next three sections.

2.7 Solution at the Fault Point

Suppose only the fault currents $\bar{I}_a^{fk}$, $\bar{I}_b^{fk}$, $\bar{I}_c^{fk}$ and phase voltages at the fault point $\bar{V}_a^{fk}$, $\bar{V}_b^{fk}$, $\bar{V}_c^{fk}$ are described from a fault study. The simplest way to solve the problem is to reduce each sequence network to its Thevenin equivalent between the fault point and the sequence neutral. For the passive zero and negative sequence networks the appropriate Thevenin equivalents will be simply impedances while the equivalent of the active positive sequence network will be a single emf (equal to the prefault phase a voltage at the fault bus) in series with an impedance. This approach will greatly reduce the complexity in analyzing the interconnected sequence networks since n-1 buses are eliminated in each sequence and the number of sources to be considered is reduced to one.

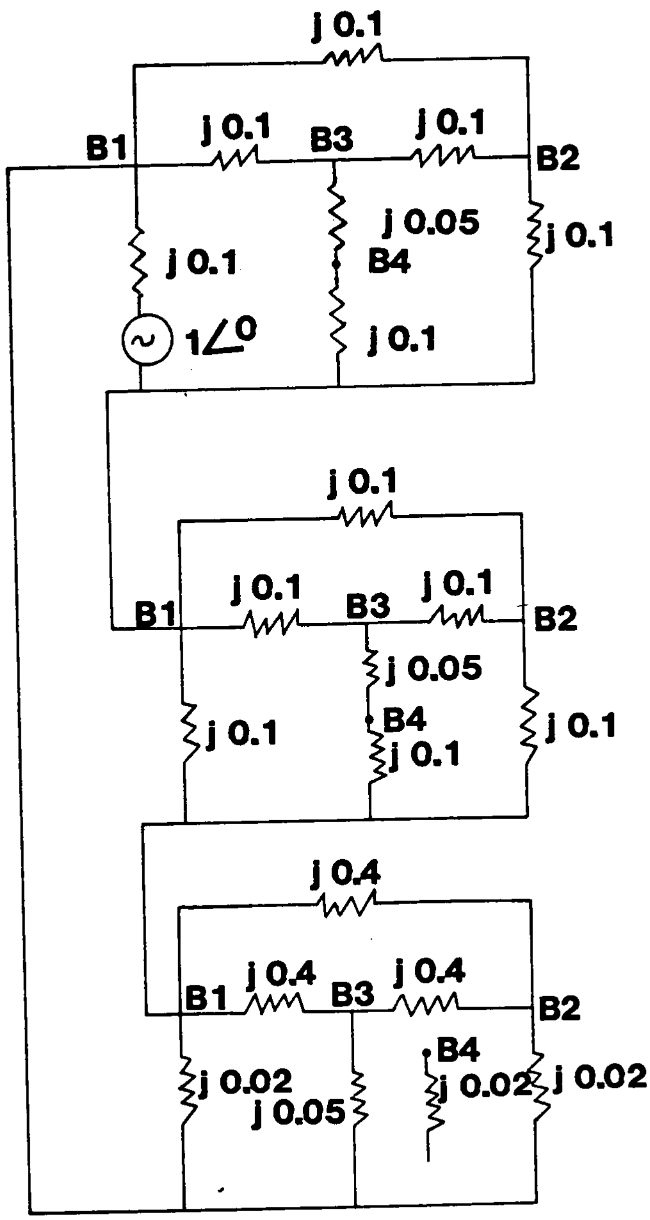


Figure 2.26. Sequence networks of the example system for a solid single line-to-ground fault at bus 1

Let the Tevenin equivalent impedances of the sequence networks be defined by

$\overline{Z}^{fk}_{TH-s}$ = Thevenin equivalent impedance of the s-sequence network between bus k and Ns. (s=0,1,2; k=1,2,...,n) (2-51)

and let the Thevenin equivalent voltage in the positive sequence network be

$\overline{V}^{fk}_{TH-1}$ = The prefault phase a voltage at bus k (2-52)

$= \overline{V}^{o}_{a-k}$

Then in terms of their Thevenin equivalents the sequence networks can be represented as shown in Figure 2.27.

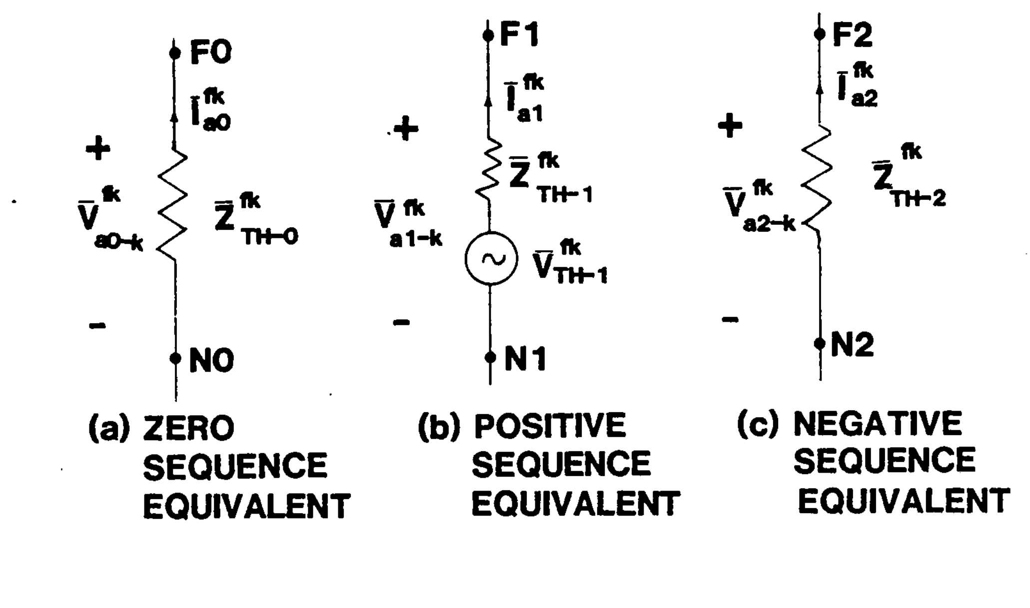


Figure 2.27. Thevenin equivalents of the sequence networks for a fault at bus k

Note that the Thevenin equivalent voltage in the positive sequence $\overline{V}^{fk}_{TH-1} = \overline{V}^{o}_{a-k}$ would be obtained from a prefault load flow solution (or the value $1\underline{/0}$ could be assumed). The Thevenin impedance can be easily established if the bus impedance matrices $\underline{Z}_0$, $\underline{Z}_1$ and $\underline{Z}_2$ are known. In particular, the driving point impedances along the diagonal of these matrices are precisely the Thevenin equivalent impedances at the various buses, i.e.

$$\overline{Z}^{fk}_{TH-s} = \{\underline{Z}_s\}_{kk} = \overline{Z}_{s-kk} \tag{2-53}$$

The analysis of the solid single line-to-ground fault at bus 1 of the example system can be greatly simplified by replacing each sequence network in Figure 2.26 by its Thevenin equivalent. Recall that (2-39) and (2-40) define $\underline{Z}_0$, $\underline{Z}_1$ and $\underline{Z}_2$ for the example. Extracting the diagonal entries corresponding to bus 1 and using the load flow data in Figure 2.11 (page 34) allows Figure 2.26 to be simplified to Figure 2.28.

The circuit in Figure 2.28 can be readily solved for the sequence quantities to obtain

$$\overline{I}^{f1}_{a0} = \overline{I}^{f1}_{a1} = \overline{I}^{f1}_{a2} = \overline{V}^{f1}_{TH-1}/(\overline{Z}^{f1}_{TH-0}+\overline{Z}^{f1}_{TH-1}+\overline{Z}^{f1}_{TH-2}+3\overline{Z}_n) = 8.110\underline{/-90} \tag{2-54}$$

$$\overline{V}^{f1}_{a0} = -\overline{I}^{f1}_{a0}\,\overline{Z}^{f1}_{TH-0} = 0.1485\underline{/180} \tag{2-55a}$$

$$\overline{V}^{f1}_{a1} = \overline{V}^{f1}_{TH-1} - \overline{I}^{f1}_{a1}\,\overline{Z}^{f1}_{TH-1} = 0.5742\underline{/0} \tag{2-55b}$$

$$\overline{V}^{f1}_{a2} = -\overline{I}^{f1}_{a2}\,\overline{Z}^{f1}_{TH-2} = 0.4258\underline{/180} \tag{2-55c}$$

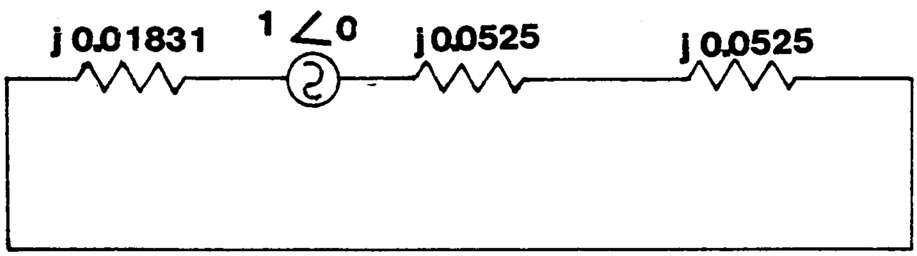


Figure 2.28. Thevenin equivalents of the sequence networks of the example system for a solid single line-to-ground fault at bus 1

which can be inverted using the ISCT (2-15) to obtain the desired fault currents

$$\bar{I}_a^{f1} = 24.329\underline{/-90} \qquad (2\text{-}56a)$$

$$\bar{I}_b^{f1} = 0.0 \qquad (2\text{-}56b)$$

$$\bar{I}_c^{f1} = 0.0 \qquad (2\text{-}56c)$$

and phase voltages at the fault point

$$\bar{V}_a^{f1} = 0.0 \qquad (2\text{-}57a)$$

$$\bar{V}_b^{f1} = 0.8942\underline{/-104.42} \qquad (2\text{-}57b)$$

$$\bar{V}_c^{f1} = 0.8942\underline{/104.42} \qquad (2\text{-}57c)$$

In summary, the fault currents and phase voltages at the fault point can easily be determined by replacing the sequence networks by their Thevenin equivalents in the interconnection diagrams appropriate for a given fault (such as Figures 2.16 on page 46, 2.19 on page 50, 2.21 on page 51, 2.23 on page 53, and 2.25 on page 54). The next two sections describe procedures for extending the information obtained at the fault point to find the faulted values of phase voltages and currents throughout the system.

2.7.1 Computing the voltage profile in each phase under faulted conditions. Recall that the sequence system bus voltage vectors $\underline{V}_{a0}$, $\underline{V}_{a1}$, $\underline{V}_{a2}$ and the phase bus voltage vectors $\underline{V}_a$, $\underline{V}_b$, $\underline{V}_c$ are defined by (2-18) and (2-19) respectively. One of the objectives of a comprehensive shunt fault study is to determine the voltage profile in each phase under faulted conditions. In other words, the vectors $\underline{V}_a^{fk}$, $\underline{V}_b^{fk}$

and $\underline{V}_c^{fk}$ are to be computed. Clearly the desired phase vectors can be found using the DSCT if the sequence system bus voltage vectors under faulted conditions $\underline{V}_{a0}^{fk}$, $\underline{V}_{a1}^{fk}$, $\underline{V}_{a2}^{fk}$ are known. As shown below, the faulted voltage profile in the sequence systems can be readily determined using an extension of Thevenin's theorem (actually Norton's theorem).

According to Norton's theorem, the changes in the s-sequence system (s=0,1,2) bus voltages due to the fault at bus k are the same as those bus voltages that would be obtained in the network with all active sources zeroed and a current source connected between bus k and N-s such that the current $\overline{I}_{as-k}$ flows out of the s-sequence network at bus k and returns through the neutral Ns. Norton's theorem can be interpreted as meaning

$\Delta\underline{V}_{as}^{fk}$ = changes in the s-sequence network bus voltages due to the fault (s=0,1,2) (2-58a)

$$= \underline{V}_{as}^{fk} - \underline{V}_{as}^{o} \quad (2\text{-}58b)$$

$$= \underline{Z}_s \, \underline{J}_{as}^{fk} \quad (2\text{-}58c)$$

where

$\underline{J}_{as}^{fk}$ = vector of currents injected into the s-sequence network by the fault (2-59a)

and

$$\{\underline{J}_{as}^{fk}\}_i = 0 \qquad i \neq k \quad (2\text{-}59b)$$

$$= -\overline{I}_{as}^{fk} \qquad i = k$$

Since all entries in $\underline{J}_{as}^{fk}$ except the kth are zero, equation (2-58c) can be written as

$$\underline{Z}_s\ \underline{J}_{as}^{fk} = -\underline{I}_{as}^{fk}\ \underline{Z}_{s\cdot k} \qquad (2\text{-}60)$$

where

$$\underline{Z}_{s\cdot k} = \text{kth column of } \underline{Z}_s \qquad (2\text{-}61)$$

Using (2-58) through (2-60) the faulted voltage profile in the sequence systems can be computed as

$$\underline{V}_{a0}^{fk} = -\bar{I}_{a0}^{fk}\ \underline{Z}_{0\cdot k} \qquad (2\text{-}62a)$$

$$\underline{V}_{a1}^{fk} = \underline{V}_a^{o} - \bar{I}_{a1}^{fk}\ \underline{Z}_{1\cdot k} \qquad (2\text{-}62b)$$

$$\underline{V}_{a2}^{fk} = -\bar{I}_{a2}^{fk}\ \underline{Z}_{2\cdot k} \qquad (2\text{-}62c)$$

Equation (2-62) makes use of the fact that $\underline{V}_{a0}^{o} = \underline{V}_{a2}^{o} = 0$ and $\underline{V}_{a1}^{o} = \underline{V}_a^{o}$.

The symmetrical components of the fault current $\bar{I}_{a0}^{fk}$, $\bar{I}_{a1}^{fk}$, $\bar{I}_{a2}^{fk}$ required in (2-62) can be determined using the simple procedure outlined in the previous section. It is important to note that for a fault study at bus k that only the kth columns of $\underline{Z}_0$, $\underline{Z}_1$ and $\underline{Z}_2$ are needed to process (2-62).

If wye-delta transformers are present the corrections must be made in the phase angles of some of the entries in the $\underline{V}_{a1}^{fk}$ and $\underline{V}_{a2}^{fk}$ vectors in order to properly account for transformer phase shift. A procedure for determining the appropriate phase shift is deferred to Chapter 3. Once any necessary changes have been made in the phase angles of the sequence network buses associated with wye-delta transformers the DSCT can be applied to the vectors in (2-62) to determine the faulted voltage profile throughout the system in each phase.

The procedure for determining the faulted voltage profile in each phase is now demonstrated on the example system of the previous section. Recall that for a solid single line-to-ground fault at bus 1 that the symmetrical components of the fault currents were computed in equation (2-59). In addition, (2-39) and (2-40) define the bus impedance matrices for the sequence networks. Applying (2-62), the voltage profile in the sequence networks under faulted conditions can be computed as

$$\underline{V}_{a0}^{f1} = -\underline{I}_{a0}^{f1}\,\underline{Z}_{0\cdot 1} \tag{2-63a}$$

$$= (8.110\underline{/90}) \begin{vmatrix} 0.01831\underline{/90} \\ 0.00092\underline{/90} \\ 0.00192\underline{/90} \\ 0.00000\underline{/0} \end{vmatrix} = \begin{vmatrix} 0.1485\underline{/180} \\ 0.0075\underline{/180} \\ 0.0156\underline{/180} \\ 0.0000\underline{/0} \end{vmatrix}$$

$$\underline{V}_{a1}^{f1} = \underline{V}_{a}^{o} - \overline{I}_{a1}^{f1}\,\underline{Z}_{1\cdot 1} \tag{2-63b}$$

$$= \begin{vmatrix} 1.0000\underline{/0} \\ 1.0000\underline{/0} \\ 1.0000\underline{/0} \\ 1.0000\underline{/0} \end{vmatrix} -(8.110/90) \begin{vmatrix} 0.0525\underline{/90} \\ 0.0275\underline{/90} \\ 0.0300\underline{/90} \\ 0.0200\underline{/90} \end{vmatrix}$$

$$= \begin{vmatrix} 0.5742\underline{/0} \\ 0.7770\underline{/0} \\ 0.7567\underline{/0} \\ 0.8378\underline{/0} \end{vmatrix}$$

$$\underline{V}_{a2}^{f1} = -\overline{I}_{a2}^{f1}\,\underline{Z}_{2\cdot 1} \tag{2-63c}$$

$$= (8.110\underline{/90})\begin{vmatrix} 0.0525\underline{/90} \\ 0.0275\underline{/90} \\ 0.0300\underline{/90} \\ 0.0200\underline{/90} \end{vmatrix} = \begin{vmatrix} 0.4258\underline{/180} \\ 0.2230\underline{/180} \\ 0.2433\underline{/180} \\ 0.1622\underline{/180} \end{vmatrix}$$

In order to properly account for wye-delta transformer phase shift 30 degrees is added to the angle of $\overline{V}_{a1-4}^{f1}$ and 30 degrees is subtracted from the angle of $\overline{V}_{a2-4}^{f1}$ in the vectors $\overline{V}_{a1}^{f1}$ and $\overline{V}_{a2}^{f1}$ respectively. Applying (2-16) to the corrected sequence network bus voltage vectors the post fault voltages at the various system buses in each phase are found to be:

$$\underline{V}_{a}^{f1} = \begin{vmatrix} 0.0000\underline{/0} \\ 0.5465\underline{/0} \\ 0.4978\underline{/0} \\ 0.7696\underline{/-40.52} \end{vmatrix} \tag{2-64a}$$

$$\underline{V}_{b}^{f1} = \begin{vmatrix} 0.8942\underline{/-104.42} \\ 0.9115\underline{/-108.18} \\ 0.9078\underline{/-107.45} \\ 0.7696\underline{/-139.48} \end{vmatrix} \tag{2-64b}$$

$$\underline{V}_{c}^{f1} = \begin{vmatrix} 0.8942\underline{/104.42} \\ 0.9115\underline{/108.18} \\ 0.9078\underline{/107.45} \\ 1.0000\underline{/90.00} \end{vmatrix} \tag{2-64c}$$

After the voltage profile under faulted conditions is known in each sequence network, the current flowing in the various system components can be determined. The procedure is described in the next section.

2.7.2 Computing phase currents under faulted conditions. To conclude a shunt fault study the phase currents flowing in each system component under faulted conditions must be computed. The symmetrical components of these currents can be determined, since the equivalent circuits and terminal voltages in each sequence are known. The sequence currents for the various network components can be calculated from the current equation for the given component. For the generator see equation (2-29); for the transmission line, equation (2-31); for the impedance type load, Figure 2.8 (page 30); for transformers, equation (2-32) and Figure 2.10 (page 33).

If delta-wye transformers are present corrections must be made in the phase angles of the currents flowing in the network. These corrections will be made with respect to the faulted bus which will be considered the reference.

The procedure for determining the fault current flow in each phase is demonstrated for the transformer between bus 3 and 4 of the example system. The sequence currents in both directions can be found by using equation (2-32), the equations in Figure 2.10 (page 33), and the voltages calculated in (2-63) as shown.

$$\begin{vmatrix} \overline{I}^{f1}_{a0-34} \\ \overline{I}^{f1}_{a0-43} \end{vmatrix} = \begin{vmatrix} 1/\overline{Z}_0 & 0 \\ 0 & 0 \end{vmatrix} \begin{vmatrix} \overline{V}^{f1}_{a0-3} \\ \overline{V}^{f1}_{a0-4} \end{vmatrix} \tag{2-65a}$$

$$= \begin{vmatrix} 1/0.05\underline{/90} & 0 \\ 0 & 0 \end{vmatrix} \begin{vmatrix} 0.0156\underline{/180} \\ 0.0000\underline{/0} \end{vmatrix}$$

$$= \begin{vmatrix} 0.3119\underline{/90} \\ 0.0000\underline{/0} \end{vmatrix}$$

$$\begin{vmatrix} \overline{I}^{f1}_{a1-34} \\ \overline{I}^{f1}_{a1-43} \end{vmatrix} = \begin{vmatrix} 1/\overline{Z}_1 & -1/\overline{Z}_1 \\ -1/\overline{Z}_1 & 1/\overline{Z}_1 \end{vmatrix} \begin{vmatrix} \overline{V}^{f1}_{a1-3} \\ \overline{V}^{f1}_{a1-4} \end{vmatrix} \tag{2-65b}$$

$$= \begin{vmatrix} 1/0.05\underline{/90} & -1/0.05\underline{/90} \\ -1/0.05\underline{/90} & 1/0.05\underline{/90} \end{vmatrix} \begin{vmatrix} 0.7567\underline{/0} \\ 0.8378/0 \end{vmatrix}$$

$$= \begin{vmatrix} 1.6219\underline{/90} \\ 1.6219\underline{/-90} \end{vmatrix}$$

$$\begin{vmatrix} \overline{I}^{f1}_{a2-34} \\ \overline{I}^{f1}_{a2-43} \end{vmatrix} = \begin{vmatrix} 1/\overline{Z}_2 & -1/\overline{Z}_2 \\ -1/\overline{Z}_2 & 1/\overline{Z}_2 \end{vmatrix} \begin{vmatrix} \overline{V}^{f1}_{a2-3} \\ \overline{V}^{f1}_{a2-4} \end{vmatrix} \tag{2-65c}$$

$$= \begin{vmatrix} 1/0.05\underline{/90} & -1/0.05\underline{/90} \\ -1/0.05\underline{/90} & 1/0.05\underline{/90} \end{vmatrix} \begin{vmatrix} 0.2433\underline{/0} \\ 0.1622\underline{/0} \end{vmatrix}$$

$$= \begin{vmatrix} 1.6219\underline{/90} \\ 1.6219\underline{/-90} \end{vmatrix}$$

In order to properly account for phase shift 30 degrees is subtracted from the angle of $\bar{I}_{a2}^{f1}$ and 30 degrees is added to the angle of $\bar{I}_{a1}^{f1}$. Applying (2-16) to the corrected sequence current components, the post fault current components in each phase are found to be

$$\begin{vmatrix} \bar{I}_{a-34}^{f1} \\ \bar{I}_{b-34}^{f1} \\ \bar{I}_{c-34}^{f1} \end{vmatrix} = \begin{vmatrix} 3.5557\underline{/90} \\ 1.3100\underline{/-90} \\ 1.3100\underline{/-90} \end{vmatrix} \qquad (2\text{-}66a)$$

$$\begin{vmatrix} \bar{I}_{a-43}^{f1} \\ \bar{I}_{b-43}^{f1} \\ \bar{I}_{c-43}^{f1} \end{vmatrix} = \begin{vmatrix} 2.8092\underline{/-90} \\ 2.8092\underline{/90} \\ 0.0000\underline{/0} \end{vmatrix} \qquad (2\text{-}66b)$$

2.8 Summary

A systematic solution for shunt faults is possible assuming that the positive, negative, and zero sequence impedance matrices are available. These matrices can be found by inverting the sequence admittance matrices which are easily formed. If a prefault load flow is available, then the prefault bus voltages at each bus are known. If a prefault load flow has not been calculated then the voltages in phase a can be set to unity with a phase angle of zero. The procedure for solving shunt fault problems is summarized in Table 2.1.

TABLE 2.1

General Fault Solution Algorithm

---

1. Find the bus impedance matrix in each sequence network ($\underline{Z}_0$, $\underline{Z}_1$, $\underline{Z}_2$).
2. Obtain the prefault bus voltage profile in all phases ($\underline{V}_a^o$, $\underline{V}_b^o$, $\underline{V}_c^o$), usually these vectors are obtained from a load flow solution.

   Note: The effects of wye-delta transformer phase shift is neglected at this time.
3. Establish the prefault voltage profile in each sequence network ($\underline{V}_{a0}^o = \underline{V}_{a2}^o = \underline{0}$; $\underline{V}_{a1}^o = \underline{V}_a^o$).
4. Select fault type and location and specify all relevant fault impedances.
5. Establish the Thevenin equivalent between the fault point and neutral in each sequence network ($\overline{Z}_{TH-0}^{fk} = \overline{Z}_{0-kk}$; $\overline{Z}_{TH-1}^{fk} = \overline{Z}_{1-kk}$; $\overline{V}_{TH-1}^{fk} = \overline{V}_{a-k}^o$; $\overline{Z}_{TH-2}^{fk} = \overline{Z}_{2-kk}$).
6. Connect the Thevenin equivalents of the sequence networks based on the specifics of the fault at bus k and solve for the symmetrical components of the fault current ($\overline{I}_{a0}^{fk}$, $\overline{I}_{a1}^{fk}$, $\overline{I}_{a2}^{fk}$).
7. Construct the sequence network current injection vectors ($\overline{J}_{a0}^{fk}$, $\overline{J}_{a1}^{fk}$, $\overline{J}_{a2}^{fk}$).
8. Determine the faulted sequence network bus voltage vectors;

   $$\underline{V}_{as}^{fk} = \underline{V}_{as}^o + \underline{Z}_s \underline{J}_{as}^{fk} \qquad (s = 0,1,2).$$

TABLE 2.1 (Continued)

9. Determine the sequence currents associated with all system components.
10. Correct sequence network voltages and currents to account for wye-delta transformer phase shift.
11. Apply the ISCT to sequence quantities (0,1,2) to find the faulted values of phase quantities (a,b,c).
12. Stop.

CHAPTER 3

MATHEMATICAL METHODS

3.1 Triangular Factorization

It was shown in the previous chapter that in order to perform a shunt fault study at the kth bus of an n-bus power system it is necessary to obtain the kth column of the positive, negative and zero sequence network impedence matrices for the system under study. The appropriate columns can be obtained by: 1) constructing the positive, negative and zero sequence admittance matrices (which are easily assembled from network data), 2) using a matrix inversion routine to invert the admittance matrices and obtain the impedance matrices, 3) extracting the desired column from each impedance matrix. Since only the kth column of the impedance matrices are required for a fault study at bus k, knowledge of the entire impedance matrix in each sequence represents a waste of computational effort. The following discussion describes the triangular factorization technique which will be used to obtain only the required columns of the sequence network impedance matrices and avoid complete inversion of the admittance matrices.

Let $\underline{Y}$ be a known nxn, symmetric, complex, nonsingular matrix with inverse $\underline{Z}$. Further, let $\underline{I}$ be the nxn identity matrix and let $\underline{1}_k$ be an nx1 column vector representing the kth column in $\underline{I}$, i.e.

$$\underline{1}_k = \underline{I}_{\cdot k} = \text{kth column of } \underline{I} \tag{3-1}$$

When $\underline{1}_k$ is premultiplied by any nxn matrix, say $\underline{Z}$, then the resulting nx1 column vector is simply the kth column in $\underline{Z}$, i.e.

$$\underline{Z1}_k = \underline{Z}_{\cdot k} = \text{kth column of } \underline{Z} \tag{3-2}$$

Given $\underline{Y}$ and $\underline{1}_k$, if the linear equation

$$\underline{Yz}_k = \underline{1}_k \tag{3-3}$$

can be solved for the unknown nx1 vector $\underline{z}_k$, then $\underline{z}_k$ represents the kth column on the inverse of $\underline{Y}$ (i.e. the kth column of $\underline{Z}$). Observe that when (3-3) is multiplied by the inverse of $\underline{Y}$ it becomes

$$\underline{z}_k = \underline{Y}^{-1}\underline{1}_k \tag{3-4}$$

$$= \underline{Z1}_k$$

and applying (3-2),

$$\underline{z}_k = \underline{Z}_{\cdot k} \tag{3-5}$$

Triangular factorization is a thechnique for solving (3-3) for the unknown vector $\underline{z}_k$ when $\underline{Y}$ is a symmetric matrix.

Let the matrix $\underline{Y}$ be written in the triangular factorized form

$$\underline{Y} = \underline{TT}^T \tag{3-6}$$

where $\underline{T}^T$ denotes the transpose of $\underline{T}$ where $\underline{T}$ is an nxn lower triangular matrix whose entries are defined by (Gastinel, 1970).

$$T_{ij} = 0 \qquad j > i \tag{3-7a}$$

$$T_{i1} = \frac{Y_{i1}}{\sqrt{Y_{11}}} \qquad i = 1,2,\ldots,n \tag{3-7b}$$

$$T_{ii} = \sqrt{Y_{ii} - \sum_{k=1}^{i-1} T_{ik}^2} \qquad i = 2,3,\ldots,n \tag{3-7c}$$

$$T_{ij} = \frac{Y_{ij} - \sum_{k=1}^{j-1} T_{ik} T_{jk}}{T_{jj}} \qquad 2 \leq j < i \leq n \qquad (3\text{-}7d)$$

Substituting (3-6) into (3-3), equation (3-3) becomes

$$\underline{T}\underline{T}^T \underline{z}_k = \underline{1}_k \qquad (3\text{-}8)$$

Define the product $\underline{T}^T \underline{z}_k$ to be the nxl vector $\hat{\underline{z}}_k$, i.e.

$$\underline{T}^T \underline{z}_k \triangleq \hat{\underline{z}}_k \qquad (3\text{-}9)$$

Substituting (3-9) into (3-8), equation (3-8) becomes

$$\underline{T}\hat{\underline{z}}_k = \underline{1}_k \qquad (3\text{-}10)$$

Equation (3-10) is known as the forward reduction equation. Given $\underline{T}$ and $\underline{1}_k$, the equation is easily solved for the entries in $\hat{\underline{z}}_k$ because $\underline{T}$ is lower triangular. Once the vector $\hat{\underline{z}}_k$ is known, equation (3-9), which is called the back substitution equation, can easily be solved for the desired unknown vector $\underline{z}_k = \underline{Z}_{\cdot k}$ because $\underline{T}^T$ is upper triangular

When delta-wye and wye-delta transformers are present in a system, it is possible for the conventions involving delta-wye and wye-delta connections to result in the formation of a singular zero sequence admittance matrix. An attempt to perform triangular factorization on a singular matrix will result in the occurence of at least one diagonal element of $\underline{T}$ which will be equal to zero. The trouble this fact may cause can be observed by looking at the forth equation of (3-7) where the diagonal element of T is zero. The value of T_{ij} (i > j) for the case of a zero T_{jj} would be infinity. This makes the problem impossible to solve on a computer because it can only store finite values. This problem can be avoided by checking all values of the diagonal elements of T and if a zero value is found replacing it by some small value near zero. In this case the value was selected to be 10^{-10}, which is large enough to guarentee accuracy and well within the range of representation of most computers.

At this point the equations in this chapter are put in the form of a computer algorithm. The triangular factorization of the matrix needs to be done only once for a system. However, since there can be multiple buses to be faulted, the equations (3-6) through (3-10) are split into two separate algorithms. One algorithm was developed for finding the factorized form of the matrix Y and a second algorithm was developed for finding the solution for (3-3). With this in mind, it is now possible to show the algorithm for triangular factorization. There are nine steps as shown in Table 3.1 (Burden,1978).

The matrix T has been calculated by the method in Table 3.1 and now (3-8) must be solved. That solution is shown by the algorithm in Table 3.2.

3.2 Example Case

The usefulness of these algorithms is shown by the following example in Figure 3.1. The Y matrix for this system may be calcultated by inspection and is shown to be

TABLE 3.1

Triangular Factorization Algorithm

1. Set all elements of the matrix $\underline{T}$ to zero.

2. Generate the first column of $\underline{T}$ by the conditions

$$T_{i1} = Y_{i1}/\sqrt{Y_{11}} \qquad \text{for each } i = 1,2,\ldots,n$$

3. Set j = 2.

4. Calculate the value of the diagonal element in column j by

$$T_{jj} = \sqrt{Y_{jj} - \sum_{k=1}^{i-1} T_{jk}^2}$$

5. If j = n then go to step 9.

6. If $T_{jj} = 0$ then $T_{jj} = 10^{-10}$.

7. Calculate the entries below the diagonal in column j by

$$T_{ij} = (Y_{ij} - \sum_{k=1}^{j-1} T_{ik}T_{jk})/T_{jj} \quad \text{for each } i = j+1,j+2,\ldots,n$$

8. Add 1 to j and go to step 4.

9. Stop, the solution has been found.

TABLE 3.2

Forward Reduction and Back Substitution Algorithm

1. Set $\hat{z}_1 = 1_k/T_{11}$.
2. Set $i = 2$.
3. Generate each element $\hat{z}_i$ by forward reduction

$$\hat{z}_i = (1_k - \sum_{j=1}^{i-1} T_{ij}\,\hat{z}_j)/T_{ii}$$

4. Add 1 to i. If $i \leq n$ go to step 3.
5. Set $z_n = \hat{z}_n/T_{nn}$.
6. Set $k = n$.
7. Generate each element z by back substitution

$$z_k = (\hat{z}_k - \sum_{j=k+1}^{n} T_{kj}z_j)/T_{kk}$$

8. Subtract 1 from k. If $k > 0$ go to step 7.
9. The solution is complete.

$$\underline{Y} = \begin{vmatrix} 2.5 & -1.0 & -1.0 \\ -1.0 & 2.5 & -0.5 \\ -1.0 & -0.5 & 2.0 \end{vmatrix} \qquad (3\text{-}11)$$

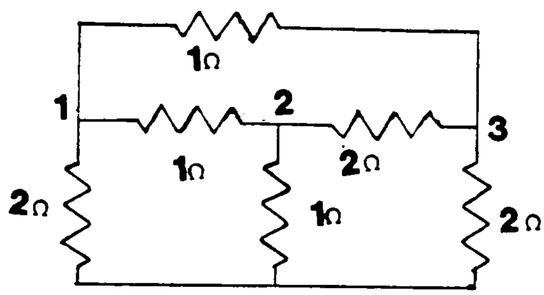


Figure 3.1. Example network

The triangular factorization algorithm can then be applied to this matrix as follows.

Initially all entries in $\underline{T}$ are set to zero per step 1 in the algorithm defined in Table 3.1. Applying step 2, the entries in the first column of $\underline{T}$ are computed as follows.

$$T_{11} = Y_{11}/\sqrt{Y_{11}} = 2.5/\sqrt{2.5} = 1.5811$$

$$T_{21} = Y_{21}/\sqrt{Y_{11}} = -1.0/\sqrt{2.5} = -0.6325 \qquad (3\text{-}12)$$

$$T_{31} = Y_{31}/\sqrt{Y_{11}} = -1.0/\sqrt{2.5} = -0.6325$$

Letting j = 2, at step 3, the entries in the second column of $\underline{T}$ at or below the main diagonal are computed by steps 4 though 8 of the algorithm with the following results.

$$T_{22} = (Y_{22} - T_{21}^2)^{1/2} = (2.5 - (-0.6325)^2)^{1/2} = 1.4491$$

$$T_{32} = (Y_{22} - T_{31}T_{32})/T_{22} = (-0.5 - (-0.6325)(-0.6325))/1.4491$$
$$= -0.6211 \qquad (3\text{-}13)$$

Letting j = 3, step 4 is repeated.

$$T_{33} = (Y_{33} - T_{31}^2 - T_{32}^2)^{1/2} \qquad (3\text{-}14)$$

$$= (2.0 - (-0.6325)^2 - (-0.6211)^2)^{1/2} = 1.1019$$

At this point all elements of the matrix have been generated so the algorithm goes to step 9 and stops. The results of the algorithm are shown in the following matrix.

$$\underline{T} = \begin{vmatrix} 1.5811 & 0.0000 & 0.0000 \\ -0.6325 & 1.4491 & 0.0000 \\ -0.6325 & -0.6211 & 1.1019 \end{vmatrix} \qquad (3\text{-}15)$$

It is now possible to use the forward reduction and back substitution algorithm, table 3.2, to find the value of the second column of the impedance matrix $\underline{Z}$ for the example admittance matrix $\underline{Y}$. The forward reduction equation (3-10) for this example is:

$$\begin{vmatrix} 1.5811 & 0.0000 & 0.0000 \\ -0.6325 & 1.4491 & 0.0000 \\ -0.6325 & -0.6211 & 1.1019 \end{vmatrix} \begin{vmatrix} \hat{z}_1 \\ \hat{z}_2 \\ \hat{z}_3 \end{vmatrix} = \begin{vmatrix} 0.0 \\ 1.0 \\ 0.0 \end{vmatrix} \qquad (3\text{-}16a)$$

or

$$\underline{T}\hat{\underline{z}}_2 = \underline{1}_2 \qquad (3\text{-}16b)$$

Applying steps 1 - 4 of the second algorithm the emtries in $\hat{\underline{z}}_2$ are computed as follows.

$$\hat{z}_1 = 0.0/1.5811 = 0.0$$

$$\hat{z}_2 = (1.0 - (-0.6325)(0.0))/1.4491 = 0.6901 \qquad (3\text{-}17)$$

$$\hat{z}_3 = (0.0 - (-0.6325)(0.0) - (-0.6211)(0.6901))/1.1019$$

$$= 0.3889$$

The back substitution problem, equation (3-9), is

$$\begin{vmatrix} 1.5811 & -0.6325 & -0.6325 \\ 0.0000 & 1.4491 & -0.6211 \\ 0.0000 & 0.0000 & 1.1019 \end{vmatrix} \begin{vmatrix} z_1 \\ z_2 \\ z_3 \end{vmatrix} = \begin{vmatrix} 0.0000 \\ 0.6901 \\ 0.3889 \end{vmatrix} \qquad (3\text{-}18a)$$

or

$$\underline{T}^T \underline{z}_2 = \hat{\underline{z}}_2 \qquad (3\text{-}18b)$$

which is solved by applying steps 5 - 8 of the algorithm. This yields:

$$z_3 = 0.3889/1.1019 = 0.3529 \qquad (3\text{-}19)$$

$$z_2 = (0.6901 - (-0.6211)(0.3529))/1.4491 = 0.6275$$

$$z_1 = (0.0 - (-0.6325)(0.3529) - (-0.6325)(0.6275))/1.5811$$

$$= 0.3922$$

The second column of $\underline{Z}$ for the example has been shown to be

$$\underline{z}_2 = \begin{vmatrix} 0.3922 \\ 0.6275 \\ 0.3529 \end{vmatrix} \qquad (3\text{-}20)$$

The reader can verify the above solution by confirming that the inverse of the example admittance matrix defined in (3-11) is $\underline{Z}$, the inverse of $\underline{Y}$.

$$\underline{Z} = \begin{vmatrix} 0.7451 & 0.3922 & 0.4706 \\ 0.3922 & 0.6275 & 0.3529 \\ 0.4706 & 0.3529 & 0.8235 \end{vmatrix} \qquad (3\text{-}21)$$

Although the example is real, the procedures shown will work for complex $\underline{Y}$ matrices.

3.3 Phase Shifing

Phase shift arising from wye-delta transformers must be considered when performing fault studies. Phase shift does not occur for delta-delta and wye-wye transformer connections. However, for delta-wye and and wye-delta connections the positive and negative sequence quantities are shifted. The amount of phase shift is minimized to 30 degrees by industrial standards which state how the windings of transformers are connected. By convention, the positive sequence quantities on the high voltage side lead the corresponding positive sequence quantities on the low voltage side by 30 degrees, while negative sequence quantities on the high voltage side lag the corresponding negative sequence quantities on the low boltage side by 30 degrees (Gross,1979). No phase shift occurs in the zero sequence network. These effects must be taken into consideration in order to calculate the correct values for both sequence and phase quantities for the fault study.

The first thing to consider when dealing with phase shifts is a reference angle. For convenience the reference is selected to be the phase a voltage phasor at the bus being faulted. The phase of all the other voltage and current phasors will be calculated with respect to this reference.

The following definitions must be made before presenting the phase shifting algorithm:

NB = The number of buses

NL = The number of lines

SB(I) = High voltage bus

EB(I) = Low voltage bus

IBUS(I) = The number of a bus

HVTC(I) = Terminal connection on high voltage side of transformer running between SB(I) and EB(I)

LVTC(I) = Terminal connection on low voltage side of transformer running between SB(I) and EB(I)

M(I) = Index of how much phase shift must be applied at IBUS(I)

FB = Index which indicates IBUS(FB) is the bus being faulted

I = Index value

The algorithm proceeds as follows. All M values are initially set to 1000 to identify them as unknown. One value, M(1), is set to zero arbitrarily. All the elements are cycled through to locate the other terminus (J) of those connected to IBUS(1). If lines, wye-wye, or delta-delta connected transformers are encountered, then M(J) is set to M(1); if a delta-wye connection is encountered, then M(J) is set to M(1) $\pm$ 1, depending on which is the high voltage bus. The process is repeated until no more undefined buses are found. The full phase shift algorithm is shown in Table 3.3 on the following pages. The values

TABLE 3.3

Phase Shift Algorithm

---

1. Set all NB elements of M equal to 1000. This will indicate the index of phase shift is undefined.
2. Set M(1) = 0.
3. Set N = 1.
4. Set IS = SB(N) and JE = EB(N).
5. Set K = 1.
6. If IBUS(K) is equal to IS then set I = K. If IBUS(K) is equal to JE hen set J = K.
7. Add 1 to K. If K is less than or equal to NB go to step 6.
8. If M(I) equals 1000 then go to step 12. If M(J) equals 1000 then go to step 9. Otherwise go to step 16.
9. Check HVTC(N) and LVTC(N), if there is a wye-delta or delta-wye connection then go to step 11.
10. Set M(J) = M(I) and then go to step 16.
11. Set M(J) = M(I) - 1 and then go to step 16.
12. If M(J) equals 1000 then go to step 16.
13. Check HVTC(N) and LVTC(N); if there is a wye-delta or delta-wye connection then go to step 15.
14. Set M(I) = M(J) and then go to step 16.
15. Set M(I) = M(J) + 1.

TABLE 3.3 (Continued)

16. Add 1 to N. If N is less than or equal to NL go to step 4.

17. If M(I) = 1000 go to step 4. I = 1,2,...,NB

18. Set M(I) = M(I) - M(FB). I = 1,2,...,NB

19. Stop.

calculated for M can be used to shift the sequence quantities by 30 degrees multiplied by M(I) for each value of I (Gross, 1979).

CHAPTER 4

COMPUTER USAGE

The program developed for this thesis was written to run on the DEC System 10 as an interactive program. The fault analysis program was also written to be part of a package for performing load flow analysis which had previously been created by the author. It is possible for the user to run only a load flow, only a fault study, or a load flow and then a fault study of the same system. The discussion in this chapter shall be restricted to the fault analysis code. The load flow code has been discussed in the report for which it was developed and will only be explained when necessary.

4.1 System Environment

The program listed in Appendix B is the DEC System 10 interactive version of the code and has been run on a KL10 processor using version 7 of the TOPS10 operating system. The FORTRAN compiler used was the FORTRAN-10 compiler. The version of the program shown in the appendix will compile and run in 160 pages of memory. This version of the program should be able to run with changes to the file open and close statements on any large computer system. A user's guide for this version is given in Appendix A.

4.2 Program Design

The algorithm developed in Chapters 2 and 3 were used to create the fault analysis code. The program was written to be combined with the previously existing load flow to provide a complete package for the analysis of small power systems. Each program can be run independently, or the load flow can be run and the results used to help calculate the fault study for the same system. Figures 4.1 and 4.2 give a flowchart of the interactive computer code.

Figure 4.1 is a flowchart of the load flow code. It is only included because the initial questions give the user the option of going immediately to the fault portion and at the end of the load flow section of the code the user has the option of performing a fault study on the same system. Figure 4.2 shows the fault analysis section of the computer code. The general form of the flowchart used in Figure 4.2 is the same as the algorithm given in Table 2.1 on pages 68 and 69. The following discussion explains the interactive features of the code.

A critical consideration in designing an interactive computer code is the degree of control that the user will have when running the program. The philosophy used in developing the program for this thesis was to give the user as much control as possible without greatly increasing the core size necessary to compile and execute the program.

There are two ways that a user can give the program a data set to analyze. As shown in Figure 4.2, the user may state that a previously created data set is to be used or a new data set can be created interactively. The interactive input of data makes it possible for the user to work with the program without going through the tedium that occurs when running a batch program of typing up cards and making certain that everything is in the correct column. The ability to work

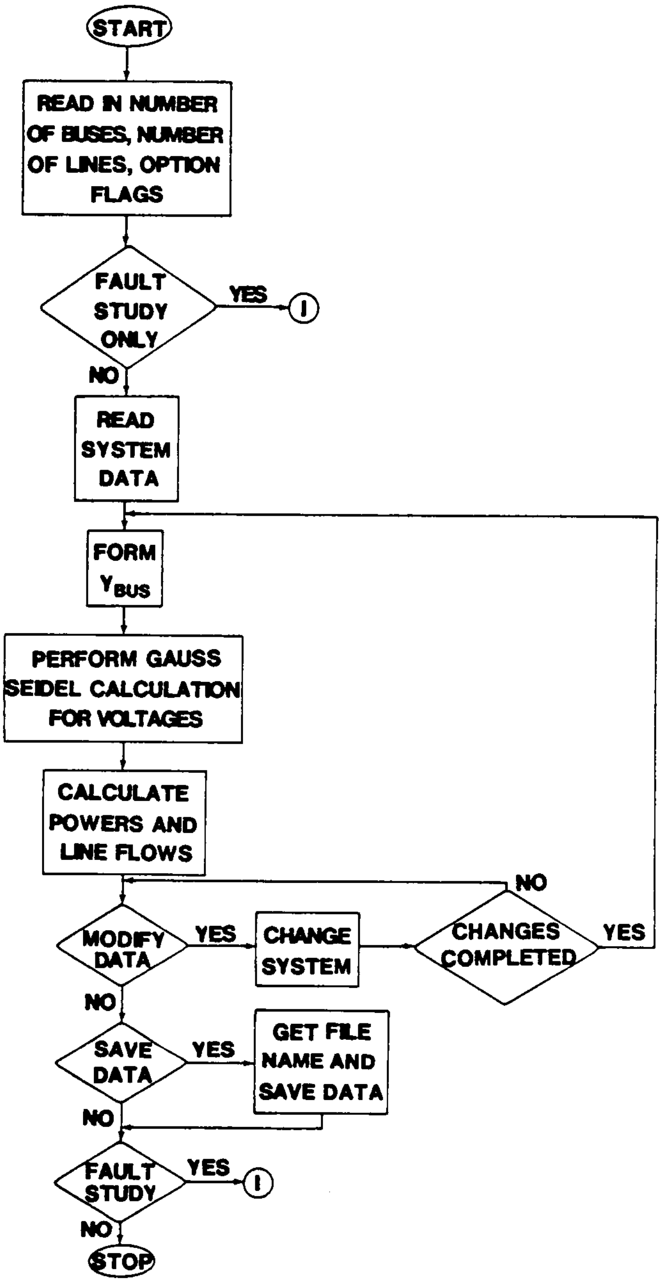


Figure 4.1. Load flow flowchart

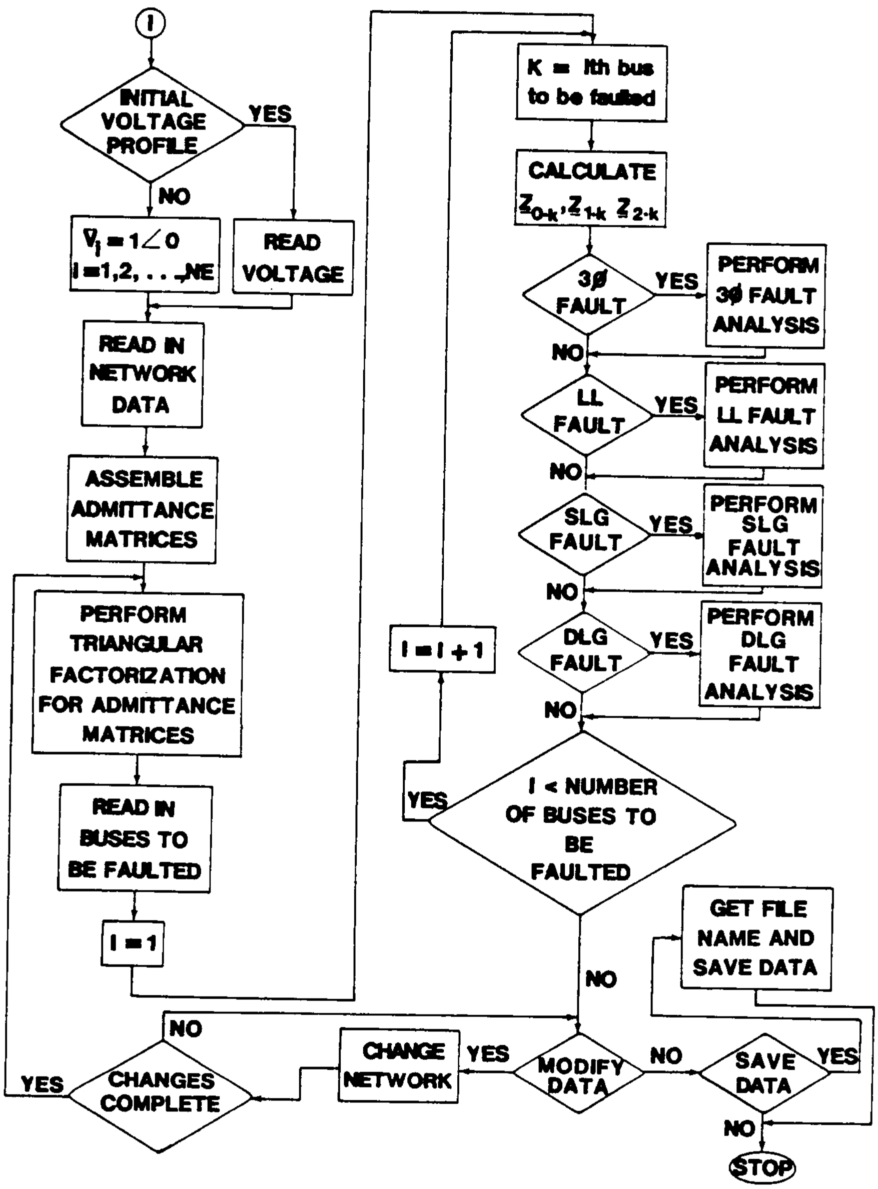


Figure 4.2. Fault study flowchart

with a previously created data set makes it possible for a system analyst to work with a nominal system configuration that has been previously input.

An important interactive feature of the code is that the user has the option of changing the system data after any analysis run is made. For example, after analyzing a given fault the user can do any or all the following before making another run:

1. Change the parameters of one or more system components.
2. Add one or more new components to the system.
3. Delete one or more old components from the system.

These features greatly enhance the user's capability to approach system design problems in an iterative way.

In order to be able to compare results with various design changes, the user has the option of creating an output file containing the system data and analytical results after each set of changes. When ready to exit the program, the user is given the chance to save the revised data set currnetly in memory.

The above options allow the user to make a wide variety of changes on the base line system rapidly and with minimum difficulty. A full description of the available options for the computer code can be found in the user's guide in Appendix A.

4.3 Example Case

The example case used in Chapter 2 can now be used to show how the interactive features of the fault analysis code may be used. Figure 4.3 shows the results of a single line-to-ground fault at bus 1 for the example case. The system planner running this case may wish

***** SINGLE LINE TO GROUND FAULT *****

FAULT OCCURS AT BUS 1

FAULT IMPEDANCE= 0.000 0.000

IFAULT= -0.0000 -8.1096 MAGNITUDE= 8.1096

SHORT CIRCUIT CAPACITY= 8.1096

POST FAULT BUS VOLTAGES

BUS NO.	PHASE A VA	PHASE A ANGLE A	PHASE B VB	PHASE B ANGLE B	PHASE C VC	PHASE C ANGLE C
1	0.0000	0.00	0.8942	-104.42	0.8942	104.42
2	0.5465	-0.00	0.9115	-108.18	0.9115	108.18
3	0.4978	-0.00	0.9078	-107.45	0.9078	107.45
4	0.7696	-40.52	0.7696	-139.48	1.0000	90.00

POST FAULT LINE CURRENTS

LINE	SB	EB	PHASE A IA	PHASE A ANGLE A	PHASE B IB	PHASE B ANGLE B	PHASE C IC	PHASE C ANGLE C
1	1	2	4.4074	90.00	1.6748	-90.00	1.6748	-90.00
1	2	1	4.4074	-90.00	1.6748	90.00	1.6748	90.00
2	1	3	3.9816	90.00	1.4924	-90.00	1.4924	-90.00
2	3	1	3.9816	-90.00	1.4924	90.00	1.4924	90.00
3	2	3	0.4258	-90.00	0.1824	90.00	0.1824	90.00
3	3	2	0.4258	90.00	0.1824	-90.00	0.1824	-90.00
10	3	4	3.5557	90.00	1.3100	-90.00	1.3100	-90.00
10	4	3	2.8092	-90.00	2.8092	90.00	0.0000	7.13

Figure 4.3. Example Case

to see if the effects of the fault on the system may be reduced by increasing the generator neutral impedance at bus 1. In this case, bus 1 was originally solidly wye grounded. A neutral reactance of 0.01 per unit is added to the generator at bus 1. This may be done interactively by following the procedure in Figure 4.4. In this case it is assumed that the original case has been run and the user is now making changes in the data. The results of this change are given in Figure 4.6.

The results in Figure 4.5 show that the fault current magnitude (6.8667) is less severe than the original case (8.1096) shown in Figure 4.3. If desired, additional changes can be made to see if the effects of the fault can be further decreased. Additional examples of how the interactive code can be used are given in the user's guide in Appendix A.

```
DO YOU WISH TO MAKE ANY CHANGES IN THE INPUT DATA ? YES

PLEASE SPECIFY BUS(GENERATOR), LINE(TRANSFORMER) ? BUS

PLEASE SPECIFY ADD, CHANGE OR DELETE ? CHANGE

BUS NUMBER = 1

BUS TYPE (B=BUS,G=GENERATOR) ? G

GENERATOR CONNECTION (Y=WYE,D=DELTA) ? Y

GENERATOR GROUNDING (G=GROUNDING, BLANK IF NONE) ? G

GENERATOR POSITIVE SEQUENCE IMPEDANCE
(REAL PART) = 0.

GENERATOR POSITIVE SEQUENCE IMPEDANCE
(IMAGINARY PART) = 0.1

GENERATOR NEGATIVE SEQUENCE IMPEDANCE
(REAL PART) = 0.

GENERATOR NEGATIVE SEQUENCE IMPEDANCE
(IMAGINARY PART) = 0.1

GENERATOR ZERO SEQUENCE IMPEDANCE
(REAL PART) = 0.

GENERATOR ZERO SEQUENCE IMPEDANCE
(IMAGINARY PART) = 0.02

GENERATOR NEUTRAL IMPEDANCE (REAL PART) = 0.

GENERATOR NEUTRAL IMPEDANCE (IMAGINARY PART) = 0.C1

DO YOU WISH TO MAKE ANY CHANGES IN THE INPUT DATA ? NO
```

Figure 4.4. Changing data interactively

***** SINGLE LINE TO GROUND FAULT *****

FAULT OCCURS AT BUS 1

FAULT IMPEDANCE= 0.000 0.000

IFAULT= -0.0000 -6.8667 MAGNITUDE= 6.8667

SHORT CIRCUIT CAPACITY= 6.8667

POST FAULT BUS VOLTAGES

	X----- PHASE QUANTITIES -----					X
BUS	X----- PHASE A -----X		X----- PHASE B -----X		X----- PHASE C -----X	
NO.	VA	ANGLE A	VB	ANGLE B	VC	ANGLE C
1	0.0000	180.0C	0.9618	-115.79	0.9618	115.79
2	0.6083	-0.00	0.9251	-11C.58	0.9251	110.58
3	0.5587	-0.00	0.9244	-11C.47	0.9244	110.47
4	0.8029	-38.52	0.8029	-141.48	1.0000	90.0C

POST FAULT LINE CURRENTS

			X----- PHASE QUANTITIES -----					X
			X---- PHASE A ---X		X----- PHASE B ----X		X---- PHASE C ----X	
LINE	SB	EB	IA	ANGLE A	IB	ANGLE B	IC	ANGLE C
1	1	2	4.0958	90.00	1.0542	-90.00	1.0542	-9C.00
1	2	1	4.0958	-90.00	1.0542	90.00	1.0542	90.00
2	1	3	3.7142	90.00	0.9208	-90.00	0.92C8	-9C.00
2	3	1	3.7142	-90.C0	C.92C8	90.00	0.92C8	90.00
3	2	3	0.3816	-90.C0	0.1334	90.00	0.1334	9C.00
3	3	2	0.3816	90.C0	0.1334	-90.00	0.1334	-9C.00
10	3	4	3.3327	90.00	0.7873	-90.00	0.7873	-9C.00
10	4	3	2.3787	-90.C0	2.3787	90.00	0.0000	6.34

Figure 4.5. Revised example case

CHAPTER 5

CONCLUSION

The primary purpose of this thesis was to develop an interactive shunt fault analysis package for small power systems. The computer code that has been developed gives satisfactory results for the analysis of the four main types of shunt faults. It is accurate and efficient and allows the user to make a wide variety of changes with ease. The version of the code in Appendix A will handle a system of 24 buses and 50 lines. The largest system that this code can handle will be dependent on the memory capacity and precision of the computer on which the code is implemented. The accuracy of the code has been verified on numerous homework and examination problems taken from power system courses at the University of Tennessee. The interactive features of the code make it possible for the package to be used for design projects where the designer's time can be spent working on the problem rather than waiting on a batch job to be returned.

In conclusion, the author offers the following suggestions for the improvement and further development of the shunt fault analysis code created for this thesis:

1. The effects of mutually coupled parallel components, such as lines, can be taken into consideration.
2. If it is desired to have the code handle larger power systems, optimal ordered triangular factorization could be used in solving the network problems. This technique requires less computer storage and execution time for large systems.

3. Include the magnetizing branch in the transformer model.
4. The code can be expanded to handle the occurance of simultaneous shunt faults.
5. The code can be made more user friendly and more tolerant of user errors. This can be done by improving the quality of the step by step explainations the code gives and by developing error recovery routines so that the user can continue working after the occurance of what is now a fatal error.
6. The code can be broken up into a number of independent seqments which can then be run on a small computer system.
7. A more accurate load model can be developed.

REFERENCES

REFERENCES

Anderson, P. M. Analysis of Faulted Power Systems. Ames, Iowa: Iowa State University Press, 1973.

Burden, R. L., et al. Numerical Analysis. Boston: Prindle, Weber & Schmidt, 1978.

Elgerd, O. I. Electric Energy Systems Theory. Second Edition. New York: McGraw Hill, 1982.

Fortesque, C. L. "Method of Symmetrical Coordinates Applied to the Solution of Polyphase Networks" Transactions of the AIEE, vol 37, pp 1027-1140, 1918.

Gross, C. A. Power System Analysis. New York: John B. Wiley & Sons, 1979.

Stagg, G. W., and A. E. El-Abiad. Computer Methods in Power System Analysis. New York: McGraw Hill, 1968.

Stevenson, W. D., Jr. Elements of Power System Analysis. Third Edition. New York: McGraw Hill, 1975

Final Report on "New Technology Integration," prepared by the University of Tennessee for the Oak Ridge National Laboratory, Oak Ridge, TN. February, 1982.

Final Report on "Load Management and Distribution Automation Studies," prepared by the University of Tennessee for the Oak Ridge National Laboratory, Oak Ridge, TN. June, 1980.

APPENDICES

APPENDIX A

University of Tennessee
Power Systems Analysis Package
User's Guide
DEC10 VERSION

By Dennis A. Reed

This program performs a load flow or fault study on a power system of up to 24 buses and 50 lines. All calculations done by this package are assumed to be done on per unit quantities.

To run the program the user must be in monitor mode on the DEC10 and type RUN LFANAL as shown below.

```
.RUN LFANAL
```

The program will respond with:

```
DO YOU WISH TO PERFORM A LOAD FLOW OR FAULT STUDY ?
```

At this point the user must refer to the proper section of this guide depending on whether they wish to perform a load flow or fault study. Before showing how to run the program for each case an example system will be given. It will then be shown how this system would be input to the program.

LOAD FLOW ANALYSIS

```
B1                                       B3
|                                        |
|               L3                       |
|----------------------------------------|
|                                        |
|                                        |
|   L1                          L2       |
|-----------          -------------------|
|          |          |                  |
|          |          |                  |
           |          |
        ----------------
              B2   |
                   |
                   V
```

For L1 Z = 0.001 + j0.01
For L2 Z = 0.0025 + j0.025
For L3 Z = 0.005 + j0.05

For B1 slack bus V=1.,PD=QD=0.
For B2 standard bus PD=12.,QD=3.,PG=QG=0.
For B3 standard bus PG=6.,QG=1.5,PD=QD=0.

FIGURE 1. Example system

To perform a load flow the user must give the following response.

DO YOU WISH TO PERFORM A LOAD FLOW OR FAULT STUDY ? LOAD FLOW

The program will respond with:

DO YOU WISH TO HAVE A OUTPUT FILE ?

Responding YES to this question will cause the program to create a disk file containing the load flows that are performed while the program is run. When finished the user may wish to print this file in order to have a hard copy of the results of his session. The following response will cause the program to create an output file called SMITH.DAT :

DO YOU WISH TO HAVE A OUTPUT FILE ? YES

INPUT FILE NAME SMITH

The next question asked by the program is:

DO YOU HAVE AN INPUT DATA FILE ?

In this case the response should be NO. If the user has a previously created data file it will be shown later how he may use it.

The program will now be ready to ask questions about the data to be input. Since this example case is 3 buses and 3 lines the questions and responses would be as follows.

INPUT THE NUMBER OF BUSES ? 3

INPUT THE NUMBER OF LINES ? 3

The next question will deal with the number of iterations to be performed by the program. In this case the number 500 is chosen. If this is not sufficient the program will give the option later of increasing the number.

INPUT THE NUMBER OF ITERATIONS ? 500

The program will now be ready to receive the data on the lines in the system. The line model for the system is a Pi model of the following form:

```
                 -------
                 |     |
-------------|  Z  |-------------
      |          |     |          |
      |          -------          |
   -------                     -------
   |     |                     |     |
   | Y/2 |                     | Y/2 |
   |     |                     |     |
   -------                     -------
      |                           |
      |                           |
---------------------------------
```

FIGURE 2. Line model

The data for line 1 is as follows:

LINE NUMBER = 1

STARTING BUS NUMBER = 1

ENDING BUS NUMBER = 2

SERIES IMPEDANCE (REAL PART) = 0.001

SERIES IMPEDANCE (IMAGINARY PART) = 0.01

SHUNT ADMITTANCE (REAL PART) = 0.

SHUNT ADMITTANCE (IMAGINARY PART) = 0.

The remaining lines of the system may be input the same way as the first line.

The next portion of the data to be input after the line data will be the bus data. This program assumes the existence of three different types of buses. They are as follows:

A type 1 bus is a standard bus, also called a load bus. The program will ask for the following information for each standard bus: power generated and power demanded.

A type 2 bus is a voltage control bus. A voltage control bus is a bus where a desired voltage magnitude must be maintained. This is done by setting a range of values for capacitor or inductor banks at the bus. That is to say, there is a maximum number of VARS that can be generated and a minimum that can be absorbed. The program will ask for the following information for each voltage control bus: voltage magnitude specified, real power generated, power demanded, maximum VARS, and minimum VARS.

A type 3 bus is the slack bus. The slack bus is the bus where any additional power to be generated by the system due to losses is added. There may only be one slack bus for any system. The program will ask for the following information for the slack bus: voltage specified and power demand.

For this example system the questions about the buses may be answered as follows.

THE NUMBER OF THE BUS = 1

THE TYPE OF BUS (1=LOAD,2=VC,3=SLACK) = 3

REAL POWER DEMAND = 0.

REACTIVE POWER DEMAND = 0.

THE BUS VOLTAGE = 1.

THE NUMBER OF THE BUS = 2

THE TYPE OF BUS (1=LOAD,2=VC,3=SLACK) = 1

REAL POWER GENERATION = 0.

REACTIVE POWER GENERATION = 0.

REAL POWER DEMAND = 12.

REACTIVE POWER DEMAND = 3.

THE NUMBER OF THE BUS = 3

THE TYPE OF BUS (1=LOAD,2=VC,3=SLACK) = 1

REAL POWER GENERATION = 6.

REACTIVE POWER GENERATION = 1.5

REAL POWER DEMAND = 0.

REACTIVE POWER DEMAND = 0.

At this point, when all bus data has been input, the program will echo out the input data to the terminal. When all data has been echoed and the load flow problem solved, the program will ask the following question:

THE NUMBER OF BUSES TO BE OUTPUT ?

The program will be asking for those buses at which the user wishes to see the load flow solution. If the user wishes to only view the results for certain buses, he may input that number and the program will then ask for the numbers of the buses to be output. For this example case, input 3 as shown.

THE NUMBER OF BUSES TO BE OUTPUT ? 3

CHANGING THE LOAD FLOW DATA

The solution of the load flow for the system will be, at this point echoed out to the terminal. When the solution has been output the program will respond with the following question:

DO YOU WISH TO MAKE ANY CHANGES IN THE INPUT DATA ?

At this point, if the user is unsatisfied with the solution to the load flow he has just seen he may modify the data and run a new load flow. The possible modifications are as follows: line add, line change, line delete, bus add, bus change, and bus delete. A user may make any number of modications to the data before running a new load flow. To run a new load flow after changing the system, respond NO to the question about changing the input data. If changes have previously been made, the program will perform a load flow on the new data.

When finished using the program the user will be asked the following questions on whether or not he wishes to save the data set he has created describing his power system. The following response will create a data set named TEST1 to store the data for future use:

DO YOU WISH THE INPUT DATA STORED IN A FILE ? YES

FILE NAME ? TEST1

At this point the program will be finished. If the user wishes he may print the output file from the program. This may be done by typing:

.PRINT SMITH.DAT

CHANGING THE NUMBER OF ITERATIONS

When running the program it is possible that the number of iterations is not sufficient to find a solution. If this occurs the program will print out the last 3 iterations for the voltages and then ask:

DO YOU WISH TO INCREASE THE NUMBER OF ITERATIONS ?

If the response is no, the program will proceed to ask about changes in input data. If the response is yes, the program will ask the following question:

THE NEW ITERATION TOTAL =

The program will then attempt to find the solution for the new maximum number of iterations.

RUNNING A SAVED DATA SET

A previously saved data set may be used by performing the following commands:

DO YOU HAVE AN INPUT DATA FILE ? YES

FILE NAME ? TEST1.DAT

This will result in the program reading the saved data set TEST1.DAT from the example case. See the section on changing the input data for information about saving a data set.

CONSTANT CHARGING ON A BUS

If you wish to have constant charging on a bus this may be done by adding a line. The program will assume that a line which has a starting bus which is part of the system and an ending bus which is numbered zero, is connected between the bus and ground and that the value of the impedance of that bus will be the desired amount of charging.

<u>FAULT ANALYSIS</u>

```
B1
|                                                  B3
|                                                  |
|                      L3                          |
|--------------------------------------------------|
|                                                  |
|                                                  |
|     L1                               L2          |
|-----------                 ---------------------|
|          |                 |                    |
|          |                 |                    |
           |                 |                    |
           ------------------
                     |      B2
                     |
                     |
                  Y  WWW   T4
                  D  MMM
                     |
                     |
                     |
                -----------
                  B4   |
                       |
                       V
```

For L1 Z1 = Z2 = 0.C01 + j0.01
        Z0 = 0.003 + j0.03
For L2 Z1 = Z2 = 0.0025 + j0.025
        Z0 = 0.01 + j0.1
For L3 Z1 = Z2 = 0.C05 + j0.05
        Z0 = 0.02 + j0.2
For T4 Z1 = Z2 = Z0 = 0.0 + j0.05

For B1 Z1 = 0.0 + j0.1, Z2 = 0.0 + j0.1, Z0 = 0.C + j0.4
        ZN = 0.0 + j0.01
For B2 Z1 = 0.0 + j0.1, Z2 = 0.0 + j0.1, Z0 = 0.C + j0.5
        ZN = 0.0 +j0.005
For B3 and B4 there are nc values because they are not generator buses.

Assume that the user wants tc check fcr all faults at bus 4.

FIGURE 3. Example system

To perform a fault study the user must give the fcllcwing response.

DO YOU WISH TO PERFORM A LOAD FLOW OR FAULT STUDY ? FAULT

The program will respond with:

DO YOU WISH TO HAVE A OUTPUT FILE ?

Responding YES to this question will cause the program to create a disk file containing the fault studies that are performed while the program is run. When finished the user may wish to print this file in order to have a hard copy of the results of his session. The following response will cause the program to create an output file called SMITH.DAT :

DO YOU WISH TO HAVE A OUTPUT FIIE ? YES

INPUT FILE NAME SMITH

The next question asked by the program is:

DO YOU HAVE AN INPUT DATA FILE ?

In this case the response should be NO. If the user has a previously created data file it will be shown later how he may use it.

The program will now be ready to ask questions about the data to be input. Since this example case is 4 buses and 4 lines the questions and responses would be as follows.

INPUT THE NUMBER OF BUSES ? 4

INPUT THE NUMBER OF LINES ? 4

The program will then ask.

DO YOU WISH TO OUTPUT THE SEQUENCE VOLTAGES AND CURRENTS ?

Answering YES to this question will result in the sequence quantities being output to terminal and disk when they are calculated. Answering NO to this question will result in only the phase quantities being output to a device. The next question will deal with whether or not the user wishes to input a nonunity voltage profile for the fault study.

DO YOU WISH TO INPUT A NONUNITY VOLTAGE PROFILE ?

If the user answers yes the next questions will be asked for all buses of the system.

THE NUMBER OF THE BUS =

THE VOLTAGE MAGNITUDE =

THE VOLTAGE PHASE ANGLE =

The program will now be ready to receive the data on the lines in the system. The line model for the system is a Pi model of the form shown in figure C.2. The data for the first line will be input as follows.

LINE/TRANSFORMER NUMBER = 1

```
LINE TYPE (L=LINE,T=TRANSFORMER) ? L

STARTING BUS NUMBER = 1

ENDING BUS NUMBER = 2

POSITIVE/NEGATIVE SEQUENCE SERIES IMPEDANCE
(REAL PART) = 0.001

POSITIVE/NEGATIVE SEQUENCE SERIES IMPEDANCE
(IMAGINARY PART) = 0.01

POSITIVE/NEGATIVE SEQUENCE SHUNT ADMITTANCE
(REAL PART) = 0.0

POSITIVE/NEGATIVE SEQUENCE SHUNT ADMITTANCE
(IMAGINARY PART) = 0.0

ZERO SEQUENCE SERIES IMPEDANCE (REAL PART) = 0.003

ZERO SEQUENCE SERIES IMPEDANCE
(IMAGINARY PART) = 0.03

ZERO SEQUENCE SHUNT ADMITTANCE (REAL PART) = 0.0

ZERO SEQUENCE SHUNT ADMITTANCE
(IMAGINARY PART) = 0.0
```

The remaining lines of the system may be input the same way as the first line. The only difference will occur when T4 is input and the results of that will be as follows.

```
LINE/TRANSFORMER NUMBER = 4

LINE TYPE (L=LINE,T=TRANSFORMER) ? T

HIGH VOLTAGE BUS NUMBER = 2

HIGH VOLTAGE SIDE CONNECTION (Y=WYE,D=DELTA) ? Y

HIGH VOLTAGE SIDE GROUND (G=GROUND, BLANK IF UNGROUNDED) ? G

LOW VOLTAGE BUS NUMBER = 4

LOW VOLTAGE SIDE CONNECTION (Y=WYE,D=DELTA) ? D

LOW VOLTAGE SIDE GROUND (G=GROUND, BLANK IF UNGROUNDED) ?

POSITIVE/NEGATIVE SEQUENCE SERIES IMPEDANCE
(REAL PART) = 0.0

POSITIVE/NEGATIVE SEQUENCE SERIES IMPEDANCE
(IMAGINARY PART) = 0.05

ZERO SEQUENCE SERIES IMPEDANCE (REAL PART) = 0.0
```

ZERO SEQUENCE SERIES IMPEDANCE
(IMAGINARY PART) = 0.05

HIGH VOLTAGE NEUTRAL IMPEDANCE (REAL PART) = 0.0

HIGH VOLTAGE NEUTRAL IMPEDANCE
(IMAGINARY PART) = 0.0

LOW VOLTAGE NEUTRAL IMPEDANCE (REAL PART) = 0.0

LOW VOLTAGE NEUTRAL IMPEDANCE
(IMAGINARY PART) = 0.0

The next portion of the data to be input is the bus and generator data. If a bus does not contain a generator no further data is required for that bus. If the bus is a generator bus it is necessary to input the sequence impedances for the generator, how the generator is connected and whether it is grounded. For the example system, these quantities would be input as follows.

BUS NUMBER = 1

BUS TYPE (B=BUS,G=GENERATOR) ? G

GENERATOR CONNECTION (Y=WYE,D=DELTA) ? Y

GENERATOR GROUNDING, G IF GROUND, BLANK IF NONE G

GENERATOR POSITIVE SEQUENCE IMPEDANCE
(REAL PART) = 0.0

GENERATOR POSITIVE SEQUENCE IMPEDANCE
(IMAGINARY PART) = 0.1

GENERATOR NEGATIVE SEQUENCE IMPEDANCE
(REAL PART) = 0.0

GENERATOR NEGATIVE SEQUENCE IMPEDANCE
(IMAGINARY PART) = 0.1

GENERATOR ZERO SEQUENCE IMPEDANCE
(REAL PART) = 0.0

GENERATOR ZERO SEQUENCE IMPEDANCE
(IMAGINARY PART) = 0.4

GENERATOR NEUTRAL IMPEDANCE (REAL PART) = 0.0

GENERATOR NEUTRAL IMPEDANCE (IMAGINARY PART) = 0.01

BUS NUMBER = 2

BUS TYPE (B=BUS,G=GENERATOR) ? G

GENERATOR CONNECTION (Y=WYE,D=DELTA) ? Y

```
GENERATOR GROUNDING, G IF GROUND, BLANK IF NONE G

GENERATOR POSITIVE SEQUENCE IMPEDANCE
(REAL PART) = 0.0

GENERATOR POSITIVE SEQUENCE IMPEDANCE
(IMAGINARY PART) = 0.1

GENERATOR NEGATIVE SEQUENCE IMPEDANCE
(REAL PART) = 0.0

GENERATOR NEGATIVE SEQUENCE IMPEDANCE
(IMAGINARY PART) = 0.1

GENERATOR ZERO SEQUENCE IMPEDANCE
(REAL PART) = 0.0

GENERATOR ZERO SEQUENCE IMPEDANCE
(IMAGINARY PART) = 0.5

GENERATOR NEUTRAL IMPEDANCE (REAL PART) = 0.0

GENERATOR NEUTRAL IMPEDANCE (IMAGINARY PART) = 0.01

BUS NUMBER = 3

BUS TYPE (B=BUS,G=GENERATOR) ? B

BUS NUMBER = 4

BUS TYPE (B=BUS,G=GENERATOR) ? B
```

At this point the program will now echo out the input data to the terminal. When all the data has been echoed the program will respond with the following question.

```
THE NUMBER OF BUSES TO BE FAULTED =
```

For the example case the response to this question will be 1. The computer will then ask the numbers of the buses to be faulted and the shunt faults to be applied at each bus. For the example case the next series of questions will be answered as follows.

```
THE NUMBER OF BUSES TO BE FAULTED = 1

THE NUMBER OF THE BUS TO BE FAULTED = 4

DO YOU WISH TO APPLY A THREE PHASE FAULT (Y/N) ? Y

DO YOU WISH TO APPLY A SINGLE LINE TO GROUND FAULT (Y/N) ? Y

DO YOU WISH TO APPLY A LINE TO LINE FAULT (Y/N) ? Y

DO YOU WISH TO APPLY A DOUBLE LINE TO GROUND FAULT (Y/N) ? Y
```

FAULT IMPEDANCE (REAL PART) = 0.0

FAULT IMPEDANCE (IMAGINARY PART) = 0.0

CHANGING THE FAULT ANALYSIS DATA

The solution of the fault study for the system will be, at this point echoed out to the terminal. When the solution has been output the program will respond with the following question:

DO YOU WISH TO MAKE ANY CHANGES IN THE INPUT DATA ?

At this point, if the user is unsatisfied with the solution to the fault study he has just seen he may modify the data and run a new fault study. The possible modifications are as follows: line add, line change, line delete, bus add, bus change, and bus delete. A user may make any number of modications to the data before running a new fault study. To run a new fault study after changing the system, respond no to the question about changing the input data. If changes have previously been made, the program will perform a fault study on the new data. Before the program performs a new fault study it will also ask what buses are to be faulted for the revised system and what faults are to be applied to the revised system.

When finished using the program the user will be asked the following questions on whether or not he wishes to save the data set he has created describing his power system. The following response will create a data set named TEST1 to store the data for future use:

DO YOU WISH THE INPUT DATA STORED IN A FILE ? YES

FILE NAME ? TEST1

At this point the program will be finished. If the user wishes he may print the output file from the program. This may be done by typing:

.PRINT SMITH.DAT

PROGRAM USAGE NOTES

This program has certain basic limitations which should be kept in mind by the user.

If an error is made while inputing data there is no immediate means of correction. The user must finish inputing whatever sequence of data he is doing and then make the correction when asked if there are any changes to be made in the input data.

The user should also keep in mind that there are certain errors which are fatal for the program. They are as follows: inputing a character when a number is asked for, inputing an unreferenced bus, inputing real number when integer is asked, inputing a negative or zero value for a bus number.

When one of the previous errors occurs the program will abnormally terminate and all data input to the program will be lost. The program will need to be run again and all data retyped.

While using this program it is strongly suggested that the user be careful what he types. If this program is used for a long session at the terminal the user should keep in mind that a fatal mistake will destroy all his results previous to that point.

APPENDIX B

```
C***********************************************************************
C
C       UNIVERSITY OF TENNESSEE POWER SYSTEM ANALYSIS PACKAGE
C                       VERSION 1.0
C
C                       WRITTEN BY
C
C                     DENNIS A. REED
C
C                     SEPT. 16, 1981
C
C***********************************************************************
C
C       THIS PROGRAM PERFORMS A LOAD FLOW STUDY AND A FAULT STUDY
C       FOR A SYSTEM OF UP TO 24 BUSES AND 50 LINES. THE GAUSS-SEIDEL
C       ITERATIVE METHOD IS USED TO SOLVE FOR THE LOAD FLOW
C
C***********************************************************************
C
C       THE DATA INPUT TO THE PROGRAM
C
C***********************************************************************
C
C       NB - THE NUMBER OF BUSES IN THE NETWORK
C
C       NL - THE NUMBER OF LINES IN THE NETWORK
C
C       ALPHA - THE ACCELERATION FACTOR FOR THE GAUSS-SEIDEL ITERATION
C               TECHNIQUE
C
C       IVG - THIS VALUE INDICATES IF THE INITIAL VOLTAGE GUESS IS
C             A UNITY VOLTAGE PROFILE OR NON UNITY
C
C       IYBUS - THIS IS SET TO ONE IF THE YBUS MATRIX IS TO BE PRINTED
C
C       ITER - THE NUMBER OF ITERATIONS TO BE DONE
C
C       EPS - THE CONVERGENCE FACTOR FOR THE SYSTEM
C
C***********************************************************************
C
C       THE FOLLOWING ITEMS ARE THE VARIABLES FOR THE LINE DATA
C
C***********************************************************************
C
C       LINE - LINE NUMBER FOR A TRANSMISSION LINE
C
C       SB - STARTING BUS FOR LINE
C
C       EB - ENDING BUS FOR LINE
C
C       LENGTH - THE LINE LENGTH. DEFAULT OF 1.0
C
C       SHTY - THE SHUNT ADMITTANCE OF A LINE
C
C       SERZ - THE SERIES IMPEDANCE OF THE LINE
C
C***********************************************************************
C
C       THESE VARIABLES ARE THE BUS DATA FOR THE PROGRAM
```

```
C
C***************************************************************************
C
C       IBUS - THE BUS NUMBER
C
C       IBTYP - THE BUS TYPE  3 - SLACK
C                             2 - VOLTAGE CONTROL
C                             1 - STANDARD
C
C       PG - REAL POWER GENERATED
C
C       QG - REACTIVE POWER GENERATED
C
C       PD - REAL POWER DEMAND
C
C       QD - REACTIVE POWER DEMAND
C
C       P - REAL POWER GENERATED MINUS REAL POWER DEMAND
C
C       Q - REACTIVE PWER GENERATED MINUS REACTIVE POWER DEMAND
C
C       VR - BUS VOLTAGE SPECIFICATION
C
C       QMIN - MINIMUM VARS FOR VOLTAGE CONTROL BUS
C
C       QMAX - MAXIMUM VARS FOR VOLTAGE CONTROL BUS
C
C       V - COMPLEX VOLTAGE FOR NON UNIT VOLTAGE PROFILE
C
C
C       Y - THE BUS ADMITTANCE MATRIX
C
C***************************************************************************
C
C       SUBROUTINES CALLED:
C
C          ERROR,BSPEC,BOUT,LINOUT,GSIFS,IDFISL,FAULT
C
        DOUBLE PRECISION   FNAME
        DOUBLE PRECISION   GNAME
        DOUBLE PRECISION   HNAME
        LOGICAL   BAL
        LOGICAL   TPRINT
        COMPLEX   SHTY(50)
        COMPLEX   Y(24,24)
        COMPLEX   SERZ(50)
        COMPLEX   V(24)
        COMPLEX   ZLINE
        COMPLEX   YLINE
        REAL   OPG(24)
        REAL   OQG(24)
        REAL   P(24)
        REAL   Q(24)
        REAL   QMAX(24)
        REAL   QMIN(24)
        REAL   VSPEC(24)
        REAL   PG(24)
        REAL   QG(24)
        REAL   PD(24)
        REAL   QD(24)
```

```
      INTEGER   SB(50)
      INTEGER   EB(50)
      INTEGER   LINE(50)
      INTEGER   IBUS(24)
      INTEGER   IFTYP(24)
      COMMON /INOU/ KTERM
      COMMON /OUTPUT/ ANSOUT,TPRINT,ANSIN
C
C     THE FOLLOWING DATA STATEMENTS MAY BE SYSTEM DEPENDENT
C
      DATA   FLT /'FAULT'/
      DATA   YES   /'YES  '/
      DATA   BUS   /'BUS  '/
      DATA   ADD   /'ADD  '/
      DATA   CHANGE/'CHANG'/
      DATA   DELETE/'DELET'/
C
C     SET TERMINAL INPUT DEVICE TO UNIT 5
C
      KTERM=5
      TPRINT=.TRUE.
C
C     DECIDE IF THIS IS LOADFLOW OR FAULT STUDY
C
      WRITE(KTERM,784)
      READ(KTERM,824)START
      IF(START.EQ.FLT)IFS=1
C
C     THIS SECTION CHECKS TO SEE HOW THE PERSON WANTS TO
C     DO HIS I/O
C
      WRITE(KTERM,804)
      READ(KTERM,824)ANSOUT
      IF(ANSOUT.NE.YES)GO TO 20
      WRITE(KTERM,844)
C
C     READ IN OUTPUT FILE NAME
C
      READ(KTERM,884)GNAME
      OPEN(UNIT=6, DEVICE='DSK', ACCESS='SEQOUT', MODE='ASCII', FILE=
     1 GNAME)
20    WRITE(KTERM,864)
      READ(KTERM,824)ANSIN
      IF(ANSIN.NE.YES)GO TO 40
      WRITE(KTERM,844)
C
C     READ IN INPUT FILE NAME
C
      READ(KTERM,884)HNAME
      OPEN(UNIT=8, DEVICE='DSK', ACCESS='SEQIN', MODE='ASCII', FILE=
     1 HNAME)
      GO TO 60
40    IF(START.EQ.FLT)GO TO 680
C
C     THIS SECTION REQUESTS THE NUMBER OF BUSES, NUMBER OF LINES,
C     NUMBER OF ITERATIONS, AND ACCELERATION FACTOR.
C
      WRITE(KTERM,904)
      READ(KTERM,*)NB
      WRITE(KTERM,924)
```

```
      READ(KTERM,*) NL
      WRITE(KTERM,944)
      READ(KTERM,*) ITER
      ALPHA=1.0
      EPS=0.00001
      GO TO 80
60    READ(8,964,ERR=9003) NB,NL,ALPHA,IVG,IYBUS,IFS,IFVP,ISQ
      IF(NB.GT. 24)CALL ERROR(12,NB,&99999)
      IF(NL.GT. 50)CALL ERROR(13,NL,&99999)
      IF(IFS.NE. 0)GO TO 680
      IF (ALPHA.LT.1.E-20)  ALPHA=1.
C
C     NUMBER OF ITERATICNS (500 IF NOT USED) AND CONVERGENCE CRITERIA
C     (.00001 IF NOT USED )
C
      READ(8,984,ERR=9004) ITER,EPS
      IF(ITER.EQ.0) ITER=500
      IF (EPS.LT.1.E-20) EPS=.00001
80    DO 120 I = 1,NB
         DO 100 J = 1,NB
            Y(I,J) = CMPLX(0.0,0.0)
100         CONTINUE
120      CONTINUE
      DO 180 I = 1,NL
C
C     READ LINE NUMBER, STARTING BUS, END BUS, LENGTH, SHUNT
C     ADMITTANCE AND SERIES IMPEDANCE IN P.U. - PER UNIT LENGTH.
C
         IF(ANSIN.EQ.YES)GO TO 140
C
C     THE LINE DATA IS INPUT
C
         WRITE(KTERM,1004)
         READ(KTERM,*)LINE(I)
         WRITE(KTERM,1024)
         READ(KTERM,*)SB(I)
         WRITE(KTERM,1044)
         READ(KTERM,*)EB(I)
         WRITE(KTERM,1084)
         READ(KTERM,*)RES
         WRITE(KTERM,1104)
         READ(KTERM,*)XIMAG
         WRITE(KTERM,1114)
         READ(KTERM,*)GIN
         WRITE(KTERM,1124)
         READ(KTERM,*)YIMAG
         SERZ(I)=CMPLX(RES,XIMAG)
         SHTY(I)=CMPLX(GIN,YIMAG)
         GO TO 180
C
C     READ IN THE LINE DATA FROM INPUT FILE
C
140      READ(8,1144,ERR=9005) LINE(I),SB(I),EB(I),SHTY(I),SERZ(I)
180      CONTINUE
C
C     READ IN SPECIFIED BUS DATA, REAL POWER, REACTIVE POWER,
C     REFERENCE VOLTAGE V1, VOLTAGE CONTROL BUS MAGNITUDES AND
C     REACTIVE POWER LIMITS.
C
      CALL BSPEC(NB,P,Q,VSPEC,QMIN,QMAX,IBTYP,IBUS,PD,PG,QD,QG,
```

```
     1 &99999)
       CALL DOUT(NB,VSPEC,QMIN,QMAX,IBTYP,IBUS,PD,PG,QD,QG)
C
C      ASSEMBLE THE BUS ADMITTANCE MATRIX.
C
220    DO 260 I=1,NB
          DO 240 J=1,NB
240          Y(I,J)=CMPLX(0.0,0.0)
260       CONTINUE
       CALL YASSEM(NB,NL,IBUS,SB,EB,SERZ,SHTY,Y,&99999)
C
C      WRITE OUT LINE DATA, SHUNT ADMITTANCE, AND THE Y BUS
C
       CALL LINOUT(NL,SB,EB,LINE,SERZ,SHTY)
       IF(IYBUS.EQ. 0) GO TO 440
       IF(ANSOUT.NE.YES) GO TO 440
       WRITE(6,1164)
       DO 420 I = 1,NB
          WRITE(6,1184)  (Y(I,J),J=1,NB)
420       CONTINUE
C
C      PERFORM THE LOAD FLOW CALCULATIONS
C
440    CALL GSLFS(NB,IVG,ALPHA,ITER,EPS,N,V,Y,P,Q,VSPEC,
     1 QMIN,QMAX,IBTYP,IBUS,&500)
       DO 460 I=1,NB
          OPG(I)=PG(I)
460       OQG(I)=QG(I)
       CALL LDFLSL(NL,NB,N,V,Y,SB,EB,LINE,SERZ,SHTY,P,Q,IBTYP,
     1 IBUS,PD,PG,QD,QG)
       DO 480 I=1,NB
          PG(I)=OPG(I)
          QG(I)=OQG(I)
          P(I)=PG(I)-PD(I)
480       Q(I)=QG(I)-QD(I)
C
C      THIS SECTION CHECKS TO SEE IF YOU WISH TO CHANGE
C      THE ORIGINAL DATA SET
C
       BAL=.FALSE.
       ICHECK=1
500    WRITE(KTERM,1204)
       READ(KTERM,824) ANSCH
C
C      IF ANSCH.NE. YES AND ICHECK NE 0 THEN THERE IS NO
C      NEED TO PERFORM A NEW LOADFLOW GO TO FAULT ROUTINE
C
       IF((ANSCH.NE.YES).AND.(ICHECK.NE.0)) GO TO 615
C
C      IF NEW LOADFLOW TO BE DONE GO PRINT OUT BUS AND LINE DATA
C
       IF(ANSCH.NE.YES)GO TO 600
       ICHECK=0
       TPRINT=.FALSE.
C
C      SPECIFY BUS OR LINE
C
       WRITE(KTERM,1224)
       READ(KTERM,824) ANSBL
C
```

```
C         ADD CHANGE OR DELETE
C
          WRITE(KTERM,1244)
          READ (KTERM,824) ANSACD
          IF(ANSBL.NE.BUS)GO TO 560
C
C         THIS SECTION GETS THE NEW DATA FOR THE CHANGES IN THE BUS DATA.
C
          WRITE(KTERM,1264)
          READ (KTERM,*) IBUSN
          IF(ANSACD.NE.DELETE)GO TO 520
          CALL BDEL(IBUSN,NB,NL,P,Q,VSPEC,QMIN,QMAX,IBTYP,IBUS,PD,PG,
     1    QD,QG,BAL,SE,EB,LINE,SHTY,SERZ,Y)
          GO TO 500
520       PGN=0.0
          QGN=0.0
          PDN=0.0
          QDN=0.0
          WRITE(KTERM,1284)
          READ (KTERM,*) IBTYPN
          WRITE(KTERM,1304)
          READ (KTERM,*) PGN
          WRITE(KTERM,1324)
          READ (KTERM,*) QGN
          WRITE(KTERM,1344)
          READ (KTERM,*) PDN
          WRITE(KTERM,1364)
          READ (KTERM,*) QDN
          VRN=0.0
          QMNN =0.0
          QMXN=0.0
          IF(IBTYPN.EQ.1)GO TO 540
          WRITE(KTERM,1384)
          READ (KTERM,*) VRN
          IF(IBTYPN.EQ.3)GO TO 540
          WRITE(KTERM,1404)
          READ (KTERM,*) QMNN
          WRITE(KTERM,1424)
          READ (KTERM,*) QMXN
540       IF(ANSACD.EQ.ADD)  CALL BADD(IBUSN,IBTYPN,PGN,QGN,PDN,QDN,VRN,
     1    QMNN,QMXN,P,Q,VSPEC,QMIN,QMAX,IBTYP,IBUS,PD,PG,QD,QG,NB,BAL)
          IF(ANSACD.EQ.CHANGE)  CALL BCHNG(IBUSN,IBTYPN,PGN,QGN,PDN,QDN,
     1    VRN,QMNN,QMXN,P,Q,VSPEC,QMIN,QMAX,IBTYP,IBUS,PD,PG,QD,QG,NB)
          GO TO 500
C
C         THIS SECTION RECEIVES THE DATA FOR CHANGES IN THE LINE DATA
C
560       WRITE(KTERM,1004)
          READ (KTERM,*) LINEN
          WRITE(KTERM,1024)
          READ (KTERM,*) ISB
          WRITE(KTERM,1044)
          READ (KTERM,*) IEB
          IF(ANSACD.NE.DELETE)GO TO 580
          CALL LDEL(LINEN,ISB,IEB,NB,NL,IBUS,SB,EB,LINE,SHTY,SERZ)
          GO TO 500
580       WRITE(KTERM,1084)
          READ (KTERM,*) RES
          WRITE(KTERM,1104)
          READ (KTERM,*) XIMAG
```

```
      ZLINE=CMPLX(RES,XIMAG)
      WRITE(KTERM,1124)
      READ(KTERM,*)YIMAG
      YLINE=CMPLX(0.,YIMAG)
      IF(ANSACD.EQ.ADD) CALL LADD(LINEN,ISB,IEB,ZLINE,YLINE,
     1 NB,NL,IBUS,SB,EB,LINE,SHTY,SERZ,Y)
      IF(ANSACD.EQ.CHANGE) CALL LCHNG(LINEN,ISB,IEB,ZLINE,YLINE,
     1 NB,NL,IBUS,LINE,SHTY,SERZ,Y)
      GO TO 500
600   CALL BOUT(NB,VSPEC,QMIN,QMAX,IBTYP,IBUS,PD,PG,QD,QG)
      IF(BAL)GO TO 220
      CALL LINOUT(NL,SB,EB,LINE,SERZ,SHTY)
      GO TO 440
C
C     THIS SECTION CHECKS TO SEE IF THE PROGRAMMER WISHES
C     TO SAVE THE NEW DATA SET HE HAS CREATED
C
615   WRITE(KTERM,1444)
      READ(KTERM,824)ANS
      IF(ANS.NE.YES)GO TO 660
      WRITE(KTERM,844)
      READ(KTERM,884)FNAME
      OPEN(UNIT=7, DEVICE='DSK', ACCESS='SEQOUT', MODE='ASCII', FILE=
     1 FNAME)
      WRITE(7,964)NB,NL,ALPHA,IVG,IYBUS,IFS,IFVP,ISQ
      WRITE(7,984)ITER,EPS
      D=0.0
      DO 640 I=1,NL
         WRITE(7,1144)LINE(I),SB(I),EB(I),SHTY(I),SERZ(I)
640      CONTINUE
      WRITE(7,1464)(IBUS(I),IBTYP(I),PG(I),QG(I),PD(I),
     1 QD(I),VSPEC(I),QMIN(I),QMAX(I),I=1,NB)
660   WRITE(KTERM,1434)
      READ(KTERM,824)ANSF
      IF(ANSF.NE.YES)GO TO 700
680   CALL FAULT(NB,NL,IFS,IFVP,ISQ,V,Y)
700   CONTINUE
      CLOSE(UNIT=5)
      CLOSE(UNIT=6)
      CLOSE(UNIT=7)
      CLOSE(UNIT=8)
      STOP
9003  CALL ERROR(3,0,&99999)
9004  CALL ERROR(4,0,&99999)
9005  CALL ERROR(5,0,&99999)
99999 STOP
C
784   FORMAT(' DO YOU WISH TO PERFORM A LOAD FLOW OR A FAULT STUDY ? '
     1        ,$)
804   FORMAT(' DO YOU WISH TO HAVE A OUTPUT FILE ? ',$)
824   FORMAT(A5)
844   FORMAT(' INPUT FILE NAME ',$)
864   FORMAT(' DO YOU HAVE AN INPUT DATA FILE ? ',$)
884   FORMAT(A10)
904   FORMAT(' INPUT THE NUMBER OF BUSES ? ',$)
924   FORMAT(' INPUT THE NUMBER OF LINES ? ',$)
944   FORMAT(' INPUT THE NUMBER OF ITERATIONS ? ',$)
964   FORMAT(2I5,F5.2,I1,1X,4I1)
984   FORMAT(I5,G10.3)
1004  FORMAT(' INPUT LINE NUMBER ',$)
```

```
1024      FORMAT(' INPUT STARTING BUS NUMBER ',$)
1044      FORMAT(' INPUT ENDING BUS NUMBER ',$)
1084      FORMAT(' INPUT IMPEDANCE (REAL PART) ',$)
1104      FORMAT(' INPUT IMPEDANCE (IMAGINARY PART) ',$)
1114      FORMAT(' INPUT ADMITTANCE (REAL PART)',$)
1124      FORMAT(' INPUT ADMITTANCE (IMAGINARY PART) ',$)
1144      FORMAT(3I5,5X,4F10.4)
1164      FORMAT(1H1,'    Y-BUS MATRIX FOR LOADFLOW CALCULATIONS'/)
1184      FORMAT(2(4(F9.4,1X,F9.4,3X)/))
1204      FORMAT(' DO YOU WISH TO MAKE ANY CHANGES IN THE INPUT DATA ? ',
     1    $)
1224      FORMAT(' PLEASE SPECIFY BUS OR LINE ',$)
1244      FORMAT(' PLEASE SPECIFY ADD, CHANGE, OR DELETE ',$)
1264      FORMAT(' INPUT THE BUS NUMBER ',$)
1284      FORMAT(' INPUT THE BUS TYPE ',$)
1304      FORMAT(' INPUT THE REAL POWER GENERATION ',$)
1324      FORMAT(' INPUT THE IMAGINARY POWER GENERATION ',$)
1344      FORMAT(' INPUT REAL POWER DEMAND ',$)
1364      FORMAT(' INPUT IMAGINARY POWER DEMAND ',$)
1384      FORMAT(' INPUT THE BUS VOLTAGE ',$)
1404      FORMAT(' INPUT Q MINIMIM FOR CONTROL BUS ',$)
1424      FORMAT(' INPUT Q MAXIMUM FOR CONTROL BUS ',$)
1434      FORMAT('DO YOU WISH TO PERFORM A FAULT STUDY (YES/NO) ? ',$)
1444      FORMAT(' DO YOU WISH THE INPUT DATA STORED IN A FILE ? ',$)
1464      FORMAT(I3,I2,7F10.3)
C
          END
          SUBROUTINE BADD(IBUSN,IBTYPN,PGN,QGN,PDN,QDN,VRN,QMNN,
     1    QMXN,P,Q,VR,QMN,QMX,IBTYP,IBUS,PD,PG,QD,QG,NB,BCHNG)
C*************************************************************************
C
C         THE SUBROUTINE BADD ADDS A NEW BUS TO THE SYSTEM. THE ROUTINE
C         ASSUMES THAT LINES CONNECTING THIS BUS WILL BE ADDED AFTER THE
C         BUS
C
C*************************************************************************
C
C         ARGUMENTS:
C
C         IBUSN - THE NEW BUS NUMBER
C
C         IBTYPN - THE NEW BUS TYPE
C
C         PGN - THE NEW REAL POWER GENERATED
C
C         QGN - THE NEW REACTIVE POWER GENERATED
C
C         PDN - THE NEW REAL POWER DEMAND
C
C         QDN - THE NEW REACTIVE POWER DEMAND
C
C         VRN - THE NEW BUS VOLTAGE SPECIFICATION
C
C         QMNN - THE NEW MINIMUM VARS FOR VOLTAGE CONTROL BUS
C
C         QMXN - THE NEW MAXIMUM VARS FOR VOLTAGE CONTROL BUS
C
C
C         NB - THE NUMBER OF BUSES IN THE SYSTEM
C
```

```
C       BCHNG - THIS IS A LOGICAL VARIABLE WHICH INDICATES IF Y BUS
C               SHOULD BE RECALCULATED
C
C       IBUS - THE BUS NUMBER
C
C       IBTYP - THE BUS TYPE  3 - SLACK
C                             2 - VOLTAGE CONTROL
C                             1 - STANDARD
C
C       PG - REAL POWER GENERATED
C
C       QG - REACTIVE POWER GENERATED
C
C       PD - REAL POWER DEMAND
C
C       QD - REACTIVE POWER DEMAND
C
C       P - REAL POWER GENERATED MINUS REAL POWER DEMAND
C
C       Q - REACTIVE POWER GENERATED MINUS REACTIVE POWER DEMAND
C
C       VR - BUS VOLTAGE SPECIFICATION
C
C       QMIN - MINIMUM VARS FOR VOLTAGE CONTROL BUS
C
C       QMAX - MAXIMUM VARS FOR VOLTAGE CONTROL BUS
C
C**********************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        LOGICAL   BCHNG
        INTEGER   IBUS(24)
        INTEGER   IBTYP(24)
        REAL   P(24)
        REAL   Q(24)
        REAL   VR(24)
        REAL   QMN(24)
        REAL   QMX(24)
        REAL   PD(24)
        REAL   QD(24)
        REAL   PG(24)
        REAL   QG(24)
        COMMON /INOU/ KTERM
C
C       FIND THE POSITION OF THE NEW BUS IN THE ARRAY
C
        DO 20 I=1,NB
           IF(IBUSN.LT.IBUS(I)) GO TO 40
20         CONTINUE
40      NB=NB+1
        IF(I.EQ.NB) GO TO 80
C
C       MOVE THE BUSES UP ONE POSITION THAT ARE LOCATED AFTER
C       WHERE THE NEW BUS SHOULD GO.
C
        N=NB-I
        DO 60 M=1,N
```

```
          J=NB-M+1
          IBUS(J)=IBUS(J-1)
          IBTYP(J)=IBTYP(J-1)
          PD(J)=PD(J-1)
          QD(J)=QD(J-1)
          PG(J)=PG(J-1)
          QG(J)=QG(J-1)
          P(J)=P(J-1)
          Q(J)=Q(J-1)
          VR(J)=VR(J-1)
          QMN(J)=QMN(J-1)
60        QMX(J)=QMX(J-1)
C
C       INSERT THE VALUES FOR THE NEW BUS IN THE ARRAY
C
80      IBUS(I)=IBUSN
        IBTYP(I)=IBTYPN
        PG(I)=PGN
        QG(I)=QGN
        PD(I)=PDN
        QD(I)=QDN
        P(I)=PG(I)-PD(I)
        Q(I)=QG(I)-QD(I)
        QMN(I)=QMNN
        QMX(I)=QMXN
        VR(I)=VRN
C
C       SET BCHNG TO TRUE TO INDICATE Y-BUS MATRIX MUST BE
C       RECALCULATED
C
        BCHNG=.TRUE.
        RETURN
        END
        SUBROUTINE BCHNG(IBUSN,IBTYPN,PGN,QGN,PDN,QDN,VRN,
     1  QMNN,QMXN,P,Q,VR,QMN,QMX,IBTYP,IBUS,PD,PG,QD,QG,NB)
C*************************************************************************
C
C       THE SUBROUTINE BCHNG CHANGES THE CHARACTERISTICS OF A BUS OF
C       THE SYSTEM
C
C*************************************************************************
C
C       ARGUMENTS:
C
C       IBUSN - THE NEW BUS NUMBER
C
C       IBTYPN - THE NEW BUS TYPE
C
C       PGN - THE NEW REAL POWER GENERATED
C
C       QGN - THE NEW REACTIVE POWER GENERATED
C
C       PDN - THE NEW REAL POWER DEMAND
C
C       QDN - THE NEW REACTIVE POWER DEMAND
C
C       VRN - THE NEW BUS VOLTAGE SPECIFICATION
C
C       QMNN - THE NEW MINIMUM VARS FOR VOLTAGE CONTROL BUS
C
```

```
C       QMXN - THE NEW MAXIMUM VARS FOR VOLTAGE CONTROL BUS
C
C
C       NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C       Y - THE SYSTEM ADMITTANCE MATRIX
C
C       IBUS - THE BUS NUMBER
C
C       IBTYP - THE BUS TYPE  3 - SLACK
C                             2 - VOLTAGE CONTROL
C                             1 - STANDARD
C
C       PG - REAL POWER GENERATED
C
C       QG - REACTIVE POWER GENERATED
C
C       PD - REAL POWER DEMAND
C
C       QD - REACTIVE POWER DEMAND
C
C       P - REAL POWER GENERATED MINUS REAL POWER DEMAND
C
C       Q - REACTIVE POWER GENERATED MINUS REACTIVE POWER DEMAND
C
C       VR - BUS VOLTAGE SPECIFICATION
C
C       QMIN - MINIMUM VARS FOR VOLTAGE CONTROL BUS
C
C       QMAX - MAXIMUM VARS FOR VOLTAGE CONTROL BUS
C
C*************************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        INTEGER   IBUS(24)
        INTEGER   IBTYP(24)
        REAL    P(24)
        REAL    Q(24)
        REAL    VR(24)
        REAL    QMN(24)
        REAL    QMX(24)
        REAL    PD(24)
        REAL    QD(24)
        REAL    PG(24)
        REAL    QG(24)
C
C       FIND THE ELEMENT TO BE CHANGED
C
        DO 20 I=1,NB
           IF(IBUSN.EQ.IBUS(I))GO TO 40
20         CONTINUE
C
C       INSERT THE NEW ONES
C
40      IBTYP(I)=IBTYPN
        PG(I)=PGN
        QG(I)=QGN
```

```
      PD(I)=PDN
      QD(I)=QDN
      P(I)=PG(I)-PD(I)
      Q(I)=QG(I)-QD(I)
      QMN(I)=QMNN
      QMX(I)=QMXN
      VR(I)=VRN
      RETURN
      END
      SUBROUTINE BDEL(IBUSN,NB,NL,P,Q,VSPEC,QMIN,QMAX,IBTYP,IBUS,
     1 PD,PG,QD,QG,BCHNG,SB,EB,LINE,SHTY,SERZ,Y)
C*********************************************************************
C
C       THE SUBROUTINE BDEL DELETES A BUS AND ALL LINES CONNECTED
C       TO IT FROM THE NETWORK
C
C*********************************************************************
C
C       ARGUMENTS:
C
C       IBUSN - THE NEW BUS NUMBER
C
C       NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C       NL - THE NUMBER OF TRANSMISSION LINES IN THE SYSTEM
C
C       BCHNG - THIS IS A LOGICAL VARIABLE WHICH INDICATES IF Y BUS
C               SHOULD BE RECALCULATED
C
C       SB - STARTING BUS FOR LINE
C
C       EB - ENDING BUS FOR LINE
C
C       LINE - LINE NUMBER FOR A TRANSMISSION LINE
C
C       SHTY - THE SHUNT ADMITTANCE OF A LINE
C
C       SERZ - THE SERIES IMPEDANCE OF THE LINE
C
C       Y - THE SYSTEM ADMITTANCE MATRIX
C
C       IBUS - THE BUS NUMBER
C
C       IBTYP - THE BUS TYPE  3 - SLACK
C                             2 - VOLTAGE CONTROL
C                             1 - STANDARD
C
C       PG - REAL POWER GENERATED
C
C       QG - REACTIVE POWER GENERATED
C
C       PD - REAL POWER DEMAND
C
C       QD - REACTIVE POWER DEMAND
C
C       P - REAL POWER GENERATED MINUS REAL POWER DEMAND
C
C       Q - REACTIVE POWER GENERATED MINUS REACTIVE POWER DEMAND
C
C       VR - BUS VOLTAGE SPECIFICATION
```

```
C
C       QMIN - MINIMUM VARS FOR VOLTAGE CONTROL BUS
C
C       QMAX - MAXIMUM VARS FOR VOLTAGE CONTROL BUS
C
C***********************************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        COMPLEX   Y(24,24)
        COMPLEX   SHTY(50)
        COMPLEX   SERZ(50)
        REAL    P(24)
        REAL    Q(24)
        REAL    VSPEC(24)
        REAL    QMIN(24)
        REAL    QMAX(24)
        REAL    PD(24)
        REAL    QD(24)
        REAL    PG(24)
        REAL    QG(24)
        INTEGER   IBUS(24)
        INTEGER   IBTYP(24)
        INTEGER   SB(50)
        INTEGER   EB(50)
        INTEGER   LINE(50)
        LOGICAL   BCHNG
C
C       FIND THE BUS TO BE DELETED
C
        NBM1=NB-1
        DO 20 I=1,NB
           IF(IBUSN.EQ.IBUS(I)) GO TO 40
20         CONTINUE
40      IF(NB.EQ.I) GO TO 80
        DO 60 J=I,NBM1
           IBUS(J)=IBUS(J+1)
           IBTYP(J)=IBTYP(J+1)
           PD(J)=PD(J+1)
           QD(J)=QD(J+1)
           PG(J)=PG(J+1)
           QG(J)=QG(J+1)
           P(J)=P(J+1)
           Q(J)=Q(J+1)
           VSPEC(J)=VSPEC(J+1)
           QMIN(J)=QMIN(J+1)
60         QMAX(J)=QMAX(J+1)
80      NB=NBM1
C
C       DELETE THE LINES CONNECTED TO THE BUS
C
100     DO 120 K=1,NL
           IF(SB(K).EQ.IBUSN)GO TO 140
120        CONTINUE
        GO TO 160
140     ISB=SB(K)
        IEB=EB(K)
        LINEN=LINE(K)
```

```
      CALL LDEL(LINEN,ISB,IEB,NB,NL,IBUS,SE,EB,LINE,SHTY,SERZ,Y)
      GO TO 100
160   DO 180 L=1,NL
         IF(EB(L).EQ.IBUSN)GO TO 200
180      CONTINUE
      GO TO 220
200   ISB=SB(L)
      IEB=EB(L)
      LINEN=LINE(L)
      CALL LDEL(LINEN,ISB,IEB,NB,NL,IBUS,SE,EB,LINE,SHTY,SERZ,Y)
      GO TO 160
220   BCHNG=.TRUE.
      RETURN
      END
      SUBROUTINE BSPEC(NB,P,Q,VR,QMN,QMX,IBTYP,IBUS,PD,PG,QD,QG,*)
C*********************************************************************
C
C     SUBROUTINE BSPEC READS IN THE BUS DATA AND SETS ANY
C     DEFAULT PARAMETERS THAT ARE INDICATED. IF AN ERROR IN
C     THE DATA OCCURS A RETURN 1 IS MADE TO THE ERROR ROUTINE
C
C*********************************************************************
C
C     ARGUMENTS:
C
C     NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C     ANSIN - THIS INDICATES WHETHER THE DATA IS BEING INPUT
C             FROM THE TERMINAL OR FROM A DISK FILE
C
C     IBUS - THE BUS NUMBER
C
C     IBTYP - THE BUS TYPE  3 - SLACK
C                           2 - VOLTAGE CONTROL
C                           1 - STANDARD
C
C     PG - REAL POWER GENERATED
C
C     QG - REACTIVE POWER GENERATED
C
C     PD - REAL POWER DEMAND
C
C     QD - REACTIVE POWER DEMAND
C
C     P - REAL POWER GENERATED MINUS REAL POWER DEMAND
C
C     Q - REACTIVE POWER GENERATED MINUS REACTIVE POWER DEMAND
C
C     VR - BUS VOLTAGE SPECIFICATION
C
C     QMIN - MINIMUM VARS FOR VOLTAGE CONTROL BUS
C
C     QMAX - MAXIMUM VARS FOR VOLTAGE CONTROL BUS
C
C*********************************************************************
C
C     SUBROUTINES CALLED:
C
C        ERROR
C
```

```
      INTEGER    IBUS(24)
      INTEGER    IBTYP(24)
      REAL    P(24)
      REAL    Q(24)
      REAL    VR(24)
      REAL    QMN(24)
      REAL    QMX(24)
      REAL    PD(24)
      REAL    QD(24)
      REAL    PG(24)
      REAL    QG(24)
      COMMON /INOU/ KTERM
      COMMON /OUTPUT/ANSOUT,PRINT,ANSIN
      DATA YES/'YES  '/
C
C     THE BUS DATA IS INPUT
C
      IF(ANSIN.NE.YES) GO TO 11
C
C     INPUT THE DATA FROM A DATA FILE
C
10    READ(8,1004,ERR=9009,END=9008)(IBUS(I),IBTYP(I),PG(I),QG(I),
     1 PD(I),QD(I),VR(I),QMN(I),QMX(I),I=1,NB)
      GO TO 16
C
C     INPUT THE DATA FROM THE TERMINAL
C
11    DO 13 I=1,NB
         WRITE(KTERM,1024)
         READ(KTERM,*)IBUS(I)
         WRITE(KTERM,1044)
         READ(KTERM,*)IBTYP(I)
         IF(IBTYP(I).NE. 3)WRITE(KTERM,1064)
         IF(IBTYP(I).NE. 3)READ(KTERM,1084)PG(I)
         IF(IBTYP(I).EQ. 1)WRITE(KTERM,1104)
         IF(IBTYP(I).EQ. 1)READ(KTERM,1084)QG(I)
         WRITE(KTERM,1124)
         READ(KTERM,1084)PD(I)
         WRITE(KTERM,1144)
         READ(KTERM,1084)QD(I)
         IF(IBTYP(I).NE. 1)WRITE(KTERM,1184)
         IF(IBTYP(I).NE. 1)READ(KTERM,1084)VR(I)
         IF(IBTYP(I).EQ. 2)WRITE(KTERM,1204)
         IF(IBTYP(I).EQ. 2)READ(KTERM,1084)QMN(I)
         IF(IBTYP(I).EQ. 2)WRITE(KTERM,1224)
         IF(IBTYP(I).EQ. 2)READ(KTERM,1084)QMX(I)
13       CONTINUE
C
C     THE BUS DATA IS SORTED
C
16    NBM1=NB-1
      DO 40 I=1,NBM1
         IP1=I+1
         DO 20 J=IP1,NB
            IF(IBUS(I).LE.IBUS(J)) GO TO 20
            ITEMP=IBTYP(I)
            IBTYP(I)=IBTYP(J)
            IBTYP(J)=ITEMP
            ITEMP=IBUS(I)
            IBUS(I)=IBUS(J)
```

```
                IBUS(J)=ITEMP
                TEMP=PG(I)
                PG(I)=PG(J)
                PG(J)=TEMP
                TEMP=PD(I)
                PD(I)=PD(J)
                PD(J)=TEMP
                TEMP=QG(I)
                QG(I)=QG(J)
                QG(J)=TEMP
                TEMP=QD(I)
                QD(I)=QD(J)
                QD(J)=TEMP
                TEMP=QMN(I)
                QMN(I)=QMN(J)
                QMN(J)=TEMP
                TEMP=VR(I)
                VR(I)=VR(J)
                VR(J)=TEMP
                TEMP=QMX(I)
                QMX(I)=QMX(J)
                QMX(J)=TEMP
20              CONTINUE
40           CONTINUE
          DO 60 I=1,NB
             P(I)=PG(I)-PD(I)
             Q(I)=QG(I)-QD(I)
C
C         QMN AND QMX ARE SET TO -100 AND 100 IF THEY ARE UNSPECIFIED
C
             IF(IBTYP(I).NE. 2) GO TO 60
             IF(QMN(I).EQ. 0.0) QMN(I)=-100.0
             IF(QMX(I).EQ. 0.0) QMX(I)=100.0
60           CONTINUE
          RETURN
9008      CALL ERROR(8,0,&99999)
9009      CALL ERROR(9,0,&99999)
99999     RETURN 1
C
1004      FORMAT(I3,I2,7F10.0)
1024      FORMAT(' INPUT THE NUMBER OF THE BUS ',$)
1044      FORMAT(' INPUT THE TYPE OF BUS ',$)
1064      FORMAT(' INPUT THE REAL POWER GENERATION ',$)
1084      FORMAT(F10.0)
1104      FORMAT(' INPUT THE REACTIVE POWER GENERATION ',$)
1124      FORMAT(' INPUT REAL POWER DEMAND ',$)
1144      FORMAT(' INPUT REACTIVE POWER DEMAND ',$)
1164      FORMAT(' INPUT BUS CAPACITANCE OR INDUCTANCE ',$)
1184      FORMAT(' INPUT THE BUS VOLTAGE ',$)
1204      FORMAT(' INPUT Q MINIMUM FOR CONTROL BUS ',$)
1224      FORMAT(' INPUT Q MAXIMUM FOR CONTROL BUS ',$)
C
          END
          SUBROUTINE BOUT(NB,VR,QMN,QMX,IBTYP,IBUS,PD,PG,QD,QG)
C***********************************************************************
C
C         THE SUBROUTINE BOUT PRINTS OUT THE INPUT BUS DATA FOR THE
C         LOADFLOW CALCULATIONS.
C
C***********************************************************************
```

```
C
C        ARGUMENTS:
C
C        NB - THE NUMBER OF BUSES OF THE SYSTEM
C
C        ANSOUT - VARIABLE TO INDICATE IF YOU ARE PRINTING TO DISK
C
C        PRINT - LOGICAL VARIABLE TO INDICATE IF YOU ARE PRINTING TO
C                TERMINAL
C
C        IBUS - THE BUS NUMBER
C
C        IBTYP - THE BUS TYPE  3 - SLACK
C                              2 - VOLTAGE CONTROL
C                              1 - STANDARD
C
C        PG - REAL POWER GENERATED
C
C        QG - REACTIVE POWER GENERATED
C
C        PD - REAL POWER DEMAND
C
C        QD - REACTIVE POWER DEMAND
C
C        VR - BUS VOLTAGE SPECIFICATION
C
C      . QMIN - MINIMUM VARS FOR VOLTAGE CONTROL BUS
C
C        QMAX - MAXIMUM VARS FOR VOLTAGE CONTROL BUS
C
C*****************************************************************************
C
C        SUBROUTINES CALLED:
C
C           NONE
C
         LOGICAL    PRINT
         INTEGER    IBUS(24)
         INTEGER    IBTYP(24)
         REAL    VR(24)
         REAL    QMN(24)
         REAL    QMX(24)
         REAL    PD(24)
         REAL    QD(24)
         REAL    PG(24)
         REAL    QG(24)
         COMMON /INOU/ KTERM
         COMMON /OUTPUT/ ANSOUT,PRINT
         DATA YES/'YES   '/
C
C        IF YOU ARE NOT TO DO AN OUTPUT FILE SKIP
C
         IF(ANSOUT.EQ.YES)WRITE(6,1004)
C
C        OUTPUT THE LINE DATA
C
         DO 20 I=1,NB
C
C        IF YOU ARE NOT TO DO AN OUTPUT FILE SKIP
C
```

```
          IF(ANSOUT.EQ.YES) WRITE(6,1024) IBUS(I),IBTYP(I),VR(I),
     1    PD(I),QD(I),PG(I),QG(I),QMN(I),QMX(I)
C
C     IF YOU ARE NOT PRINTING TO TERMINAL SKIP
C
          IF(.NOT.PRINT) GO TO 20
          WRITE(KTERM,1044) IBUS(I),IBTYP(I),VR(I),PD(I),QD(I),PG(I),
     1    QG(I),QMN(I),QMX(I)
20        CONTINUE
      RETURN
C
1004  FORMAT('1BASE CASE BUS DATA ENTERED'/'0X------ BUS ------X',
     1 'X-- VOLTAGE --XX--- LOAD ----XX---- GEN ----X',5X,'QMIN',3X,
     2 'QMAX'/4X,'BUS NO.',3X,'TYPE',6X,'M(PU)',6X,
     3 3X,'REAL',3X,'REACT',3X,'REAL',3X,'REACT',6X,'VAR',4X,'VAR'//)
1024  FORMAT(4X,'BUS',I3,3X,I3,6X,F7.3,6X,F7.3,1X,2F7.3,1X,F7.3,
     1 2X,2F8.3)
1044  FORMAT(' BUS NUMBER ',5X,I5,2X,' BUS TYPE ',I5,/
     2 ' VOLTAGE ',5X,F8.3,/
     3 ' POWER DEMAND ',2F8.3,/
     4 ' POWER GENER. ',2F8.3,/
     5 ' QMIN ',8X,F8.3,3X,'QMIN ',F9.3,/)
C
      END
      SUBROUTINE ERROR(NERR,IER,*)
      COMMON /INOU/ KTERM
C*********************************************************************
C
C     THE SUBROUTINE ERROR PRINTS THE VARIOUS ERROR MESSAGES FOR THE
C     DIFFERENT TYPES OF ERRORS THAT OCCUR IN THE PROGRAM
C
C*********************************************************************
C
C     ARGUMENTS:
C
C     NERR - THE NUMBER OF THE ERROR THAT HAS OCCURED
C
C     IER - NUMBER DEALING WITH THE ERROR. THIS NUMBER CAN BE BUS
C           NUMBER, LINE NUMBER, TOTAL NUMBER OF BUSES OR LINES, ETC.
C
C*********************************************************************
C
C     SUBROUTINES CALLED:
C
C        NONE
C
      GO TO(20,40,60,80,100,120,140,160,180,200,220,240,260),NERR
20    WRITE(KTERM,1004)
1004  FORMAT(//' *** ERROR *** THE BUS NUMBERS IN THE FAULT DATA',/
     1 '              ARE NOT THE SAME AS THE NUMBERS FOR THE LOAD ',/
     2 '              FLOW STUDY')
      GO TO 99999
40    WRITE(KTERM,1024) IER
1024  FORMAT(' YOU MUST REMEMBER TO TELL ME EVERYTHING',/
     1 ' BUS NUMBER',I3,' IN LINE DATA IS UNDEFINED')
      GO TO 99999
60    WRITE(KTERM,1044)
1044  FORMAT('0THE DATA IS NOT IN CORRECT FORM FOR CARD 1')
      GO TO 99999
80    WRITE(KTERM,1064)
```

```
1064    FORMAT('0THE DATA IS NOT IN CORRECT FORM FOR CARD 2')
        GO TO 99999
100     WRITE(KTERM,1084)
1084    FORMAT('0THE DATA IS NOT IN CORRECT FORM ON LINE DATA CARD')
        GO TO 99999
120     WRITE(KTERM,1104)
1104    FORMAT('0THE DATA IS NOT IN CORRECT FORM ON THE CAPACITOR CARD')
        GO TO 99999
140     WRITE(KTERM,1124)
1124    FORMAT('0*** ERROR *** Z-BUS MATRIX IS SINGULAR')
        GO TO 99999
160     WRITE(KTERM,1144)
1144    FORMAT(' YOU DO NCT HAVE THE NUMBER OF BUS CARDS YOU THOUGHT')
        GO TO 99999
180     WRITE(KTERM,1164)
1164    FORMAT(' YOU FORGOT TO INCLUDE THE SHUNT CAPACITOR CARDS')
        GO TO 99999
200     WRITE(KTERM,1184)
1184    FORMAT('0YOU DID NOT INCLUDE THE NON UNIT PROFILE DATA')
        GO TO 99999
220     WRITE(KTERM,1204)
1204    FORMAT('0*** ERROR *** NONUNIT PROFILE BUS NUMBERS DO NCT MATCH
     1   THE NUMBERS GIVEN IN BUS DATA')
        GO TO 99999
240     WRITE(KTERM,1224)IER
1224    FORMAT(' *** ERROR *** INPUT ERROR. YOU ARE ATTEMPTING'/
     1         '              TO ANALYSE A SYSTEM OF ',I6,' BUSES')
        GO TO 99999
260     WRITE(KTERM,1244)IER
1244    FORMAT(' *** ERROR *** INPUT ERROR. YOU ARE ATTEMPTING'/
     1         '              TO ANALYSE A SYSTEM OF ',I6,' LINES')
99999   RETURN 1
        END
        SUBROUTINE GSLFS(NB,IVG,ALPHA,ITER,EPS,N,V,Y,P,Q,VSPEC,
     1  QMIN,QMAX,IETYP,IBUS,*)
C**********************************************************************
C
C       THE SUBROUTINE GSLFS PERFORMS THE GAUSS SEIDEL SOLUTION
C       OF THE LOADFLOW EQUATIONS FOR THE SYSTEM
C
C**********************************************************************
C
C       ARGUMENTS:
C
C       NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C       IVG - INITIAL VOLTAGE GUESS FOR THE LOADFLOW IS TO
C             BE READ IN BEFORE THE LOADFLOW IS PERFORMED
C
C       ALPHA - IF ALPHA IS NONZERO AN ACCELERATION FACTOR
C               IS TO BE USED WHILE PERFORMING GAUSS SEIDEL
C               ON THE SYSTEM
C
C       ITER - THE NUMBER OF ITERATIONS TO BE PERFORMED
C
C       EPS - THE CONVERGENCE CRITERIA FOR A SOLUTION
C
C       N - THE NUMBER OF ITERATIONS IT REQUIRED FOR A SOLUTION
C           TO CONVERGE
C
```

```
C       V - THE BUS VOLTAGES
C
C       Y - THE SYSTEM ADMITTANCE MATRIX
C
C       IBUS - THE BUS NUMBER
C
C       IBTYP - THE BUS TYPE  3 - SLACK
C                             2 - VOLTAGE CONTROL
C                             1 - STANDARD
C
C       P - REAL POWER GENERATED MINUS REAL POWER DEMAND
C
C       Q - REACTIVE POWER GENERATED MINUS REACTIVE POWER DEMAND
C
C       VSPEC - BUS VOLTAGE SPECIFICATION
C
C       QMIN - MINIMUM VARS FOR VOLTAGE CONTROL BUS
C
C       QMAX - MAXIMUM VARS FOP VOLTAGE CONTROL BUS
C
C*********************************************************************
C
C       LOCAL VARIABLES:
C
C       NNB - INTEGER ARRAY OF THE BUSES FOR THE NON - UNITY
C             VOLTAGE PROFILE
C
C       DELV - THE DIFFERENCE BETWEEN THE PRESENT VOLTAGE APPROXIMATION
C              AND THE LAST ONE
C
C       DVMAX - THE MAXIMUM VALUE FOUND OF DVMAX
C
C
C*********************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        COMPLEX   Y(24,24)
        COMPLEX   A(24)
        COMPLEX   B(24,24)
        COMPLEX   V(24)
        COMPLEX   VI
        COMPLEX   VII
        COMPLEX   SUM
        COMPLEX   DX
        COMPLEX   VN(24)
        INTEGER   IBUS(24)
        INTEGER   IBTYP(24)
        INTEGER   NNB(24)
        REAL   P(24)
        REAL   Q(24)
        REAL   VSPEC(24)
        REAL   QMAX(24)
        REAL   QMIN(24)
        COMMON /INOU/ KTERM
        COMMON /OUTPUT/ ANSOUT,PRINT
        DATA YES/'YES  '/
        DO 20 I=1,NB
```

```
            IF(IBTYP(I).EQ. 3)V(I)=CMPLX(VSPEC(I),0.)
20          CONTINUE
C
C        INITIALIZE UNKNOWN VOLTAGES AND REACTIVE POWERS.
C
         IF (IVG.EQ.0) GO TO 60
         READ(5,1004,END=9010) (NNB(I),V(I),I=1,NB)
         CALL SORT2V(NB,NNB,V)
         DO 40 I=1,NB
            IF(IBUS(I).NE.NNB(I)) CALL ERROR(11,0,89999)
40          CONTINUE
60       DO 120 I=1,NB
            IF(IBTYP(I).EQ. 2) Q(I)=0.0
            IF(IVG.EQ.1) GO TO 80
            IF(IBTYP(I).NE. 3) V(I)=CMPLX(1.0,0.0)
C
C        CALCULATE NECESSARY CONSTANTS A(I) AND B(I).
C
80          IF(IBTYP(I).EQ. 1) A(I)=(CMPLX(P(I),-Q(I)))/Y(I,I)
            DO 100 J = 1,NB
100            IF(I.NE.J) B(I,J) = Y(I,J)/Y(I,I)
120         CONTINUE
C
C        INITIALIZE CONSTANTS AND BEGIN VOLTAGE ITERATIONS.
C
         N = 0
140      DVMAX = 0.0
         I=1
160      VII = V(I)
         IF(IBTYP(I).EQ. 3) GO TO 340
         IF(IBTYP(I).EQ. 1) GO TO 300
C
C        FOR VOLTAGE CONTROL BUSSES ADJUST VOLTAGE TO SPECIFIED
C        MAGNITUDE AND CALCULATE REACTIVE POWER, IF Q LIMITS ARE
C        EXCEEDED SET Q EQUAL TO THE LIMIT AND RETURN VOLTAGE TO
C        PREVIOUS VALUE, CALCULATE A(I).
C
         V(I) = (V(I)/CABS(V(I))) * VSPEC(I)
         SUM = CMPLX(0.0,0.0)
         DO 180 L = 1,NB
180         SUM = SUM + Y(I,L) * V(L)
         Q(I) = -AIMAG(SUM * CONJG(V(I)))
         IF(Q(I) - QMAX(I)) 200,280,220
200      IF(Q(I) - QMIN(I)) 240,280,280
220      Q(I) = QMAX(I)
         GO TO 260
240      Q(I) = QMIN(I)
260      V(I) = VII
280      A(I) = (CMPLX(P(I), - Q(I)))/Y(I,I)
C
C        CALCULATE NU + 1 VOLTAGES.
C
300      SUM = CMPLX(0.0,0.0)
         VI = V(I)
         DO 320 L = 1,NB
320         IF(L.NE.I) SUM = SUM + B(I,L) * V(L)
         VN(I) = A(I)/CONJG(V(I)) - SUM
         DX = VN(I) - VI
         VN(I) = VI + ALPHA*DX
C
```

```
C         DETERMINE MAXIMUM VOLTAGE DIFFERENCE BETWEEN ITERATIONS.
C
          DELV = CABS(VN(I) - VII)
          IF(DELV.GE.DVMAX) DVMAX = DELV
C
C         UPDATE VOLTAGES BY ONE ITERATION.
C
          V(I) =VN(I)
340       I = I + 1
          IF(I.GT.NB) GO TO 360
          IF(IBTYP(I).EQ. 3) GO TO 340
          IF(I.LE.NB) GO TO 160
360       N = N + 1
C
C         COMPARE MAXIMUM VOLTAGE DIFFERENCE AGAINST CONVERGENCE
C         CRITERIA.
C
          IF (DVMAX.LE.EPS) GO TO 99999
C
C         LIMIT ITERATIONS AS A PROTECTION AGAINST DIVERGENCE.
C
          IF (N.LT.ITER) GO TO 140
          IF(ANSOUT.NE.YES)GO TO 380
          WRITE(6,1024) N
          WRITE(6,1044) (V(I),I=1,NB)
380       WRITE(KTERM,1024)N
          WRITE(KTERM,1064) (V(I),I=1,NB)
          LLL=N-ITER
          IF (LLL.LT.2) GO TO 140
          WRITE(KTERM,1084)
          READ(KTERM,1104)RESP
          IF(RESP.NE.YES)GO TO 9999
          WRITE(KTERM,1124)
          READ(KTERM,*)ITER
          GO TO 140
99999     RETURN
9010      CALL ERROR(10,0,&9999)
9999      RETURN 1
C
1004      FORMAT(2(I10,2F10.0))
1024      FORMAT(////,T10,'CONVERGENCE NOT OBTAINED IN',I5,' ITERATIONS',
     1    //T10, 'THE VOLTAGE - REAL AND IMAGINARY'//)
1044      FORMAT(T10,//4(2F12.3))
1064      FORMAT(//4(2F9.3))
1084      FORMAT(' DO YOU WISH TO INCREASE THE NUMBER OF ITERATIONS ? ',$)
1104      FORMAT(A5)
1124      FORMAT(' INPUT THE NEW ITERATION TOTAL ',$)
C
          END
          SUBROUTINE MAGANG(COM1,COM2,COM0,MAG1,MAG2,MAG0,ANG1,ANG2,ANG0)
C*****************************************************************************
C
C         THE SUBROUTINE MAGANG CALCULATES THE MAGNITUDE AND ANGLE OF
C         THREE COMPLEX QUANTITIES.
C
C*****************************************************************************
C         ARGUMENTS:
C
C         COM1,COM2,COM0 - THREE COMPLEX QUANTITIES
C
```

```
C       MAG1,MAG2,MAG0 - THREE MAGNITUDES OF COMPLEX NUMBERS
C
C       ANG1,ANG2,ANG0 - THREE ANGLES OF POLAR FORMS OF COMPLEX NUMBERS
C
C***********************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        COMPLEX   COM1
        COMPLEX   COM2
        COMPLEX   COM0
        REAL   MAG1
        REAL   MAG2
        REAL   MAG0
        REAL   ANG1
        REAL   ANG2
        REAL   ANG0
        MAG1=CABS(COM1)
        MAG2=CABS(COM2)
        MAG0=CABS(COM0)
        ANG1=0.
        ANG2=0.
        ANG0=0.
        IF((REAL(COM1).NE. 0.).OR.(AIMAG(COM1).NE. 0.))
     1  ANG1=ATAN2(AIMAG(COM1),REAL(COM1))*57.29578
        IF((REAL(COM2).NE. 0.).OR.(AIMAG(COM2).NE. 0.))
     1  ANG2=ATAN2(AIMAG(COM2),REAL(COM2))*57.29578
        IF((REAL(COM0).NE. 0.).OR.(AIMAG(COM0).NE. 0.))
     1  ANG0=ATAN2(AIMAG(COM0),REAL(COM0))*57.29578
        RETURN
        END
        SUBROUTINE FAULT(NB,NL,IFS,IFVP,ISQ,V,Y1)
C***********************************************************************
C
C       THE SUBROUTINE FAULT PREPARES THE DATA FOR THE FOUR
C       DIFFERENT TYPES OF FAULT ROUTINES
C
C***********************************************************************
C
C       ARGUMENTS:
C
C       NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C       NL - THE NUMBER OF LINES IN THE SYSTEM
C
C       IFS - IS NONZERO IF ONLY A FAULT IS TO BE PERFORMED AND
C             NOT A LOADFLOW
C
C       IFVP - IS THERE AN INITIAL VOLTAGE PROFILE TO BE READ IN
C
C       ISQ - THIS ARGUEMENT IS NONZERO IF THE SEQUENCE VOLTAGES AND
C             CURRENTS FOR THE FAULT ARE TO BE PRINTED OUT.
C
C       V - THE PREFAULT BUS VOLTAGES FOR THE SYSTEM
C
C       Y1 - THE POSITIVE SEQUENCE ADMITTANCE MATRIX FROM THE
C            LOADFLOW STUDY
C
```

```
C*************************************************************************
C
C       LOCAL VARIABLES:
C
C       Z1 - THIS VECTOR CONTAINS THE POSITIVE SEQUENCE IMPEDANCES
C            OF ALL THE BUSES IN THE NETWORK WITH RESPECT TO THE
C            BUS THAT IS BEING FAULTED.
C
C       Z2 - THIS VECTOR CONTAINS THE NEGATIVE SEQUENCE IMPEDANCES
C            OF ALL THE BUSES IN THE NETWORK WITH RESPECT TO THE
C            BUS THAT IS BEING FAULTED.
C
C       Z0 - THIS VECTOR CONTAINS THE ZERO SEQUENCE IMPEDANCES
C            OF ALL THE BUSES IN THE NETWORK WITH RESPECT TO THE
C            BUS THAT IS BEING FAULTED.
C
C       ZF - THE IMPEDANCE OF THE FAULT AT THE PARTICULAR BUS BEING
C            FAULTED.
C
C       BTYP - THE TYPE OF BUS INDICATOR 'B' IS A STANDARD BUS,
C              'G' IS A GENERATOR BUS
C
C       BZ0,BZ1,BZ2 - THE ZERO,POSITIVE AND NEGATIVE SEQUENCE VECTORS
C                     FOR THE BUSES
C
C       BZN - THE NEUTRAL IMPEDANCE FOR A BUS
C
C       LTYPE - LINE TYPE. 'L' FOR A LINE. 'T' FOR A TRANSFORMER
C               BEING MODELED AS A LINE
C
C       ZSER1 - POSITIVE AND NEGATIVE SEQUENCE IMPEDANCE OF LINE
C
C       ZSERO - ZERO SEQUENCE IMPEDANCE OF LINE
C
C       YSHT1 - POSITIVE AND NEGATIVE SEQUENCE ADMITTANCE OF LINE
C
C       YSHT0 - ZERO SEQUENCE ADMITTANCE OF LINE
C
C       F1 - THE POSITIVE SEQUENCE ADMITTANCE MATRIX
C
C       F2 - THE NEGATIVE SEQUENCE ADMITTANCE MATRIX
C
C       F0 - THE ZERO SEQUENCE ADMITTANCE MATRIX
C
C       T1 - THE TRIANGULAR FACTORIZED POSITIVE SEQUENCE ADMITTANCE
C            MATRIX
C
C       T2 - THE TRIANGULAR FACTORIZED NEGATIVE SEQUENCE ADMITTANCE
C            MATRIX
C
C       T0 - THE TRIANGULAR FACTORIZED ZERO SEQUENCE ADMITTANCE MATRIX
C
C       HVTC - THIS INDICATES HOW A HIGH VOLTAGE SIDE OF A TRANSFORMER
C              IS CONNECTED 'D' FOR DELTA 'Y' FOR WYE.
C
C       LVTC - THIS INDICATES HOW A LOW VOLTAGE SIDE OF A TRANSFORMER
C              IS CONNECTED 'D' FOR DELTA 'Y' FOR WYE.
C
C       IBUS - THE NUMBERS OF THE BUSES IN THE SYSTEM
C
```

```
C       SB - THE STARTING BUS OF A LINE. ALSO THE HIGH VOLTAGE BUS OF A
C            TRANSFORMER
C
C       LB - THE ENDING BUS OF A LINE. ALSO THE LOW VOLTAGE BUS OF A
C            TRANSFORMER
C
C       LINE - THE NUMBER OF EACH LINE OR TRANSFORMER WITHIN THE SYSTEM
C
C
C*************************************************************************
C
C       SUBROUTINES CALLED:
C
C          SCRT2V,ERROR,TRIFAC,VECTOR,F3PHAS,FLNLN,FSLG,F2LG
C
        DOUBLE PRECISION   FILEN
        LOGICAL    BAL
        COMPLEX    BI1
        COMPLEX    BI2
        COMPLEX    BI0
        COMPLEX    BIN
        COMPLEX    ZS1
        COMPLEX    YS1
        COMPLEX    ZS0
        COMPLEX    YS0
        COMPLEX    ZNH
        COMPLEX    ZNL
        COMPLEX    T1(24,24)
        COMPLEX    T2(24,24)
        COMPLEX    T0(24,24)
        COMPLEX    Z1(24)
        COMPLEX    Z2(24)
        COMPLEX    Z0(24)
        COMPLEX    JV(24)
        COMPLEX    Y1(24,24)
        COMPLEX    F1(24,24)
        COMPLEX    F2(24,24)
        COMPLEX    F0(24,24)
        COMPLEX    ZF(24)
        COMPLEX    V(24)
        COMPLEX    CTEMP
        COMPLEX    ZSER1(50)
        COMPLEX    ZSER0(50)
        COMPLEX    YSHT1(50)
        COMPLEX    YSHT0(50)
        COMPLEX    BZ0(24)
        COMPLEX    BZ1(24)
        COMPLEX    BZ2(24)
        COMPLEX    SERY0
        INTEGER    SB(50)
        INTEGER    EB(50)
        INTEGER    IBUS(24)
        REAL    LT
        REAL    LV
        REAL    LG
        REAL    LIN
        REAL    BTYP(24)
        REAL    LTYPE(50)
        REAL    LTY
        REAL    HVTC(50)
```

```
      REAL    LVTC(50)
      REAL    HGND(50)
      REAL    LGND(50)
      REAL    RCON(24)
      REAL    BGND(24)
      COMPLEX    BZN(24)
      COMPLEX    ZNHV(50)
      COMPLEX    ZNLV(50)
      REAL    TLET
      REAL    LLET
      INTEGER    IBF(24)
      INTEGER    F3P(24)
      INTEGER    FSLGN(24)
      INTEGER    FLL(24)
      INTEGER    F2LGN(24)
      INTEGER    LINE(50)
      COMMON /INOU/ KTERM
      COMMON /OUTPUT/ANSOUT,TPRINT,ANSIN
      DATA    YES/'YES'/
      DATA    BUS/'BUS'/
      DATA    GEN/'GENER'/
      DATA    LIN/'LINE'/
      DATA    TRANS/'TRANS'/
      DATA    ADD/'ADD'/
      DATA    CHANGE/'CHANG'/
      DATA    DELETE/'DELET'/
      DATA    B/'B'/
      DATA    G/'G'/
      DATA    D/'D'/
      DATA    Y/'Y'/
      DATA    TLET/'T'/
      DATA    LLET/'L'/
      TPRINT=.TRUE.
C
C     INPUT THE NUMBER OF BUSES AND LINE FOR INTERACTIVE RESPONSE
C
      IF(ANSIN.EQ.YES)GO TO 5
      WRITE(KTERM,704)
      READ(KTERM,*) NB
      WRITE(KTERM,724)
      READ(KTERM,*) NL
      WRITE(KTERM,834)
      READ(KTERM,764) ANS
      ISQ=0
      IF(ANS.EQ.YES) ISQ=1
C
C     THE VARIABLES ARE INITIALIZED TO ZERO
C
5     DO 10 I=1,NL
         YSHTO(I)=CMPLX(0.,0.)
10       CONTINUE
      DO 40 I=1,NB
         BZ0(I)=CMPLX(0.,0.)
         BZ1(I)=CMPLX(0.,0.)
         BZ2(I)=CMPLX(0.,0.)
         BZN(I)=CMPLX(0.,0.)
         ZF(I)=CMPLX(0.,0.)
40       CONTINUE
C
C     READ IN SYSTEM DATA
```

```
C
      IF(IFS.EQ.0)GO TO 44
      IF(ANSIN.EQ.YES)GC TO 401
      WRITE(KTERM,744)
      READ(KTERM,764) ANS
C
C     INPUT INITIAL FAULT VOLTAGE PROFILE IF NOT UNITY
C
      IFVP=0
      IF(ANS.EQ.YES) IFVP=1
401   IF(IFVP.EQ.0)GO TC 42
      DO 41 I=1,NB
         IF(ANSIN.EQ.YES) GO TO 402
         WRITE(KTERM,784)
         READ(KTERM,*)IBF(I)
         WRITE(KTERM,804)
         READ(KTERM,*)VM
         WRITE(KTERM,824) ANGLE
         V(I)=CMPLX(VM*COS(ANGLE/57.29578),VM*SIN(ANGLE/57.29578))
         GO TO 41
402      READ(8,864)IBF(I),V(I)
41       CONTINUE
      CALL SCRT2V(NB,IBF,V)
      GO TO 44
42    DO 43 I=1,NB
43       V(I)=CMPLX(1.,C.)
C
C     INPUT LINE AND TRANSFORMER DATA
C
44    DO 50 I=1,NL
         IF(ANSIN.EQ.YES) GO TO 441
         CALL FLNDAT(LINE(I),SB(I),EB(I),ZSER1(I),
     1         YSHT1(I),LTYPE(I),HVTC(I),LVTC(I),HGND(I),LGND(I))
         GO TO 442
441      READ(8,874,END=99999) LTYPE(I),LINE(I),SB(I),HVTC(I),
     1   HGND(I),EB(I),LVTC(I),LGND(I),ZSER1(I),YSHT1(I)
442      IF(LTYPE(I).EQ. LLET)GC TO 45
         ZSERO(I)=ZSER1(I)
         YSHTO(I)=YSHT1(I)
         IF(ANSIN.EQ.YES)GO TO 443
         WRITE(KTERM,884)
         READ(KTERM,*)R
         WRITE(KTERM,894)
         READ(KTERM,*)X
         ZNHV(I)=CMPLX(R,X)
         WRITE(KTERM,904)
         READ(KTERM,*)R
         WRITE(KTERM,914)
         READ(KTERM,*)X
         ZNLV(I)=CMPLX(R,X)
         GO TO 50
443      READ(8,924)LTY,ZNHV(I),ZNLV(I)
         IF(LTY.NE. TIET)CALL ERRCR(14,C,&99999)
         GO TO 50
45       IF(ANSIN.EQ.YES)GO TO 451
         WRITE(KTERM,9244)
         READ(KTERM,*)R
         WRITE(KTERM,9254)
         READ(KTERM,*)X
         ZSERO(I)=CMPLX(R,X)
```

```
            WRITE(KTERM,9264)
            READ(KTERM,*)GG
            WRITE(KTERM,9274)
            READ(KTERM,*)BB
            YSHTO(I)=CMPLX(GG,BB)
            GO TO 50
451         READ(8,924)LTY,ZSERO(I),YSHTC(I)
            IF(LTY.NE. LIET)CALL ERRCR(14,C,&99999)
50          CONTINUE
C
C       INPUT BUS AND GENERATOR DATA
C
55      DO 56 I=1,NB
            IF(ANSIN.EQ.YES) GO TO 551
            CALL FBSDAT(IBUS(I),BTYP(I),BCCN(I),BGND(I),BZ1(I),BZ2(I),
     1                   BZC(I),BZN(I))
            GO TO 56
551         READ(8,964)BTYP(I),IBUS(I),BCCN(I),BGND(I),BZ1(I),BZ2(I)
            IF(BTYP(I).EQ.E)GO TO 56
            READ(8,974)BZ0(I),BZN(I)
56          CONTINUE
C
C       BUSES TO BE FAULTED
C
        IF(ANSIN.EQ.YES)GO TO 562
5601    WRITE(KTERM,9744)
        READ(KTERM,*)NF
        DO 561 I=1,NF
            WRITE(KTERM,9754)
            READ(KTERM,*)IBF(I)
            WRITE(KTERM,9764)
            READ(KTERM,764)ANS
            IF((ANS.EQ.YES).OR.(ANS.EQ.Y)) F3P(I)=1
            WRITE(KTERM,9774)
            READ(KTERM,764)ANS
            IF((ANS.EQ.YES).OR.(ANS.EQ.Y))FSLGN(I)=1
            WRITE(KTERM,9784)
            READ(KTERM,764)ANS
            IF((ANS.EQ.YES).OR.(ANS.EQ.Y))FLL(I)=1
            WRITE(KTERM,9794)
            READ(KTERM,764)ANS
            IF((ANS.EQ.YES).OR.(ANS.EQ.Y))F2LGN(I)=1
            WRITE(KTERM,9804)
            READ(KTERM,*)R
            WRITE(KTERM,9814)
            READ(KTERM,*)X
            ZF(I)=CMPLX(R,X)
561         CONTINUE
        GO TO 591
562     I=1
57      READ(8,984,END=59)IBF(I),F3P(I),FSLGN(I),FLL(I),F2LGN(I),ZF(I)
        I=I+1
        GO TO 57
59      NF=I-1
C
C       OUTPUT SYSTEM DATA
C
591     CALL FLNOUT(NL,SB,EB,LINE,ZSER1,YSHT1,LVTC,HVTC,LTYPE,1)
        CALL FLNOUT(NL,SB,EB,LINE,ZSERO,YSHTC,LVTC,HVTC,LTYPE,0)
        CALL GENOUT(NB,BTYP,IBUS,BZ1,BZ2,BZ0,BZN)
```

```
C
C     SORT THE INPUTS FOR THE FAULT STUDY
C
      NBM1=NB-1
      DO 80 L=1,NBM1
         LP1=L+1
         DO 60 J=LP1,NB
            IF(IBUS(L).LE.IBUS(J)) GO TO 60
            ITEMP=IBUS(L)
            IBUS(L)=IBUS(J)
            IBUS(J)=ITEMP
            TEMP=BTYP(L)
            BTYP(L)=BTYP(J)
            BTYP(J)=TEMP
            CTEMP=BZ0(L)
            BZ0(L)=BZ0(J)
            BZ0(J)=CTEMP
            CTEMP=BZ1(L)
            BZ1(L)=BZ1(J)
            BZ1(J)=CTEMP
            CTEMP=BZ2(L)
            BZ2(L)=BZ2(J)
            BZ2(J)=CTEMP
            CTEMP=BZN(L)
            BZN(L)=BZN(J)
            BZN(J)=TEMP
60          CONTINUE
80       CONTINUE
C
C     INITIALIZE ADMITTANCE MATRICES TO ZERO
C
      DO 84 I=1,NB
         DO 82 J=1,NB
            Y1(I,J)=CMPLX(0.,0.)
            F0(I,J)=CMPLX(0.0,0.0)
            F1(I,J)=CMPLX(0.0,0.0)
82          F2(I,J)=CMPLX(0.0,0.0)
84       CONTINUE
      CALL YASSEM(NB,NL,IBUS,SB,EB,ZSER1,YSHT1,Y1,&99999)
      DO 242 I=1,NL
         SERY0=1.0/ZSER0(I)
C
C     ASSEMBLE THE ZERO SEQUENCE ADMITTANCE MATRIX.
C
         DO 162 J1=1,NB
            IF(IBUS(J1).EQ.SB(I)) GO TO 182
162         CONTINUE
         CALL ERROR(2,SB(I),&99999)
182      L=J1
         DO 202 J2=1,NB
            IF(IBUS(J2).EQ.EB(I)) GO TO 222
202         CONTINUE
         CALL ERROR(2,EB(I),&99999)
222      M=J2
         IF((HVTC(I).EQ.D).AND.(LVTC(I).EQ.D)) GO TO 242
         IF(((HVTC(I).EQ.Y).AND.(LVTC(I).EQ.D)).OR.
     1   ((HVTC(I).EQ.D).AND.(LVTC(I).EQ.Y))) GO TO 232
         IF(((HGND(I).NE.G).OR.(LGND(I).NE.G)).AND.(LTYPE(I).EQ.TLET))
     1   GO TO 242
         F0(L,M) = F0(L,M) - SERY0
```

```
                FO(M,L) = FO(M,L) - SERYO
                FO(L,L) = FO(L,L) + SERYO + YSHTO(I)/2.
                FO(M,M) = FO(M,M) + SERYO + YSHTO(I)/2.
                GO TO 242
  232           IF(LVTC(I).NE.Y)GO TO 234
                IF(LGND(I).NE.G)GO TO 242
                FO(M,M) = FO(M,M) + SERYO
                GO TO 242
  234           IF(HGND(I).NE.G)GO TO 242
                FO(L,L) = FO(L,L) + SERYO
  242           CONTINUE
C
C         ASSEMBLE THE Y-BUS MATRIX FOR FAULT STUDY
C
          DO 160 I=1,NB
             DO 140 J=1,NB
                F1(I,J)=Y1(I,J)
                F2(I,J)=Y1(I,J)
                IF (BTYP(I).EQ.B) GO TO 140
                IF(I.NE.J)GO TO 140
                F1(I,J)=Y1(I,J)+1./BZ1(I)
                F2(I,J)=Y1(I,J)+1./BZ2(I)
                IF(BGND(I).NE.G)GO TO 140
                FO(I,I) = FO(I,I) + 1./(BZO(I)+3.*BZN(I))
  140           CONTINUE
  160        CONTINUE
C
C         WRITE OUT THE FAULTED YBUS MATRIX FOR THE SYSTEM
C
          IF(ANSOUT.NE.YES)GO TO 193
          WRITE(6,1024)
          DO 180 I=1,NB
             WRITE(6,1044)(F1(I,J),J=1,NB)
  180        CONTINUE
          WRITE(6,1034)
          DO 190 I=1,NB
             WRITE(6,1044)(F2(I,J),J=1,NB)
  190        CONTINUE
          WRITE(6,1054)
          DO 192 I=1,NB
             WRITE(6,1044)(FO(I,J),J=1,NB)
  192        CONTINUE
C
C         FIND KTH COLUMN OF Z MATRICES
C
  193     CALL TRIFAC(FO,TO,NB)
          CALL TRIFAC(F1,T1,NB)
          CALL TRIFAC(F2,T2,NB)
          DO 340 I=1,NF
             DO 195 K=1,NB
                IF(IBF(I).EQ.IBUS(K))GO TO 197
  195           CONTINUE
  197        DO 200 J=1,NB
  200           JV(J)=CMPLX(0.,0.)
             JV(K)=CMPLX(1.,0.)
             CALL VECTOR(TO,ZO,JV,NB)
             CALL VECTOR(T1,Z1,JV,NB)
             CALL VECTOR(T2,Z2,JV,NB)
C
C         PERFORM FAULT ANALYSIS
```

```
C
          IF(F3P(I).NE.0)CALL F3PHAS(Z1,ZF(I),F1,V,HVTC,IVTC,IBUS,SB,
     1                          EB,LINE,K,NB,NL,ISQ)
          IF(FLL(I).NE.0)CALL FLNLN(Z1,Z2,ZF(I),F1,F2,V,HVTC,LVTC,IBUS,
     1                          SB,EB,LINE,K,NB,NL,ISQ)
          IF(FSLGN(I).NE.0)CALL FSLG(Z1,Z2,Z0,ZF(I),F1,F2,F0,ZSERO,V,
     1                          HVTC,LVTC,IBUS,SB,EB,LINE,K,NB,
     2                          NL,ISQ)
          IF(F2LGN(I).NE.0)CALL F2LG(Z1,Z2,Z0,ZF(I),F1,F2,F0,ZSERO,V,
     1                          HVTC,LVTC,IBUS,SB,EB,LINE,K,NB,
     2                          NL,ISQ)
340       CONTINUE
C
C       THIS SECTION CHECKS TO SEE IF YOU WISH TO CHANGE
C       THE ORIGINAL DATA SET
C
        ICHECK=1
400     WRITE(KTERM,1504)
        READ(KTERM,764)ANS
C
C       IF ANS NE YES AND ICHECK NE 0 THEN THERE IS NO
C       NEED TO PERFORM A NEW FAULT STUDY. QUIT.
C
        IF((ANS.NE.YES).AND.(ICHECK.NE.0))GO TO 640
C
C       IF NEW FAULT STUDY IS TO BE DONE GO PRINT OUT
C       COMPONENT DATA
C
        IF(ANS.NE.YES)GO TO 5601
        ICHECK=0
        TPRINT=.FALSE.
C
C       SPECIFIC BUS(GENERATOR) OR LINE(TRANSFORMER)
C
405     WRITE(KTERM,1524)
        READ(KTERM,764)ANSBL
C
C       ADD, CHANGE OR DELETE
C
410     WRITE(KTERM,1544)
        READ(KTERM,764)ANSACD
C
C       THIS SECTION GETS THE NEW DATA FOR THE CHANGES IN THE BUS DATA
C
        IF((ANSBL.NE.BUS).AND.(ANSBL.NE.GEN))GO TO 460
C
C       DELETING A BUS OR GENERATOR
C
        IF(ANSACD.NE.DELETE)GO TO 420
        WRITE(KTERM,1564)
        READ(KTERM,*)IB
        CALL FBDEL(IB,LINE,SB,EB,LTYPE,HVTC,HGND,LVTC,LGND,ZSER1,YSHT1,
     1             ZSERO,YSHTO,ZNHV,ZNLV,NL,IBUS,BTYP,BCCN,BGND,BZ1,BZ2,
     2             BZ0,BZN,NB)
        GO TO 400
C
C       CHANGING A BUS OR GENERATOR
C
420     IF(ANSACD.NE.CHANGE)GO TO 440
        CALL FBSDAT(IB,BT,BC,BG,BI1,BI2,BI0,BIN)
```

```
      CALL FBCHNG(IB,BT,BC,BG,BI1,BI2,BI0,BIN,IBUS,BTYP,BCON,BGND,BZ1,
     1            BZ2,BZ0,BZN,NB)
      GO TO 400
C
C     ADDING A BUS OR GENERATOR
C
440   IF(ANSACD.NE.ADD) GO TO 410
      CALL FBSDAT(IB,BT,BC,BG,BI1,BI2,BI0,BIN)
      CALL FBADD(IB,BT,BC,BG,BI1,BI2,BI0,BIN,IBUS,BTYP,BCON,BGND,BZ1,
     1            BZ2,BZ0,BZN,NB)
      GO TO 400
C
C     THIS SECTION RECEIVES THE CHANGES FOR THE LINE DATA
C
460   IF((ANSBL.NE.LIN).AND.(ANSBL.NE.TRANS)) GO TO 405
C
C     DELETING A LINE OR TRANSFORMER
C
      IF(ANSACD.NE.DELETE)GO TO 480
      WRITE(KTERM,1584)
      READ(KTERM,*) LN
      CALL FLDEL(LN,LINE,SB,EB,LTYPE,HVTC,HGND,LVTC,LGND,
     1            ZSER1,YSHT1,ZSER0,YSHT0,ZNHV,ZNLV,NL)
      GO TO 400
C
C     CHANGING A LINE OR TRANSFORMER
C
480   IF(ANSACD.NE.CHANGE)GO TO 500
      CALL FLNDAT(LN,IS,IE,ZS1,YS1,LT,HV,LV,HG,LG)
      ZNH=CMPLX(0.,0.)
      ZNL=CMPLX(0.,0.)
      ZS0=CMPLX(0.,0.)
      YS0=CMPLX(0.,0.)
      IF(LT.EQ.LLET)GO TO 485
      WRITE(KTERM,884)
      READ(KTERM,*) R
      WRITE(KTERM,894)
      READ(KTERM,*) X
      ZNH=CMPLX(R,X)
      WRITE(KTERM,904)
      READ(KTERM,*) R
      WRITE(KTERM,914)
      READ(KTERM,*) X
      ZNL=CMPLX(R,X)
      GO TO 490
485   WRITE(KTERM,9244)
      READ(KTERM,*) R
      WRITE(KTERM,9254)
      READ(KTERM,*) X
      ZS0=CMPLX(R,X)
      WRITE(KTERM,9264)
      READ(KTERM,*) GG
      WRITE(KTERM,9274)
      READ(KTERM,*) BB
      YS0=CMPLX(GG,BB)
490   CALL FLCHNG(LN,IS,IE,ZS1,YS1,ZS0,YS0,ZNH,ZNL,LT,HV,LV,
     1            HG,LG,LINE,SB,EB,LTYPE,HVTC,HGND,LVTC,LGND,
     2            ZSER1,YSHT1,ZSER0,YSHT0,ZNHV,ZNLV,NL)
      GO TO 400
C
```

```
C       ADDING A LINE OR TRANSFORMER
C
500     IF(ANSACD.NE.ADD) GO TO 410
        CALL FLNDAT(LN,IS,IE,ZS1,YS1,II,HV,IV,HG,IG)
        ZNH=CMPLX(0.,0.)
        ZNL=CMPLX(0.,0.)
        ZS0=CMPLX(0.,0.)
        YS0=CMPLX(0.,0.)
        IF(LT.EQ.LLET) GO TO 505
        WRITE(KTERM,884)
        READ(KTERM,*) R
        WRITE(KTERM,894)
        READ(KTERM,*) X
        ZNH=CMPLX(R,X)
        WRITE(KTERM,904)
        READ(KTERM,*) R
        WRITE(KTERM,914)
        READ(KTERM,*) X
        ZNL=CMPLX(R,X)
        GO TO 510
505     WRITE(KTERM,9244)
        READ(KTERM,*) R
        WRITE(KTERM,9254)
        READ(KTERM,*) X
        ZS0=CMPLX(R,X)
        WRITE(KTERM,9264)
        READ(KTERM,*) GG
        WRITE(KTERM,9274)
        READ(KTERM,*) BB
        YS0=CMPLX(GG,BB)
510     CALL FLADD(LN,IS,IE,ZS1,YS1,ZS0,YS0,ZNH,ZNL,LT,HV,IV,
     1              HG,IG,LINE,SE,EE,LTYPE,HVTC,HGND,LVTC,LGND,
     2              ZSER1,YSHT1,ZSER0,YSHT0,ZNHV,ZNLV,NL)
        GO TO 400
C
C       SAVE THE INPUT DATA IN A DATA SET
C
640     WRITE(KTERM,2004)
        READ(KTERM,764) ANS
        IF(ANS.NE.YES) GO TO 99999
        IF(IFS.EQ.0) GO TO 650
        WRITE(KTERM,2024)
        READ(KTERM,2044) FILEN
        OPEN(UNIT=7, DEVICE='DSK', ACCESS='SEQOUT', MODE='ASCII',FILE=
     1  FILEN)
        WRITE(7,2064) NB,NL,ALPHA,IVG,IYBUS,IFS,IFVP,ISQ
650     DO 660 I=1,NL
           WRITE(7,874) LTYPE(I),LINE(I),SE(I),HVTC(I),HGND(I),EE(I),
     1                  LVTC(I),LGND(I),ZSER1(I),YSHT1(I)
           IF(LTYPE(I).EQ.TLET) WRITE(7,924) LTYPE(I),ZNHV(I),ZNLV(I)
           IF(LTYPE(I).EQ.LLET) WRITE(7,924) LTYPE(I),ZSER0(I),YSHT0(I)
660        CONTINUE
        DO 670 I=1,NB
           WRITE(7,964) BTYP(I),IBUS(I),BCCN(I),BGND(I),BZ1(I),BZ2(I)
           IF(BTYP(I).NE.B) WRITE(7,974) BZ0(I),BZN(I)
670        CONTINUE
        DO 680 I=1,NF
           WRITE(7,984) IBF(I),F3P(I),FSIGN(I),FLL(I),F2LGN(I),ZF(I)
680        CONTINUE
99999   RETURN
```

```
C
704      FORMAT(' THE NUMBER OF BUSES = ',$)
724      FORMAT(' THE NUMBER OF LINES = ',$)
744      FORMAT(' DO YOU WISH TO INPUT A NONUNITY VOLTAGE PROFILE ? ',$)
764      FORMAT(A5)
784      FORMAT(' THE NUMBER OF THE BUS = ',$)
804      FORMAT(' THE VOLTAGE MAGNITUDE = ',$)
824      FORMAT(' THE VOLTAGE PHASE ANGLE = ',$)
834      FORMAT(' DO YOU WISH TO OUTPUT THE SEQUENCE QUANTITIES ? ',$)
864      FORMAT(I10,2G10.4)
874      FORMAT(A1,I4,I5,2A1,I3,2A1,3X,4G10.4)
884      FORMAT(' HIGH VOLTAGE NEUTRAL IMPEDANCE (REAL PART) = ',$)
894      FORMAT(' HIGH VOLTAGE NEUTRAL IMPEDANCE'/
     1          ' (IMAGINARY PART) = ',$)
904      FORMAT(' LOW VOLTAGE NEUTRAL IMPEDANCE (REAL PART) = ',$)
914      FORMAT(' LOW VOLTAGE NEUTRAL IMPEDANCE'/
     1          ' (IMAGINARY PART) = ',$)
924      FORMAT(A1,19X,4G10.4)
9244     FORMAT(' ZERO SEQUENCE SERIES IMPEDANCE (REAL PART) = ',$)
9254     FORMAT(' ZERO SEQUENCE SERIES IMPEDANCE',/
     1          ' (IMAGINARY PART) = ',$)
9264     FORMAT(' ZERO SEQUENCE SHUNT ADMITTANCE (REAL PART) = ',$)
9274     FORMAT(' ZERO SEQUENCE SHUNT ADMITTANCE',/
     1          ' (IMAGINARY PART) = ',$)
964      FORMAT(A1,I4,2A1,3X,4G10.4)
974      FORMAT(10X,4G10.4)
9744     FORMAT(' THE NUMBER OF BUSES TO BE FAULTED = ',$)
9754     FORMAT(' THE NUMBER OF THE BUS TO BE FAULTED = ',$)
9764     FORMAT(' DO YOU WISH TO APPLY A THREE PHASE FAULT (Y/N) ? ',$)
9774     FORMAT(' DO YOU WISH TO APPLY A SINGLE LINE TO GROUND FAULT',
     1          ' (Y/N) ? ',$)
9784     FORMAT(' DO YOU WISH TO APPLY A LINE TO LINE FAULT (Y/N) ? ',$)
9794     FORMAT(' DO YOU WISH TO APPLY A DOUBLE LINE TO GROUND FAULT',
     1          ' (Y/N) ? '$)
9804     FORMAT(' FAULT IMPEDANCE (REAL PART) = ',$)
9814     FORMAT(' FAULT IMPEDANCE (IMAGINARY PART) = ',$)
984      FORMAT(I5,4I1,1X,2G10.4)
1024     FORMAT(1H1,///,' Y-BUS MATRIX FOR FAULT CALCULATION',
     1   ' (POSITIVE SEQUENCE NETWORK)',/)
1034     FORMAT(1H1,///,' Y-BUS MATRIX FOR FAULT CALCULATION',
     1   ' (NEGATIVE SEQUENCE NETWORK)',/)
1044     FORMAT(/,(1X,6(F9.4,1X,F9.4,3X)))
1054     FORMAT(1H1,///,' Y-BUS MATRIX FOR FAULT CALCULATION',
     1   ' (ZERO SEQUENCE NETWORK)',/)
1504     FORMAT(' DO YOU WISH TO MAKE ANY CHANGES IN THE INPUT DATA ? ',
     1   $)
1524     FORMAT(' PLEASE SPECIFY BUS(GENERATOR) OR LINE(TRANSFORMER) ? '
     1   ,$)
1544     FORMAT(' PLEASE SPECIFY ADD, CHANGE, OR DELETE ? ',$)
1564     FORMAT(' BUS NUMBER = ',$)
1584     FORMAT(' LINE/TRANSFORMER NUMBER = ',$)
2004     FORMAT(' DO YOU WISH THE FAULT DATA STORED IN A FILE ? ',$)
2024     FORMAT(' FILE NAME ? ',$)
2044     FORMAT(A10)
2064     FORMAT(2I5,F5.2,I1,1X,4I1)
C
         END
         SUBROUTINE FLNDAT(ILIN,ISB,IEB,ZSLIN1,YSLIN1,LINTYP,HV,LV,HG,LG)
C*****************************************************************************
C
```

```
C       THE SUBROUTINE FLNDAT READS IN DATA FROM TERMINAL EQUIVALENT TO
C       READING IN THE FIRST CARD FOR A LINE OR TRANSFORMER
C
C*************************************************************************
C
C       ARGUMENTS:
C
C       ILIN - LINE NUMBER OF COMPONENT
C
C       ISB - STARTING BUS OR HV BUS NUMBER
C
C       IEB - ENDING BUS OR LOW VOLTAGE BUS NUMBER
C
C       ZSLIN1 - POSITIVE/NEGATIVE SEQUENCE IMPEDANCE OF A LINE
C
C       YSLIN1 - POSITIVE/NEGATIVE SEQUENCE ADMITTANCE OF A LINE
C
C       LINTYP - LINE TYPE. L FOR LINE T FOR TRANSFORMER.
C
C       HV - HIGH VOLTAGE SIDE WINDING CONNECTION, D OR Y.
C
C       LV - LOW VOLTAGE SIDE WINDING CONNECTION, D OR Y.
C
C       HG - HIGH VOLTAGE SIDE GROUND. G IF GROUND, BLANK IF NONE.
C
C
C       LG - LOW VOLTAGE SIDE GROUND. G IF GROUND, BLANK IF NONE.
C
C*************************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
      COMPLEX   ZSLIN1
      COMPLEX   YSLIN1
      REAL   LINTYP
      REAL   HV
      REAL   LV
      REAL   HG
      REAL   LG
      REAL   L
      COMMON /INOU/ KTERM
      DATA   T/'T'/
      DATA   L/'L'/
      DATA   G/'G'/
      DATA   BLANK/' '/
      HV=BLANK
      LV=BLANK
      HG=BLANK
      LG=BLANK
      ZSLIN1=CMPLX(0.,0.)
      YSLIN1=CMPLX(0.,0.)
      WRITE(KTERM,1004)
      READ(KTERM,*) ILIN
      WRITE(KTERM,1024)
      READ(KTERM,1044)LINTYP
      IF(LINTYP.EQ.T) WRITE(KTERM,1064)
      IF(LINTYP.EQ.L) WRITE(KTERM,1084)
      READ(KTERM,*) ISB
```

```
      IF(LINTYP.NE.T)GO TO 20
      WRITE(KTERM,1104)
      READ(KTERM,1044)HV
      WRITE(KTERM,1124)
      READ(KTERM,1044)HG
20    IF(LINTYP.EQ.T)WRITE(KTERM,1144)
      IF(LINTYP.EQ.L)WRITE(KTERM,1164)
      READ(KTERM,*)IEE
      IF(LINTYP.NE.T)GO TO 40
      WRITE(KTERM,1184)
      READ(KTERM,1044)LV
      WRITE(KTERM,1204)
      READ(KTERM,1044)LG
40    WRITE(KTERM,1224)
      READ(KTERM,*)R
      WRITE(KTERM,1244)
      READ(KTERM,*)X
      ZSLIN1=CMPLX(R,X)
      WRITE(KTERM,1264)
      READ(KTERM,*)G
      WRITE(KTERM,1284)
      READ(KTERM,*)B
      YSLIN1=CMPLX(G,B)
      RETURN
C
1004  FORMAT(' LINE/TRANSFORMER NUMBER = ',$)
1024  FORMAT(' LINE TYPE (L=LINE,T=TRANSFORMER) ? ',$)
1044  FORMAT(A1)
1064  FORMAT(' HIGH VOLTAGE BUS NUMBER = ',$)
1084  FORMAT(' STARTING BUS NUMBER = ',$)
1104  FORMAT(' HIGH VOLTAGE SIDE CONNECTION (Y=WYE,D=DELTA) ? ',$)
1124  FORMAT(' HIGH VOLTAGE SIDE GROUND (G=GROUND',
     1       ' BLANK IF UNGROUNDED) ? ',$)
1144  FORMAT(' LOW VOLTAGE BUS NUMBER = ',$)
1164  FORMAT(' ENDING BUS NUMBER = ',$)
1184  FORMAT(' LOW VOLTAGE SIDE CONNECTION (Y=WYE,D=DELTA) ? ',$)
1204  FORMAT(' LOW VOLTAGE SIDE GROUND (G=GROUND',
     1       ' BLANK IF UNGROUNDED) ? ',$)
1224  FORMAT(' POSITIVE/NEGATIVE SEQUENCE SERIES IMPEDANCE',/
     1       ' (REAL PART) = ',$)
1244  FORMAT(' POSITIVE/NEGATIVE SEQUENCE SERIES IMPEDANCE',/
     1       ' (IMAGINARY PART) = ',$)
1264  FORMAT(' POSITIVE/NEGATIVE SEQUENCE SHUNT ADMITTANCE',/
     1       ' (REAL PART) = ',$)
1284  FORMAT(' POSITIVE/NEGATIVE SEQUENCE SHUNT ADMITTANCE',/
     1       ' (IMAGINARY PART) = ',$)
C
      END
      SUBROUTINE FBSDAT(IBNUM,BUSTYP,EC,EG,BIMP1,BIMP2,BIMP0,BIMPN)
C*********************************************************************
C
C     THE SUBROUINE FBSDAT READS IN DATA FROM THE TERMINAL EQUIVALENT
C     TO READING IN THE CARDS FOR A BUS OR GENERATOR
C
C*********************************************************************
C
C     ARGUMENTS:
C
C     IBNUM - THE NUMBER OF THE BUS OR GENERATOR
C
```

```
C        BUSTYP - BUS TYPE. B IF STANDARD BUS. G IF GENERATOR BUS.
C
C        BC - METHOD BY WHICH GENERATOR IS CONNECTED Y OR D.
C
C        BG - BUS GROUND. G IF GROUNDED.
C
C        BIMP1 - POSITIVE SEQUENCE GENERATOR IMPEDANCE
C
C        BIMP2 - NEGATIVE SEQUENCE GENERATOR IMPEDANCE
C
C        BIMP0 - ZERO SEQUENCE GENERATOR IMPEDANCE
C
C        BIMPN - NEUTRAL IMPEDANCE OF GENERATOR
C
C***********************************************************************
C
C        SUBROUTINES CALLED:
C
C           NONE
C
         COMPLEX   BIMP1
         COMPLEX   BIMP2
         COMPLEX   BIMP0
         COMPLEX   BIMPN
         REAL    BUSTYP
         REAL    BC
         REAL    BG
         INTEGER   IBNUM
         COMMON /INOU/ KTERM
         DATA    G/'G'/
         DATA    BLANK/' '/
         BC=BLANK
         BG=BLANK
         BIMP1=CMPLX(0.,0.)
         BIMP2=CMPLX(0.,0.)
         BIMP0=CMPLX(0.,0.)
         BIMPN=CMPLX(0.,0.)
         WRITE(KTERM,984)
         READ(KTERM,*) IBNUM
         WRITE(KTERM,1004)
         READ(KTERM,1024)BUSTYP
         IF(BUSTYP.NE.G)GO TO 20
         WRITE(KTERM,1044)
         READ(KTERM,1024)BC
         WRITE(KTERM,1064)
         READ(KTERM,1024)BG
         WRITE(KTERM,1084)
         READ(KTERM,*) R
         WRITE(KTERM,1104)
         READ(KTERM,*) X
         BIMP1=CMPLX(R,X)
         WRITE(KTERM,1124)
         READ(KTERM,*) R
         WRITE(KTERM,1144)
         READ(KTERM,*) X
         BIMP2=CMPLX(R,X)
         WRITE(KTERM,1164)
         READ(KTERM,*) R
         WRITE(KTERM,1184)
         READ(KTERM,*) X
```

```
      BIMPO=CMPLX(R,X)
      IF(BG.NE.G) GO TO 20
      WRITE(KTERM,1204)
      READ(KTERM,*) R
      WRITE(KTERM,1224)
      READ(KTERM,*) X
      BIMPN=CMPLX(R,X)
20    RETURN
C
984   FORMAT(' BUS NUMBER = ',$)
1004  FORMAT(' BUS TYPE (B=BUS,G=GENERATOR) ? ',$)
1024  FORMAT(A1)
1044  FORMAT(' GENERATOR CONNECTION (Y=WYE,D=DELTA) ? ',$)
1064  FORMAT(' GENERATOR GROUNDING, G IF GROUND, BLANK IF NONE ? ',$)
1084  FORMAT(' GENERATOR POSITIVE SEQUENCE IMPEDANCE',/
     1 ' (REAL PART) = ',$)
1104  FORMAT(' GENERATOR POSITIVE SEQUENCE IMPEDANCE',/
     1 ' (IMAGINARY PART) = ',$)
1124  FORMAT(' GENERATOR NEGATIVE SEQUENCE IMPEDANCE',/
     1 ' (REAL PART) = ',$)
1144  FORMAT(' GENERATOR NEGATIVE SEQUENCE IMPEDANCE',/
     1 ' (IMAGINARY PART) = ',$)
1164  FORMAT(' GENERATOR ZERO SEQUENCE IMPEDANCE',/
     1 ' (REAL PART) = ',$)
1184  FORMAT(' GENERATOR ZERO SEQUENCE IMPEDANCE',/
     1 ' (IMAGINARY PART) = ',$)
1204  FORMAT(' GENERATOR NEUTRAL IMPEDANCE',
     1 ' (REAL PART) = ',$)
1224  FORMAT(' GENERATOR NEUTRAL IMPEDANCE',
     1 ' (IMAGINARY PART) = ',$)
C
      END
      SUBROUTINE FBADD(IB,BT,BC,BG,BI1,BI2,BIO,BIN,IBUS,BTYP,
     1                 BCON,BGND,BZ1,BZ2,BZO,BZN,NB)
C*************************************************************************
C
C       THE SUBROUTINE FBADD ADDS A NEW BUS TO THE FAULTED SYSTEM
C
C*************************************************************************
C
C       ARGUMENTS:
C
C       IB - THE NEW BUS NUMBER
C
C       BT - THE NEW BUS TYPE
C
C       BC - THE NEW GENERATOR CONNECTION TO GROUND
C
C       BG - THE NEW GENERATOR GROUND
C
C       BI1 - THE NEW GENERATOR POSITIVE SEQUENCE IMPEDANCE
C
C       BI2 - THE NEW GENERATOR NEGATIVE SEQUENCE IMPEDANCE
C
C       BIO - THE NEW GENERATOR ZERO SEQUENCE IMPEDANCE
C
C       BIN - THE NEW GENERATOR NEUTRAL IMPEDANCE
C
C       IBUS - THE BUS NUMBER
C
```

```
C         BTYP - THE BUS TYPE
C
C         BCON - THE GENERATOR CONNECTION, Y OR D
C
C         BGND - THE GENERATOR GROUND
C
C         BZ1 - THE GENERATOR POSITIVE SEQUENCE IMPEDANCE
C
C         BZ2 - THE GENERATOR NEGATIVE SEQUENCE IMPEDANCE
C
C         BZ0 - THE GENERATOR ZERO SEQUENCE IMPEDANCE
C
C         BZN - THE GENERATOR NEUTRAL IMPEDANCE
C
C         NB - THE NUMBER OF BUSES AND GENERATORS IN THE SYSTEM
C
C*****************************************************************************
C
C         SUBROUTINES CALLED:
C
C            NONE
C
          COMPLEX    BI1
          COMPLEX    BI2
          COMPLEX    BI0
          COMPLEX    BIN
          COMPLEX    BZ1(24)
          COMPLEX    BZ2(24)
          COMPLEX    BZ0(24)
          COMPLEX    BZN(24)
          REAL    BTYP(24)
          REAL    BCON(24)
          REAL    BGND(24)
          INTEGER    IBUS(24)
C
C         FIND THE POSITION OF THE NEW BUS IN THAT ARRAY
C
          DO 20 I=1,NB
             IF(IB.EQ.IBUS(I))GO TO 40
20           CONTINUE
40        NB=NB+1
          IF(I.EQ.NB) GO TO 80
C
C         MOVE THE BUSES UP ONE POSITION THAT ARE LOCATED AFTER
C         WHERE THE NEW BUS SHOULD GO
C
          N=NB-I
          DO 60 M=1,N
             J=NB-M+1
             IBUS(J)=IBUS(J-1)
             BTYP(J)=BTYP(J-1)
             BCON(J)=BCON(J-1)
             BGND(J)=BGND(J-1)
             BZ1(J)=BZ1(J-1)
             BZ2(J)=BZ2(J-1)
             BZ0(J)=BZ0(J-1)
             BZN(J)=BZN(J-1)
60           CONTINUE
C
C         INSERT THE VALUES FOR THE NEW BUS IN THE ARRAY
```

```
C
80      IBUS(I)=IB
        BTYP(I)=BT
        BCON(I)=BC
        BGND(I)=BG
        BZ1(I)=BI1
        BZ2(I)=BI2
        BZ0(I)=BI0
        BZN(I)=BIN
C
        RETURN
        END
        SUBROUTINE FBCHNG(IB,BT,BC,BG,BI1,BI2,BI0,BIN,IBUS,BTYP,
     1                   BCON,BGND,BZ1,BZ2,BZ0,BZN,NB)
C*********************************************************************
C
C       THE SUBROUTINE FBCHNG CHANGES A BUS OR GENERATORS PARAMETERS
C       IN THE FAULTED SYSTEM
C
C*********************************************************************
C
C       ARGUMENTS:
C
C       IB - THE NEW BUS NUMBER
C
C       BT - THE NEW BUS TYPE
C
C       BC - THE NEW GENERATOR CONNECTION TO GROUND
C
C       BG - THE NEW GENERATOR GROUND
C
C       BI1 - THE NEW GENERATOR POSITIVE SEQUENCE IMPEDANCE
C
C       BI2 - THE NEW GENERATOR NEGATIVE SEQUENCE IMPEDANCE
C
C       BI0 - THE NEW GENERATOR ZERO SEQUENCE IMPEDANCE
C
C       BIN - THE NEW GENERATOR NEUTRAL IMPEDANCE
C
C       IBUS - THE BUS NUMBER
C
C       BTYP - THE BUS TYPE
C
C       BCON - THE GENERATOR CONNECTION, Y OR D
C
C       BGND - THE GENERATOR GROUND
C
C       BZ1 - THE GENERATOR POSITIVE SEQUENCE IMPEDANCE
C
C       BZ2 - THE GENERATOR NEGATIVE SEQUENCE IMPEDANCE
C
C       BZ0 - THE GENERATOR ZERO SEQUENCE IMPEDANCE
C
C       BZN - THE GENERATOR NEUTRAL IMPEDANCE
C
C       NB - THE NUMBER OF BUSES AND GENERATORS IN THE SYSTEM
C
C*********************************************************************
C
C       SUBROUTINES CALLED:
```

```
C
C          NONE
C
        COMPLEX    BI1
        COMPLEX    BI2
        COMPLEX    BI0
        COMPLEX    BIN
        COMPLEX    BZ1(24)
        COMPLEX    BZ2(24)
        COMPLEX    BZ0(24)
        COMPLEX    BZN(24)
        REAL    BTYP(24)
        REAL    BCON(24)
        REAL    BGND(24)
        INTEGER    IBUS(24)
C
C       FIND THE BUS TO BE CHANGED
C
        DO 20 I=1,NB
           IF(IB.EQ.IBUS(I))GO TO 40
20         CONTINUE
C
C       REPLACE THE PARAMETERS
C
40      IBUS(I)=IB
        BTYP(I)=BT
        BCON(I)=BC
        BGND(I)=BG
        BZ1(I)=BI1
        BZ2(I)=BI2
        BZ0(I)=BI0
        BZN(I)=BIN
C
        RETURN
        END
        SUBROUTINE FBDEL(IB,LINE,SB,EB,LTYPE,HVTC,HGND,LVTC,LGND,
     1                    ZSER1,YSHT1,ZSER0,YSHT0,ZNHV,ZNLV,NL,IBUS,
     2                    BTYP,BCON,BGND,BZ1,BZ2,BZ0,BZN,NB)
C*************************************************************************
C
C       THE SUBROUTINE FBDEL DELETES A BUS OR GENERATOR FROM THE SYSTEM
C
C*************************************************************************
C
C       ARGUMENTS:
C
C       IB - THE NEW BUS NUMBER
C
C       LINE - LINE NUMBER FOR A TRANSFORMER OR TRANSMISSION LINE
C
C       SB - STARTING BUS OR HV BUS FOR A LINE OR TRANSFORMER
C
C       EB - ENDING BUS OR LV BUS FOR A LINE OR TRANSFORMER
C
C       LTYPE - SPECIFIES WHETHER COMPONENT IS A LINE OR TRANSFORMER
C
C       HVTC - THIS INDICATES HIGH VOLTAGE TERMINAL CONNECTION OF A
C              TRANSFORMER
C
C       HGND - THIS INDICATES HOW THE HIGH VOLTAGE SIDE IS GROUNDED
```

```
C
C       LVTC - THIS INDICATES LOW VOLTAGE TERMINAL CONNECTION OF A
C              TRANSFORMER
C
C       LGND - THIS INDICATES HOW THE LOW VOLTAGE SIDE IS GROUNDED
C
C       ZSER1 - POSITIVE AND NEGATIVE SEQUENCE IMPEDANCE OF A LINE
C               OR TRANSFORMER
C
C       YSHT1 - POSITIVE AND NEGATIVE SEQUENCE IMPEDANCE OF A LINE
C
C       ZSERO - ZERO SEQUENCE IMPEDANCE OF A LINE OR TRANSFORMER
C
C       YSHTO - ZERO SEQUENCE IMPEDANCE OF A LINE
C
C       ZNHV - HIGH VOLTAGE NEUTRAL IMPEDANCE OF A TRANSFORMER
C
C       ZNLV - LOW VOLTAGE NEUTRAL IMPEDANCE OF A TRANSFORMER
C
C       NL - NUMBER OF LINES AND TRANSFORMERS IN THE SYSTEM
C
C       IBUS - THE BUS NUMBER
C
C       BTYP - THE BUS TYPE
C
C       BCON - THE GENERATOR CONNECTION, Y OR D
C
C       BGND - THE GENERATOR GROUND
C
C       BZ1 - THE GENERATOR POSITIVE SEQUENCE IMPEDANCE
C
C       BZ2 - THE GENERATOR NEGATIVE SEQUENCE IMPEDANCE
C
C       BZ0 - THE GENERATOR ZERO SEQUENCE IMPEDANCE
C
C       BZN - THE GENERATOR NEUTRAL IMPEDANCE
C
C       NB - THE NUMBER OF BUSES AND GENERATORS IN THE SYSTEM
C
C*****************************************************************************
C
C       SUBROUTINES CALLED:
C
C          FLDEL
C
        COMPLEX   ZSER1(50)
        COMPLEX   YSHT1(50)
        COMPLEX   ZSERO(50)
        COMPLEX   YSHTO(50)
        COMPLEX   ZNHV(50)
        COMPLEX   ZNLV(50)
        COMPLEX   BZ1(24)
        COMPLEX   BZ2(24)
        COMPLEX   BZ0(24)
        COMPLEX   BZN(24)
        REAL   LTYPE(50)
        REAL   HVTC(50)
        REAL   HGND(50)
        REAL   LVTC(50)
        REAL   LGND(50)
```

```
      REAL    BTYP(24)
      REAL    BCON(24)
      REAL    BGND(24)
      INTEGER    LINE(50)
      INTEGER    SB(50)
      INTEGER    EB(50)
      INTEGER    IBUS(24)
C
C     FIND THE POSITION OF THE BUS TO BE DELETED
C
      NBM1=NB-1
      DO 20 I=1,NB
         IF(IB.EQ.IBUS(I))GO TO 40
20       CONTINUE
40    IF(I.EQ.NB) GO TO 80
C
C     MOVE THE BUSES UP ONE POSITION THAT ARE LOCATED AFTER
C     WHERE THE OLD BUS WAS DELETED
C
      DO 60 J=I,NBM1
         IBUS(J)=IBUS(J+1)
         BTYP(J)=BTYP(J+1)
         BCON(J)=BCON(J+1)
         BGND(J)=BGND(J+1)
         BZ1(J)=BZ1(J+1)
         BZ2(J)=BZ2(J+1)
         BZ0(J)=BZ0(J+1)
         BZN(J)=BZN(J+1)
60       CONTINUE
80    NB=NBM1
C
C     DELETE THE LINES CONNECTED TO THE BUS
C
100   DO 120 K=1,NL
         IF(SB(K).EQ.IB)GO TO 140
120      CONTINUE
      GO TO 160
140   IS=SB(K)
      IE=EB(K)
      LN=LINE(K)
      CALL FLDEL(LN,LINE,SB,EB,LTYPE,HVTC,HGND,LVTC,LGND,ZSER1,
     1           YSHT1,ZSERO,YSHTO,ZNHV,ZNLV,NL)
      GO TO 100
160   DO 180 L=1,NL
         IF(EB(L).EQ.IB)GO TO 200
180      CONTINUE
      GO TO 220
200   IS=SB(L)
      IE=EB(L)
      LN=LINE(L)
      CALL FLDEL(LN,LINE,SB,EB,LTYPE,HVTC,HGND,LVTC,LGND,ZSER1,
     1           YSHT1,ZSERO,YSHTO,ZNHV,ZNLV,NL)
      GO TO 160
C
220   RETURN
      END
      SUBROUTINE FLADD(LN,IS,IE,ZS1,YS1,ZS0,YS0,ZNH,ZNL,
     1                 LT,HV,LV,HG,LG,LINE,SB,EB,LTYPE,
     2                 HVTC,HGND,LVTC,LGND,ZSER1,YSHT1,ZSERO,
     3                 YSHTO,ZNHV,ZNLV,NL)
```

```
C***********************************************************************
C
C       THE SUBROUTINE FLADD ADDS A NEW LINE TO THE FAULTED SYSTEM
C
C***********************************************************************
C
C       ARGUMENTS:
C
C       LN - THE NEW LINE OR TRANSFORMER ADDED TO THE SYSTEM
C
CC      IS - THE NEW STARTING OR HV BUS FOR A LINE OR TRANSFORMER
C
C       IE - THE NEW ENDING OR LV BUS FOR A LINE OR TRANSFORMER
C
C       ZS1 - THE NEW LINE'S POSITIVE AND NEGATIVE SEQUENCE IMPEDANCE
C
C       YS1 - THE NEW LINE'S POSITIVE AND NEGATIVE SEQUENCE ADMITTANCE
C
C       ZS0 - THE NEW LINE'S ZERO SEQUENCE IMPEDANCE
C
C       YS0 - THE NEW LINE'S ZERO SEQUENCE ADMITTANCE
C
C       ZNH - THE NEW LINE'S HIGH VOLTAGE NEUTRAL IMPEDANCE
C
C       ZNL - THE NEW LINE'S LOW VOLTAGE NEUTRAL IMPEDANCE
C
C       LT - NEW LINE TYPE
C
C       HV - NEW LINE HV TERMINAL CONNECTION
C
C       LV - NEW LINE LV TERMINAL CONNECTION
C
C       HG - NEW LINE HV SIDE GROUND
C
C       LG - NEW LINE LV SIDE GROUND
C
C       LINE -LINE NUMBER FOR A TRANSFORMER OR TRANSMISSION LINE
C
C       SB - STARTING BUS OR HV BUS FOR A LINE OR TRANSFORMER
C
C       EB - ENDING BUS OR LV BUS FOR A LINE OR TRANSFORMER
C
C       LTYPE - SPECIFIES WHETHER COMPONENT IS A LINE OR TRANSFORMER
C
C       HVTC - THIS INDICATES HIGH VOLTAGE TERMINAL CONNECTION OF A
C              TRANSFORMER
C
C       HGND - THIS INDICATES HOW THE HIGH VOLTAGE SIDE IS GROUNDED
C
C       LVTC - THIS INDICATES LOW VOLTAGE TERMINAL CONNECTION OF A
C              TRANSFORMER
C
C       LGND - THIS INDICATES HOW THE LOW VOLTAGE SIDE IS GROUNDED
C
C       ZSER1 - POSITIVE AND NEGATIVE SEQUENCE IMPEDANCE OF A LINE
C               OR TRANSFORMER
C
C       YSHT1 - POSITIVE AND NEGATIVE SEQUENCE IMPEDANCE OF A LINE
C
C       ZSERO - ZERO SEQUENCE IMPEDANCE OF A LINE OR TRANSFORMER
```

```
C
C       YSHTO - ZERO SEQUENCE IMPEDANCE OF A LINE
C
C       ZNHV - HIGH VOLTAGE NEUTRAL IMPEDANCE OF A TRANSFORMER
C
C       ZNLV - LOW VOLTAGE NEUTRAL IMPEDANCE OF A TRANSFORMER
C
C       NL - NUMBER OF LINES AND TRANSFORMERS IN THE SYSTEM
C
C***********************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        COMPLEX   ZS1
        COMPLEX   YS1
        COMPLEX   ZS0
        COMPLEX   YS0
        COMPLEX   ZNH
        COMPLEX   ZNL
        COMPLEX   ZSER1(50)
        COMPLEX   YSHT1(50)
        COMPLEX   ZSER0(50)
        COMPLEX   YSHTO(50)
        COMPLEX   ZNHV(50)
        COMPLEX   ZNLV(50)
        REAL   LT
        REAL   HV
        REAL   LV
        REAL   HG
        REAL   LG
        REAL   LTYPE(50)
        REAL   HVTC(50)
        REAL   HGND(50)
        REAL   LVTC(50)
        REAL   LGND(50)
        INTEGER   LN
        INTEGER   IS
        INTEGER   IE
        INTEGER   LINE(50)
        INTEGER   SB(50)
        INTEGER   EB(50)
        INTEGER   IBUS(50)
C
C       FIND THE POSITION OF THE NEW LINE IN THE ARRAY
C
        DO 20 I=1,NL
           IF(LN.EQ.LINE(I))GO TO 40
20         CONTINUE
40      NL=NL+1
        IF(I.EQ.NL)GO TO 80
C
C       MOVE THE LINES UP ONE POSITION THAT ARE LOCATED AFTER WHERE
C       THE NEW LINE SHOULD GO
C
        N=NL-I
        DO 60 M=1,N
           J=NL-M+1
           LINE(J)=LINE(J-1)
```

```
            SB(J)=SB(J-1)
            EB(J)=EB(J-1)
            LTYPE(J)=LTYPE(J-1)
            HVTC(J)=HVTC(J-1)
            HGND(J)=HGND(J-1)
            LVTC(J)=LVTC(J-1)
            LGND(J)=LGND(J-1)
            ZSER1(J)=ZSER1(J-1)
            YSHT1(J)=YSHT1(J-1)
            ZSERO(J)=ZSERO(J-1)
            YSHTO(J)=YSHTO(J-1)
            ZNHV(J)=ZNHV(J-1)
            ZNLV(J)=ZNLV(J-1)
60          CONTINUE
C
C       INSERT VALUES FOR THE NEW LINE IN THE ARRAY
C
80      LINE(I)=LN
        SB(I)=IS
        EB(I)=IE
        LTYPE(I)=LT
        HVTC(I)=HV
        HGND(I)=HG
        LVTC(I)=LV
        LGND(I)=LG
        ZSER1(I)=ZS1
        YSHT1(I)=YS1
        ZSER0(I)=ZS0
        YSHT0(I)=YS0
        ZNHV(I)=ZNH
        ZNLV(I)=ZNL
C
        RETURN
        END
        SUBROUTINE FLCHNG(LN,IS,IE,ZS1,YS1,ZS0,YS0,ZNH,ZNL,
     1                    LT,HV,LV,HG,LG,LINE,SB,EB,LTYPE,
     2                    HVTC,HGND,LVTC,LGND,ZSER1,YSHT1,ZSERO,
     3                    YSHTO,ZNHV,ZNLV,NL)
C**********************************************************************
C
C       THE SUBROUTINE FLCHNG CHANGES THE PARAMETERS OF A LINE
C       OR TRANSFORMER IN THE FAULTED SYSTEM
C
C**********************************************************************
C
C       ARGUMENTS:
C
C       LN - THE NEW LINE OR TRANSFORMER ADDED TO THE SYSTEM
C
CC      IS - THE NEW STARTING OR HV BUS FOR A LINE OR TRANSFORMER
C
C       IE - THE NEW ENDING OR LV BUS FOR A LINE OR TRANSFORMER
C
C       ZS1 - THE NEW LINE'S POSITIVE AND NEGATIVE SEQUENCE IMPEDANCE
C
C       YS1 - THE NEW LINE'S POSITIVE AND NEGATIVE SEQUENCE ADMITTANCE
C
C       ZS0 - THE NEW LINE'S ZERO SEQUENCE IMPEDANCE
C
C       YS0 - THE NEW LINE'S ZERO SEQUENCE ADMITTANCE
```

```
C
C       ZNH - THE NEW LINE'S HIGH VOLTAGE NEUTRAL IMPEDANCE
C
C       ZNL - THE NEW LINE'S LOW VOLTAGE NEUTRAL IMPEDANCE
C
C       LT - NEW LINE TYPE
C
C       HV - NEW LINE HV TERMINAL CONNECTION
C
C       LV - NEW LINE LV TERMINAL CONNECTION
C
C       HG - NEW LINE HV SIDE GROUND
C
C       LG - NEW LINE LV SIDE GROUND
C
C       LINE -LINE NUMBER FOR A TRANSFORMER OR TRANSMISSION LINE
C
C       SB - STARTING BUS OR HV BUS FOR A LINE OR TRANSFORMER
C
C       EB - ENDING BUS OR LV BUS FOR A LINE OR TRANSFORMER
C
C       LTYPE - SPECIFIES WHETHER COMPONENT IS A LINE OR TRANSFORMER
C
C       HVTC - THIS INDICATES HIGH VOLTAGE TERMINAL CONNECTION OF A
C              TRANSFORMER
C
C       HGND - THIS INDICATES HOW THE HIGH VOLTAGE SIDE IS GROUNDED
C
C       LVTC - THIS INDICATES LOW VOLTAGE TERMINAL CONNECTION OF A
C              TRANSFORMER
C
C       LGND - THIS INDICATES HOW THE LOW VOLTAGE SIDE IS GROUNDED
C
C       ZSER1 - POSITIVE AND NEGATIVE SEQUENCE IMPEDANCE OF A LINE
C               OR TRANSFORMER
C
C       YSHT1 - POSITIVE AND NEGATIVE SEQUENCE IMPEDANCE OF A LINE
C
C       ZSER0 - ZERO SEQUENCE IMPEDANCE OF A LINE OR TRANSFORMER
C
C       YSHT0 - ZERO SEQUENCE IMPEDANCE OF A LINE
C
C       ZNHV - HIGH VOLTAGE NEUTRAL IMPEDANCE OF A TRANSFORMER
C
C       ZNLV - LOW VOLTAGE NEUTRAL IMPEDANCE OF A TRANSFORMER
C
C       NL - NUMBER OF LINES AND TRANSFORMERS IN THE SYSTEM
C
C*********************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        COMPLEX   ZS1
        COMPLEX   YS1
        COMPLEX   ZS0
        COMPLEX   YS0
        COMPLEX   ZNH
        COMPLEX   ZNL
```

```
      COMPLEX    ZSER1(50)
      COMPLEX    YSHT1(50)
      COMPLEX    ZSER0(50)
      COMPLEX    YSHT0(50)
      COMPLEX    ZNHV(50)
      COMPLEX    ZNLV(50)
      REAL   LT
      REAL   HV
      REAL   LV
      REAL   HG
      REAL   LG
      REAL   LTYPE(50)
      REAL   HVTC(50)
      REAL   HGND(50)
      REAL   LVTC(50)
      REAL   LGND(50)
      INTEGER   LN
      INTEGER   IS
      INTEGER   IE
      INTEGER   LINE(50)
      INTEGER   SB(50)
      INTEGER   EE(50)
      INTEGER   IBUS(50)
C
C     FIND THE POSITION OF THE LINE TO BE CHANGED
C
      DO 20 I=1,NI
         IF(LN.EQ.LINE(I))GO TO 40
20       CONTINUE
C
C     REPLACE THE PARAMETERS
C
40    LINE(I)=LN
      SB(I)=IS
      EB(I)=IE
      LTYPE(I)=LT
      HVTC(I)=HV
      HGND(I)=HG
      LVTC(I)=LV
      LGND(I)=LG
      ZSER1(I)=ZS1
      YSHT1(I)=YS1
      ZSER0(I)=ZS0
      YSHT0(I)=YS0
      ZNHV(I)=ZNH
      ZNLV(I)=ZNL
C
      RETURN
      END
      SUBROUTINE FLDEL(LN,LINE,SB,EB,ITYPE,HVTC,HGND,LVTC,LGND,
     1                 ZSER1,YSHT1,ZSERC,YSHT0,ZNHV,ZNLV,NL)
C*****************************************************************
C
C     THE SUBROUTINE FLDEL DELETES A LINE FROM THE FAULTED SYSTEM
C
C*****************************************************************
C
C     ARGUMENTS:
C
C     LN - THE NEW LINE OR TRANSFORMER ADDED TO THE SYSTEM
```

```
C
C       LINE -LINE NUMBER FOR A TRANSFCRMER CR TRANSMISSICN LINE
C
C       SB - STARTING BUS OR HV BUS FOR A LINE CR TRANSFCRMER
C
C       EB - ENDING BUS OR LV BUS FOR A LINE CR TRANSFCRMER
C
C       LTYPE - SPECIFIES WHETHER CCMPCNENT IS A LINE OR TRANSFCRMER
C
C       HVTC - THIS INDICATES HIGH VCLTAGE TERMINAL CCNNECTICN CF A
C              TRANSFCRMER
C
C       HGND - THIS INDICATES HOW THE HIGH VCLTAGE SIDE IS GROUNDED
C
C       LVTC - THIS INDICATES LCW VCLTAGE TERMINAL CONNECTICN OF A
C              TRANSFORMER
C
C       LGND - THIS INDICATES HOW THE LCW VCLTAGE SIDE IS GRCUNDED
C
C       ZSER1 - POSITIVE AND NEGATIVE SEQUENCE IMPEDANCE CF A LINE
C               CR TRANSFCRMER
C
C       YSHT1 - POSITIVE AND NEGATIVE SEQUENCE IMPEDANCE CF A LINE
C
C       ZSERO - ZERC SEQUENCE IMPEDANCE OF A LINE OR TRANSFORMER
C
C       YSHTO - ZERC SEQUENCE IMPEDANCE OF A LINE
C
C       ZNHV - HIGH VCLTAGE NEUTRAL IMPEDANCE OF A TRANSFCRMER
C
C       ZNLV - LOW VCLTAGE NEUTRAL IMPEDANCE OF A TRANSFORMER
C
C       NL - NUMBER OF LINES AND TRANSFCRMERS IN THE SYSTEM
C
C*************************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        CCMPLEX    ZSER1(50)
        CCMPLEX    YSHT1(50)
        CCMPLEX    ZSERO(50)
        COMPLEX    YSHTO(50)
        CCMPLEX    ZNHV(50)
        CCMPLEX    ZNLV(50)
        REAL    LTYPE(50)
        REAL    HVTC(50)
        REAL    HGND(50)
        REAL    LVTC(50)
        REAL    LGND(50)
        INTEGER    LN
        INTEGER    LINE(50)
        INTEGER    SB(50)
        INTEGER    EB(50)
        INTEGER    IBUS(50)
C
C       FIND THE POSITICN OF THE LINE TC BE DELTED
C
        NLM1=NL-1
```

```
      DO 20 I=1,NL
         IF(LN.EQ.LINE(I))GO TO 40
20       CONTINUE
40    IF(I.EQ.NL) GO TO 80
C
C     MOVE THE LINES UP ONE POSITION THAT ARE LOCATED AFTER WHERE
C     THE OLD BUS WAS DELETED
C
      DO 60 J=I,NLM1
         LINE(J)=LINE(J+1)
         SB(J)=SB(J+1)
         EB(J)=EB(J+1)
         LTYPE(J)=LTYPE(J+1)
         HVTC(J)=HVTC(J+1)
         HGND(J)=HGND(J+1)
         LVTC(J)=LVTC(J+1)
         LGND(J)=LGND(J+1)
         ZSER1(J)=ZSER1(J+1)
         YSHT1(J)=YSHT1(J+1)
         ZSER0(J)=ZSER0(J+1)
         YSHT0(J)=YSHT0(J+1)
         ZNHV(J)=ZNHV(J+1)
         ZNLV(J)=ZNLV(J+1)
60       CONTINUE
80    NL=NLM1
C
      RETURN
      END
      SUBROUTINE F3PHAS(Z1,ZF,F,V,HVTC,LVTC,IBUS,SB,EB,LINE,IFF,NB,
     1 NL,ISQ)
C*********************************************************************
C
C     THE SUBROUTINE F3PHAS SOLVES FOR A THREE PHASE FAULT AT
C     AT PARTICULAR BUS OF A POWER SYSTEM.
C
C*********************************************************************
C
C     ARGUMENTS:
C
C     Z1 - THIS VECTOR CONTAINS THE POSITIVE SEQUENCE IMPEDANCES
C          OF ALL THE BUSES IN THE NETWORK WITH RESPECT TO THE
C          BUS THAT IS BEING FAULTED.
C
C     ZF - THE IMPEDANCE OF THE FAULT AT THE PARTICULAR BUS BEING
C          FAULTED.
C
C     F - THE POSITIVE SEQUENCE ADMITTANCE MATRIX
C
C     V - THE PREFAULT BUS VOLTAGES FOR THE SYSTEM
C
C     HVTC - THIS INDICATES HOW A HIGH VOLTAGE SIDE OF A TRANSFORMER
C              IS CONNECTED 'D' FOR DELTA 'Y' FOR WYE.
C
C     LVTC - THIS INDICATES HOW A LOW VOLTAGE SIDE OF A TRANSFORMER
C              IS CONNECTED 'D' FOR DELTA 'Y' FOR WYE.
C
C     IBUS - THE NUMBERS OF THE BUSES IN THE SYSTEM
C
C     SB - THE STARTING BUS OF A LINE. ALSO THE HIGH VOLTAGE BUS OF A
C          TRANSFORMER
```

```
C
C       EB - THE ENDING BUS OF A LINE. ALSO THE LOW VOLTAGE BUS OF A
C            TRANSFORMER
C
C       LINE - THE NUMBER OF EACH LINE OR TRANSFORMER WITHIN THE SYSTEM
C
C       IFP - THE NUMBER OF THE BUS BEING FAULTED
C
C       NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C       NL - THE NUMBER OF LINES IN THE SYSTEM.
C
C       ISQ - THIS ARGUEMENT IS NONZERO IF THE SEQUENCE VOLTAGES AND
C             CURRENTS FOR THE FAULT ARE TO BE PRINTED OUT.
C
C*********************************************************************
C
C       LOCAL VARIABLES:
C
C       V0,V1,V2 - THE COMPLEX VECTORS FOR THE ZERO, POSITIVE, AND
C                  NEGATIVE SEQUENCE VOLTAGES
C
C       VMAG0,VMAG1,VMAG2 - THE MAGNITUDES OF THE THREE SEQUENCE
C                           VOLTAGES
C
C       ANG0,ANG1,ANG2 - THE VOLTAGE ANGLES OF THE ZERO,POSITIVE,
C                        AND NEGATIVE SEQUENCE VOLTAGES AND CURRENTS
C
C       ILF0,ILF1,ILF2 - THE CURRENT FLOWING FORWARD ALONG A LINE
C                        (FROM STARTING BUS TO ENDING BUS)
C
C       ILFM0,ILFM1,ILFM2 - THE MAGNITUDE OF THE SEQUENCE CURRENT IN A
C                           LINE
C
C       V0S,V1S,V2S - THE COMPLEX VALUE OF THE VOLTAGE WITH PHASE
C                     SHIFTING DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                     TAKEN INTO ACCOUNT
C
C       I0SF,I1SF,I2SF - THE COMPLEX VALUE OF THE CURRENT WITH PHASE
C                        SHIFT DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                        TAKEN INTO ACCOUNT. THIS IS FOR CURRENTS GOING
C                        FROM STARTING BUS TO ENDING BUS.
C
C       I0SB,I1SB,I2SB - THE COMPLEX VALUE OF THE CURRENT WITH PHASE
C                        SHIFT DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                        TAKEN INTO ACCOUNT. THIS IS FOR CURRENTS GOING
C                        FROM ENDING BUS TO STARTING BUS
C
C       IFAULT - COMPLEX FAULT CUURENT
C
C       MAG - REAL MAGNITUDE OF IFAULT
C
C       SCC - SHORT CIRCUIT CAPACITANCE
C
C       SHIFT - THIS INTEGER VECTOR IS RETURNED BY DYFIX. IT CONTAINS
C               THE NUMBER OF TIMES A BUS VOLTAGE IS TO BE SHIFTED BY
C               30 DEGREES.
C
C       A30 - THIS CONSTANT IS TAKEN TO THE POWER OF THE VALUE IN SHIFT
C             FOR THAT BUS
```

```
C
C***************************************************************************
C
C       SUBROUTINES CALLED:
C
C          DYFIX,PHPRNT
C
        COMPLEX   Z1(24)
        COMPLEX   F(24,24)
        COMPLEX   IFAULT
        COMPLEX   V0(24)
        COMPLEX   V1(24)
        COMPLEX   V2(24)
        COMPLEX   ILF0
        COMPLEX   ILF1
        COMPLEX   ILF2
        COMPLEX   ZF
        COMPLEX   V(24)
        COMPLEX   I0SF(50)
        COMPLEX   I1SF(50)
        COMPLEX   I2SF(50)
        COMPLEX   I0SB(50)
        COMPLEX   I1SB(50)
        COMPLEX   I2SB(50)
        COMPLEX   V0S(24)
        COMPLEX   V1S(24)
        COMPLEX   V2S(24)
        COMPLEX   A30
        REAL   HVTC(24)
        REAL   LVTC(24)
        REAL   MAG
        REAL   ILFM0
        REAL   ILFM1
        REAL   ILFM2
        INTEGER   SB(50)
        INTEGER   EB(50)
        INTEGER   IBUS(24)
        INTEGER   SHIFT(24)
        INTEGER   LINE(50)
        LOGICAL   PRINT
        COMMON /INOU/ KTERM
        COMMON /OUTPUT/ ANSOUT,PRINT,ANSIN
        DATA   YES/'YES'/
        A30=CMPLX(0.8660254,0.5)
        CALL DYFIX(SHIFT,HVTC,LVTC,SB,EB,LINE,IBUS,NB,NL,IFP)
        IFAULT=V(IFP)/(Z1(IFP)+ZF)
        MAG=CABS(IFAULT)
        SCC =CABS(V(IFP))*MAG
        IF(ISQ.EQ. 0)GO TO 20
        WRITE(KTERM,1064)
        WRITE(KTERM,1084)  IBUS(IFP)
        WRITE(KTERM,1104)  ZF
        WRITE(KTERM,1124)  IFAULT, MAG
        WRITE(KTERM,1144)  SCC
        WRITE(KTERM,1164)
        IF(ANSOUT.NE.YES)GO TO 20
        WRITE(6,1064)
        WRITE(6,1084)  IBUS(IFP)
        WRITE(6,1104)  ZF
        WRITE(6,1124)  IFAULT, MAG
```

```
      WRITE(6,1144) SCC
      WRITE(6,1164)
C
C     CALCULATE THE SEQUENCE VOLATAGES FOR THE FAULT
C
20    DO 220 K=1,NB
         V0(K)=CMPLX(0.,0.)
         V1(K)=V(K)-(Z1(K)/(ZF+Z1(IFP)))*V(IFP)
         V2(K)=CMPLX(0.,0.)
         IF(I.EQ.K)V1(K)=(ZF/(ZF+Z1(IFP)))*V(IFP)
         V0S(K)=V0(K)
         V1S(K)=V1(K)
         V2S(K)=V2(K)
C
C     PERFORM PHASE SHIFTS IF NECESSARY
C
         IF(SHIFT(K).NE.0)V1S(K)=V1(K)*A30**SHIFT(K)
         IF(ISQ.EQ.0)GO TO 220
         CALL MAGANG(V1S(K),V2S(K),V0S(K),VMAG1,VMAG2,VMAG0,
     1   ANG1,ANG2,ANG0)
         WRITE(KTERM,1184)IBUS(K),VMAG0,ANG0,VMAG1,ANG1,VMAG2,ANG2
         IF(ANSOUT.EQ.YES)WRITE(6,1184)IBUS(K),VMAG0,ANG0,VMAG1,ANG1,
     1                                  VMAG2,ANG2
220      CONTINUE
      IF(ISQ.NE.0)WRITE(KTERM,1204)
      IF((ISQ.NE.0).AND.(ANSOUT.EQ.YES))WRITE(6,1204)
C
C     CALCULATE THE SEQUENCE CURRENTS FOR THE NETWORK
C
      DO 320 N=1,NL
         DO 240 J=1,NB
            IF(IBUS(J).EQ.SB(N))  GO TO 260
240         CONTINUE
260      L=J
         DO 280 J=1,NB
            IF(IBUS(J).EQ.EB(N))  GO TO 300
280         CONTINUE
300      M=J
         ILF1=(V1(L)-V1(M))*(-F(L,M))
         I0SF(N)=CMPLX(0.,0.)
         I1SF(N)=ILF1
         I2SF(N)=CMPLX(0.,0.)
         IF(SHIFT(L).NE.0)I1SF(N)=I1SF(N)*A30**SHIFT(L)
         IF(ISQ.EQ.0)GO TO 320
         CALL MAGANG(I1SF(N),I2SF(N),I0SF(N),ILFM1,ILFM2,ILFM0,
     1   ANG1,ANG2,ANG0)
         WRITE(KTERM,1224)LINE(N),SB(N),EB(N),ILFM0,ANG0,ILFM1,
     1                    ANG1,ILFM2,ANG2
         IF(ANSOUT.EQ.YES)WRITE(6,1224)LINE(N),SB(N),EB(N),ILFM0,ANG0,
     1                                 ILFM1,ANG1,ILFM2,ANG2
         I0SB(N)=CMPLX(0.,0.)
         I1SB(N)=-ILF1
         I2SB(N)=CMPLX(0.,0.)
         IF(SHIFT(M).NE.0)I1SB(N)=I1SB(N)*A30**SHIFT(M)
         CALL MAGANG(I1SB(N),I2SB(N),I0SB(N),ILFM1,ILFM2,ILFM0,
     1   ANG1,ANG2,ANG0)
         WRITE(KTERM,1234)LINE(N),EB(N),SB(N),ILFM0,ANG0,ILFM1,
     1                    ANG1,ILFM2,ANG2
         IF(ANSOUT.EQ.YES)WRITE(6,1234)LINE(N),EB(N),SB(N),ILFM0,
     1                                 ANG0,ILFM1,ANG1,ILFM2,ANG2
```

```
320       CONTINUE
C
C       CALCULATE PER PHASE VALUES
C
        WRITE(KTERM,1064)
        WRITE(KTERM,1084) IBUS(IFP)
        WRITE(KTERM,1104) ZF
        WRITE(KTERM,1124) IFAULT, MAG
        WRITE(KTERM,1144) SCC
        IF(ANSOUT.NE.YES)GO TO 340
        WRITE(6,1064)
        WRITE(6,1084) IBUS(IFP)
        WRITE(6,1104) ZF
        WRITE(6,1124) IFAULT, MAG
        WRITE(6,1144) SCC
340     CALL PHPRNT(NB,NL,IBUS,LINE,SB,EB,VOS,V1S,V2S,IOSF,I1SF,I2SF,
     1  IOSB,I1SB,I2SB)
C
1064    FORMAT('1',20X,'***** BALANCED THREE PHASE FAULT *****')
1084    FORMAT (1H0,'FAULT OCCURS AT BUS     ',I2)
1104    FORMAT(1H0,'FAULT IMPEDANCE=', 2F10.3)
1124    FORMAT (1H0,'IFAULT=',2F10.4,4X,'MAGNITUDE=',F10.4)
1144    FORMAT(1H0,'SHORT CIRCUIT CAPACITY=',F10.4)
1164    FORMAT(1H0,
     1  /,' POST FAULT BUS VOLTAGES',//,
     1  6X,'X',20('-'),' SEQUENCE QUANTITIES ',20('-'),'X'/
     2  ' BUS',2X,'X------ ZERO -------XX---- POSITIVE -----X',
     3  'X---- NEGATIVE -----X'/' NC.'6X,
     2  'V0',6X,'ANGLE 0',6X,'V1',6X,'ANGLE 1',6X,'V2',6X,'ANGLE 2'/)
1184    FORMAT (1H ,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,F10.2)
1204    FORMAT (1H0,/,' POST FAULT LINE CURRENTS',
     1  //,17X,'X',20('-'),' SEQUENCE QUANTITIES ',
     1  20('-'),'X',/17X,'X------ ZERO -------X',
     1  'X---- POSITIVE -----XX---- NEGATIVE -----X',/
     2  ' LINE',4X,'SB',2X,'EB',5X,
     2  'IO',6X,'ANGLE C',6X,'I1',6X,'ANGLE 1',6X,'I2',6X,'ANGLE 2'/)
1224    FORMAT (1H ,I3,4X,I2,2X,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,
     1          F10.2)
1234    FORMAT (1H ,I3,4X,I2,2X,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,
     1          F10.2/)
C
        RETURN
        END
        SUBROUTINE FLNLN(Z1,Z2,ZF,F1,F2,V,HVTC,LVTC,IBUS,SB,EB,
     1  LINE,IFP,NB,NL,ISQ)
C***********************************************************************
C
C       THE SUBROUTINE FLNLN SOLVES FOR A LINE TO LINE FAULT AT
C       AT PARTICULAR BUS OF A POWER SYSTEM.
C
C***********************************************************************
C
C       ARGUMENTS:
C
C       Z1 - THIS VECTOR CONTAINS THE POSITIVE SEQUENCE IMPEDANCES
C            OF ALL THE BUSES IN THE NETWORK WITH RESPECT TO THE
C            BUS THAT IS BEING FAULTED.
C
C       Z2 - THIS VECTOR CONTAINS THE NEGATIVE SEQUENCE IMPEDANCES
C            OF ALL THE BUSES IN THE NETWORK WITH RESPECT TO THE
```

```
C               BUS THAT IS BEING FAULTED.
C
C         ZF - THE IMPEDANCE OF THE FAULT AT THE PARTICULAR BUS BEING
C               FAULTED.
C
C         F1 - THE POSITIVE SEQUENCE ADMITTANCE MATRIX
C
C         F2 - THE NEGATIVE SEQUENCE ADMITTANCE MATRIX
C
C         V - THE PREFAULT BUS VOLTAGES FOR THE SYSTEM
C
C         HVTC - THIS INDICATES HOW A HIGH VOLTAGE SIDE OF A TRANSFORMER
C                IS CONNECTED 'D' FOR DELTA 'Y' FOR WYE.
C
C         LVTC - THIS INDICATES HOW A LOW VOLTAGE SIDE OF A TRANSFORMER
C                IS CONNECTED 'D' FOR DELTA 'Y' FOR WYE.
C
C         IBUS - THE NUMBERS OF THE BUSES IN THE SYSTEM
C
C         SB - THE STARTING BUS OF A LINE. ALSO THE HIGH VOLTAGE BUS OF A
C              TRANSFORMER
C
C         EB - THE ENDING BUS OF A LINE. ALSO THE LOW VOLTAGE BUS OF A
C              TRANSFORMER
C
C         LINE - THE NUMBER OF EACH LINE OR TRANSFORMER WITHIN THE SYSTEM
C
C         IFP - THE NUMBER OF THE BUS BEING FAULTED
C
C         NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C         NL - THE NUMBER OF LINES IN THE SYSTEM.
C
C         ISQ - THIS ARGUEMENT IS NONZERO IF THE SEQUENCE VOLTAGES AND
C               CURRENTS FOR THE FAULT ARE TO BE PRINTED OUT.
C
C*************************************************************************
C
C         LOCAL VARIABLES:
C
C         V0,V1,V2 - THE COMPLEX VECTORS FOR THE ZERO, POSITIVE, AND
C                    NEGATIVE SEQUENCE VOLTAGES
C
C         VMAG0,VMAG1,VMAG2 - THE MAGNITUDES OF THE THREE SEQUENCE
C                             VOLTAGES
C
C         ANG0,ANG1,ANG2 - THE VOLTAGE ANGLES OF THE ZERO,POSITIVE,
C                          AND NEGATIVE SEQUENCE VOLTAGES AND CURRENTS
C
C         ILF0,ILF1,ILF2 - THE CURRENT FLOWING FORWARD ALONG A LINE
C                          (FROM STARTING BUS TO ENDING BUS)
C
C         ILFM0,ILFM1,ILFM2 - THE MAGNITUDE OF THE SEQUENCE CURRENT IN A
C                             LINE
C
C         V0S,V1S,V2S - THE COMPLEX VALUE OF THE VOLTAGE WITH PHASE
C                       SHIFTING DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                       TAKEN INTO ACCOUNT
C
C         I0SF,I1SF,I2SF - THE COMPLEX VALUE OF THE CURRENT WITH PHASE
```

```
C                        SHIFT DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                        TAKEN INTO ACCOUNT. THIS IS FOR CURRENTS GOING
C                        FROM STARTING BUS TO ENDING BUS.
C
C        IOSB,I1SB,I2SB - THE COMPLEX VALUE OF THE CURRENT WITH PHASE
C                        SHIFT DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                        TAKEN INTO ACCOUNT. THIS IS FOR CURRENTS GOING
C                        FROM ENDING BUS TO STARTING BUS
C
C        IFAULT - COMPLEX FAULT CUURENT
C
C        MAG - REAL MAGNITUDE OF IFAULT
C
C        SCC - SHORT CIRCUIT CAPACITANCE
C
C        SHIFT - THIS INTEGER VECTOR IS RETURNED BY DYFIX. IT CONTAINS
C                THE NUMBER OF TIMES A BUS VOLATGE IS TO BE SHIFTED BY
C                30 DEGREES.
C
C        A30 - THIS CONSTANT IS TAKEN TO THE POWER OF THE VALUE IN SHIFT
C              FOR THAT BUS
C
C*************************************************************************
C
C        SUBROUTINES CALLED:
C
C           DYFIX,PHPRNT
C
      COMPLEX   Z1(24)
      COMPLEX   Z2(24)
      COMPLEX   F1(24,24)
      COMPLEX   F2(24,24)
      COMPLEX   IFAULT
      COMPLEX   V0(24)
      COMPLEX   V1(24)
      COMPLEX   V2(24)
      COMPLEX   ILF0
      COMPLEX   ILF1
      COMPLEX   ILF2
      COMPLEX   ZF
      COMPLEX   V(24)
      COMPLEX   IOSF(50)
      COMPLEX   I1SF(50)
      COMPLEX   I2SF(50)
      COMPLEX   IOSB(50)
      COMPLEX   I1SB(50)
      COMPLEX   I2SB(50)
      COMPLEX   V0S(24)
      COMPLEX   V1S(24)
      COMPLEX   V2S(24)
      COMPLEX   A30
      REAL   HVTC(24)
      REAL   LVTC(24)
      REAL   MAG
      REAL   ILFM0
      REAL   ILFM1
      REAL   ILFM2
      INTEGER   SB(50)
      INTEGER   EB(50)
      INTEGER   IBUS(24)
```

```
      INTEGER    SHIFT(24)
      INTEGER    LINE(50)
      LOGICAL    PRINT
      COMMON /INOU/ KTERM
      COMMON /OUTPUT/ ANSOUT,PRINT,ANSIN
      DATA   YES/'YES'/
      A30=CMPLX(0.8660254,0.5)
      CALL DYFIX(SHIFT,HVTC,LVTC,SB,EB,LINE,IBUS,NB,NL,I)
      IFAULT=V(IFP)/(Z1(IFP)+Z2(IFP)+ZF)
      MAG=CABS(IFAULT)
      SCC =CABS(V(IFP))*MAG
      IF(ISQ.EQ. 0)GO TO 20
      WRITE(KTERM,1064)
      WRITE(KTERM,1084) IBUS(IFP)
      WRITE(KTERM,1104) ZF
      WRITE(KTERM,1124) IFAULT, MAG
      WRITE(KTERM,1144) SCC
      WRITE(KTERM,1164)
      IF(ANSOUT.NE.YES)GO TO 20
      WRITE(6,1064)
      WRITE(6,1084) IBUS(IFP)
      WRITE(6,1104) ZF
      WRITE(6,1124) IFAULT, MAG
      WRITE(6,1144) SCC
      WRITE(6,1164)
20    DO 220 K=1,NB
         V0(K)=CMPLX(0.,0.)
         V1(K)=V(K)-(Z1(K)/(ZF+Z1(IFP)+Z2(IFP)))*V(IFP)
         V2(K)=(Z2(K)/(ZF+Z1(IFP)+Z2(IFP)))*V(IFP)
         V0S(K)=V0(K)
         V1S(K)=V1(K)
         V2S(K)=V2(K)
         IF(SHIFT(K).NE.0)V1S(K)=V1(K)*A30**SHIFT(K)
         IF(SHIFT(K).NE.0)V2S(K)=V2(K)*A30**(-SHIFT(K))
         IF(ISQ.EQ.0)GO TO 220
         CALL MAGANG(V1S(K),V2S(K),V0S(K),VMAG1,VMAG2,VMAG0,
     1   ANG1,ANG2,ANG0)
         WRITE(KTERM,1184)IBUS(K),VMAG0,ANG0,VMAG1,ANG1,VMAG2,ANG2
         IF(ANSOUT.EQ.YES)WRITE(6,1184)IBUS(K),VMAG0,ANG0,VMAG1,ANG1,
     1                                  VMAG2,ANG2
220      CONTINUE
      IF(ISQ.NE.0)WRITE(KTERM,1204)
      IF((ISQ.NE.0).AND.(ANSOUT.EQ.YES))WRITE(6,1204)
      DO 320 N=1,NL
         DO 240 J=1,NB
            IF(IBUS(J).EQ.SB(N)) GO TO 260
240         CONTINUE
260      L=J
         DO 280 J=1,NB
            IF(IBUS(J).EQ.EB(N)) GO TO 300
280         CONTINUE
300      M=J
         ILF0=CMPLX(0.,0.)
         ILF1=(V1(L)-V1(M))*(-F1(L,M))
         ILF2=(V2(L)-V2(M))*(-F2(L,M))
         I0SF(N)=ILF0
         I1SF(N)=ILF1
         I2SF(N)=ILF2
         IF(SHIFT(L).NE.0)I1SF(N)=I1SF(N)*A30**SHIFT(L)
         IF(SHIFT(L).NE.0)I2SF(N)=I2SF(N)*A30**(-SHIFT(L))
```

```
      IF(ISQ.EQ.0)GO TO 320
      CALL MAGANG(I1SF(N),I2SF(N),IOSF(N),ILFM1,ILFM2,ILFMO,
   1  ANG1,ANG2,ANGO)
      WRITE(KTERM,1224)LINE(N),SB(N),EB(N),ILFMO,ANGO,ILFM1,
   1                 ANG1,ILFM2,ANG2
      IF(ANSOUT.EQ.YES)WRITE(6,1224)LINE(N),SB(N),EB(N),ILFMO,ANGO,
   1                             ILFM1,ANG1,ILFM2,ANG2
      IOSB(N)=ILFO
      I1SB(N)=-ILF1
      I2SB(N)=-ILF2
      IF(SHIFT(M).NE.0)I1SB(N)=I1SB(N)*A30**SHIFT(M)
      IF(SHIFT(M).NE.0)I2SB(N)=I2SB(N)*A30**(-SHIFT(M))
      CALL MAGANG(I1SB(N),I2SB(N),IOSB(N),ILFM1,ILFM2,ILFMO,
   1  ANG1,ANG2,ANGO)
      WRITE(KTERM,1234)LINE(N),EB(N),SB(N),ILFMO,ANGO,ILFM1,
   1                 ANG1,ILFM2,ANG2
      IF(ANSOUT.EQ.YES)WRITE(6,1234)LINE(N),EB(N),SB(N),ILFMO,
   1                             ANGO,ILFM1,ANG1,ILFM2,ANG2
320     CONTINUE
      WRITE(KTERM,1064)
      WRITE(KTERM,1084)  IBUS(IFP)
      WRITE(KTERM,1104)  ZF
      WRITE(KTERM,1124)  IFAULT, MAG
      WRITE(KTERM,1144)  SCC
      IF(ANSOUT.NE.YES)GO TO 340
      WRITE(6,1064)
      WRITE(6,1084)  IBUS(IFP)
      WRITE(6,1104)  ZF
      WRITE(6,1124)  IFAULT, MAG
      WRITE(6,1144)  SCC
340   CALL PHPRNT(NB,NL,IBUS,LINE,SB,EB,VOS,V1S,V2S,IOSF,I1SF,I2SF,
   1  IOSB,I1SB,I2SB)
C
1064  FORMAT('1',20X,'***** LINE TO LINE FAULT *****')
1084  FORMAT (1H0,'FAULT OCCURS AT BUS     ',I2)
1104  FORMAT(1H0,'FAULT IMPEDANCE=', 2F10.3)
1124  FORMAT (1H0,'IFAULT=',2F10.4,4X,'MAGNITUDE=',F10.4)
1144  FORMAT(1H0,'SHORT CIRCUIT CAPACITY=',F10.4)
1164  FORMAT(1H0,
   1  /,' POST FAULT BUS VOLTAGES',//,
   1  6X,'X',20('-'),' SEQUENCE QUANTITIES ',20('-'),'X'/
   2  ' BUS',2X,'X------ ZERO -------XX---- POSITIVE -----X',
   3  'X---- NEGATIVE -----X'/' NO.'6X,
   2  'V0',6X,'ANGLE 0',6X,'V1',6X,'ANGLE 1',6X,'V2',6X,'ANGLE 2'/)
1184  FORMAT (1H ,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,F10.2)
1204  FORMAT (1H0,/,' POST FAULT LINE CURRENTS',
   1  //,17X,'X',20('-'),' SEQUENCE QUANTITIES ',
   1  20('-'),'X',/17X,'X------ ZERO -------X',
   1  'X---- POSITIVE -----XX---- NEGATIVE -----X',/
   2  ' LINE',4X,'SB',2X,'EB',5X,
   2  'I0',6X,'ANGLE 0',6X,'I1',6X,'ANGLE 1',6X,'I2',6X,'ANGLE 2'/)
1224  FORMAT (1H ,I3,4X,I2,2X,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,
   1         F10.2)
1234  FORMAT (1H ,I3,4X,I2,2X,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,
   1         F10.2/)
C
      RETURN
      END
      SUBROUTINE FSLG(Z1,Z2,Z0,ZF,F1,F2,F0,ZSERO,V,HVTC,LVTC,IBUS,
   1  SB,EB,LINE,IFP,NB,NL,ISQ)
```

```
C*****************************************************************************
C
C       THE SUBROUTINE FSIG SOLVES FOR A SINGLE LINE TC GROUND FAULT
C       AT PARTICULAR BUS OF A POWER SYSTEM.
C
C*****************************************************************************
C
C       ARGUMENTS:
C
C       Z1 - THIS VECTOR CONTAINS THE POSITIVE SEQUENCE IMPEDANCES
C            OF ALL THE BUSES IN THE NETWORK WITH RESPECT TC THE
C            BUS THAT IS BEING FAULTED.
C
C       Z2 - THIS VECTOR CONTAINS THE NEGATIVE SEQUENCE IMPEDANCES
C            OF ALL THE BUSES IN THE NETWORK WITH RESPECT TC THE
C            BUS THAT IS BEING FAULTED.
C
C       Z0 - THIS VECTOR CONTAINS THE ZERO SEQUENCE IMPEDANCES
C            OF ALL THE BUSES IN THE NETWORK WITH RESPECT TC THE
C            BUS THAT IS BEING FAULTED.
C
C       ZF - THE IMPEDANCE OF THE FAULT AT THE PARTICULAR BUS BEING
C            FAULTED.
C
C       F1 - THE POSITIVE SEQUENCE ADMITTANCE MATRIX
C
C       F2 - THE NEGATIVE SEQUENCE ADMITTANCE MATRIX
C
C       F0 - THE ZERO SEQUENCE ADMITTANCE MATRIX
C
C       ZSERO - ZERO SEQUENCE IMPEDANCE OF LINE
C
C       V - THE PREFAULT BUS VOLTAGES FOR THE SYSTEM
C
C       HVTC - THIS INDICATES HOW A HIGH VOLTAGE SIDE OF A TRANSFORMER
C              IS CONNECTED 'D' FOR DELTA 'Y' FOR WYE.
C
C       LVTC - THIS INDICATES HOW A LOW VOLTAGE SIDE OF A TRANSFORMER
C              IS CONNECTED 'D' FOR DELTA 'Y' FOR WYE.
C
C       IBUS - THE NUMBERS OF THE BUSES IN THE SYSTEM
C
C       SB - THE STARTING BUS OF A LINE. ALSO THE HIGH VOLTAGE BUS OF A
C            TRANSFORMER
C
C       EB - THE ENDING BUS OF A LINE. ALSO THE LOW VOLTAGE BUS OF A
C            TRANSFORMER
C
C       LINE - THE NUMBER OF EACH LINE OR TRANSFORMER WITHIN THE SYSTEM
C
C       IFP - THE NUMBER OF THE BUS BEING FAULTED
C
C       NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C       NL - THE NUMBER OF LINES IN THE SYSTEM.
C
C       ISQ - THIS ARGUEMENT IS NONZERO IF THE SEQUENCE VOLTAGES AND
C             CURRENTS FOR THE FAULT ARE TO BE PRINTED OUT.
C
C*****************************************************************************
```

```
C
C       LOCAL VARIABLES:
C
C       V0,V1,V2 - THE COMPLEX VECTORS FOR THE ZERO, POSITIVE, AND
C                  NEGATIVE SEQUENCE VOLTAGES
C
C       VMAG0,VMAG1,VMAG2 - THE MAGNITUDES OF THE THREE SEQUENCE
C                           VOLTAGES
C
C       ANG0,ANG1,ANG2 - THE VOLTAGE ANGLES OF THE ZERO,POSITIVE,
C                        AND NEGATIVE SEQUENCE VOLTAGES AND CURRENTS
C
C       ILF0,ILF1,ILF2 - THE CURRENT FLOWING FORWARD ALONG A LINE
C                        (FROM STARTING BUS TO ENDING BUS)
C
C       ILFM0,ILFM1,ILFM2 - THE MAGNITUDE OF THE SEQUENCE CURRENT IN A
C                           LINE
C
C       V0S,V1S,V2S - THE COMPLEX VALUE OF THE VOLTAGE WITH PHASE
C                     SHIFTING DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                     TAKEN INTO ACCOUNT
C
C       I0SF,I1SF,I2SF - THE COMPLEX VALUE OF THE CURRENT WITH PHASE
C                        SHIFT DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                        TAKEN INTO ACCOUNT. THIS IS FOR CURRENTS GOING
C                        FROM STARTING BUS TO ENDING BUS.
C
C       I0SB,I1SB,I2SB - THE COMPLEX VALUE OF THE CURRENT WITH PHASE
C                        SHIFT DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                        TAKEN INTO ACCOUNT. THIS IS FOR CURRENTS GOING
C                        FROM ENDING BUS TO STARTING BUS
C
C       IFAULT - COMPLEX FAULT CUURENT
C
C       MAG - REAL MAGNITUDE OF IFAULT
C
C       SCC - SHORT CIRCUIT CAPACITANCE
C
C       SHIFT - THIS INTEGER VECTOR IS RETURNED BY DYFIX. IT CONTAINS
C               THE NUMBER OF TIMES A BUS VOLATGE IS TO BE SHIFTED BY
C               30 DEGREES.
C
C       A30 - THIS CONSTANT IS TAKEN TO THE POWER OF THE VALUE IN SHIFT
C             FOR THAT BUS
C
C*****************************************************************************
C
C       SUBROUTINES CALLED:
C
C          DYFIX,PHPRNT
C
      COMPLEX   Z1(24)
      COMPLEX   Z2(24)
      COMPLEX   Z0(24)
      COMPLEX   F1(24,24)
      COMPLEX   F2(24,24)
      COMPLEX   F0(24,24)
      COMPLEX   ZSERO(50)
      COMPLEX   IFAULT
      COMPLEX   V0(24)
```

```
      COMPLEX    V1(24)
      COMPLEX    V2(24)
      COMPLEX    ILF0
      COMPLEX    ILF1
      COMPLEX    ILF2
      COMPLEX    ZF
      COMPLEX    V(24)
      COMPLEX    I0SF(50)
      COMPLEX    I1SF(50)
      COMPLEX    I2SF(50)
      COMPLEX    I0SB(50)
      COMPLEX    I1SB(50)
      COMPLEX    I2SB(50)
      COMPLEX    V0S(24)
      COMPLEX    V1S(24)
      COMPLEX    V2S(24)
      COMPLEX    A30
      REAL    HVTC(24)
      REAL    LVTC(24)
      REAL    MAG
      REAL    ILFM0
      REAL    ILFM1
      REAL    ILFM2
      REAL    Y
      INTEGER    SB(50)
      INTEGER    EB(50)
      INTEGER    IBUS(24)
      INTEGER    SHIFT(24)
      INTEGER    LINE(50)
      LOGICAL    PRINT
      COMMON /INOU/ KTERM
      COMMON /OUTPUT/ ANSOUT,PRINT,ANSIN
      DATA    YES/'YES'/
      DATA    Y/'Y'/
      A30=CMPLX(0.8660254,0.5)
      CALL DYFIX(SHIFT,HVTC,LVTC,SB,EB,LINE,IBUS,NB,NL,I)
      IFAULT=V(IFP)/(Z1(IFP)+Z2(IFP)+Z0(IFP)+3.*ZF)
      MAG=CABS(IFAULT)
      SCC =CABS(V(IFP))*MAG
      IF(ISQ.EQ. 0)GO TO 20
      WRITE(KTERM,1064)
      WRITE(KTERM,1084) IBUS(IFP)
      WRITE(KTERM,1104) ZF
      WRITE(KTERM,1124) IFAULT, MAG
      WRITE(KTERM,1144) SCC
      WRITE(KTERM,1164)
      IF(ANSOUT.NE.YES)GO TO 20
      WRITE(6,1064)
      WRITE(6,1084) IBUS(IFP)
      WRITE(6,1104) ZF
      WRITE(6,1124) IFAULT, MAG
      WRITE(6,1144) SCC
      WRITE(6,1164)
20    DO 220 K=1,NB
         V0(K)=-(Z0(K)/(3.*ZF+Z1(IFP)+Z2(IFP)+Z0(IFP)))*V(IFP)
         V1(K)=V(K)-(Z1(K)/(3.*ZF+Z1(IFP)+Z2(IFP)+Z0(IFP)))*V(IFP)
         V2(K)=-(Z2(K)/(3.*ZF+Z1(IFP)+Z2(IFP)+Z0(IFP)))*V(IFP)
         V0S(K)=V0(K)
         V1S(K)=V1(K)
         V2S(K)=V2(K)
```

```
            IF(SHIFT(K).NE.0) V1S(K)=V1(K)*A30**SHIFT(K)
            IF(SHIFT(K).NE.0) V2S(K)=V2(K)*A30**(-SHIFT(K))
            IF(ISQ.EQ.0) GO TO 220
            CALL MAGANG(V1S(K),V2S(K),V0S(K),VMAG1,VMAG2,VMAG0,
     1      ANG1,ANG2,ANG0)
            WRITE(KTERM,1184) IBUS(K),VMAG0,ANG0,VMAG1,ANG1,VMAG2,ANG2
            IF(ANSOUT.EQ.YES) WRITE(6,1184) IBUS(K),VMAG0,ANG0,VMAG1,ANG1,
     1                                          VMAG2,ANG2
220         CONTINUE
         IF(ISQ.NE.0) WRITE(KTERM,1204)
         IF((ISQ.NE.0).AND.(ANSOUT.EQ.YES))WRITE(6,1204)
         DO 320 N=1,NL
            DO 240 J=1,NB
               IF(IBUS(J).EQ.SB(N))  GO TO 260
240            CONTINUE
260         L=J
            DO 280 J=1,NB
               IF(IBUS(J).EQ.EB(N))  GO TO 300
280            CONTINUE
300         M=J
            ILF0=(V0(L)-V0(M))*(-F0(L,M))
            IF((CABS(F0(L,M)).EQ. 0.).AND.(HVTC(N).EQ.Y))
     1      ILF0=(V0(L)-V0(M))/ZSERO(N)
            ILF1=(V1(L)-V1(M))*(-F1(L,M))
            ILF2=(V2(L)-V2(M))*(-F2(L,M))
            I0SF(N)=ILF0
            I1SF(N)=ILF1
            I2SF(N)=ILF2
            IF(SHIFT(L).NE.0) I1SF(N)=I1SF(N)*A30**SHIFT(L)
            IF(SHIFT(L).NE.0) I2SF(N)=I2SF(N)*A30**(-SHIFT(L))
            IF(ISQ.EQ.0) GO TO 320
            CALL MAGANG(I1SF(N),I2SF(N),I0SF(N),ILFM1,ILFM2,ILFM0,
     1      ANG1,ANG2,ANG0)
            WRITE(KTERM,1224) LINE(N),SB(N),EB(N),ILFM0,ANG0,ILFM1,
     1                       ANG1,ILFM2,ANG2
            IF(ANSOUT.EQ.YES) WRITE(6,1224) LINE(N),SB(N),EB(N),ILFM0,ANG0,
     1                                         ILFM1,ANG1,ILFM2,ANG2
            I0SB(N)=(V0(M)-V0(L))*(-F0(M,L))
            IF((CABS(F0(M,L)).EQ. 0.).AND.(LVTC(N).EQ.Y))
     1      I0SB(N)=(V0(M)-V0(L))/ZSERO(N)
            I1SB(N)=-ILF1
            I2SB(N)=-ILF2
            IF(SHIFT(M).NE.0) I1SB(N)=I1SB(N)*A30**SHIFT(M)
            IF(SHIFT(M).NE.0) I2SB(N)=I2SB(N)*A30**(-SHIFT(M))
            CALL MAGANG(I1SB(N),I2SB(N),I0SB(N),ILFM1,ILFM2,ILFM0,
     1      ANG1,ANG2,ANG0)
            WRITE(KTERM,1234) LINE(N),EB(N),SB(N),ILFM0,ANG0,ILFM1,
     1                       ANG1,ILFM2,ANG2
            IF(ANSOUT.EQ.YES) WRITE(6,1234) LINE(N),EB(N),SB(N),ILFM0,
     1                                         ANG0,ILFM1,ANG1,ILFM2,ANG2
320         CONTINUE
         WRITE(KTERM,1064)
         WRITE(KTERM,1084)  IBUS(IFP)
         WRITE(KTERM,1104)  ZF
         WRITE(KTERM,1124)  IFAULT, MAG
         WRITE(KTERM,1144)  SCC
         IF(ANSOUT.NE.YES) GO TO 340
         WRITE(6,1064)
         WRITE(6,1084)  IBUS(IFP)
         WRITE(6,1104)  ZF
```

```
      WRITE(6,1124) IFAULT, MAG
      WRITE(6,1144) SCC
340   CALL PHPENT(NB,NL,IBUS,LINE,SB,EB,V0S,V1S,V2S,I0SF,I1SF,I2SF,
     1 I0SB,I1SB,I2SB)
C
1064  FORMAT('1',20X,'***** SINGLE LINE TO GROUND FAULT *****')
1084  FORMAT (1H0,'FAULT OCCURS AT BUS     ',I2)
1104  FORMAT(1H0,'FAULT IMPEDANCE=', 2F10.3)
1124  FORMAT (1H0,'IFAULT=',2F10.4,4X,'MAGNITUDE=',F10.4)
1144  FORMAT(1H0,'SHORT CIRCUIT CAPACITY=',F10.4)
1164  FORMAT(1H0,
     1 /,' POST FAULT BUS VOLTAGES',//,
     1 6X,'X',20('-'),' SEQUENCE QUANTITIES ',20('-'),'X'/
     2 ' BUS',2X,'X------ ZERO -------XX---- POSITIVE -----X',
     3 'X---- NEGATIVE -----X'/' NO.'6X,
     2 'V0',6X,'ANGLE 0',6X,'V1',6X,'ANGLE 1',6X,'V2',6X,'ANGLE 2'/)
1184  FORMAT (1H ,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,F10.2)
1204  FORMAT (1H0,/,' POST FAULT LINE CURRENTS',
     1 //,17X,'X',20('-'),' SEQUENCE QUANTITIES ',
     1 20('-'),'X',/17X,'X------ ZERO -------X',
     1 'X---- POSITIVE -----XX---- NEGATIVE -----X',/
     2 ' LINE',4X,'SB',2X,'EB',5X,
     2 'I0',6X,'ANGLE 0',6X,'I1',6X,'ANGLE 1',6X,'I2',6X,'ANGLE 2'/)
1224  FORMAT (1H ,I3,4X,I2,2X,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,
     1        F10.2)
1234  FORMAT (1H ,I3,4X,I2,2X,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,
     1        F10.2/)
C
      RETURN
      END
      SUBROUTINE F2LG(Z1,Z2,Z0,ZF,F1,F2,FC,ZSERO,V,HVTC,LVTC,IBUS,
     1 SB,EB,LINE,IFP,NB,NL,ISQ)
C*********************************************************************
C
C       THE SUBROUTINE F2LG SOLVES FOR A DOUBLE LINE TO GROUND FAULT
C       AT PARTICULAR BUS OF A POWER SYSTEM.
C
C*********************************************************************
C
C       ARGUMENTS:
C
C       Z1 - THIS VECTOR CONTAINS THE POSITIVE SEQUENCE IMPEDANCES
C            OF ALL THE BUSES IN THE NETWORK WITH RESPECT TO THE
C            BUS THAT IS BEING FAULTED.
C
C       Z2 - THIS VECTOR CONTAINS THE NEGATIVE SEQUENCE IMPEDANCES
C            OF ALL THE BUSES IN THE NETWORK WITH RESPECT TO THE
C            BUS THAT IS BEING FAULTED.
C
C       Z0 - THIS VECTOR CONTAINS THE ZERO SEQUENCE IMPEDANCES
C            OF ALL THE BUSES IN THE NETWORK WITH RESPECT TO THE
C            BUS THAT IS BEING FAULTED.
C
C       ZF - THE IMPEDANCE OF THE FAULT AT THE PARTICULAR BUS BEING
C            FAULTED.
C
C       F1 - THE POSITIVE SEQUENCE ADMITTANCE MATRIX
C
C       F2 - THE NEGATIVE SEQUENCE ADMITTANCE MATRIX
C
```

```
C        FO - THE ZERO SEQUENCE ADMITTANCE MATRIX
C
C        ZSERO - ZERO SEQUENCE IMPEDANCE OF LINE
C
C        V - THE PREFAULT BUS VOLTAGES FOR THE SYSTEM
C
C        HVTC - THIS INDICATES HOW A HIGH VOLTAGE SIDE OF A TRANSFORMER
C               IS CONNECTED 'D' FOR DELTA 'Y' FOR WYE.
C
C        LVTC - THIS INDICATES HOW A LOW VOLTAGE SIDE OF A TRANSFORMER
C               IS CONNECTED 'D' FOR DELTA 'Y' FOR WYE.
C
C        IBUS - THE NUMBERS OF THE BUSES IN THE SYSTEM
C
C        SB - THE STARTING BUS OF A LINE. ALSO THE HIGH VOLTAGE BUS OF A
C             TRANSFORMER
C
C        EB - THE ENDING BUS OF A LINE. ALSO THE LOW VOLTAGE BUS OF A
C             TRANSFORMER
C
C        LINE - THE NUMBER OF EACH LINE OR TRANSFORMER WITHIN THE SYSTEM
C
C        IFP - THE NUMBER OF THE BUS BEING FAULTED
C
C        NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C        NL - THE NUMBER OF LINES IN THE SYSTEM.
C
C        ISQ - THIS ARGUEMENT IS NONZERO IF THE SEQUENCE VOLTAGES AND
C              CURRENTS FOR THE FAULT ARE TO BE PRINTED OUT.
C
C***********************************************************************
C
C        LOCAL VARIABLES:
C
C        V0,V1,V2 - THE COMPLEX VECTORS FOR THE ZERO, POSITIVE, AND
C                   NEGATIVE SEQUENCE VOLTAGES
C
C        VMAG0,VMAG1,VMAG2 - THE MAGNITUDES OF THE THREE SEQUENCE
C                            VOLTAGES
C
C        ANG0,ANG1,ANG2 - THE VOLTAGE ANGLES OF THE ZERO,POSITIVE,
C                         AND NEGATIVE SEQUENCE VOLTAGES AND CURRENTS
C
C        ILF0,ILF1,ILF2 - THE CURRENT FLOWING FORWARD ALONG A LINE
C                         (FROM STARTING BUS TO ENDING BUS)
C
C        ILFM0,ILFM1,ILFM2 - THE MAGNITUDE OF THE SEQUENCE CURRENT IN A
C                            LINE
C
C        V0S,V1S,V2S - THE COMPLEX VALUE OF THE VOLTAGE WITH PHASE
C                      SHIFTING DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                      TAKEN INTO ACCOUNT
C
C        I0SF,I1SF,I2SF - THE COMPLEX VALUE OF THE CURRENT WITH PHASE
C                         SHIFT DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                         TAKEN INTO ACCOUNT. THIS IS FOR CURRENTS GOING
C                         FROM STARTING BUS TO ENDING BUS.
C
C        I0SB,I1SB,I2SB - THE COMPLEX VALUE OF THE CURRENT WITH PHASE
```

```
C                        SHIFT DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                        TAKEN INTO ACCCUNT. THIS IS FOR CURRENTS GOING
C                        FRCM ENDING BUS TC STARTING BUS
C
C       IFAULT - COMPLEX FAULT CUURENT
C
C       MAG - REAL MAGNITUDE OF IFAULT
C
C       SCC - SHORT CIRCUIT CAPACITANCE
C
C       SHIFT - THIS INTEGER VECTOR IS RETURNED BY DYFIX. IT CONTAINS
C               THE NUMBER OF TIMES A BUS VCLATGE IS TO BE SHIFTED BY
C               30 DEGREES.
C
C       A30 - THIS CONSTANT IS TAKEN TC THE POWER OF THE VALUE IN SHIFT
C             FOR THAT BUS
C
C***********************************************************************
C
C       SUBROUTINES CALLED:
C
C         DYFIX,PHERNT
C
        COMPLEX   Z1(24)
        COMPLEX   Z2(24)
        COMPLEX   Z0(24)
        COMPLEX   F1(24,24)
        COMPLEX   F2(24,24)
        COMPLEX   F0(24,24)
        COMPLEX   ZSERO(50)
        COMPLEX   IFAULT
        COMPLEX   V0(24)
        COMPLEX   V1(24)
        COMPLEX   V2(24)
        COMPLEX   ILF0
        COMPLEX   ILF1
        COMPLEX   ILF2
        COMPLEX   ZF
        COMPLEX   V(24)
        COMPLEX   I0SF(50)
        COMPLEX   I1SF(50)
        COMPLEX   I2SF(50)
        COMPLEX   I0SB(50)
        COMPLEX   I1SB(50)
        COMPLEX   I2SB(50)
        COMPLEX   V0S(24)
        COMPLEX   V1S(24)
        COMPLEX   V2S(24)
        COMPLEX   A30
        COMPLEX   DELT
        REAL    HVTC(24)
        REAL    LVTC(24)
        REAL    MAG
        REAL    ILFM0
        REAL    ILFM1
        REAL    ILFM2
        INTEGER   SB(50)
        INTEGER   EB(50)
        INTEGER   IBUS(24)
        INTEGER   SHIFT(24)
```

```
      INTEGER   LINE(50)
      LOGICAL   PRINT
      COMMON /INOU/ KTERM
      COMMON /OUTPUT/ ANSOUT,PRINT,ANSIN
      DATA    YES/'YES'/
      A30=CMPLX(0.8660254,0.5)
      CALL DYFIX(SHIFT,HVTC,LVTC,SE,EE,LINE,IBUS,NB,NL,I)
      DELT=Z1(IFP)*Z2(IFP)+Z1(IFP)*(3.*ZF+Z0(IFP))+Z2(IFP)*
     1 (3.*ZF+Z0(IFP))
      IFAULT=((Z2(IFP)+3.*ZF+Z0(IFP))*V(IFP))/DELT
      MAG=CABS(IFAULT)
      SCC =CABS(V(IFP))*MAG
      IF(ISQ.EQ. 0)GO TO 20
      WRITE(KTERM,1064)
      WRITE(KTERM,1084) IBUS(IFP)
      WRITE(KTERM,1104) ZF
      WRITE(KTERM,1124) IFAULT, MAG
      WRITE(KTERM,1144) SCC
      WRITE(KTERM,1164)
      IF(ANSOUT.NE.YES)GO TO 20
      WRITE(6,1064)
      WRITE(6,1084) IBUS(IFP)
      WRITE(6,1104) ZF
      WRITE(6,1124) IFAULT, MAG
      WRITE(6,1144) SCC
      WRITE(6,1164)
20    DO 220 K=1,NB
         V0(K)=Z0(K)*Z2(IFP)*V(K)/DELT
         V1(K)=((DELT-Z1(K)*(Z2(IFP)+3.*ZF+Z0(IFP)))/DELT)*V(K)
         V2(K)=Z2(K)*(3.*ZF+Z0(IFP))*V(K)/DELT
         V0S(K)=V0(K)
         V1S(K)=V1(K)
         V2S(K)=V2(K)
         IF(SHIFT(K).NE.0) V1S(K)=V1(K)*A30**SHIFT(K)
         IF(SHIFT(K).NE.0) V2S(K)=V2(K)*A30**(-SHIFT(K))
         IF(ISQ.EQ.0)GO TO 220
         CALL MAGANG(V1S(K),V2S(K),V0S(K),VMAG1,VMAG2,VMAG0,
     1   ANG1,ANG2,ANG0)
         WRITE(KTERM,1184)IBUS(K),VMAG0,ANG0,VMAG1,ANG1,VMAG2,ANG2
         IF(ANSOUT.EQ.YES) WRITE(6,1184) IBUS(K),VMAG0,ANG0,VMAG1,ANG1,
     1                                   VMAG2,ANG2
220      CONTINUE
      IF(ISQ.NE.0)WRITE(KTERM,1204)
      IF((ISQ.NE.0).AND.(ANSOUT.EQ.YES))WRITE(6,1204)
      DO 320 N=1,NL
         DO 240 J=1,NB
            IF(IBUS(J).EQ.SB(N)) GO TO 260
240         CONTINUE
260      L=J
         DO 280 J=1,NB
            IF(IBUS(J).EQ.EB(N)) GO TO 300
280         CONTINUE
300      M=J
         ILF0=(V0(L)-V0(M))*(-F0(L,M))
         IF((CABS(F0(L,M)).EQ. 0.).AND.(HVTC(N).EQ.Y))
     1   ILF0=(V0(L)-V0(M))/ZSERO(N)
         ILF1=(V1(L)-V1(M))*(-F1(L,M))
         ILF2=(V2(L)-V2(M))*(-F2(L,M))
         I0SF(N)=ILF0
         I1SF(N)=ILF1
```

```
         I2SF(N)=ILF2
         IF(SHIFT(L).NE.0)I1SF(N)=I1SF(N)*A30**SHIFT(L)
         IF(SHIFT(L).NE.0)I2SF(N)=I2SF(N)*A30**(-SHIFT(L))
         IF(ISQ.EQ.0)GO TO 320
         CALL MAGANG(I1SF(N),I2SF(N),IOSF(N),ILFM1,ILFM2,ILFM0,
     1   ANG1,ANG2,ANG0)
         WRITE(KTERM,1224)LINE(N),SB(N),EB(N),ILFM0,ANG0,ILFM1,
     1                  ANG1,ILFM2,ANG2
         IF(ANSOUT.EQ.YES)WRITE(6,1224)LINE(N),SB(N),EB(N),ILFM0,ANG0,
     1                             ILFM1,ANG1,ILFM2,ANG2
         IOSB(N)=(VO(M)-VO(L))*(-FO(M,L))
         IF((CABS(FO(M,L)).EQ. 0.).AND.(LVIC(N).EQ.Y))
     1   IOSB(N)=(VO(M)-VO(L))/ZSERO(N)
         I1SB(N)=-ILF1
         I2SB(N)=-ILF2
         IF(SHIFT(M).NE.0)I1SB(N)=I1SB(N)*A30**SHIFT(M)
         IF(SHIFT(M).NE.0)I2SB(N)=I2SB(N)*A30**(-SHIFT(M))
         CALL MAGANG(I1SB(N),I2SB(N),IOSB(N),ILFM1,ILFM2,ILFM0,
     1   ANG1,ANG2,ANG0)
         WRITE(KTERM,1234)LINE(N),EB(N),SB(N),ILFM0,ANG0,ILFM1,
     1                  ANG1,ILFM2,ANG2
         IF(ANSOUT.EQ.YES)WRITE(6,1234)LINE(N),EB(N),SB(N),ILFM0,
     1                             ANG0,ILFM1,ANG1,ILFM2,ANG2
 320     CONTINUE
      WRITE(KTERM,1064)
      WRITE(KTERM,1084)  IBUS(IFP)
      WRITE(KTERM,1104)  ZF
      WRITE(KTERM,1124)  IFAULT, MAG
      WRITE(KTERM,1144)  SCC
      IF(ANSOUT.NE.YES)GO TO 340
      WRITE(6,1064)
      WRITE(6,1084)  IBUS(IFP)
      WRITE(6,1104)  ZF
      WRITE(6,1124)  IFAULT, MAG
      WRITE(6,1144)  SCC
 340  CALL PHPRNT(NB,NL,IBUS,LINE,SB,EB,VOS,V1S,V2S,IOSF,I1SF,I2SF,
     1 IOSB,I1SB,I2SB)
C
 1064 FORMAT('1',20X,'***** DOUBLE LINE TO GROUND FAULT *****')
 1084 FORMAT (1H0,'FAULT OCCURS AT BUS    ',I2)
 1104 FORMAT(1H0,'FAULT IMPEDANCE=', 2F10.3)
 1124 FORMAT  (1H0,'IFAULT=',2F10.4,4X,'MAGNITUDE=',F10.4)
 1144 FORMAT(1H0,'SHORT CIRCUIT CAPACITY=',F10.4)
 1164 FORMAT(1H0,
     1 /,' POST FAULT BUS VOLTAGES',//,
     1 6X,'X',20('-'),' SEQUENCE QUANTITIES ',20('-'),'X'/
     2 ' BUS',2X,'X------ ZERO -------XX---- POSITIVE -----X',
     3 'X---- NEGATIVE -----X'/' NO.'6X,
     2 'V0',6X,'ANGLE 0',6X,'V1',6X,'ANGLE 1',6X,'V2',6X,'ANGLE 2'/)
 1184 FORMAT (1H ,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,F10.2)
 1204 FORMAT (1H0,/,' POST FAULT LINE CURRENTS',
     1 //,17X,'X',20('-'),' SEQUENCE QUANTITIES ',
     1 20('-'),'X',/17X,'X------ ZERO -------X',
     1 'X---- POSITIVE -----XX---- NEGATIVE -----X',/
     2 ' LINE',4X,'SB',2X,'EB',5X,
     2 'I0',6X,'ANGLE 0',6X,'I1',6X,'ANGLE 1',6X,'I2',6X,'ANGLE 2'/)
 1224 FORMAT (1H ,I3,4X,I2,2X,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,
     1        F10.2)
 1234 FORMAT (1H ,I3,4X,I2,2X,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,
     1        F10.2/)
```

```
C
      RETURN
      END
      SUBROUTINE PHPRNT(NB,NL,IBUS,LINE,SB,EB,VOS,V1S,V2S,IOSF,I1SF,
     1 I2SF,IOSB,I1SB,I2SB)
C***********************************************************************
C
C     THE SUBROUTINE PHPRNT PRINTS OUT THE PHASE VOLTAGES AND
C     CURRENTS FOR THE FAULTED SYSTEM
C
C***********************************************************************
C
C     ARGUMENTS:
C
C     NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C     NL - THE NUMBER OF LINES IN THE SYSTEM.
C
C     IBUS - THE NUMBERS OF THE BUSES IN THE SYSTEM
C
C     LINE - THE NUMBER OF EACH LINE OR TRANSFORMER WITHIN THE SYSTEM
C
C     SB - THE STARTING BUS OF A LINE. ALSO THE HIGH VOLTAGE BUS OF A
C          TRANSFORMER
C
C     EB - THE ENDING BUS OF A LINE. ALSO THE LOW VOLTAGE BUS OF A
C          TRANSFORMER
C
C     VOS,V1S,V2S - THE COMPLEX VALUE OF THE VOLTAGE WITH PHASE
C                   SHIFTING DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                   TAKEN INTO ACCOUNT
C
C     IOSF,I1SF,I2SF - THE COMPLEX VALUE OF THE CURRENT WITH PHASE
C                      SHIFT DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                      TAKEN INTO ACCOUNT. THIS IS FOR CURRENTS GOING
C                      FROM STARTING BUS TO ENDING BUS.
C
C     IOSB,I1SB,I2SB - THE COMPLEX VALUE OF THE CURRENT WITH PHASE
C                      SHIFT DUE TO DELTA-WYE TRANSFORMER CONNECTIONS
C                      TAKEN INTO ACCOUNT. THIS IS FOR CURRENTS GOING
C                      FROM ENDING BUS TO STARTING BUS
C
C***********************************************************************
C
C     LOCAL VARIABLES:
C
C     VA,VB,VC - THE COMPLEX VECTORS FOR THE THREE PHASES OF THE
C                VOLTAGE.
C
C     VMAGA,VMAGB,VMAGC - THE MAGNITUDES OF THE THREE PHASE
C                         VOLTAGES
C
C     ANGA,ANGB,ANGC - THE ANGLES OF THE THREE PHASES OF
C                      VOLTAGE AND CURRENT
C
C     ILFMA,ILFMB,ILFMC - THE MAGNITUDE OF THE PHASE CURRENT IN A
C                         LINE
C
C     ILFAF,ILFBF,ILFCF - THE COMPLEX VALUE OF THE PHASE CURRENT WITH
C                         PHASE SHIFT DUE TO DELTA-WYE TRANSFORMER
```

```
C                           CONNECTIONS TAKEN INTO ACCOUNT. THIS IS FOR
C                           CURRENTS GOING FROM STARTING BUS TO ENDING
C                           BUS
C
C       ILFAB,ILFBB,ILFCB - THE COMPLEX VALUE OF THE PHASE CURRENT WITH
C                           PHASE SHIFT DUE TO DELTA-WYE TRANSFORMER
C                           CONNECTIONS TAKEN INTO ACCOUNT. THIS IS FOR
C                           CURRENTS GOING FROM ENDING BUS TO STARTING
C                           BUS
C
C***********************************************************************
C
C       SUBROUTINES CALLED:
C
C          SQPHAS
C
        COMPLEX   VA(24)
        COMPLEX   VB(24)
        COMPLEX   VC(24)
        COMPLEX   ILFAF(50)
        COMPLEX   ILFBF(50)
        COMPLEX   ILFCF(50)
        COMPLEX   ILFAB(50)
        COMPLEX   ILFBB(50)
        COMPLEX   ILFCB(50)
        COMPLEX   I0SF(50)
        COMPLEX   I1SF(50)
        COMPLEX   I2SF(50)
        COMPLEX   I0SB(50)
        COMPLEX   I1SB(50)
        COMPLEX   I2SB(50)
        COMPLEX   V0S(24)
        COMPLEX   V1S(24)
        COMPLEX   V2S(24)
        REAL   ILFMA
        REAL   ILFMB
        REAL   ILFMC
        INTEGER   SB(50)
        INTEGER   EB(50)
        INTEGER   IBUS(24)
        INTEGER   LINE(50)
        COMMON /INOU/ KTERM
        COMMON /OUTPUT/ ANSOUT,PRINT,ANSIN
        DATA   YES/'YES  '/
        WRITE(KTERM,1244)
        IF(ANSOUT.EQ.YES) WRITE(6,1244)
        CALL SQPHAS(V0S,V1S,V2S,VA,VB,VC,NB)
        DO 321 K=1,NB
           VMAGA=CABS(VA(K))
           VMAGB=CABS(VB(K))
           VMAGC=CABS(VC(K))
           ANGA=0.
           IF((REAL(VA(K)).NE. 0.).OR.(AIMAG(VA(K)).NE. 0.))
     1     ANGA=ATAN2(AIMAG(VA(K)),REAL(VA(K)))*57.29578
           ANGB=0.
           IF((REAL(VB(K)).NE. 0.).OR.(AIMAG(VB(K)).NE. 0.))
     1     ANGB=ATAN2(AIMAG(VB(K)),REAL(VB(K)))*57.29578
           ANGC=0.
           IF((REAL(VC(K)).NE. 0.).OR.(AIMAG(VC(K)).NE. 0.))
     1     ANGC=ATAN2(AIMAG(VC(K)),REAL(VC(K)))*57.29578
```

```
            WRITE(KTERM,1184) IBUS(K),VMAGA,ANGA,VMAGB,ANGB,VMAGC,ANGC
            IF(ANSOUT.EQ.YES) WRITE(6,1184) IBUS(K),VMAGA,ANGA,VMAGB,
     1                       ANGB,VMAGC,ANGC
 321        CONTINUE
         CALL SQPHAS(IOSF,I1SF,I2SF,ILFAF,ILFBF,ILFCF,NL)
         CALL SQPHAS(IOSB,I1SB,I2SB,ILFAB,ILFBB,ILFCB,NL)
         WRITE(KTERM,1264)
         IF(ANSOUT.EQ.YES) WRITE(6,1264)
         DO 322 N=1,NL
            DO 241 J=1,NE
               IF(IBUS(J).EQ.SB(N)) GO TO 261
 241           CONTINUE
 261        L=J
            DO 281 J=1,NE
               IF(IBUS(J).EQ.EB(N)) GO TO 301
 281           CONTINUE
 301        M=J
            ILFMA=CABS(ILFAF(N))
            ILFMB=CABS(ILFBF(N))
            ILFMC=CABS(ILFCF(N))
            ANGA=0.
            IF((REAL(ILFAF(N)).NE. 0.).OR.(AIMAG(ILFAF(N)).NE. 0.))
     1      ANGA=ATAN2(AIMAG(ILFAF(N)),REAL(ILFAF(N)))*57.29578
            ANGB=0.
            IF((REAL(ILFBF(N)).NE. 0.).OR.(AIMAG(ILFBF(N)).NE. 0.))
     1      ANGB=ATAN2(AIMAG(ILFBF(N)),REAL(ILFBF(N)))*57.29578
            ANGC=0.
            IF((REAL(ILFCF(N)).NE. 0.).OR.(AIMAG(ILFCF(N)).NE. 0.))
     1      ANGC=ATAN2(AIMAG(ILFCF(N)),REAL(ILFCF(N)))*57.29578
            WRITE(KTERM,1224) LINE(N),SB(N),EB(N),ILFMA,ANGA,ILFMB,
     1      ANGB,ILFMC,ANGC
            IF(ANSOUT.EQ.YES) WRITE(6,1224) LINE(N),SB(N),EB(N),ILFMA,
     1                       ANGA,ILFMB,ANGB,ILFMC,ANGC
            ILFMA=CABS(ILFAB(N))
            ILFMB=CABS(ILFBB(N))
            ILFMC=CABS(ILFCB(N))
            ANGA=0.
            IF((REAL(ILFAB(N)).NE. 0.).OR.(AIMAG(ILFAB(N)).NE. 0.))
     1      ANGA=ATAN2(AIMAG(ILFAB(N)),REAL(ILFAB(N)))*57.29578
            ANGB=0.
            IF((REAL(ILFBB(N)).NE. 0.).OR.(AIMAG(ILFBB(N)).NE. 0.))
     1      ANGB=ATAN2(AIMAG(ILFBB(N)),REAL(ILFBB(N)))*57.29578
            ANGC=0.
            IF((REAL(ILFCB(N)).NE. 0.).OR.(AIMAG(ILFCB(N)).NE. 0.))
     1      ANGC=ATAN2(AIMAG(ILFCB(N)),REAL(ILFCB(N)))*57.29578
            WRITE(KTERM,1234) LINE(N),EB(N),SB(N),ILFMA,ANGA,ILFMB,
     1      ANGB,ILFMC,ANGC
            IF(ANSOUT.EQ.YES) WRITE(6,1224) LINE(N),EB(N),SB(N),ILFMA,
     1                       ANGA,ILFMB,ANGB,ILFMC,ANGC
 322        CONTINUE
C
 1184    FORMAT (1H ,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,F10.2)
 1224    FORMAT (1H ,I3,4X,I2,2X,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,
     1           F10.2)
 1234    FORMAT (1H ,I3,4X,I2,2X,I2,2X,F10.4,F10.2,F11.4,F10.2,F11.4,
     1           F10.2/)
 1244    FORMAT(1H0,
     1   /,' POST FAULT BUS VOLTAGES',//,
     1   6X,'X',22('-'),' PHASE QUANTITIES ',21('-'),'X',/
     2   ' BUS',2X,'X----- PHASE A -----XX----- PHASE B -----X',
```

```
     3  'X----- PHASE C -----X',/' NO.',6X,
     2  'VA',6X,'ANGLE A',6X,'VB',6X,'ANGLE B',6X,'VC',6X,'ANGLE C'/)
1264    FORMAT (1H0,/,' POST FAULT LINE CURRENTS',
     1  //,17X,'X',22('-'),' PHASE QUANTITIES ',21('-'),
     1  'X',/,17X,'X----- PHASE A -----X',
     1  'X----- PHASE B -----XX----- PHASE C -----X',/
     2  ' LINE',4X,'SB',2X,'EB',5X,
     C  'IA',6X,'ANGLE A',6X,'IB',6X,'ANGLE B',6X,'IC',6X,'ANGLE C'/)
C
        RETURN
        END
        SUBROUTINE DYFIX(M,HVTC,LVTC,SB,EB,LINE,IBUS,NB,NL,FB)
C*******************************************************************
C
C       THE SUBROUTINE DYFIX CALCULATES THE NUMBER OF TIMES A PHASE
C       SHIFT OF 30 DEGREES MUST BE APPLIED TO EACH BUS DUE TO THE
C       PRESENCE OF DELTA WYE TRANSFORMER CONNECTIONS
C
C*******************************************************************
C
C       ARGUMENTS:
C
C       M - THIS VECTOR RETURNS THE NUMBER OF TIMES EACH BUS VOLTAGE
C           ANGLE MUST BE SHIFTED BY 30 DEGREES
C
C       HVTC - THIS VECTOR INDICATES WHETHER OR NOT THE HIGH VOLTAGE
C              BUS SIDE OF A TRANSFORMER WAS CONNECTED DELTA OR WYE
C
C       LVTC - THIS VECTOR INDICATES WHETHER OR NOT THE LOW VOLTAGE
C              BUS SIDE OF A TRANSFORMER WAS CONNECTED DELTA OR WYE
C
C       SB - THIS VECTOR IS THE HIGH VOLTAGE BUS CONNECTION SIDE FOR A
C            TRANSFORMER
C
C       EB - THIS VECTOR IS THE LOW VOLTAGE BUS CONNECTION SIDE FOR A
C            TRANSFORMER
C
C       LINE - THIS VECTOR CONTAINS ALL THE LINE NUMBERS FOR THE SYSTEM
C
C       IBUS - THIS VECTOR CONTAINS ALL THE BUS NUMBERS FOR THE SYSTEM
C
C       NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C       NL - THE NUMBER OF LINES IN THE SYSTEM
C
C       FB - INTEGER NUMBER OF THE BUS BEING FAULTED
C
C*******************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        REAL     HVTC(50)
        REAL     LVTC(50)
        INTEGER    M(24)
        INTEGER    SB(50)
        INTEGER    EB(50)
        INTEGER    IBUS(24)
        INTEGER    FB
```

```
      DATA D/'D'/
      DATA Y/'Y'/
      DO 20 I=1,NB
20       M(I)=1000
      M(1)=0
40       DO 160 N=1,NL
         IS=SB(N)
         JE=EB(N)
         DO 60 K=1,NB
            IF(IBUS(K).EQ.IS) I=K
            IF(IBUS(K).EQ.JE) J=K
60          CONTINUE
         IF(M(I).EQ. 1000)GO TO 120
         IF(M(J).EQ. 1000)GO TO 80
         GO TO 160
80       IF(((HVTC(N).EQ.Y).AND.(LVTC(N).EQ.D)).OR.
     1   ((HVTC(N).EQ.D).AND.(LVTC(N).EQ.Y)))GO TO 100
         M(J)=M(I)
         GO TO 160
100      M(J)=M(I)-1
         GO TO 160
120      IF(M(J).EQ. 1000)GO TO 160
         IF(((HVTC(N).EQ.Y).AND.(LVTC(N).EQ.D)).OR.
     1   ((HVTC(N).EQ.D).AND.(LVTC(N).EQ.Y)))GO TO 140
         M(I)=M(J)
         GO TO 160
140      M(I)=M(J)+1
160      CONTINUE
      DO 180 I=1,NB
         IF(M(I).EQ. 1000)GO TO 40
180      CONTINUE
      DO 200 I=1,NB
200      M(I)=M(I)-M(FB)
      RETURN
      END
      SUBROUTINE TRIFAC(F,T,NB)
C***********************************************************************
C
C     THE SUBROUTINE TRIFAC PERFORMS A TRIANGULAR FACTORIZATION OF
C     THE MATRIX F STORING THE FACTORED MATRIX IN THE MATRIX T
C
C***********************************************************************
C
C     ARGUMENTS:
C
C     F - THE MATRIX TO BE FACTORED
C
C     T - THE FACTORIZED VERSION OF MATRIX F
C
C     NB - THE SIZE OF THE MATRICES
C
C***********************************************************************
C
C     SUBROUTINES CALLED:
C
C        NONE
C
      COMPLEX F(24,24)
      COMPLEX T(24,24)
      COMPLEX SUM
```

```
      DO 50 I=1,NB
         DO 40 J=1,NB
40          T(I,J)=CMPLX(0.,0.)
50       CONTINUE
      DO 60 I=1,NB
60       T(I,1)=F(I,1)/CSQRT(F(1,1))
      DO 90 J=2,NB
         DO 80 I=J,NB
            IF(I.NE.J)GO TO 75
               IM1=I-1
               SUM=CMPLX(0.,0.)
               DO 70 K=1,IM1
70                SUM=SUM+T(I,K)**2
               T(I,I)=CSQRT(F(I,I)-SUM)
               IF((REAL(T(I,I)).EQ. 0.).AND.(AIMAG(T(I,I)).EQ. 0.))
     1         T(I,I)=CMPLX(1.E-10,0.)
               GO TO 80
75          JM1=J-1
            SUM=CMPLX(0.,0.)
            DO 77 K=1,JM1
77             SUM=SUM+T(I,K)*T(J,K)
            IF((REAL(T(J,J)).EQ. 0.).AND.(AIMAG(T(J,J)).EQ. 0.))
     1      T(J,J)=CMPLX(1.E-10,0.)
            T(I,J)=(F(I,J)-SUM)/T(J,J)
80          CONTINUE
90       CONTINUE
      RETURN
      END
      SUBROUTINE VECTOR(T,Z,JV,NB)
C***********************************************************************
C
C      THE SUBROUTINE VECTOR SOLVES FOR THE IMPEDANCE VECTOR OF
C      THE SYSTEM FOR A FAULT AT A PARTICULAR BUS
C
C***********************************************************************
C
C      ARGUMENTS:
C
C      T - THE TRIANGULAR FACTORIZED VERSION OF THE ADMITTANCE
C          MATRIX OF THE SYSTEM.
C
C      Z - THE IMPEDANCE VECTOR OF THE SYSTEM FOR A FAULT AT
C          A GIVEN BUS
C
C      JV - THIS VECTOR IS ZERO EXCEPT FOR THE POSITION OF THE
C           BUS TO BE FAULTED
C
C      NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C***********************************************************************
C
C      LOCAL VARIABLES:
C
C      VHAT - VECTOR USED IN INTERMEDIATE CALCULATIONS
C
      COMPLEX T(24,24)
      COMPLEX Z(24)
      COMPLEX JV(24)
      COMPLEX VHAT(24)
      COMPLEX SUM
```

```
      DO 30 I=1,NB
         SUM=CMPLX(0.,0.)
         IF(I.EQ. 1)GO TO 20
         IM1=I-1
         DO 10 J=1,IM1
10          SUM=SUM+T(I,J)*VHAT(J)
20       VHAT(I)=(JV(I)-SUM)/T(I,I)
30       CONTINUE
      DO 60 I=1,NB
         K=NB-I+1
         SUM=CMPLX(0.,0.)
         IF(K.EQ. NB)GO TO 50
         KP1=K+1
         DO 40 J=KP1,NB
40          SUM=SUM+T(J,K)*Z(J)
50       Z(K)=(VHAT(K)-SUM)/T(K,K)
60       CONTINUE
      RETURN
      END
      SUBROUTINE SQPHAS(V0,V1,V2,VA,VB,VC,NB)
C***********************************************************************
C
C     THE SUBROUTINE SQPHAS CALCULATES THE THREE PHASE VALUES
C     FROM THE SEQUENCE VALUES FOR VOLTAGE AND CURRENT
C
C***********************************************************************
C
C     ARGUMENTS:
C
C     V0 - THE ZERO SEQUENCE VOLTAGE OR CURRENT VECTOR
C
C     V1 - THE POSITIVE SEQUENCE VOLTAGE OR CURRENT VECTOR
C
C     V2 - THE NEGATIVE SEQUENCE VOLTAGE OR CURRENT VECTOR
C
C     VA - THE A PHASE VOLTAGE OR CURRENT VECTOR
C
C     VB - THE B PHASE VOLTAGE OR CURRENT VECTOR
C
C     VC - THE C PHASE VOLTAGE OR CURRENT VECTOR
C
C     NB - THE LENGTH OF THE VECTORS V0,V1,V2,VA,VB,VC
C
C     LOCAL VARIABLES:
C
C     A - COMPLEX VERSION OF 1 AT ANGLE OF 120 DEGREES POLAR
C
C***********************************************************************
C
C     SUBROUTINES CALLED:
C
C        NONE
C
      COMPLEX   V0(NB)
      COMPLEX   V1(NB)
      COMPLEX   V2(NB)
      COMPLEX   VA(NB)
      COMPLEX   VB(NB)
      COMPLEX   VC(NB)
      COMPLEX   A
```

```
      A=CMPLX(-0.5,0.8660254)
      DO 20 I=1,NB
         VA(I)=V0(I)+V1(I)+V2(I)
         VB(I)=V0(I)+A*A*V1(I)+A*V2(I)
20       VC(I)=V0(I)+A*V1(I)+A*A*V2(I)
      RETURN
      END
      SUBROUTINE FLNOUT(NL,SB,EB,LINE,ZSER1,YSHT1,LVTC,HVTC,LTYPE,IS)
C**********************************************************************
C
C       THE SUBROUTINE FLNOUT PRINTS OUT THE LINE DATA FOR THE
C       FAULT CALCULATIONS
C
C**********************************************************************
C
C       NL - THE NUMBER OF LINES IN THE SYSTEM
C
C       SB - STARTING BUS FOR LINE
C
C       EB - ENDING BUS FOR LINE
C
C       LINE - LINE NUMBER FOR A TRANSMISSION LINE
C
C       ZSER1 - THE SERIES IMPEDANCE OF THE LINE
C
C       YSHT1 - THE SHUNT ADMITTANCE OF A LINE
C
C       HVTC - THIS INDICATES HOW A HIGH VOLTAGE SIDE OF A TRANSFORMER
C              IS CONNECTED 'D' FOR DELTA 'Y' FOR WYE.
C
C       LVTC - THIS INDICATES HOW A LOW VOLTAGE SIDE OF A TRANSFORMER
C              IS CONNECTED 'D' FOR DELTA 'Y' FOR WYE.
C
C       LTYPE - LINE TYPE. 'L' FOR A LINE. 'T' FOR A TRANSFORMER
C               BEING MODELED AS A LINE
C
C       IS - THE SEQUENCE NETWORK BEING PRINTED OUT BY THE PROGRAM.
C            0 IS ZERO. 1 IS POSITIVE
C
C**********************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
      COMPLEX   YSHT1(50)
      COMPLEX   ZSER1(50)
      REAL   HVTC(50)
      REAL   LVTC(50)
      REAL   LTYPE(50)
      REAL   L
      INTEGER   SB(50)
      INTEGER   EB(50)
      INTEGER   LINE(50)
      LOGICAL   PRINT
      COMMON /INOU/ KTERM
      COMMON /OUTPUT/ ANSOUT,PRINT,ANSIN
      DATA   YES/'YES  '/
      DATA   T/'T'/
      DATA   L/'L'/
```

```
C
C       WRITEOUT LINE DATA, SHUNT ADMITTANCE, AND THE Y BUS
C
        IF(IS.NE.1) GO TO 12
        IF(ANSOUT.EQ.YES) WRITE(6,1004)
        IF(PRINT) WRITE(KTERM,1004)
12      IF(IS.NE.0) GO TO 14
        IF(ANSOUT.EQ.YES) WRITE(6,1024)
        IF(PRINT) WRITE(KTERM,1024)
14      DO 20 I = 1,NL
           IF(ANSOUT.EQ.YES) WRITE(6,1064) HVTC(I),LVTC(I),SB(I),EB(I),
     1                   LINE(I),ZSER1(I),YSHT1(I)
           IF(PRINT) WRITE(KTERM,1064) HVTC(I),LVTC(I),SB(I),EB(I),
     1             LINE(I),ZSER1(I),YSHT1(I)
20         CONTINUE
        RETURN
1004    FORMAT('1TRANSMISSION LINE AND TRANSFORMER DATA ASSEMBLY'//
     1  22X,'DATA FOR POSITIVE SEQUENCE',/,14X,
     2  'X--------------- ACTUAL ---------------X'/7X,'LINE'/6X,'SB',2X,
     3  'EB',2X,'NO.',4X,'R(PU)',4X,'X(PU)',4X,'B(PU)',4X,'G(PU)'//)
1024    FORMAT('1TRANSMISSION LINE AND TRANSFORMER DATA ASSEMBLY'//
     1  24X,'DATA FOR ZERO SEQUENCE',/,14X,
     2  'X--------------- ACTUAL ---------------X'/7X,'LINE'/6X,'SB',2X,
     3  'EB',2X,'NO.',4X,'R(PU)',4X,'X(PU)',4X,'B(PU)',4X,'G(PU)'//)
1064    FORMAT(1X,2A1,2X,I3,1X,I4,1X,I3,4F9.4)
        END
        SUBROUTINE GENOUT(NB,BTYP,IBUS,BZ1,BZ2,BZ0,BZN)
C**********************************************************************
C
C       THE SUBROUTINE GENOUT PRINTS OUT THE GENERATOR DATA FOR THE
C       FAULTED SYSTEM.
C
C**********************************************************************
C
C       NB - THE NUMBER OF BUSES IN THE NETWORK
C
C       BTYP - THE TYPE OF BUS INDICATOR 'B' IS A STANDARD BUS,
C             'G' IS A GENERATOR BUS
C
C       IBUS - THE BUS NUMBER
C
C       BZ0,BZ1,BZ2 - THE ZERO,POSITIVE AND NEGATIVE SEQUENCE VECTORS
C                   FOR THE BUSES
C
C       BZN - THE NEUTRAL IMPEDANCE FOR A BUS
C
C**********************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        COMPLEX    BZ1(24)
        COMPLEX    BZ2(24)
        COMPLEX    BZ0(24)
        COMPLEX    BZN(24)
        REAL    BTYP(24)
        INTEGER    IBUS(24)
        LOGICAL    PRINT
        COMMON /INOU/ KTERM
```

```
      COMMON /OUTPUT/ANSOUT,PRINT,ANSIN
      DATA    YES/'YES   '/
      DATA    G/'G'/
      ICOUNT=0
      IF(ANSOUT.EQ.YES)WRITE(6,1004)
      DO 20 I=1,NB
         IF(BTYP(I).NE.G)GO TO 20
         IF(ANSOUT.EQ.YES)WRITE(6,1024)IBUS(I),BZ0(I),BZ1(I),BZ2(I),
     1                     BZN(I)
         IF(PRINT)WRITE(KTERM,1034)IBUS(I),BZ0(I),BZ1(I),BZ2(I),
     1             BZN(I)
         ICOUNT=ICOUNT+1
20       CONTINUE
      IF(ICOUNT.NE.0)GO TO 99999
      IF(ANSOUT.EQ.YES)WRITE(6,1044)
      IF(PRINT)WRITE(KTERM,1044)
99999 RETURN
C
1004  FORMAT('1GENERATOR IMPEDANCES',//
     1 6X,'X',20('-'),' SEQUENCE QUANTITIES ',20('-'),'X'/
     2 ' BUS',2X,'X------ ZERO -------XX---- POSITIVE -----X',
     3 'X---- NEGATIVE -----XX----- NEUTRAL -----X'
     4 /' NO.'4(5X,'R(PU)',6X,'X(PU)'))
1024  FORMAT(2X,I3,1X,8F10.4)
1034  FORMAT('0GENERATOR NUMBER',10X,I5,/
     1          ' ZERO      SEQUENCE IMPEDANCE',5X,2F8.4,/
     2          ' POSITIVE SEQUENCE IMPEDANCE',5X,2F8.4,/
     3          ' NEGATIVE SEQUENCE IMPEDANCE',5X,2F8.4,/
     4          ' NEUTRAL           IMPEDANCE',5X,2F8.4)
1044  FORMAT(//' NO GENERATOR BUSES')
C
      END
      SUBROUTINE LADD(LINEN,ISB,IEB,ZLINE,YLINE,NB,NL,
     1 IBUS,SB,EB,LINE,SHTY,SERZ,Y)
C*****************************************************************
C
C     THE SUBROUTINE LADD ADDS A LINE TO THE NETWORK
C
C*****************************************************************
C
C     ARGUMENTS:
C
C     LINEN - LINE NUMBER FOR  TRANSMISSION LINE TO BE ADDED
C
C     ISB - STARTING BUS FOR LINE TO BE ADDED
C
C     IEB - ENDING BUS FOR LINE TO BE ADDED
C
C     ZLINE - THE SERIES IMPEDANCE OF THE LINE TO BE ADDED
C
C     YLINE - THE SHUNT ADMITTANCE OF THE LINE TO BE ADDED
C
C     NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C     NL - THE NUMBER OF TRANSMISSION LINES IN THE SYSTEM
C
C     IBUS - THE BUS NUMBER
C
C     SB - STARTING BUS FOR LINE
C
```

```
C       EB - ENDING BUS FOR LINE
C
C       LINE - LINE NUMBER FOR A TRANSMISSION LINE
C
C       SHTY - THE SHUNT ADMITTANCE OF A LINE
C
C       SERZ - THE SERIES IMPEDANCE OF THE LINE
C
C       Y - THE SYSTEM ADMITTANCE MATRIX
C
C************************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        COMPLEX   SERY
        COMPLEX   SHTY(50)
        COMPLEX   Y(24,24)
        COMPLEX   SERZ(50)
        COMPLEX   ZLINE
        COMPLEX   YLINE
        INTEGER   SB(50)
        INTEGER   EB(50)
        INTEGER   LINE(50)
        INTEGER   IBUS(24)
C
C       FIND POSITION OF STARTING BUS AND ENDING BUS IN IBUS ARRAY
C
        DO 20 J=1,NB
           IF(ISB.EQ.IBUS(J)) K=J
           IF(IEB.EQ.IBUS(J)) L=J
20         CONTINUE
        NLP1=NL+1
        I=NLP1
C
C       ADD IN NEW BUS
C
        SERZ(I)=ZLINE
        SHTY(I)=YLINE
        SERY=1.0/SERZ(I)
        SB(I)=ISB
        EB(I)=IEB
        LINE(I)=LINEN
        Y(K,K) = Y(K,K) + SERY + SHTY(I)/2.
        Y(L,L) = Y(L,L) + SERY + SHTY(I)/2.
        Y(K,L) = Y(K,L) - SERY
        Y(L,K) = Y(L,K) - SERY
        NL=NLP1
        RETURN
        END
        SUBROUTINE LCHNG(LINEN,ISB,IEB,ZLINE,YLINE,NB,NL,
     1  IBUS,LINE,SHTY,SERZ,Y)
C************************************************************************
C
C       THE SUBROUTINE LCHNG CHANGES A LINE IN THE NETWORK
C
C************************************************************************
C
C       ARGUMENTS:
```

```
C
C       LINEN - LINE NUMBER FOR  TRANSMISSION LINE TO BE CHANGED
C
C       ISB - STARTING BUS FOR LINE TO BE CHANGED
C
C       IEB - ENDING BUS FOR LINE TO BE CHANGED
C
C       ZLINE - THE SERIES IMPEDANCE OF THE LINE TO BE ADDED
C
C       YLINE - THE SHUNT ADMITTANCE OF THE LINE TO BE ADDED
C
C       NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C       NL - THE NUMBER OF TRANSMISSION LINES IN THE SYSTEM
C
C       IBUS - THE BUS NUMBER
C
C       LINE - LINE NUMBER FOR A TRANSMISSION LINE
C
C       SHTY - THE SHUNT ADMITTANCE OF A LINE
C
C       SERZ - THE SERIES IMPEDANCE OF THE LINE
C
C       Y - THE SYSTEM ADMITTANCE MATRIX
C
C**********************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        COMPLEX   SERY
        COMPLEX   SHTY(50)
        COMPLEX   Y(24,24)
        COMPLEX   SERZ(50)
        COMPLEX   ZLINE
        COMPLEX   YLINE
        INTEGER   LINE(50)
        INTEGER   IBUS(24)
C
C       FIND THE LINE
C
        DO 20 I=1,NL
           IF(LINEN.EQ.LINE(I)) GO TO 40
20         CONTINUE
40      DO 60 J=1,NB
           IF(ISB.EQ.IBUS(J)) K=J
           IF(IEB.EQ.IBUS(J)) L=J
60         CONTINUE
C
C       CHANGE THE LINE DATA
C
        SERY=1./SERZ(I)
        Y(K,K)  = Y(K,K)  - SERY  - SHTY(I)/2.
        Y(L,L)  = Y(L,L)  - SERY  - SHTY(I)/2.
        Y(K,L)  = Y(K,L)  + SERY
        Y(L,K)  = Y(L,K)  + SERY
        SERZ(I)=ZLINE
        SHTY(I)=YLINE
        SERY=1.0/SERZ(I)
```

```
      Y(K,K)  = Y(K,K)  + SERY + SHTY(I)/2.
      Y(L,L)  = Y(L,L)  + SERY + SHTY(I)/2.
      Y(K,L)  = Y(K,L)  - SERY
      Y(L,K)  = Y(L,K)  - SERY
      RETURN
      END
      SUBROUTINE LDEL(LINEN,ISB,IEB,NE,NL,IBUS,SB,EB,LINE,SHTY,SERZ)
C******************************************************************************
C
C      THE SUBROUTINE LDEL DELETES A LINE FROM THE NETWORK
C
C******************************************************************************
C
C      ARGUMENTS:
C
C      LINEN - LINE NUMBER FOR  TRANSMISSION LINE TO BE DELETED
C
C      ISB - STARTING BUS FOR LINE TO BE DELETED
C
C      IEB - ENDING BUS FOR LINE TO BE DELETED
C
C      NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C      NL - THE NUMBER OF TRANSMISSION LINES IN THE SYSTEM
C
C      IBUS - THE BUS NUMBER
C
C      SB - STARTING BUS FOR LINE
C
C      EB - ENDING BUS FOR LINE
C
C      LINE - LINE NUMBER FOR A TRANSMISSION LINE
C
C      SHTY - THE SHUNT ADMITTANCE OF A LINE
C
C      SERZ - THE SERIES IMPEDANCE OF THE LINE
C
C      Y - THE SYSTEM ADMITTANCE MATRIX
C
C******************************************************************************
C
C      SUBROUTINES CALLED:
C
C         NONE
C
      COMPLEX   SERY
      COMPLEX   SHTY(50)
      COMPLEX   Y(24,24)
      COMPLEX   SERZ(50)
      INTEGER   SB(50)
      INTEGER   EB(50)
      INTEGER   LINE(50)
      INTEGER   IBUS(24)
C
C      FIND THE LINE
C
      DO 20 I=1,NL
         IF(LINEN.EQ.LINE(I)) GO TO 40
20       CONTINUE
40    DO 60 J=1,NB
```

```
            IF(ISB.EQ.IBUS(J))  K=J
            IF(IEB.EQ.IBUS(J))  L=J
60          CONTINUE
C
C       DELETE THE LINE
C
        SERY=1./SERZ(I)
        Y(K,K) = Y(K,K) - SERY - SHTY(I)/2.
        Y(L,L) = Y(L,L) - SERY - SHTY(I)/2.
        Y(K,L) = Y(K,L) + SERY
        Y(L,K) = Y(L,K) + SERY
        NLM1=NL-1
        IF(NL.EQ.I) GO TC 100
        DO 80 J=I,NLM1
           LINE(J)=LINE(J+1)
           SB(J)=SB(J+1)
           EB(J)=EB(J+1)
           SERZ(J)=SERZ(J+1)
80         SHTY(J)=SHTY(J+1)
100     NL=NLM1
        RETURN
        END
        SUBROUTINE IDFLSL(NL,NB,N,V,Y,SE,EB,LINE,SERZ,SHTY,P,Q,
     1  IBTYP,IBUS,PD,PG,QD,QG)
C*********************************************************************
C
C       THE SUBROUTINE IDFSLSL CALCULATES THE POWER FLOWS FROM THE
C       LOADFLOW SOLUTICN AND PRINTS OUT THE VOLTAGES AND POWERS
C
C*********************************************************************
C
C       ARGUMENTS:
C
C       NL - THE NUMBER OF TRANSMISSICN LINES IN THE SYSTEM
C
C       NB - THE NUMBER OF BUSES IN THE SYSTEM
C
C       N - THE NUMBER OF ITERATICNS IT REQUIRED FOR A SOLUTICN
C           TO CONVERGE
C
C       V - THE BUS VCLTAGES
C
C       Y - THE SYSTEM ADMITTANCE MATRIX
C
C       SB - STARTING BUS FOR LINE
C
C       EB - ENDING BUS FOR LINE
C
C       LINE - LINE NUMBER FOR A TRANSMISSICN LINE
C
C       SERZ - THE SERIES IMPEDANCE OF THE LINE
C
C       SHTY - THE SHUNT ADMITTANCE OF A LINE
C
C       IBUS - THE BUS NUMBER
C
C       IBTYP - THE BUS TYPE  3 - SLACK
C                             2 - VCLTAGE CONTROL
C                             1 - STANDARD
C
```

```
C         PG - REAL POWER GENERATED
C
C         QG - REACTIVE POWER GENERATED
C
C         PD - REAL POWER DEMAND
C
C         QD - REACTIVE POWER DEMAND
C
C         P - REAL POWER GENERATED MINUS REAL POWER DEMAND
C
C         Q - REACTIVE PWER GENERATED MINUS REACTIVE POWER DEMAND
C
C         VSPEC - BUS VOLTAGE SPECIFICATION
C
C         QMIN - MINIMUM VARS FOR VOLTAGE CONTROL BUS
C
C         QMAX - MAXIMUM VARS FOR VOLTAGE CONTROL BUS
C
C*************************************************************************
C
C         SUBROUTINES CALLED:
C
C            NONE
C
          COMPLEX   SUM
          COMPLEX   V(24)
          COMPLEX   SERZ(50)
          COMPLEX   SHTY(50)
          COMPLEX   S
          COMPLEX   R
          COMPLEX   Y(24,24)
          INTEGER   SB(50)
          INTEGER   EB(50)
          INTEGER   IBTYP(24)
          INTEGER   IBUS(24)
          INTEGER   LINE(50)
          INTEGER   PBUS(24)
          LOGICAL   SKIP
          REAL    P(24)
          REAL    Q(24)
          REAL    PG(24)
          REAL    PD(24)
          REAL    QG(24)
          REAL    QD(24)
          REAL    MAGV
          COMMON /INOU/ KTERM
          COMMON /OUTPUT/ ANSOUT,PRINT
          DATA YES/'YES  '/
C
C       SLACK BUS POWER CALCULATION
C
          DO 20 I=1,NB
             IF(IBTYP(I).EQ. 3) GO TO 40
20           CONTINUE
40        M=I
          SUM = CMPLX(0.0,0.0)
          DO 60 I = 1,NB
60           SUM = SUM + Y(M,I) * V(I)
          P(M)  = REAL(SUM*CONJG(V(M)))
          Q(M) = -AIMAG(SUM*CONJG(V(M)))
```

```
C
C       WRITE OUT BUS DATA.
C
        ISWB=IBUS(M)
        WRITE(KTERM,944)
        READ(KTERM,*) NBUS
        IF(NBUS.EQ.NB) GO TO 66
        DO 63 IC=1,NBUS
           WRITE(KTERM,964)
           READ(KTERM,*) PBUS(IC)
63         CONTINUE
66      WRITE(KTERM,984) N,ISWB
        IF(ANSOUT.NE.YES) GO TO 69
        WRITE(6,1004) N,ISWB
        WRITE(6,1024)
69      DO 300 N=1,NB
           SKIP=.TRUE.
           IF(NBUS.EQ.NB) SKIP=.FALSE.
           DO 72 IC=1,NBUS
              IF(IBUS(N).EQ.PBUS(IC)) SKIP=.FALSE.
72            CONTINUE
           QG(N)=Q(N)+QD(N)
           PG(N)=P(N)+PD(N)
           MAGV = CABS(V(N))
           DELTA = ATAN2(AIMAG(V(N)),REAL(V(N))) * 57.29578
           IF(SKIP) GO TO 75
           WRITE(KTERM,1034) IBUS(N),MAGV,DELTA,PD(N),QD(N),PG(N),
     1     QG(N)
           WRITE(KTERM,1054)
75         IF(ANSOUT.NE.YES)GO TO 78
           WRITE(6,1044) IBUS(N),MAGV,DELTA,PD(N),QD(N),PG(N),QG(N)
C
C       LINE FLOWS ARE CALCULATED
C
78         KN=1
80         DO 100 K=KN,NL
              IF(IBUS(N).EQ.SB(K)) GO TO 180
100           CONTINUE
120        LN=1
140        DO 160 K=LN,NL
              IF(IBUS(N).EQ.EB(K)) GO TO 240
160           CONTINUE
           GO TO 300
180        J=K
           IC=EB(K)
           DO 200 L=1,NB
              IF(IBUS(L).EQ. IC) GO TO 220
200           CONTINUE
220        R=V(N)*CONJG((V(N)-V(L))/SERZ(J)+V(N)*(SHTY(J)/2.0))
           IF(ANSOUT.NE.YES) GO TO 225
           WRITE(6,1064) IC,LINE(K),R
225        IF(SKIP) GO TO 230
           WRITE(KTERM,1074) IBUS(N),IC,LINE(K),R
230        KN=K+1
           IF(KN.LE.NL) GO TO 80
           GO TO 120
240        J=K
           IC=SB(K)
           DO 260 L=1,NB
              IF(IBUS(L).EQ. IC) GO TO 280
```

```
260             CONTINUE
280          S=V(N)*CONJG((V(N)-V(L))/SERZ(J)+V(N)*(SHTY(J)/2.0))
             IF(ANSOUT.NE.YES)GO TO 285
             WRITE(6,1064)IC,LINE(K),S
285          IF(SKIP)GO TO 290
             WRITE(KTERM,1074)IBUS(N),IC,LINE(K),S
290          LN=K+1
             IF(LN.LE.NL)GO TO 140
300          CONTINUE
          PDSUM=0.0
          QDSUM=0.0
          PGSUM=0.0
          QGSUM=0.0
          DO 320 I=1,NB
             PDSUM=PDSUM+PD(I)
             QDSUM=QDSUM+QD(I)
             PGSUM=PGSUM+PG(I)
320          QGSUM=QGSUM+QG(I)
          IF(ANSOUT.NE.YES)GO TO 99999
          WRITE(6,1084)PDSUM,QDSUM,PGSUM,QGSUM
99999     RETURN
C
944       FORMAT('0INPUT THE NUMBER OF BUSES TO BE OUTPUT ',$)
964       FORMAT(' INPUT BUS NUMBER ',$)
984       FORMAT(/' CONVERGENCE OCCURED IN ',I4,' ITERATIONS',/
     1' BUS NUMBER ',I5,' WAS THE SWING BUS')
1004      FORMAT('1REPORT OF LOADFLOW CALCULATIONS',20X,
     1'TOTAL ITERATIONS =',I5,' SWING BUS =',I5/' X--------------------
     2--- B U S - D A T A --------------------X  X----------- L I N E -
     3 D A T A --------X'/7X,'FROM',6X,'VOLTS',2X,'ANGLE',7X,'LOAD',
     49X,'GENERATION',12X,'TO',7X,'LINE','LINE FLOW')
1024      FORMAT(3X,'BUS    NAME',4X,'PU',5X,'DEG',7X,'REAL',2X,'REACT',
     1    5X,'REAL',2X,'REACT',5X,'BUS    NAME',3X,'NO.',5X,'REAL',3X,
     2    'REACT'//)
1034      FORMAT(/1X,'BUS NUMBER',15X,I10/
     1    ' VOLTAGE MAGNITUDE',8X,F10.4/
     2    ' VOLTAGE ANGLE',12X,F10.4/
     3    ' POWER DEMAND',13X,2F10.4/
     5    ' POWER GENER.',13X,2F10.4,/)
1044      FORMAT(5X,'BUS',I3,3X,F7.3,1X,F7.2,1X,F7.3,1X,F7.3,1X,F7.3,1X,
     1    F7.3,2X,43('-'))
1054      FORMAT(/4X,'SB',4X,'EB',2X,'LINE',4X,'POWER FLOW')
1064      FORMAT(66X,'BUS',I3,6X,I3,2X,F7.3,2X,F7.3)
1074      FORMAT(3(1X,I5),2F8.3)
1084      FORMAT('0',29X,7('*'),1X,7('*'),1X,7('*'),1X,7('*')/
     1            ' TOTALS',23X,F7.3,1X,F7.3,1X,F7.3,1X,F7.3)
C
          END
          SUBROUTINE LINOUT(NL,SB,EB,LINE,SERZ,SHTY)
C*********************************************************************
C
C         THE SUBROUTINE LINOUT PRINTS OUT THE LINE DATA FOR THE
C         LOADFLOW CALCULATIONS
C
C*********************************************************************
C
C         NL - THE NUMBER OF LINES IN THE SYSTEM
C
C         ANSOUT - VARIABLE TO INDICATE IF YOU ARE PRINTING TO DISK
C
```

```
C       PRINT - LOGICAL VARIABLE TO INDICATE IF YOU ARE PRINTING TO
C               TERMINAL
C
C       SB - STARTING BUS FOR LINE
C
C       EB - ENDING BUS FOR LINE
C
C       LINE - LINE NUMBER FOR A TRANSMISSION LINE
C
C       SERZ - THE SERIES IMPEDANCE OF THE LINE
C
C       SHTY - THE SHUNT ADMITTANCE OF A LINE
C
C***********************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        COMPLEX   SHTY(50)
        COMPLEX   SERZ(50)
        LOGICAL   PRINT
        INTEGER   SB(50)
        INTEGER   EB(50)
        INTEGER   LINE(50)
        COMMON /INOU/ KTERM
        COMMON /OUTPUT/ ANSOUT,PRINT
        DATA YES/'YES  '/
C
C       WRITE OUT LINE DATA, SHUNT ADMITTANCE, AND THE Y BUS
C
        IF(ANSOUT.EQ.YES) WRITE(6,1004)
        IF(PRINT) WRITE(KTERM,1004)
        DO 20 I = 1,NL
           IF(.NOT. PRINT) GO TO 15
           WRITE(KTERM,1024) SB(I),EB(I),LINE(I),SERZ(I),SHTY(I)
15         IF(ANSOUT.NE.YES) GO TO 20
           WRITE(6,1024)SB(I),EB(I),LINE(I),SERZ(I),SHTY(I)
20         CONTINUE
        RETURN
1004    FORMAT('1TRANSMISSION LINE DATA ASSEMBLY'/11X,
     1  'X--------------- ACTUAL --------------X'/4X,'LINE'/3X,'SB',2X,
     2  'EB',2X,'NO.',4X,'R(PU)',4X,'X(PU)',4X,'B(PU)',4X,'G(PU)'//)
1024    FORMAT(2X,I3,1X,I4,1X,I3,4F9.4)
        END
        SUBROUTINE SORT2V(NB,NNB,V)
C***********************************************************************
C
C       THE SUBROUTINE SORT2V PERFORMS A BUBBLE SORT FOR 2 VECTORS
C       WITH RESPECT TO THE VECTOR NNB.
C
C***********************************************************************
C
C       ARGUMENTS:
C
C       NNB - THE INTEGER VECTOR WHICH IS TO BE SORTED IN ASCENDING
C             ORDER
C
C       V - THE REAL VECTOR TO BE SORTED WITH NNB
C
```

```
C       NB - THE NUMBER OF ELEMENTS IN THE VECTORS
C
C***********************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        COMPLEX   V(24)
        COMPLEX   CTEMP
        INTEGER   NNB(24)
        NBM1=NB-1
        DO 40 L=1,NBM1
           LP1=L+1
           DO 20 J=LP1,NB
              IF(NNB(L).LE.NNB(J)) GO TO 20
              ITEMP=NNB(L)
              NNB(L)=NNB(J)
              NNB(J)=ITEMP
              CTEMP=V(L)
              V(L)=V(J)
              V(J)=CTEMP
20            CONTINUE
40         CONTINUE
        RETURN
        END
        SUBROUTINE YASSEM(NB,NL,IBUS,SB,EB,SERZ,SHTY,Y,*)
C***********************************************************************
C
C       THE SUBROUTINE YASSEM ASSEMBLES THE Y BUS MATRIX
C       FOR THE LOADFLOW AND POSITIVE SEQUENCE NETWORK FOR THE
C       FAULT STUDY
C
C***********************************************************************
C
C       NB - THE NUMBER OF BUSES IN THE NETWORK
C
C       NL - THE NUMBER OF LINES IN THE NETWORK
C
C       IBUS - THE BUS NUMBER
C
C       SB - STARTING BUS FOR LINE
C
C       EB - ENDING BUS FOR LINE
C
C       SERZ - THE SERIES IMPEDANCE OF THE LINE
C
C       SHTY - THE SHUNT ADMITTANCE OF A LINE
C
C       Y - THE BUS ADMITTANCE MATRIX
C
C***********************************************************************
C
C       SUBROUTINES CALLED:
C
C          NONE
C
        COMPLEX   Y(24,24)
        COMPLEX   SERZ(50)
        COMPLEX   SHTY(50)
```

```
      COMPLEX   SERY
      INTEGER   NE
      INTEGER   NL
      INTEGER   IBUS(24)
      INTEGER   SB(50)
      INTEGER   EB(50)
      DO 240 I=1,NL
         SERY = 1.0/SERZ(I)
C
C     ASSEMBLE THE BUS ADMITTANCE MATRIX.
C
         J1=0
         IF(SB(I).EQ. 0)GO TO 180
         DO 160 J1=1,NB
            IF(IBUS(J1).EQ.SB(I)) GO TO 180
160         CONTINUE
         CALL ERROR(2,SB(I),&99999)
180      L=J1
         J2=0
         IF(EB(I).EQ. 0)GO TO 220
         DO 200 J2=1,NB
            IF(IBUS(J2).EQ.EB(I)) GO TO 220
200         CONTINUE
         CALL ERROR(2,EB(I),&99999)
220      M=J2
         IF(L.EQ. 0)GO TO 230
         IF(M.EQ. 0)GO TO 235
         Y(L,L) = Y(L,L) + SERY + SHTY(I)/2.
         Y(M,M) = Y(M,M) + SERY + SHTY(I)/2.
         Y(L,M) = Y(L,M) - SERY
         Y(M,L) = Y(M,L) - SERY
         GO TO 240
230      Y(M,M) = Y(M,M) + SERY + SHTY(I)
         GO TO 240
235      Y(L,L) = Y(L,L) + SERY + SHTY(I)
240      CONTINUE
      RETURN
99999 RETURN 1
      END
```

VITA

The author was born on December 21, 1954 in Knoxville, Tennessee. He graduated from Everett High Scool in Maryville in 1972. Later that year he entered the University of Tennessee at Knoxville.

He received a bachelor's degree in computer science in 1977. In 1979 he entered the electrical engineering graduate program. As a graduate student he worked as a research assistant, teaching assistant and manager of the Image Processing and Analysis Laboratory. He received his master's degree in electrical engineering in December 1982.