

AD-A035 292

AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 10/2
A USER-ORIENTED POWER DISTRIBUTION SYSTEM ANALYSIS PROGRAM. (U)
DEC 76 J A UNDERWOOD

UNCLASSIFIED

6E/EE/76-43

NL

1 of 3

AD
A035292

1

9 Master's / thesis

8

A USER-ORIENTED POWER
DISTRIBUTION SYSTEM
ANALYSIS PROGRAM

14

GE/EE/76-43

10

THESIS

Jesse Allen Underwood, Jr.
Major USAF

11 Dec 76

12 258p.

DDC
RECEIVED
FEB 8 1977
A

Approved for public release; distribution unlimited.

Copies available to DDC from the
Annual Army Institute of Production

012225

VB

A USER-ORIENTED POWER DISTRIBUTION SYSTEM
ANALYSIS PROGRAM

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology

Air University

in Partial Fulfillment of the
Requirements for the Degree of

Master of Science

by

Jesse A. Underwood, Jr., B.S.

Major

USAF

Graduate Electrical Engineering

December 1976

Approved for public release; distribution unlimited.

Preface

This thesis is basically a revision to a digital computer program originally prepared at the Air Force Institute of Technology in 1975. Although minor corrections have been made and additional features added, the basic program structure has remained unchanged. Since the original program was completed, students and faculty at the Air Force Institute of Technology have used the program for analysis of actual and theoretical power distribution systems. The many comments and suggestions received from the users have served as one of the motivating forces for the revision.

This thesis is written to be used in conjunction with the thesis for the original program. As only theoretical development is given in this thesis for the changes and added features, both these should be studied in order to obtain a complete description of the theory and operation of the program. The highlight of this thesis is the User's Guide, listed as Appendix A. The thesis was purposely written so the guide could be withdrawn and used separately. Anyone only interested in using the program should find Appendix A complete and sufficient for their needs.

I wish to acknowledge all those who contributed to the success of this thesis project with my special thanks to my advisor, Capt. Frederick C. Brockhurst; my readers, Capt. Michael A. Aimone and Mr. Charles W. Richard, Jr.; and to the original program author, Capt. Michael C. Heer. The

comments and guidance from my advisor and the original program author were especially helpful in doing the background work and research. I also wish to thank my typist, Mrs. Charlette Kjesbo for her help in preparing this thesis.

Jesse A. Underwood, Jr.

Contents

	Page
Preface	ii
List of Figures	vi
List of Tables	viii
List of Symbols and Abbreviations	x
Abstract	xi
I. Introduction	1
Background	1
Problem Statement and Scope	1
Assumptions	2
Approach	2
Presentation	3
II. Development	4
Introduction	4
Control Cards	4
Line Element Input Routines	5
Load Flow Routine	9
Short Circuit Routine	10
Revision of User's Guide	13
III. Tests and Results	14
Introduction	14
Line Element Input Routines	14
Load Flow Tests	15
Short Circuit Testing	17
Summary	19
IV. Conclusions and Recommendations	21
Introduction	21
Conclusions	21
Program Capabilities	21
Limitations	21
Recommendations	23
General	23
Input Routines	24
Load Flow	25
Short Circuit	25

Bibliography	26
Appendix A. User's Guide	28
Appendix B. Program Logic Flow Charts	128
Appendix C. Program Variables and Data Flow Charts	162
Appendix D. Program Listing	177

List of Figures

Figure		Page
A-1	PDSAP Card Deck Structure	38
A-2	Title, Input-Output, and System Parameter Control Cards	40
A-3	Line Element Card Format for LINEZ Routine	47
A-4	Conductor Data Card Format for LINDATA Routine	50
A-5	Data Card Format for LINDATA Routine	50
A-6	Delta Equivalent Three-Winding Transformer	55
A-7	Load Flow Control Card	58
A-8	Load Flow Bus Card Format	60
A-9	Load Change Control Card	61
A-10	Short Circuit Control Card	65
A-11	Current Source Control Card	67
A-12	Current Source Data Card	67
A-13	Mutual Coupling Control and Data Cards . .	69
A-14	Fault Impedance Control Card	72
A-15	Bus List Format for Fault Impedance and Subsystem Buses	73
A-16	Subsystem Control Card	73
A-17	FORTTRAN Error Example	80
A-18	Example 1 System Diagram	86
A-19	Data Card Format, Example 1	87
A-20	System Diagram, Example 2	97
A-21	Data Card Format, Example 2	98
A-22	System Diagram, Example 3	106
A-23	Data Card Format, Example 3	107
A-24	System Diagram, Example 4	115

Figure		Page
A-25	Data Card Format, Example 4	116
B-1	EXECUTIVE Routine Flow Chart	129
B-2	INPUT Routine Flow Chart	131
B-3	LINDATA Routine Flow Chart	132
B-4	LINEZ Routine Flow Chart	138
B-5	Load Flow Routine Flow Chart	141
B-6	LIMIT Subroutine Flow Chart	146
B-7	Short Circuit (FAULT) Routine Flow Chart .	149
B-8	BUS Subroutine Flow Chart	156
B-9	BUSØ Subroutine Flow Chart	156
B-10	MUTEST Subroutine Flow Chart	161
C-1	COMMON COMA Variables and Data Flow	164
C-2	COMMON COMB Arrays and Data Flow	168
C-3	COMMON COMC Arrays and Data Flow	171
C-4	COMMON CONST Variables and Data Flow . . .	173
C-5	COMMON SAVE Variables and Data Flow	175

List of Tables

Table		Page
I	Wire Impedance Value Comparisons	16
II	Comparison of Bus Voltages	16
III	Comparison of Fault Currents-DSPM System	18
IV	Comparison of Fault Currents-Westinghouse System	20
A-I	Input-Output Control Card Variables . . .	41
A-II	System Parameter Control Card Variables .	41
A-III	Codes for OUT Variable	42
A-IV	ID Code Values	44
A-V	Transformer Connection Codes	45
A-VI	Line Element Variables for LINEZ Routine	47
A-VII	Wire Sizes	49
A-VIII	Line Element Variables for LINDATA Routine	51
A-IX	Load Flow Control Card Variables	58
A-X	Load Flow Bus Card Variables	60
A-XI	Short Circuit Control Card Variables . .	65
A-XII	Current Source Control Card Variables . .	67
A-XIII	Current Source Data Card Variables . . .	68
A-XIV	Current Source Equations	70
A-XV	Mutual Coupling Variables	70
A-XVI	Fault Impedance Control Card Variables .	72
A-XVII	Bus List Card Variables	73
A-XVIII	Subsystem Control Card Variables	74
A-XIX	Error Messages and Causes	82
C-I	Common COMA Variables	163

C-II	Common COMB Variables	165
C-III	Common COMC Variables	170
C-IV	Common CONST Variables	172
C-V	Common SAVE Variables	174
C-VI	Common ZCONST Variables	174
C-VII	Common ZERO Arrays	176

List of Symbols and Abbreviations

ACSR	Aluminum Cable Steel Reinforced.
ANG	Angle, usually degrees.
CM	Circular Mils
D_e	Equivalent spacing between conductors or neutrals in feet.
D_s	Geometric mean radius of neutrals in feet.
f	Frequency in hertz.
GMR	Geometric Mean Radius
KVA	Kilovolt-Amperes
KW	Kilowatt
r_a	AC resistance in ohms
r_e	Zero sequence and earth resistance component of impedance
ρ (rho)	Resistivity in meter-ohms
TCUL	Tap Changing Under Load Transformer
X/R	Impedance ratio of reactive to real impedance.
x_a	Inductive reactance out to one foot radius.
x_e	Zero sequence and earth inductive component of impedance.
ZOA	Zero sequence of a conductor without neutrals.
ZOAG	Zero sequence self impedance of neutrals or ground wires.
ZOG	Zero sequence mutual impedance between neutrals and conductors.

Abstract

This paper is a revision of a digital computer program written to perform a load flow and/or short circuit analysis of a power distribution system. The program has been named Power Distribution System Analysis Program (PDSAP). The program capacity is 250 buses and 500 line elements, with 250 of the line elements being transformers. Input routines accept data as impedances (ohms or per-unit), or as descriptive information such as wire size, length, or transformer ratings. For descriptive data, the program uses pre-calculated approximations to derive the impedance values for the various line elements and the program will adjust impedance values due to neutral conductors in the system. The load flow routine uses the fast-decoupled Newton-Raphson technique and has the capability of changing loads to represent load growth within the system. The short circuit routine analyzes systems in 50 bus groups, simulating various types of faults for each bus. Bus voltages and line currents in the system are calculated for each simulated fault. The paper contains a comprehensive User's Guide which provides clear and concise instructions for operating the program. The PDSAP program is intended for use by anyone in the Air Force with an electrical engineering background and concerned with power distribution.

I. Introduction

Background

In 1975 a digital computer program was written at the Air Force Institute of Technology that was capable of simulating power distribution systems such as found at most Air Force installations (Ref 2:xii). The program was successfully designed to provide the Air Force with the capability to analyze power distribution systems without incurring excessive costs. However, the original version of the program had some characteristics that limited its applicability in certain cases.

Since efficient operation of power distribution systems continues to be of great interest and concern, the need for comprehensive analysis capability at minimum cost remains a valid goal. During the past year, several power distribution systems have been simulated with the computer program. As a result of certain problems encountered in attempting to simulate actual systems, the necessity of revising the original program became apparent.

Problem Statement and Scope

The problem was to analyze the existing digital computer program and investigate possible changes and additions to improve the capability of simulating power distribution systems as generally found throughout the Air Force. The program was re-named the Power Distribution System Analysis Program, PDSAP, and modified to overcome certain limitations

that existed in the original version. Two of these limitations were the adjustment of impedance values for systems with neutrals and the ability to do short circuit analysis on non-radial systems.

The scope of the work was limited to modifications and changes of the existing program structure, endeavoring to retain the basic logic structure as used in the original version of the program. Additionally, a new User's Guide, Appendix A, was written to aid and encourage the use of the program by Air Force electrical engineering personnel in resolving problems and increasing efficiency of power distribution systems with as little expense as possible.

Assumptions: The assumptions used in the original program preparation were also used in this modification of the program. Basically, these assumptions were that a one-line diagram of the power system to be studied would be available for a data source, that accuracies in load-flow calculations within ± 10 percent would be reasonable, and that the program would be used on the CDC CYBER computer system. An additional assumption was that systems being simulated in the program were either actual systems or systems that were designed to accepted standards.

Approach: The existing digital program was used as the basis for the new program. Each area of the program was examined and tested with additional system simulations taken from textbooks, actual systems or theoretical systems. Problems identified by use of the additional systems were cate-

gorized into groups for each routine in the program. Each group of problems was then considered in making the revision to the program. In addition, several different persons utilized the program and difficulties encountered in coding input data provided suggestions to be incorporated into the new User's Guide.

The overall goal was to modify the existing program to make it more versatile and useful to the Air Force by assisting electrical engineering personnel in power distribution system analysis and design.

Presentation: The paper is organized into 4 chapters with appendices which include a copy of the revised program and the new User's Guide. Chapter II, "Development," discusses the changes to each routine in the program with the inclusion of the theoretical basis for information that has been added. Chapter III, "Tests and Results," details the results of test data that was used to verify the accuracy of the program. This includes comparison with the test results from the original program, plus additional tests used for new routines that have been added. The final chapter, "Conclusion and Recommendations," discusses future changes that would enhance the value of the program and limitations that have been found to exist with the revised version.

In addition, information contained in the four appendices are the User's Guide, program logic charts, program data flow charts and program listing.

II. Development

Introduction

In this chapter, each of the major routines in the original program is discussed in terms of changes and additions that have been made. The major subdivisions of the chapter are Control Cards, Line Element Input Routines, Load Flow Routine, Short Circuit Routine and User's Guide.

Equations utilized have been adapted from the original program or from additional sources as cited. The reader is referred to the original program (Ref 2:190) for a complete discussion on basic program theory and logic structure. In cases where there was a significant modification to the original program logic, there is additional documentation to support the changes.

Control Cards

One of the significant changes in the program was to introduce a new standardized format for all program control cards. Previously, control cards were varied in both location and format within the program. The revision attempted to standardize all program control cards to prevent confusion. Additionally the control cards are used as an error checking routine to help prevent erroneous input data. The program control cards all use a keyword that is read and compared to the expected word before the program will continue. Errors and omission in the use of the keyword will cause the program to halt and print an appropriate error message. An-

other change to the program control cards was the location relative to the data cards. By the addition of one program control card, all control cards now appear in front of the data cards for each routine and help identify the data cards which follow. This scheme also facilitates finding control cards in a large data deck.

Line Element Input Routines

Two line element routines are in the PDSAP program, LINEZ and LINDATA. The LINEZ routine utilizes input data in ohms or per-unit to calculate all impedance values for line elements. The LINDATA routine uses line element information such as wire size, transformer rating and similar variables to calculate impedance values.

In the LINEZ routine, changes were made to compensate for neutral, or ground wires, that would affect the zero sequence impedance values. The method used to adjust the zero sequence values was to multiply the given values by a constant, 2.7. The constant value was taken as the higher value for a representative range (Ref 7:21). The higher value was chosen to give the lowest value for the phase-to-ground fault current that might be expected within the system. The lowest phase-to-ground fault current is of primary interest in being able to detect this type of fault within the system relative to normal load current values.

Also in the LINEZ routine, a constant value of 3.5 was used to create zero sequence impedance when no input value

was given. Again, the constant was used as a representative value based on an average range (Ref 7:21). The higher value was chosen for the same reason as cited in the preceding paragraph.

In the LINDATA routine, changes were made to accommodate a larger number of standard wire sizes for both aerial conductors and underground cable. A table was added in the User's Guide listing all the wire sizes available in the program. The impedance values for the additional wire sizes were based on interpolation of curves generated by using representative impedance values for wire sizes used in the original program (Ref 3:19-20)

Also in the LINDATA program, provision was made to include the effects of grounds, or neutrals, on the impedance values. The program uses the values given for the phase conductors, and information for the neutral conductors, to determine the proper zero sequence impedance values for aerial lines. An additional section was added to the LINDATA routine to perform this calculation as detailed in the following three paragraphs.

The zero sequence impedance is adjusted as follows:
The input line element card is coded to reflect the presence of neutrals with information as to number, size, and spacing. The program uses this information to find the zero sequence self-impedance of the ground wires, Eq (3), and the zero sequence mutual impedance between the three-phase circuit as one group of conductors and the neutrals as the other con-

ductor group, Eq (2). The equation for one circuit with n neutrals (Ref 8:44-45) is as follows:

$$Z_0 = Z_{0A} - \frac{(Z_{0AG})^2}{Z_{0G}} \quad (1)$$

where Z_{0A} is the zero sequence impedance without neutrals, Z_{0G} is the zero sequence self impedance of the neutrals, and Z_{0AG} is the zero sequence mutual impedance between the three-phase circuit as one group and the neutrals as another group. With this equation, the zero sequence impedance for n neutrals, or ground wires, can be determined with relative accuracy.

The addition to the LINDATA routine was constructed to utilize values and logic from the original routine as much as possible. The value of Z_{0A} was used from the original program. Z_{0AG} is calculated as follows:

$$Z_{0AG} = r_e + j(x_e - 3x_d) \quad (2)$$

where

$$r_e = .004764(f)$$

$$x_e = .006985(f) \log_{10} \left(\frac{4.6655(10^6)\rho}{f} \right)$$

$$x_d = .004657(f) \log_{10} D_e$$

f = frequency (hertz)

ρ = resistivity (meter-ohms)

D_e = equivalent spacing (feet)

The value for Z_{0G} is calculated as follows:

$$Z_{0G} = 3/n(r_a) + r_e + j \left(x_e + \frac{3x_a}{n} - \frac{3x_d}{n} \right) \quad (3)$$

where r_a = ac resistance (ohms)
 n = number of neutrals
 $r_e = .004764(f)$
 $x_a = .004657(f) \log_{10} \left(\frac{1}{D_s} \right)$
 and f = frequency (hertz)
 D_s = GMR of neutrals

Adjustment to the zero sequence value can also be made when neutrals are present but their size and spacing are unknown. When neutrals are present and only the number is specified, the program assumes an equivalent spacing from the conductors of 4 feet and a spacing between neutrals of 15 feet for more than one neutral. The assumed wire size for neutrals is 105,500 CM (Circular Mils) for copper and 133,100 CM for aluminum when size is not specified. This addition was added to the LINDATA routine between statements 230 and 300 of the program listing (Appendix D).

In both the LINEZ and LINDATA routines, minor changes were necessary to the calculated impedance values for transformers. The original program contained impedance values for three-winding transformers which could not be used with the present input data format. Therefore, all three-winding transformer impedance values were blocked, except for three-winding phase-shifting transformers, and all references were deleted from the User's Guide. Three-winding phase-shifter transformers that have only two windings connected externally are compatible and codes for these have been retained. To simulate a three-winding transformer, a procedure was

added to the User's Guide to use three two-winding transformers. Although blocked from use, the values for all three-winding transformers were left in the program for possible future use.

Load Flow Routine

In the Load Flow routine, convergence with certain systems can be a problem. By extensive testing with several simulated systems, it became apparent that if the loads within the system are not reasonably well distributed or the R/X ratio was greater than approximately 1.5, convergence sometimes is impossible regardless of the tolerance or number of iterations allowed. From limited analysis of several systems, under these conditions the simulated system in the program appears to diverge. As a result, the difference between the specified and calculated power becomes greater with each additional iteration. Eventually, the numbers become larger than allowed by the program and a processing error, or mode error, results. To prevent mode errors, limitation on the voltage array elements, $V(I)$, was set at ± 100.00 per-unit and the angle array elements, $ANG(I)$, at $\pm 30\pi$ radians. Also, a limitation was placed on Theta, the angle between the start bus and end bus of each line, at $\pm 30\pi$ radians. By using these extreme limits, most systems being simulated should not be affected, but systems that will not converge will not halt the program with a mode error.

Another change to the Load Flow routine was to print the voltage magnitude, voltage angle, DLP and DLQ values

when convergence is not obtained. DLP and DLQ are the differences between the calculated and specified real and reactive power respectively. This additional information can aid the user in being able to detect what parts of the simulated system are causing the convergence problem. By using this information, the system configuration can usually be altered to obtain a satisfactory solution.

A significant addition to the Load Flow routine was the addition of the load change capability. After the initial system has converged and the results printed, any number of Type I load buses can be changed and the Load Flow routine run with the revised data. Type 1 load buses have no restrictions on voltage limits. Only changes to Type 1 buses are allowed as to change voltage controlled buses, Type 2, would require recalculating the B'' matrix (Ref 4:863). After the new solution is obtained, it is printed with a list of buses that were changed and the new load values for all buses in the system. This routine is included in statements 580 thru 600 of the program (Appendix D).

Short Circuit Routine

The Short Circuit routine as originally designed had several limitations. It was only capable of analyzing a radial system and the process used to select various buses as subsystems for fault analysis did not have repetitive capability. Lastly, some of the transformers connection codes were apparently in error, occasionally causing invalid results for the zero sequence impedance values.

The first change was to modify the BUS and and BUSØ sub-routines to allow a system of any configuration to be satisfactorily simulated. This required extensive analysis of these routines and considerable testing with various simulated systems to insure the new logic was correct. Additionally, the necessary changes were made to initialize the arrays each time the BUS and BUSØ routines were called, which corrected the problem of not being able to run successive subsystem studies.

Some minor changes were necessary in the routine where the transformer connection codes, or C values, adjust the line table entries to properly reflect the transformer winding configuration. The changes were made and testing done with various simulated systems to insure the impedance values in the line tables were being correctly adjusted (Ref 3:25).

Another minor change was to the equations used for the phase-phase-ground faults as there was an apparent error in the original program. The present equations used in this part of the program were carefully checked for accuracy (Ref 3:10). Logic changes can be noted by comparison of the logic flow charts, Appendix B or the program listing, Appendix D, with the original program (Ref 2:84).

A major addition to the Short Circuit routine was the section from statements 135 to 148 of the program (Appendix D). Using the elements of the ZBUS matrix as developed by the BUS routine, the three-phase fault currents for lines were calculated using the equation:

$$I_{se} = \frac{Z_{sk} - Z_{ek}}{Z_{line\ se}} \frac{1}{Z_{kk}} \quad (4)$$

where I_{se} is fault current, in per-unit, on the line between S and E (SB and EB) with the fault at bus K. Z_{sk} and Z_{ek} are the calculated impedance values between bus K and bus S or E. Z_{kk} is the driving point impedance value or diagonal value, of bus K (Ref 1:26-27). The lines are any existing lines between buses listed in the subsystem being analyzed.

Likewise, the phase-ground fault currents were calculated using the equation

$$I_{se} = 2I_{se}^+ + I_{se}^0 \quad (5)$$

where I_{se}^+ and I_{se}^0 are the positive and zero sequence currents respectively (Ref 1:78-79).

The positive sequence current, I_{se}^+ is found by

$$I_{se}^+ = \frac{1}{2Z_{kk}^+ + Z_{kk}^0} \frac{Z_{sk}^+ - Z_{ek}^+}{Z_{line\ se}^+} \quad (6)$$

and, the zero sequence current is found by

$$I_{se}^0 = \frac{1}{2Z_{kk}^+ + Z_{kk}^0} \frac{Z_{sk}^0 - Z_{ek}^0}{Z_{line\ se}^0} \quad (7)$$

where K is the bus at fault and S and E are the SB and EB of the line in question.

The line current values are stored in arrays CU and CUR for three-phase and phase-ground respectively these values are then printed following the bus voltages when the proper OUT code is selected.

User's Guide

The User's Guide has been completely re-written. The revision was written with the concept that anyone in the electrical engineering field could follow the instructions for utilization of the program to include preparation of the input data and analysis of the results.

All changes and additions discussed in the preceding paragraphs have been incorporated into the new guide. The layout of figures and tables was done with the expectation that these might be extracted and used separately, therefore much information contained in the writing was purposely duplicated in the tables and figures.

Some additional information, such as wire tables and current equations, has been added to assist the user in preparing the data for input to the program. Although the CDC CYBER computer is the only system presently capable of using PDSAP, the guide is designed so that persons at any location can prepare the input data based on instructions in the User's Guide. Likewise, the output products can be analyzed at any location with the aid of the guide.

III. Tests and Results

Introduction

The overall PDSAP program had many changes and additions during this revision. Each change or addition was implemented and tested until the desired accuracy was achieved. Although the results cited in the following paragraphs do not compare exactly with published results for the same problems, it is noteworthy that the degree of error is tolerable given to inaccuracies that are inherent with such a computer program.

Testing of simulated large systems was done to a limited degree. Systems with up to 167 buses have been run with satisfactory results. However, due to time limitations, not all functions of the revised program have been completely tested on systems larger than 130 buses. Changes in each routine were tested and the results summarized in the following paragraphs.

Line Element Input Routines

In the LINEZ routine, the significant changes were to modify the zero sequence impedance values when neutrals were present and to assume a zero sequence impedance value when no input value was specified. Testing for the change consisted of using an example problem (Ref 7:2-34 to 2-43) where the zero sequence impedance was assumed to be 3.5 times the positive sequence impedance. The program values were identical to the example problem values. Likewise, the same problem

was modified and the effect of neutrals on the impedance values was tested. The results were as expected; since a constant, 2.7, is used when neutrals are considered.

In the LINDATA routine, the changes were more extensive and designed to give greater accuracy in calculating the zero sequence impedance values. To test the routine designed to adjust the zero sequence impedance value for the affect of neutrals, a problem was especially designed to compare approximate values from a published table (Ref 3:14-15). The results of the comparison are given in Table I. The top figure is the published value and lower figure the program calculated value. The first two columns are the positive sequence impedance values, the next two the zero sequence values, and the last two the zero sequence values with neutrals. It should be noted that having the wide range of wire sizes available necessitates some approximations. However, as shown in the table, the difference is not of the magnitude that the program results would be in jeopardy.

Load Flow Tests

In the Load Flow routine, since minor changes were made to prevent the program from going into a mode error condition for certain systems, some of the same problems used for testing in the original program were again tested. The Stagg and El-Abiad System (Ref 5:284-299), shown in Fig. A-18, give the identical results as previously reported (Ref 2:41).

One of the major changes in the routine was the addition

Table I
Wire Impedance Value Comparisons

Wire Size (CM)	Type	R ₁	X ₁		R ₀		X ₀		R _{OG}	X _{OG}
			(ohms per mile)							
500,000	Cu	.1229	.6310	.4161	2.9600	.6922	1.9388			
		.1190	.6332	.4047	2.9566	.6761	1.8711			
300,000	ACSR	.3421	.6442	.6278	3.1590	1.0259	2.2429			
		.3095	.6618	.5951	2.9852	1.0141	2.0901			
250,000	Cu	.2571	.6732	.5428	3.0022	.8189	1.9811			
		.2333	.6761	.5190	2.9994	.7903	1.9139			
105,500	ACSR	1.1200	.8422	1.4061	3.3570	1.8200	2.5312			
		.8856	.8284	1.1712	3.1518	1.6283	2.3853			
83,690	Cu	.7651	.7461	1.0512	3.0751	1.3881	1.9811			
		.6999	.7475	.9855	3.0708	1.3236	2.0758			
66,370	ACSR	1.6901	.8511	1.8850	3.3660	2.3910	2.5402			
		1.4093	.8570	1.6949	3.1803	2.1139	2.4235			

Table II
Comparison of Bus Voltages

Bus No.	1st Run (Original)		2nd Run (Modified)		3rd Run (Original)	
	MAG(p.u.)	ANG(deg)	MAG(p.u.)	ANG(deg)	MAG(p.u.)	ANG(deg)
1	1.06	0.0	1.06	0.0	1.06	0.0
2	1.037	-2.6357	1.0364	-2.3942	1.037	-2.640
3	1.009	-4.7990	1.0082	-4.3160	1.009	-4.807
4	1.008	-5.1239	1.0064	-4.5358	1.007	-5.133
5	1.002	-5.9684	1.0012	-5.6227	1.002	-5.981

of the load change capability. This was tested using the same system by changing to new bus loads on the second run and then back to the original values on the third run. As noted in Table II, there are slight differences between the first and last result. However, the differences, probably introduced by rounding errors, should not measurably affect the overall program accuracy.

Short Circuit Testing

To test the Short Circuit routine, a comparison was made with the example problem called the DSPM System (Ref 3:40-44). The DSPM system is an 18 bus radial system with copper aerial conductors and aluminum underground cable, Fig. A-21. This system uses the LINDATA input routine to calculate impedance values. Table III shows the published values on the top line for fault currents and the PDSAP program calculated value on the second line. There was no zero fault impedance on faults for the underground cable and 20 ohms resistance for the aerial faults.

Analysis of the data reveals that the program values are very close to the published values in all cases. The largest error is approximately 2.2% on the Ph-Ph-Gnd(B) value for Bus 6. Considering the slight difference in conductor impedance values calculated by the program as shown in Table I, the accuracy of the Short Circuit routine is considered very good. This same problem was used for the original program (Ref 2:43). The calculated values for buses 1, 12, and 13 are very similar. The others vary as zero

Table III
Comparison of Fault Currents - DSPM System

Bus No.	3Ph-Gnd	Ph-Gnd	Ph-Ph	Ph-Ph-Gnd(B)	Ph-Ph-Gnd(C)
	(Values in Amperes)				
1	3123	3793	2705	3506	3720
	3123	3793	2705	3506	3720
2	2506	2947	2170	3068	2497
	2504	2944	2169	3065	2496
3	335	332	533	1822	1665
	336	332	535	1828	1670
5	294	277	414	913	799
	297	279	421	928	810
6	276	254	372	730	633
	280	257	379	746	643
7	0	240	0	0	0
	0	244	0	0	0
10	2244	2579	1944	2756	2083
	2243	2576	1943	2753	2082
11	2059	2322	1783	2518	1831
	2059	2319	1783	2514	1830
12	1924	2138	1666	2340	1650
	1923	2135	1665	2337	1665
13	1729	1879	1497	2082	1448
	1728	1876	1496	2078	1449
16	1972	2202	1707	2403	1721
	1971	2199	1707	2399	1721
17	0	2125	0	0	0
	0	2124	0	0	0

fault impedance was used previously for all buses.

Another test problem run with the Short Circuit routine was the Westinghouse example problem (Ref 7:2-34 to 2-43) to illustrate that a three-winding transformer can be simulated as three two-winding transformers and to verify the new line current routine. Fig. A-24, page 115 is a one line diagram of the problem. Using the procedure detailed in Appendix A, the three-winding transformer impedance values were converted to three two-winding transformers and input to the program through the LINEZ routine. The results are shown in Table IV. In this example problem, the text values were altered to reflect the positive and negative sequence values being equal. In Table IV, the text values are on the top line and the program values on the following line. The error between the values is less than 1% on the average. As the three-winding transformer was modeled differently in the program, no current magnitude comparison was made for the three-winding transformer buses.

Summary

The additions and changes to PDSAP appear to greatly enhance the value of the program. Although extensive testing in all areas was not permitted due to time constraints, several additional examples consisting of large and small systems have been simulated with success. In some cases, comparison data is not available, but indications tend to support the supposition that the program results are well within acceptable tolerances for all functions.

Table IV.

Comparison of Fault Currents - Westinghouse System

Line SB - EB	3Ph-Gnd Current Magnitude in Amperes	Ph-Gnd
1 - 3	572.5	359.1*
	571.7	358.7
2 - 3	201.8	126.5
	202.0	126.7
3 - 4	1095.6	1124.9
	1095.1	1125.0
4 - 5	793.7	652.0
	793.6	652.2
5 - 8	535.5	335.6
	534.9	335.5

*All text values estimated on positive and negative sequence impedances being equal.

IV. Conclusions and Recommendations

Introduction

The purpose of this chapter is to summarize the significant results obtained from the revised PDSAP program. The chapter is divided into two areas, conclusions and recommendations. The conclusions will detail information gained from the project regarding capabilities and limitations of the program. Recommendations will detail suggested changes and additions that might be added at some future time.

Conclusions

Program Capabilities. The goal to expand the capability of the PDSAP program to simulate certain power distribution systems not possible with the original version was realized. Systems with neutrals and non-radial systems can now be successfully simulated with the program. Accuracy of the program improves with the quality of the input data, but results obtained from using the minimum data required will often suffice. The program has been exercised on simulating several different distribution systems, large and small, with the largest being 167 buses. Comparison of the program results with published data on sample problems shows that the PDSAP program has acceptable accuracy in both load flow analysis (Ref 2:41) and short circuit analysis (Ref 3:44).

Limitations. The PDSAP program has certain limitations. As pointed out by the original author (Ref 2:50), the pro-

gram uses a series impedance representation for the Load Flow routine and this may not be adequate for systems with large shunt admittances. Also, by experience, it has been proven that some systems can not be successfully simulated by the Load Flow routine. Changes to the program will prevent termination due to processing, or mode, errors; but the output data will be of questionable accuracy.

Another limitation with the Load Flow routine, is the program capability to simulate low voltage systems. The program was primarily designed to simulate high voltage (above 600 V) systems where the impedance ratio, X/R , was less than approximately 0.4. Problems where the R value was large compared to the X value tended to cause convergence problems. This limitation is not unusual, and has been encountered by others (Ref 4:868).

In the input data, certain limitations had to be placed on transformer winding configurations. As a result, three-winding transformers can not be simulated unless used as three two-winding transformers. With the limited number of three-winding transformers in use, this program limitation should affect relatively few prospective users.

At present, the program has no capability for transient stability analysis. Certain information, such as generator reactances, needed to determine the stability constraints of a system would have to be added before this feature could be included.

Most power distribution systems on Air Force installa-

tions use a mixture of single and three-phase circuits. At present, there is no provision in the program to simulate single phase branch circuits on a specified phase of a three-phase system. This can cause problems in adequately simulating the system where the attachment of single phase branch circuits will noticeably alter the load balance between phases.

Recommendations

General. PDSAP is an excellent program for power distribution system analysis, but additional capability would enhance its usefulness even further. As noted in the foregoing material, several additions and changes were made to the original program. Where practical, additions and changes were integrated into the existing logic structure to require as little additional computer memory space as possible. Although the revised program is apparently working accurately in all respects, it is recommended that before many more additions or changes are made, an attempt be made to condense the present program. For example, adoption of a dynamic memory allocation procedure would greatly increase the efficiency of the program. With this method, memory space would be allocated according to the size of the problem instead of using the maximum for every problem.

The recommendations cited by the original author remain valid (Ref 2:51-52) with the exception of those items incorporated into this revision. Especially the need for transient stability analysis and a line outage routine. Modifi-

cation to the network representation from a series to a model in the Load Flow routine would also be considered a desirable change.

One general recommendation that would greatly enhance the value of the program is to add some method of specifying how single phase branch currents are connected to three-phase systems. At many installations in the Air Force, the power distribution system will have several single phase branch circuits. Presently, single phase loads on branch circuits have to be simulated as three-phase loads or ignored when using the Load Flow routine. This kind of approximation could have an undesirable effect for accurate analysis of the system.

To further enhance the credibility of the PDSAP program, extensive testing using the IEEE Standard Test System and then a comparison with published results for these test systems is recommended. Some of the logic in the PDSAP program is based on published articles that used the IEEE standard systems (Ref 4:864), and a comparison of the results from different programs would be noteworthy.

Input Routines. Often errors on the line element data cards cause premature termination of the program. Development of a separate data scan program, using minimum computer memory space and processing time, is recommended. With the use of a separate input error checking routine, computer usage would be considerably more efficient than the present methods with error checking in the main program.

In the LINDATA routine, adding the capability to use all aluminum wire or cable would be a worthwhile addition. As more aluminum comes into use, the need for this change may become more urgent.

Also in the LINDATA routine, expanding the wire tables to lower voltages and smaller wire sizes is recommended. From data submitted to date, there appears a need for a computer program to analyze power distribution systems for individual buildings. As often only sketchy information is available on drawings, use of the LINDATA routine would be very valuable in these situations.

Load Flow Routine. As presently configured, the Load Flow routine has certain limitations. It is often difficult to satisfactorily simulate the systems with X/R ratios less than approximately 0.4. Using the fast decoupled method and trying to keep the program condensed as much as possible, the low X/R ratios can cause convergence problems when the 0.4 ratio is exceeded. This type limitation is most likely to be encountered with low voltage (less than 600 V) systems. If it is desired to extensively utilize the PDSAP on low voltage systems, this problem would have to be overcome since X/R ratios less than .4 are very common at lower voltages.

Short Circuit. In addition to the line outage routine previously recommended, a routine to change certain short circuit input data would be useful. In this regard, if fault impedances could be changed, similar to the load change capability in the Load Flow routine, the time required for additional computer runs could be minimized.

Bibliography

1. Brown, H. E. Solutions of Large Networks by Matrix Methods. New York: John Wiley and Sons, 1975.
2. Heer, Michael A. A User-Simplified Power Distribution System Analysis Program. Unpublished thesis, Wright-Patterson Air Force Base, Ohio: Air Force Institute of Technology, December 1975.
3. McGraw-Edison Company, Power Systems Division. Distribution Systems Protection Manual. Canonsburg, Pennsylvania: McGraw-Edison Co.
4. Scott, B. and O. Alsac. "Fast Decoupled Load Flow." IEEE Transactions on Power Apparatus and Systems, PAS 93:859-869 (May/June 1974).
5. Stagg, G. W. and A. El-Abiad. Computer Methods in Power Analysis. New York: McGraw Hill Book Co., 1968.
6. Stevenson, W. D., Jr. Elements of Power System Analysis (Third Edition). New York: McGraw-Hill, Inc., 1975.
7. Westinghouse Electric Corporation. Applied Protective Relaying. Newark: Westinghouse Electric Corporation.
8. Westinghouse Electric Corporation. Electrical Transmission and Distribution Reference Book (Fourth Edition). East Pittsburgh, Pennsylvania: Westinghouse Electric Corporation, 1964.

Supplemental Bibliography

1. Clarke, Edith. Circuit Analysis of A-C Power Systems. Vol I: Symmetrical and Related Components. New York, New York: John Wiley and Sons, 1943.
2. Essex International, Power Conductor Division. Essex Underground Cable Engineering Handbook. Marion, Indiana: Essex International, Inc. 1971.
3. Myatt, L. J. Symmetrical Components. New York: Pergamon Press Ltd., 1968.
4. Peterson, Norris, M., William F. Tinney, and Donald W. Bree, Jr. "Iterative Linear AC Power Flow Solution for Fast Approximate Outage Studies." IEEE Transactions on Power Apparatus and Systems, PAS-91:2048-2056 (Sept/Oct 1972).
5. Wylie, R. A. "Digital Calculation of Sequence Impedance and Fault Currents for Radial Primary Distribution Circuits." IEEE Transactions, Part III, Power Apparatus and Systems. Vol 79:1264-1277 (Feb 61).

Appendix A. User's Guide

This appendix contains a User's Guide for the PDSAP program. The guide is written so that it is a complete document that can be withdrawn and used separately with the PDSAP program.

Power Distribution System Analysis Program
(PDSAP)
User's Guide

<u>Chapter</u>	<u>Subject</u>	<u>Page</u>
1	Introduction	30
2	Program Control	35
3	Line Element Data	43
4	Load Flow Analysis	57
5	Short Circuit Analysis	64
6	Troubleshooting	77
7	Example Problems	84

1. Introduction

Purpose

The purpose of the Power Distribution System Analysis Program, PDSAP, is to provide analysis capability for various electrical distribution systems located on Air Force installations and to provide information to aid in the study and operation of these systems. The program has been developed so that it can easily be used to model electrical systems of the type normally operated and maintained by Base Civil Engineering personnel.

Capabilities and Limitations

The PDSAP program is capable of simulating systems with up to 250 buses and 500 lines. The program is formatted for the CDC CYBER 74 computer system and is written in FORTRAN. With necessary programing changes, it could be adapted to other computer systems of similar size. In addition to the lines, 250 transformers can be represented of which 50 may be Tap-Changing Under Load (TCUL) and 25 phase-shifting transformers. Series reactors and capacitors can also be modeled. Up to 50 of the 250 buses can be voltage controlled buses.

The program has 3 major functions: Calculate line impedance values; calculate power flows in lines throughout the system; and calculate fault currents for simulated faults at selected buses in the system.

Although the program is quite large and can simulate

almost any electrical system found with the Air Force, there are certain limitations that are necessary. These limitations are usually minor and will not adversely affect the overall accuracy of the analysis. The specific limitations are noted at appropriate locations in the following chapters. As presently configured, the program has no capability of transient stability analysis nor simulating line outages. Line outages can be simulated by alterations to the input data.

The following chapters of the guide, with the exception of the last two chapters, are entitled to represent the sections of the program: Program Control, Line Element Data, Load Flow Analysis and Short Circuit Analysis. It is suggested that the first two chapters be read in detail prior to attempting to use the program. Each chapter, with the exception of the last two, discusses input requirements and output products unique to that section of the program. Once the user has selected the desired functions in the control section, subsequent sections of little interest can be passed with a brief overview.

Input Data Requirements

The power system analysis program is designed to develop a computer simulation of a power system based in information usually found on a complete one-line system diagram. The program has the flexibility of using this information in almost any form normally found. Depending upon the desire of the user, some input data may be omitted if not using all

the major functions of the program. Information not available can often be approximated by the user. For some variables, preset values within the program can be used if actual values are unknown. Variables with the preset option will be so noted throughout the following chapters. However, for the best results, the more detailed the input information, the greater the accuracy of the result.

Input data for the program is entered by the use of the standard 80 column computer card. Coding of the data on standard coding forms such as AF Form 1531, Punch Card Transcript, is highly recommended before punching of the data cards is attempted. As is true with most computer data, accuracy in the coding is very important. As a general rule, after the coding forms have been prepared, a cross check should be made with the one-line diagram of other source data to insure correctness. Throughout the instructions, the various input requirements will be specified using one of 3 FORTRAN formats: F, I or R. The F format is used where real numbers, such as 875.53, may be used. The I format is used where an integer, such as 9, is required. With an integer format, no decimal points are allowed. No entry always implies an integer value of 0. Integers can be used in F formats, but care should be taken as the decimal point will be inserted automatically. An F5.2 format and an input value of 4839 will be used as 48.39 for example. For integers used with I or F formats, if the number is not right justified, or started in the right most column, zeros will be

added in the empty columns to the right to complete the format. For example, if the format was I3 and a 4 was placed in column 2, a value of 40, would be assumed by the program. Use of commas in large numbers is not permitted; 1,043 should be 1043 as an example.

The R format is an alphanumeric format where various combinations may be used as desired. For example, an R7 format could be Alpha 2. The program has several important control cards in each function. The cards are started with an R format and use a key word such as PGMCON. The purpose of the key word is to insure the control cards are properly identified and located in the program. The spelling and location of the key word on the control cards is essential and it is imperative that the data be entered in the columns specified for proper computation.

Output Data and Errors

The output data will vary in format and quantity depending on which of several options is selected by the user. In each of the following chapters, the output products are discussed for each major function of the program.

The PDSAP program contains a fairly comprehensive data checking and error detection routine. Should a problem be encountered with the input data or in computation, the program will output a message to identify the problem. Usage of the error messages in resolving problems is discussed in the next to last chapter of the guide.

Examples

The last chapter presents several examples showing the various input and output formats. These examples are necessarily brief due to space limitations, but can be used as a guide in modeling larger systems.

2. Control

Introduction

In the PDSAP program there are two types of control cards, FORTRAN and program. FORTRAN control cards are used to load the program into the computer's permanent file or to access the program if already on file. The FORTRAN control cards should be prepared by the person who will be submitting the information to the computer center. Program control cards determine the functions of the program to be used, output products and system parameters. Program control cards should be prepared by the person who is preparing the data for computation.

FORTRAN Control Cards

The PDSAP program exists as a source deck, or data card deck, consisting of some 3700 standard computer cards. The program must be loaded into the computer and a binary file created before data can be submitted for processing. To load the source deck, the following cards are required:

1. xx,T40,IO80,CM164000,STCSB. Txxxxxx,Name,Tel.Ext.
2. REQUEST,PDSAP,*PF.
3. MAP,PART.
4. FTN,OPT=2,B=PDSAP,R=3.
5. LOAD,PDSAP.
6. NOGO.
7. REWIND,PDSAP.
8. CATALOG,PDSAP,CY=xx,RP=xxx,ID=Txxxxxx,XR=xxxx.

9. *EOR
10. Source deck
11. *EOR
12. *EOF

For detailed information on FORTRAN cards, consult the User's Guide available at the computer terminal or see the FORTRAN Reference Manual. Alternative options are available on the FTN card as noted in the preceding references.

The following FORTRAN control cards are used when the source deck has been previously loaded, compiled and permanently filed in the computer.

1. xxx,Txxx,IOxx,CM164000,STCSB. Txxxxxx,Name,Tel.Ext.
2. ATTACH,PDSAP,CY=xx,ID=Txxxxxx.
3. PDSAP.
4. DISPOSE,TAPE1,PR=IBB.
5. DISPOSE,TAPE2,PR=IBB.
6. *EOR
7. Data Cards
8. *EOR
9. *EOF

The first Txxx is processing time requested. For impedance calculations and/or short circuit analysis, T50 is usually sufficient. For load flow analysis a larger Txx, up to T200 or more, may be required for time to obtain the desired convergence on a large system. IOxx is the amount of input/output time required. For programs of 50 buses or less, IO20 should be sufficient. Systems with more buses

may require IO50.

Other combinations of control cards are possible. For a one-time job, the source deck and data deck could be run at the same time, however this method might prove to be slightly more expensive and less efficient. The source deck is not designed to be repeatedly loaded and great care must be used not to get the deck out of sequence.

The program uses two output files, Tape 1 and Tape 2. Information printed on the tapes will be detailed in later chapters and illustrated with the example problems.

Program Control Cards

The program is designed to perform three separate functions as described in the introduction. The order and format of the program control cards is very important as all calculations will be affected by these values. Program control cards will be referred to as "control cards" throughout the following chapters and do not include the FORTRAN control cards previously discussed. The first three control cards title the system, select the functions desired and establish parameters to be used throughout the program. Figure A-1 will show the overall program card deck arrangement. This figure and the examples in the last chapter should be used to arrange the cards in proper order.

The first control card, which follows the first *EOR FORTRAN control card, is the Title card. This card can have 80 alphanumeric characters, in two groups of 40 columns each.

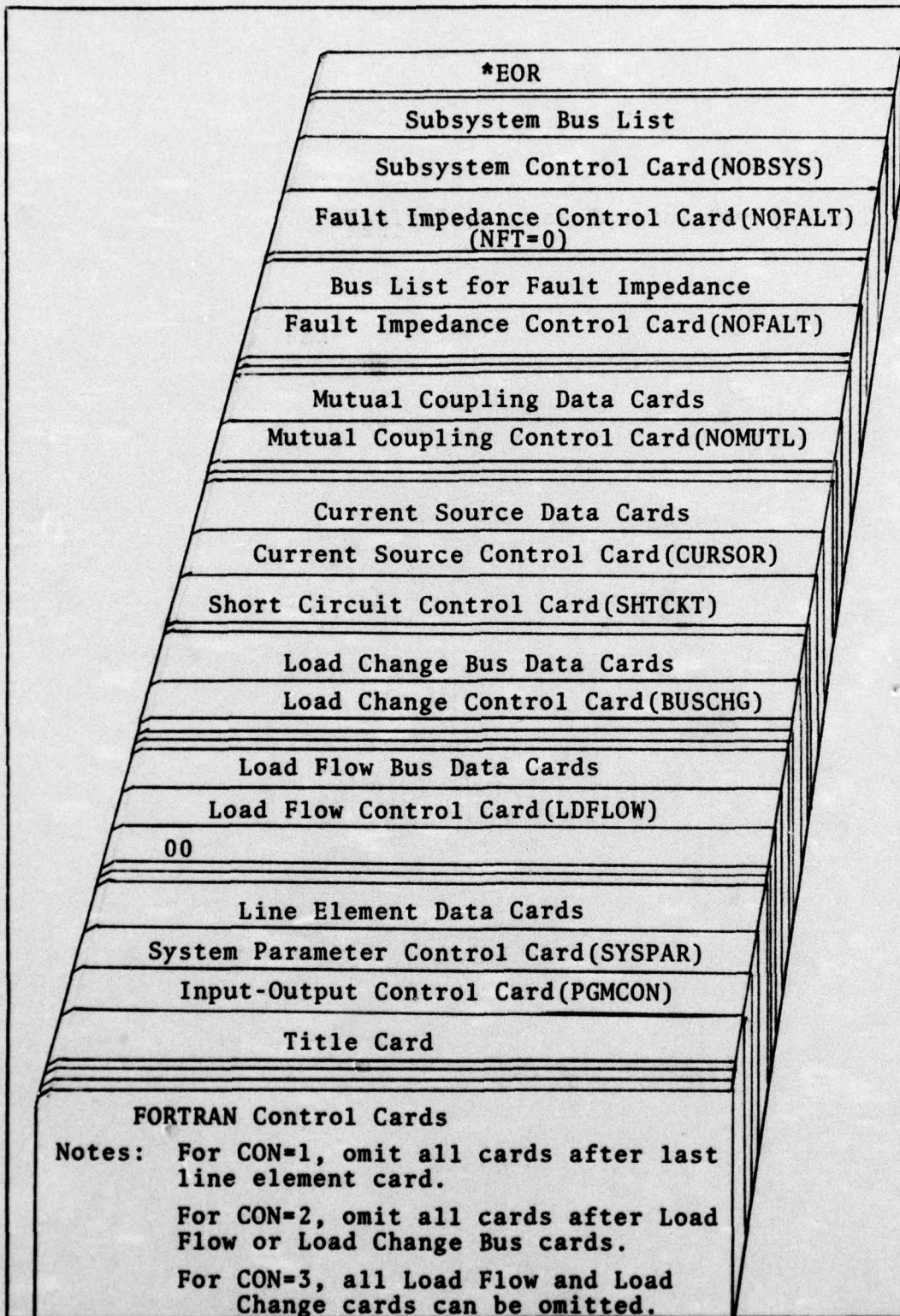


Fig. A-1. PDSAP Card Deck Structure

This card provides two "title" lines for the program. The first 40 columns will appear on one line and the second 40 columns on the following line.

The second control card is the Input-Output card. This card controls the selection of the program function, input format to be used, and output products produced. Four variables: CON, INP, OUT, and CHG are used on this card. The CON variable is for the function control, the INP variable is for line data input control, the OUT variable is for output selection, and the CHG variable is not used at present. Table A-I details the code values for the Input-Output card.

The third control card is the System Parameter card. This card is used to select the base KVA, frequency, temperature and earth resistivity. The program uses pre-selected values for any parameter not specified on this card. Table A-II specifies the variable location for the System Parameter card.

Fig. A-2 shows the location of the variables on the first three control cards. Examples of actual control cards are in the last chapter.

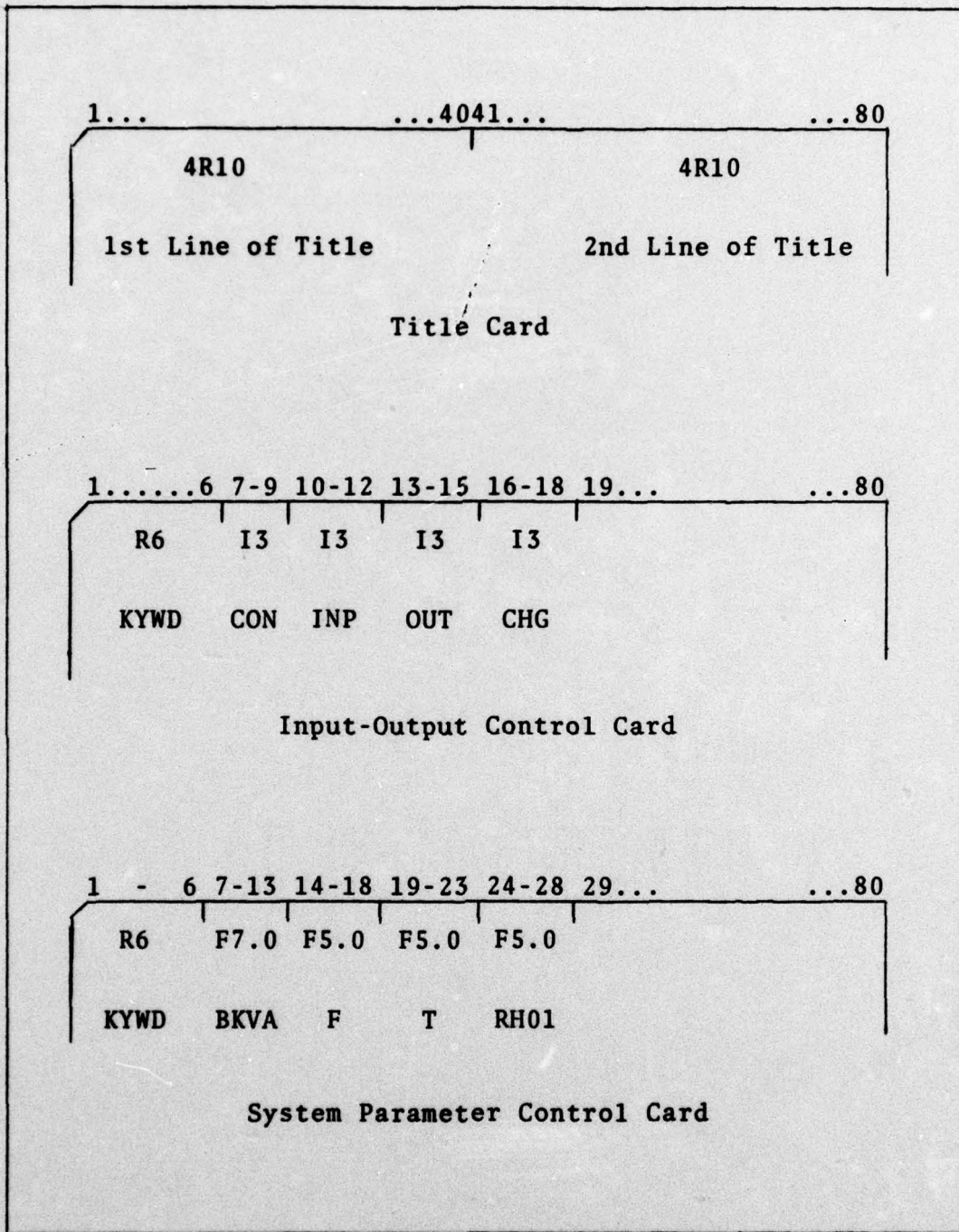


Fig. A-2. Title, Input-Output, and System Parameter Cards

Table A-1
Input-Output Control Card Variables

Variable (Card Col)	Definition/Use
KYWD (1-6)	Keyword, use PGMCON.
CON (7-9)	Function selector. Choose 1, 2, 3, or 5. 1=Assemble line input data only. 2=Assemble line input data and run Load Flow analysis. 3=Assemble line input data and run Short Circuit analysis. 4=Not used at present. 5=Assemble line input data and run both Load Flow and Short Circuit analysis.
INP (10-12)	Input selector for line data. Choose 1 or 0. No entry implies 0. Use 1 when input impedances in ohms or per-unit. See chapter on line data.
OUT (13-15)	Output selector. Choose value from Table A-III for desired output lists. No entry implies 0.
CHG (16-18)	Not used at present, leave blank.

Table A-II
System Parameter Control Card Variables

Variable	Definition/Use
KYWD (1-6)	Keyword, use SYSPAR.
BKVA (7-13)	Base KVA. Use selected base. No entry implies 100,000 KVA. Base KVA range: 1 to 100,000 KVA.
F (14-18)	Frequency in hertz. No entry implies 60 hz.
T (19-23)	Temperature in degrees C. No entry implies 25 C.
RH01 (24-28)	Earth resistivity in meter-ohms. No entry implies 100 meter-ohm.

Table A-III
Codes for OUT Variable

Code	Definition
0	No outputs suppressed. Assembled line data printed in per-unit values.
1	No outputs suppressed. Assembled line data printed in ohms.
2	Suppress Tape 1 data. This includes all line data, input bus lists, input short circuit data.
3	Suppress assembled line data only. Sorted list of line data still printed.
4	Suppress sorted line data (G and B list). Assembled line data printed in per-unit values.
5	Suppress both assembled and sorted line data.
6	Suppress line flows in Load Flow function. Output bus data printed.
7	Suppress reordered bus list, input bus list printed.
8	Suppress assembled and sorted line data, reordered bus list and input bus list.
9	Suppress transformer output data. Line list and bus list printed.
10	Suppress all outputs except Load Flow bus summaries and short circuit fault summaries.
11	Start new page for each fault summary.
12	Suppress Short Circuit voltage and line current summaries.
13	Suppress Short Circuit voltage summaries.
14	Suppress Short Circuit voltage summaries and start new page for each fault summary.
15	Applies to LINDATA routine only. Input data is printed exactly as appears on data cards, program halts.

3. Line Element Data

Introduction

There are two line element routines available, LINEZ and LINDATA. The LINEZ routine should be used where input impedance values are in per-unit, ohms or a combination of the two. The LINDATA routine should be used where impedance values are unknown. The size, rating, length and other variables with LINDATA routine are used to calculate the correct impedances. The two routines cannot be used in the same system simulation. Therefore, the data for the line elements should be reviewed and the routine selected that will result in as little manual computation by the user as possible. As noted in the preceding chapter; on the Input-Output control card (PGMCON) a value of 1 for INP selects the LINEZ routine and a 0 selects LINDATA.

Both the LINEZ and LINDATA routines use several variables to compute the line impedance values which are then used by other functions of the program. Each data card is used to represent a line element. Line elements are aerial lines, underground cables, transformers, series capacitors, and series reactors. All line elements must be connected between two buses. Bus numbers must start with 1 and run consecutively until all buses are numbered. The location of the bus numbers is not significant, but some logical order will facilitate cross checking for errors and omissions. Bus locations should be chosen at points of interest and where line

elements change. Locations would normally be ends of lines, both sides of transformers, both sides of switches, and where loads are connected.

The line element data cards can be assembled in any order except the last data card which represents the ground or reference. This card should have a start bus (SB) and end bus (EB) of 0. No other entry is used on this card.

Neutral or ground wires in close proximity to phase conductors will have an effect on zero sequence impedance values. The presence of neutrals can be coded in both routines when simulating aerial conductors. The routines adjust the zero sequence impedance values to account for the presence of the neutrals.

Both routines use the same codes for line element identifiers (ID) and transformer windings (C). Codes for these variables are listed in Tables A-IV and A-V.

Table A-IV
ID Code Values

ID Code	Line Element Description
1	Copper, aerial
2	ACSR, aerial
3	Underground cable, aluminum conductor
4	Underground cable, copper conductor
5	Transformer, fixed up
6	Transformer, auto
7	Transformer, TCUL
8	Transformer, Phase-shifter
9	Series Capacitor
10	Series Reactor

Table A-V
Transformer Connection Codes

Type	Code	Winding Description (SB/EB)
Fixed or TCUL	1	Y/Y, Y/Y _T , X /Y, Y/Δ, Δ/Y
	2	Δ/Δ
	3	Y _T /Y _T
	4	X /Δ
	5	Δ/Y _T
Auto	1	Y/Y (Not 3-ph core type)
	2	Y _T /Y _T (Not 3-ph core type)
	3	Y _T /Y _T (3-ph core type)
	4	Y _T /Y _T two series windings (Not 3-ph core type)
	5	Y _T /Y _T two series windings (3-ph core type)
Phase-Shifter	1	Series; Y/Y auto exciting (grounded or ungrounded neutral).
	2	Series; Y/Y auto exciting (grounded neutral, 3-ph core type).
	3	Series; Y/Y auto exciting (grounded neutral/tertiary).
	4	Series; Y/star exciting (grounded or ungrounded neutral).
	5	Series; Y/star exciting (grounded neutral, 3-ph core type).
	6	Series; Y/star exciting (grounded neutral, tertiary).
	7	Excited series.
	8	Star series; Δ/star exciting (grounded or ungrounded neutral).

LINEZ Routine

The LINEZ input routine, INP=1, is capable of accepting only values in per-unit or ohms for all line elements. Values for both positive and zero sequence, operating voltage, and bus connections are entered on each line element data card. If the zero sequence impedance is not specified, the program assumes a value for lines based on 3.5 times the positive sequence impedance for lines with neutrals. For transformers, capacitors, and reactors, the zero sequence impedance value is set equal to the positive sequence impedance value unless specified otherwise on the data card.

The format for the LINEZ line element data card is shown in Fig. A-3. The variables and the definitions for the LINEZ data card are given in Table A-VI.

LINDATA Routine

The LINDATA routine, INP=0, is designed to accept line element information that may be available when the impedance values are unknown. The conductor size, length, spacing and other information is used to calculate line impedances. Table A-VII shows the various sizes and strands of conductors that can be simulated in the program. The routine has the capability to use different size wire for phase conductors than that used for the neutral. If data on neutral conductors is not available, an approximation is made of 133,100 CM for ACSR neutrals and 105,500 CM for copper neutrals. KVA ratings, voltages, and winding information are used to

1-3	4-6	7-11	12-13	14	15-20	21-26	27-32	33-38	39-44
I3	I3	F5.0	I2	I1	F6.0	F6.0	F6.0	F6.0	F6.0
SB	EB	VP	ID	PH	L	ReZZ	ImZZ	ReZZO	ImZZO
45-48	49-53	54-58	59-63	64-65	66-67	68-73	74-78	79	80
F4.0	F5.0	F5.0	F5.0	I2	I2	F6.0	F6.0	I1	
PHI	TPI	TMNN	TMXX	C	IUNIT	ReZN	ImZN	CH	

Fig. A-3. Line Element Card Format for LINEZ Routine

Table A-VI
Line Element Variables for LINEZ Routine

Variable (Card Column)	Definition/Use
SB (1-3)	Start bus; bus number to which one end of line element is connected. Also called "from bus." Must be tapping end of a TCUL element.
EB (4-6)	End bus; bus number of other end of line element. Also called "to bus."
VP (7-11)	For conductors, capacitors, and reactors: Voltage (KV) line-to-line if 3-ph or line to-neutral if 1-ph. For Transformers: Voltage (KV) high side rating.
ID (12-13)	Line element identifier; Choose code from Table A-IV.
PH (14)	Phase. Choose 1 or 3.

Table A-VI (Cont'd)

Variable (Card Column)	Definition/Use
L (15-20)	For conductors: Length in feet. For other elements: KVA rating in KVA.
ZZ (21-26) (27-32)	Positive sequence impedance. Complex, enter as two numbers in ohms or per-unit values.
ZZ0 (33-38) (39-44)	Zero sequence impedance. Complex, enter as two numbers. If unknown, leave blank; program will assume values based on positive sequence values.
PHI (45-53)	Phase angle (degrees). For phase-shifter only. All other elements: Leave blank.
TPI (49-53)	For conductors: Enter 1 for zero sequence adjustment for ground or neutrals, otherwise skip. For transformers: Initial tap setting in per unit. No entry implies 1.0. For other elements: leave blank.
TMNN (54-58)	For TCUL transformer, minimum tap limit in per unit. No entry implies .9. For other elements: Leave blank.
TMXX (59-63)	For TCUL transformer, maximum tap limit in per unit. No entry implies 1.1. For other elements: Leave blank.
C (64-65)	For conductors, capacitors and reactors: Leave blank. For transformers: Choose connection code from Table A-V.
IUNIT (66-67)	Impedance units identifier. For all elements: Blank or 0 for impedance values in ohms. 1 implies per-unit on system base KVA. For transformers capacitors and reactors: 2 implies per-unit on element base KVA.
ZN (68-73) (74-78)	For conductors, capacitors and reactors: Leave blank. For transformers, neutral impedance if other than zero. Use ohms or per-unit values.
CH (79)	Not used presently, leave blank.

Table A-VII
Wire Sizes

Circular Mils (CM)	A.W.G.	Comments
1,000,000	-	If UG, 3-ph only
750,000	-	If UG, 3-ph only
500,000	-	If UG, 3-ph only
350,000	-	
300,000	-	
250,000	-	
211,600	4/0	
167,800	3/0	
133,100	2/0	
105,500	1/0	
83,690	1	
66,370	2	
41,740	4	
26,250	6	
16,510	8	

Number of Strands:

Copper: 37,19,12,7,3,1, or 0

ACSR (Aluminum part): 54,30,26,6, or 0

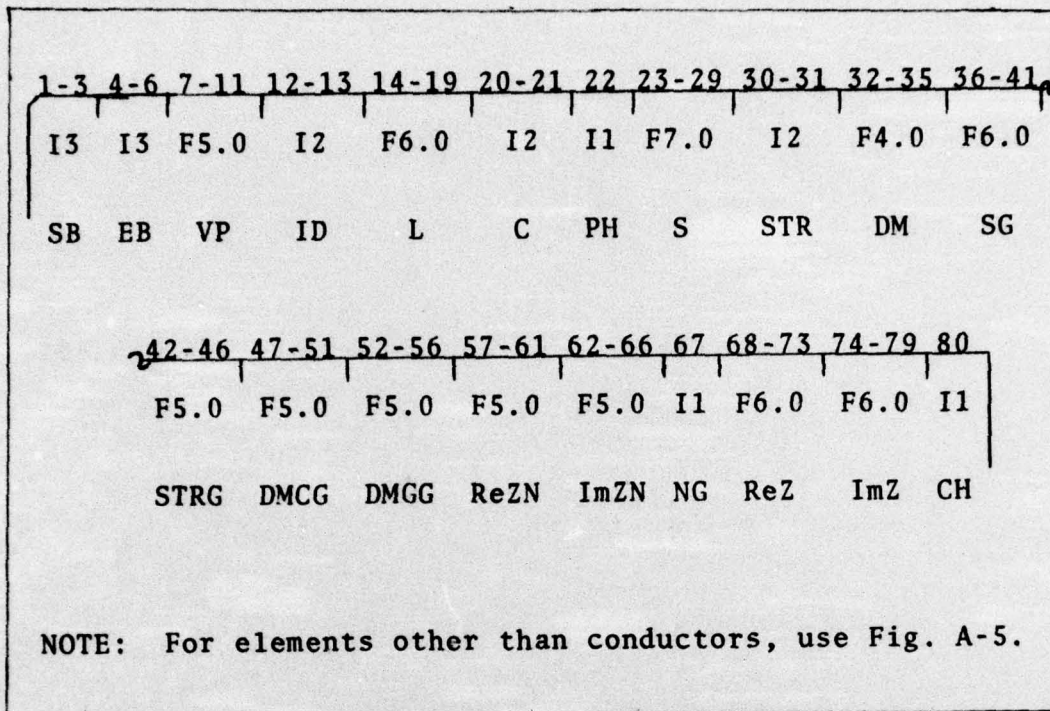


Fig. A-4. Conductor Data Card Format for LINDATA Routine.

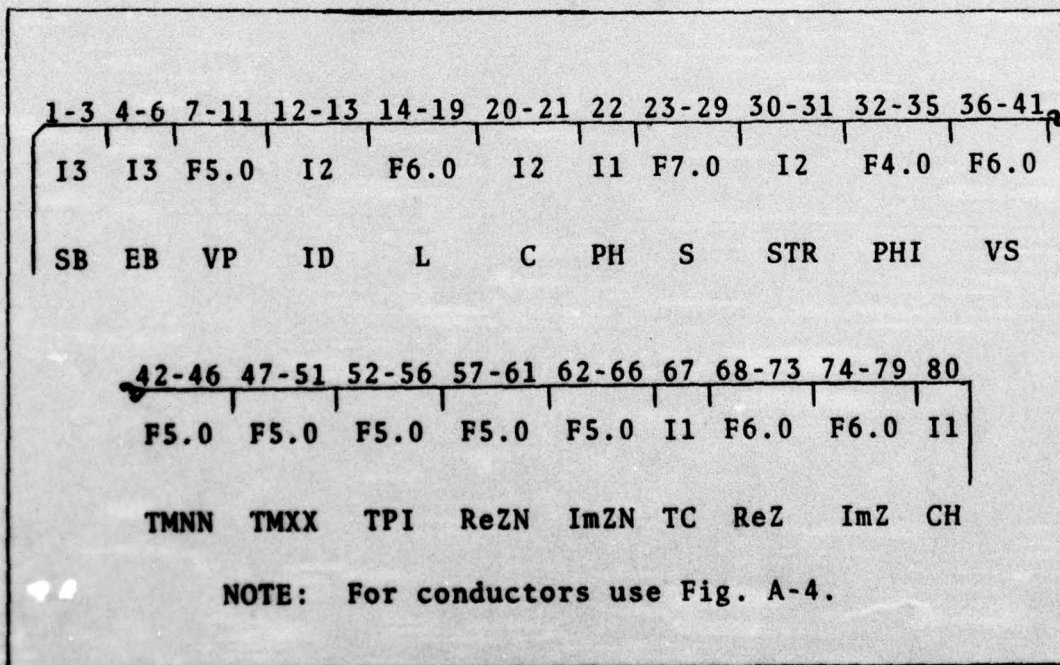


Fig. A-5. Data Card Format for LINDATA Routine.

Table A-VIII
Line Element Variables for LINDATA Routine

Variable (Card Column)	Definition/Use
SB (1-3)	Start bus: Bus number to which one end of line element is connected. Also called "from bus." Must be tapping end of a TCUL element.
EB (4-6)	End bus: Bus number of other end of line element. Also called "to bus."
VP (7-11)	For conductors: Voltage (KV) line-to-line if 3 ph or line-to-neutral if 1 ph. For transformers: Voltage (KV) high side.
ID (12-13)	Line element identifier: Choose code from Table A-IV.
L (14-19)	For conductors: Length in feet. For other elements: Leave blank.
C (20-21)	Connection code. For conductors: Leave blank. For transformers: Choose code from Table A-V.
PH (22)	Phase. Choose 1 or 3.
S (23-29)	For conductors: Diameter in circular mils (CM). See Table A-VII for choices on wire sizes. For transformers and others: Rating in KVA.
STR (30-31)	For conductors: Number of strands. See Table A-VII for choices. If unknown, leave blank. For other elements: Leave blank.
DM or PHI (32-35)	For conductors: Equivalent spacing in feet. If unknown, leave blank and value assumed as 4.69 for 25 KV or less, 12.57 for above 25 KV. For phase-shifter transformers: Phase angle. For other elements: Leave blank.
SG or VS (36-41)	For conductors: Size of neutral in circular mils. If none or unknown, leave blank. For transformers: Secondary voltage in KV. For other elements: Leave blank.

Table A-VIII (Cont'd)

Variable (Card Column)	Definition/Use
STRG or TMNN (42-46)	For conductors: Number of strands in neutral, if known. If unknown or none, leave blank. For transformers: Minimum tap setting for TCUL in per-unit. If blank, assumes .9. For other elements: Leave blank.
DMCG or TMXX (47-51)	For conductors: Equivalent spacing between conductors and neutral (ft). If unknown or none, leave blank. For transformers: Maximum tap setting for TCUL in per-unit. If blank, assumes 1.1. For other elements: Leave blank.
DMGG or TPI (52-56)	For conductors: Equivalent spacing between neutrals if more than one. If unknown or none, leave blank. For transformers: Initial tap setting in per-unit. If blank, assumes 1.0. For other elements: Leave blank.
ZN (57-61) (62-66)	For conductors: Leave blank. For transformers: Neutral impedance, if other than zero, in ohms. Enter as real and imaginary. For other elements: Leave blank.
NG or TC (67)	For conductors: Number of neutral of ground wires. For other elements: Positive sequence impedance selector. 0 will use program calculated value. 1 will use value specified by Z variable as ohms. 2 will use value specified by Z variable as per-unit on base KVA of line element.
Z (68-73) (74-79)	For conductors: Leave blank. For transformers: Positive sequence impedance in ohms or per-unit. Complex value, enter as two numbers in ohms or per-unit. If unknown, insure TC variable is 0.
CH (80)	Not used presently, leave blank.

determine the impedance values for transformers and other elements.

Two formats are used for the LINDATA routine, one for conductors and the other for transformers, series capacitors and series reactors. Zero sequence impedance values are calculated in the routine based on the positive sequence values. Fig. A-4 shows the format for the conductors and Fig. A-5 shows the format for other elements. Variables for both formats are defined in Table A-VIII.

Limitations

Both LINEZ and LINDATA routines have certain limitations. Due to the algorithm used with the Load Flow routine, input line elements usually cannot have an X/R ratio less than .4, but this restriction can be waived if only using the Short Circuit routine. With the LINEZ routine, the data can be checked before being entered on the data cards. With the LINDATA routine it may be necessary to run the program, using CON=1 on the Input-Output control card, to calculate impedance values in order to determine the X/R ratio.

The line element routines do not have the capability to simulate a three-winding transformer except those used as phase shifting transformers can be simulated if only two windings are connected to the system. The problem of simulating a three-winding transformer can be resolved by modeling the transformer as three separate two-winding transformers as described by the following procedure:

1. Identify the impedance of the three windings; Hi (ZH), Med (ZM), and Lo (ZL). Note winding information such as Hi/Med is (Wye-Delta).

2. Using the impedance values given for Zh1, Zmh, and Zml, calculate vales for ZH, ZM, and ZL as shown. Zh1, Zml, Zmh are the given impedance values for Hi/Lo, Med/Lo and Med/Hi windings respectively.

$$Z_H = \frac{Z_{h1} + Z_{mh} - Z_{ml}}{2} \quad (8)$$

$$Z_L = \frac{Z_{ml} + Z_{h1} - Z_{mh}}{2} \quad (9)$$

$$Z_M = \frac{Z_{mh} + Z_{ml} - Z_{h1}}{2} \quad (10)$$

3. Use a wye configuration for the three impedance values, ZH, ZL, and ZM as shown in Fig. A-6. Use bus numbers, such as 1, 2 and 3 for each end of the wye and use N for the common middle point.

4. Convert the wye configuration to an equivalent delta by any standard method (Ref 6:155-156 as follows:

$$Z_{12} = \frac{(Z_H)(Z_M) + (Z_M)(Z_L) + (Z_H)(Z_L)}{Z_L} \quad (11)$$

$$Z_{13} = \frac{(Z_H)(Z_M) + (Z_M)(Z_L) + (Z_H)(Z_L)}{Z_M} \quad (12)$$

$$Z_{23} = \frac{(Z_H)(Z_M) + (Z_M)(Z_L) + (Z_H)(Z_L)}{Z_H} \quad (13)$$

5. Using the winding characteristics asso-

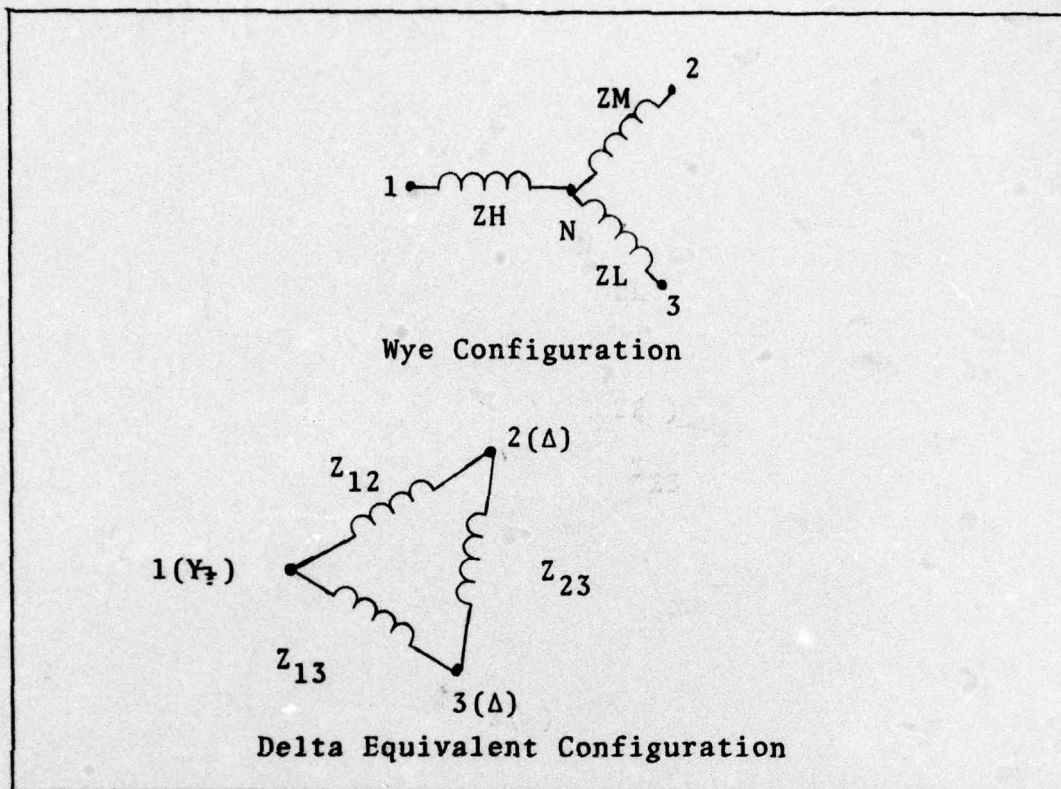


Fig. A-6 Delta Equivalent Three-Winding Transformer

ciated with each bus number, such as Hi(1)(wye) or Med(2)(delta), assign appropriate winding characteristics to each bus as shown in Fig. A-6. Winding characteristics should be chosen from the given information for the transformer.

6. Code the input data for the program using the equivalent delta configuration as three two-winding transformers with connection codes as appropriate. Use the highest voltage rating of the three-winding transformer as the high side voltage on the three two-winding transformers.

7. Unless other information is given, let the

LINEZ or LINDATA routine calculate the zero sequence impedance values.

Output

Output from both the LINEZ and LINDATA routines is similar. The first item is the program title and printout designation, Tape 1 or Tape 2. The second item is the system title chosen by the user. The third item is the program control constants as chosen on the Input-Output control card. A -0 indicates no entry was made and is the same as 0. The fourth item is the program parameter constants as coded on the System Parameter control card. These four items appear on both Tape 1 and Tape 2 printouts.

Next, on the Tape 1 printout only, appears the LINEZ or LINDATA header format stating the assembled line element data is in per-unit or ohms. The entries in this listing are in the same order as the line data input cards. Following the assembled line input data is the sorted line data list. The list is sorted by ascending line numbers and lists the start and end bus, the G (Conductance) and B (Susceptance) values, plus the zero sequence impedance values. If either the Load Flow or the Short Circuit routine is utilized, other entries appear between the assembled line list and the sorted line list.

The output data from the LINEZ and LINDATA routines is controlled by the OUT variable on the Input-Output control card. A review of Table A-III indicates output content is on Tape 1 and Tape 2 when a value of OUT other than 0 is selected.

4. Load Flow Analysis

Introduction

The Load Flow function of the program is used to calculate power flow throughout the system. Inputs required, in addition to the line element data, are the loads and generating sources in the system. Output is the power distribution throughout the system which shows the load requirements at each bus, line flows, and the amount of power imbalance that is accounted for on the slack bus.

Inputs

The first card in the Load Flow function is the Load Flow control card, LDFLOW, which is used to select the tolerance desired for the comparison between the calculated and specified power requirements on each bus. The tolerances, expressed in per-unit, are labeled PTOL and QTOL, for real and reactive power. Accuracies to about 1 KVA are possible. The tolerances should be consistent with the magnitude of the loads on the system. On the same card, the number of iterations that will be permitted to reach the selected tolerance is also specified as ITR1 and ITR2. ITR1 is the iteration counter for PTOL and ITR2 the counter for QTOL. One other item, NLC, is also specified on the LDFLOW control card. This variable is used to select the load change routine. Any number, other than zero, will select the number of times the basic program is to be modified by load changes within the system and a new load flow result calculated.

The format and variable description for the Load Flow control card are found in Table A-IX and Fig. A-7.

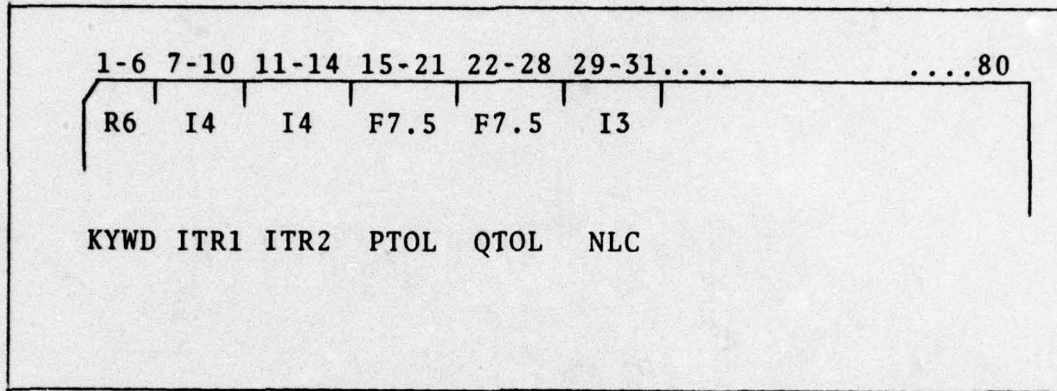


Fig. A-7. Load Flow Control Card

Table A-IX

Load Flow Control Card Variables

Variable (Card Column)	Description/Use
KYWD (1-6)	Key word. Always use LDFLOW.
ITR1 (7-10)	Maximum number of iterations to reach specified tolerance in PTOL. Choose number between 1 and 9999.
ITR2 (11-14)	Maximum number of iterations to reach specified tolerance in QTOL. Choose number between 1 and 9999.
PTOL (15-21)	Tolerance, difference between P specified and P calculated in per-unit. Usually between .1 and .0001.
QTOL (22-28)	Tolerance, difference between Q specified and Q calculated in per-unit. Usually between .1 and .0001.
NLC (29-31)	Number of times load buses will be changed and load flows re-calculated. Choose number between 0 and 999.

Following the Load Flow control card, LDFLOW, are the bus data cards. One bus data card is required for each bus. If the bus does not have a load or a source, the P and Q values may be left blank, but the bus number and type are required. If more than one load is located at a bus, the loads must be combined as only one bus card is permitted for each bus. Power flowing into a bus, such as from a generator, is considered positive. Power flowing out of a bus, such as to a load, is listed as negative values. The format and definition of variables used for the Load Flow Bus Data cards are shown in Fig. A-8 and Table A-X.

Three types of buses are used as shown in Table A-X. The most numerous are the load buses, or Type 1 buses. Type 2 buses, voltage controlled buses, are normally for sources where the voltage should be held constant. The program assumes that whatever reactive power required to do this will be available as long as it does not exceed the QMIN or QMAX limits selected. If the limits are exceeded, the voltage will be adjusted and a message printed with the results noting the voltage has not been held constant. The slack bus, Type 3, is to absorb all overages and shortages throughout the system. There must be one, and only one, slack bus for each system. The voltage must be specified for the slack bus but not the P and Q values.

The load change feature changes any number of loads in the simulated system and then calculates a revised load flow system summary. The same PTOL, QTOL, ITR1, and ITR2 will be

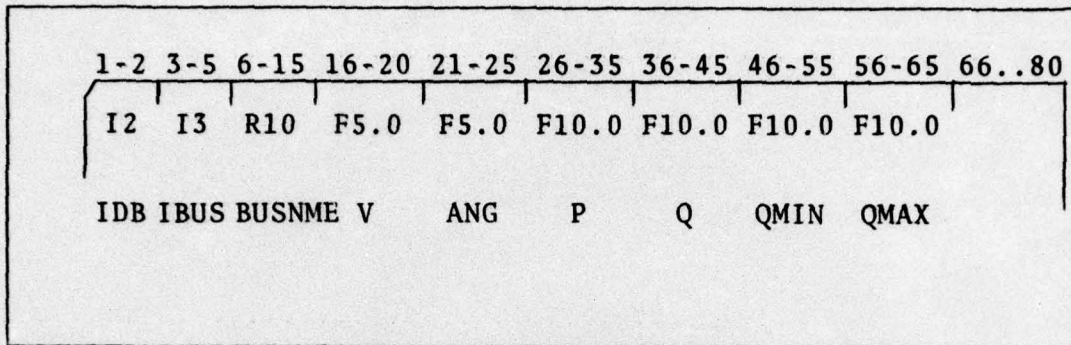


Fig. A-8. Load Flow Bus Card Format

Table A-X

Load Flow Bus Card Variables

Variable (Card Column)	Description/Use
IDB	Bus type. Choose 1,2,or 3.
(1-2)	1 Load bus. Specify P and Q. 2 Voltage Control Bus. Specify P and V. 3 Slack Bus. Specify V and ANG. (Only one type 3 bus allowed and required)
IBUS (3-5)	Bus number. Use consecutive numbers between 1 and 250, same as used for lines.
BUSNME (6-15)	Name of bus, up to 10 numbers and/or letters. May leave blank if desired.
V (16-20)	Bus voltage magnitude, in per-unit. No entry implies 1.0.
ANG (21-25)	Bus voltage angle, in degrees. No entry implies 0 degrees.
P (26-35)	Bus real power specified in KVA. Injected power is positive, loads are negative.
Q (36-45)	Bus reactive power specified in KVAR. Injected is positive, loads are negative.
QMIN (46-55)	Minimum reactive power limit. Specified for type 2 buses in KVAR. Other types, 0.
QMAX (56-65)	Maximum reactive power limit. Specified for type 2 buses in KVAR. Other types, 0 or leave blank.

used. The line data used is also the same as for the initial calculation. To vary the loads, select the value of NLC on the LDFLOW control card to indicate how many times the system will be modified and re-calculated. Next, following the last bus data card, add the load change control card, BUSCHG, with a value for NC representing the number of buses to be modified with this change. Following the load change control card, add the bus cards as changed. The bus cards must only be Type 1 and use the same bus number as the bus to be replaced. Fig. A-9 illustrates the load change control card, BUSCHG. Bus data cards should be in format as shown in Fig. A-8. If more than one change is selected, NLC greater than 1, additional BUSCHG control cards and bus data cards are added as required.

If several load changes are used or the system is very large, an increase in the computation time, TXXX, as allowed by the FORTRAN Control card may be required.

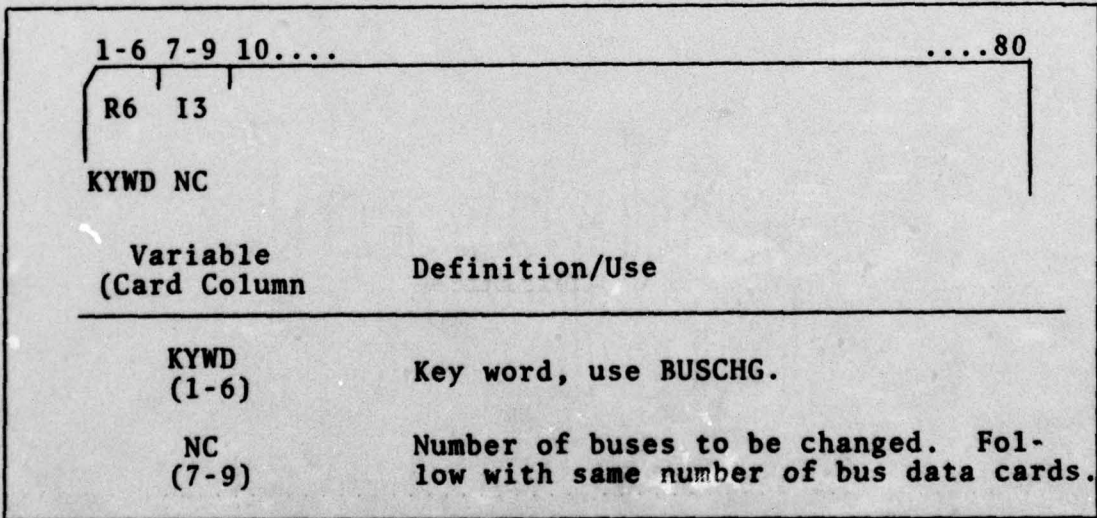


Fig. A-9. Load Change Control Card

Limitations

As noted previously, only one bus card is allowed per bus. Therefore, if both loads and sources are present at the same bus, only the net difference can be entered on the bus card.

The program uses an iterative routine to balance the system between calculated and specified power at each bus. If the number of buses is large, a larger time on the first FORTRAN control card, TXXX, may be required for the desired tolerance.

Convergence to any selected tolerance is usually obtainable unless the X/R ratio is .4 or less. When the X/R ratio approaches .4, convergence becomes more difficult and sometimes impossible regardless of the tolerance selected. When the specified tolerance is not reached with the number of iterations allowed, the output will not show line flows, but will show the DLP and DLQ values. The DLP and DLQ values are the difference between the specified and calculated power at each bus when the iteration limit was reached. Load changes will not be processed when convergence to the selected tolerance has not been obtained.

There is no way of assigning a single phase load to a particular phase in a three-phase system. All loads are considered as balanced between phases.

Output

The output from the Load Flow routine is varied by the

OUT value selected on the Input-Output control card, PGMCON. If an OUT value of 0 is selected, all tables are printed. On Tape 1 printout appears an input bus list, showing the input load values for each bus and the type of bus. On Tape 2 printout appears the output bus data showing the calculated loads for each bus, the number of iterations required for convergence, the line flowpower between all buses, and the slack bus power. If the load change routine is used, on the Tape 2 printout appears the number of the load change, the new bus data being added, and the re-calculated load flow result based on the modified system. All values are in per-unit notation and should be multiplied by the appropriate base for actual KW and KVA values.

5. Short Circuit Analysis

Introduction

The third function of the PDSAP program is the Short Circuit Analysis routine. The short circuit routine selectively simulates faults of various types at each bus and then examines the other buses and lines in the system to show fault currents available at these locations. The types of faults that can be simulated are three-phase, phase-ground, phase-phase, and phase-phase-ground.

Input

The first card in the Short Circuit routine is the Short Circuit control card, SHTCKT. On this card are entered 2 values, ISYS and SCOP. ISYS determines the network configuration for the short circuit study. The short circuit study can be run on any system up to 500 lines and 250 buses. However, simulated faults can only be calculated on 50 bus segments or less. ISYS indicates the number of bus groups to be analyzed. Each group can have from 1 to 50 buses and one bus may appear in more than one group. It is important to note, that although only 50 buses are shown, the entire system is used in the fault simulation. Changing the bus numbers in the subsystem does not in any way alter the system configuration. If changes in line connections are desired, the line element cards must be changed and the program re-run. The SCOP variable is used to select the type of fault conditions to be used. A blank implies a value

of 0. For most occasions, values of 0 or 2 should be used for SCOP. Table A-XI and Fig. A-10 illustrate the format and variable codes for the Short Circuit Control card, SHTCKT.

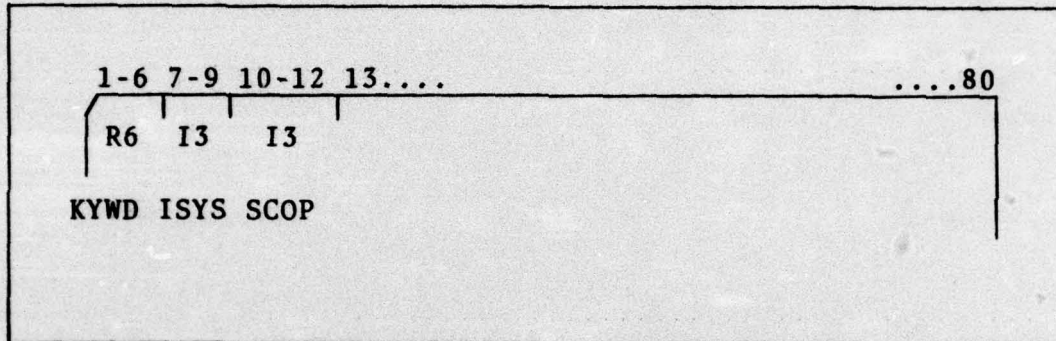


Fig. A-10. Short Circuit Control Card

Table A-XI

Short Circuit Control Card Variables

Variable (Card Column)	Definition/Use
KYWD (1-6)	Key word, use SHTCKT.
ISYS (7-9)	Number of subsystems to be used. If 0 or blank, all buses will be faulted in groups of 50.
SCOP (10-12)	Type of short circuit analysis. Choose 0,1,2, or 3. <ul style="list-style-type: none"> 0=All fault types used (3 Ph-Gnd, Ph-Gnd, Ph-Ph, Ph-Ph-Gnd). Bus Voltages are set to 1.0 per-unit. 1=Same as 0 except voltages are result of Load Flow routine. 2=Only Ph-Gnd and 3 Ph-Gnd faults used with bus voltages at 1.0 per unit. 3=Same as 2, except bus voltages are result of Load Flow routine.

The second card read by the Short Circuit routine is the Current Source Control card, CURSOR. This card specifies the number of buses that will contribute short circuit current to the system. Current sources will normally include the system source and large loads such as motors. It should include anything that would contribute current into the system in the event of a fault. If no current sources are present, the value on the CURSOR card may be 0 or blank. If there are no current sources, there must be at least one path to ground, or reference, through a transformer or other line element in the system being simulated. The last line element card is not considered part of the system although the SB and EB are 0. Fig. A-11 and Table A-XII show the format for current source control card.

Following the Current Source Control card are the Current Source Data cards. The variables on the Current Source Data cards are shown in Fig. A-12 and defined in Table A-XIII. The fault current, both FA3 and FA1, should normally have a large negative imaginary part based on the assumption of a positive reactance value for the source being simulated. If the data available for the system does not list short circuit current for the source, it may be calculated by knowing two of the following: KVA, voltage, or impedance of the source. Table A-XIV lists equations to use to find the short circuit current. The fault impedance, ZF, is any additional impedance, other than source impedance, prior to entering the system at the designated bus. Likewise, the neutral impe-

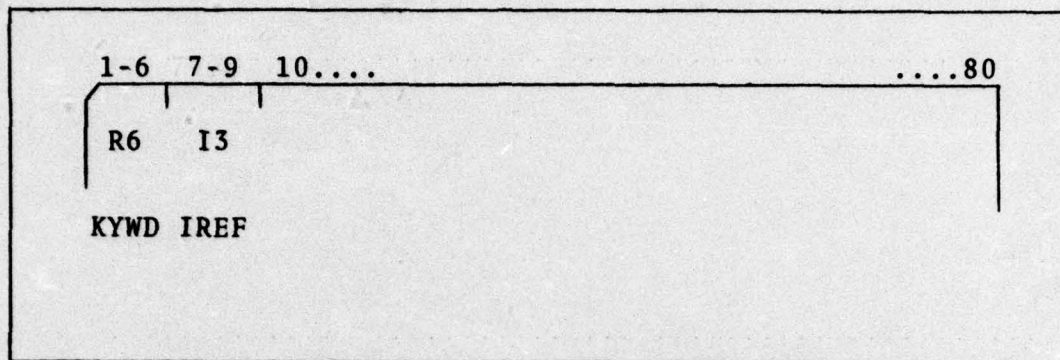


Fig. A-11. Current Source Control Card

Table A-XII
Current Source Control Card Variables

Variable (Card Column)	Definition/Use
KYWD (1-6)	Key word, use CURSOR.
IREF (7-9)	Number of current source data cards. If none, leave blank; but system then must have at least one path to ground through a transformer.

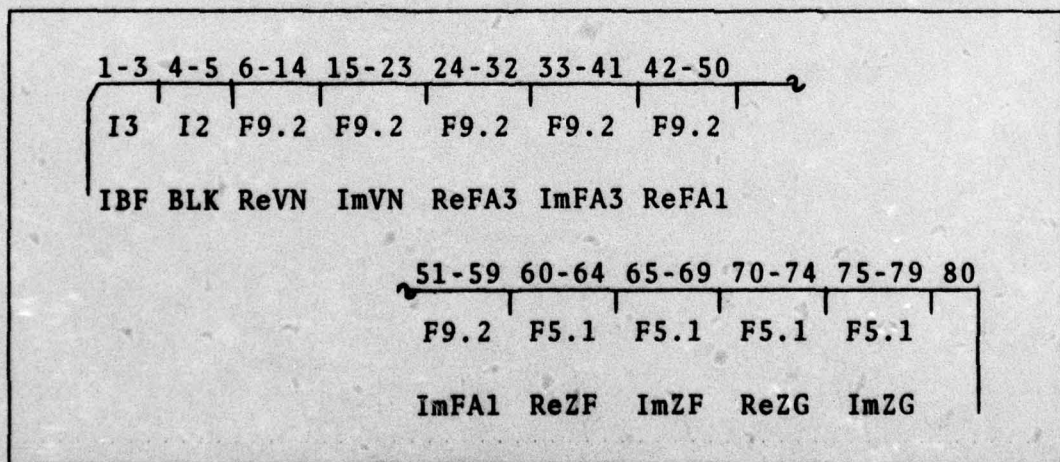


Fig. A-12. Current Source Data Card

Table A-XIII
Current Source Data Card Variables

Variable (Card Column)	Definition/Use
IBF (1-3)	Bus number of current source. Same as used for line elements.
BLK (4-5)	Not used. Leave blank.
VN (6-14) (15-23)	Voltage (KV) line-to-neutral as to a complex number. Usually all real.
FA3 (24-32) (33-41)	Short circuit input current (amps) as complex number. Usually all negative imaginary.
FA1 (42-50) (51-59)	Short circuit input current (amps) for Ph-Gnd fault. Usually all negative imaginary.
ZF (60-64) (65-69)	Fault impedance (ohms) as a complex number. Should be in addition to values used to calculate FA3 or FA1. Usually none.
ZG (70-74) (75-79)	Neutral impedance (ohms) as complex number. If any, should be in addition to values used to calculate FA3 or FA1.

Table A-IV
Current Source Equations

Three Phase	Single Phase or Per-Unit
$I = KVA/V_{LL}\sqrt{3}$	$I = KVA/V$
$I = V_{LL}/2\sqrt{3}$	$I = V/Z$
$I^2 = KVA/3Z$	$I^2 = KVA/Z$

Voltage values in KV or per-unit. Impedance values in ohms or per-unit.

dance, Z_G , is any impedance, other than zero, between the source and the reference or ground.

Following the current Source Data cards is the Mutual Coupling Control card, NOMUTL. This card indicates the number of mutually coupled lines in the system being simulated. The routine has the capacity to simulate up to 25 mutually coupled lines. If there are no mutually coupled lines, the value on the card should be 0 or blank. Fig. A-13 illustrates the format for the Mutual Coupling Control card. When mutual coupling is present, following the Mutual Coupling Control card will be the Mutual Coupling Data cards. These cards should have the format as illustrated in Fig. A-13 and defined in Table A-XV.

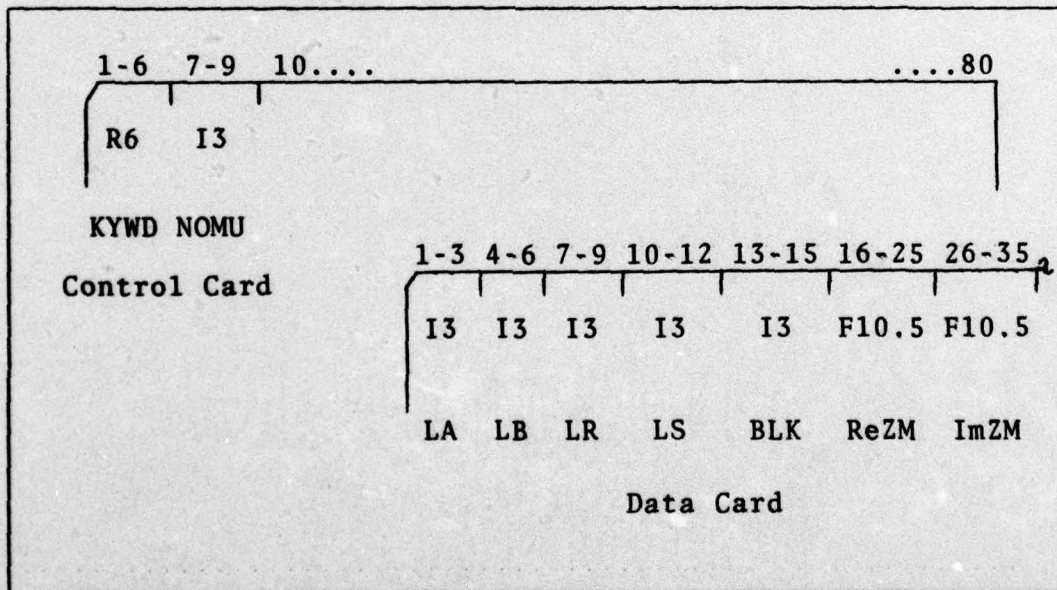


Fig. A-13. Mutual Coupling Control and Data Cards

Table A-XV
Mutual Coupling Variables

Variable (Card Column)	Definition/Use
KYWD (1-6)	Key word, use NOMUTL.
NOMU (7-9)	Number of mutually coupled lines. Same number of data cards should follow. If none, leave blank or enter 0.

LA (1-3)	Start bus of mutually coupled line. Use same numbers as in line element data.
LB (4-6)	End bus of mutually coupled line.
LR (7-9)	Start bus of line to which coupled.
LS (10-12)	End bus of line to which coupled.
BLK (13-135)	Not used, leave blank.
ZM (16-25) (26-35)	Mutual impedance between two lines in per-unit, complex.

After the mutual coupling cards are the fault impedance cards. The first fault impedance card is a Fault Impedance Control card, NOFALT. This card specifies the voltage, phase and fault impedance for a series of buses. Fault impedance, ZF, is the value of impedance between the fault and ground. Usually this is zero or some real value in ohms. If the fault impedance is assumed to be zero for the entire system, only one Fault Impedance Control card is required.

If there is more than one fault impedance value or different phases and voltages for the same value, additional Fault Impedance Control cards are needed. Fig. A-14 illustrates the format for the Fault Impedance Control card, and Table A-XVI lists the definition of the variables. With each Fault Impedance Control card that specifies an impedance other than zero, a bus list must immediately follow indicating to what buses the fault impedance applies. Fig. A-14 illustrates the bus list format and Table A-XVII the variables. The last card in this group must be a Fault Impedance Control card with NFT of 0.

The last group of cards in the Short Circuit routine are the subsystem cards. The Subsystem Control Card, NOBSYS, specifies the number of buses in the subsystem as shown in Fig. A-16 with the variables defined in Table A-XVIII. Up to 50 buses can be in any one subsystem. The bus numbers are listed on a bus list following the NOBSYS card. Fig. A-15 illustrates the format for the bus list. If there is more than one subsystem, ISYS greater than 1, then additional NOBSYS cards and bus lists are used. It is important to insure the number of NOBSYS cards equals the ISYS value on the Short Circuit Control card.

The use of the subsystem bus list should be considered carefully. When working with large systems, over 50 buses, and line currents in particular lines are of interest, the subsystem bus list should be used. When using the automatic feature (ISYS=0), there is no assurance that buses for lines

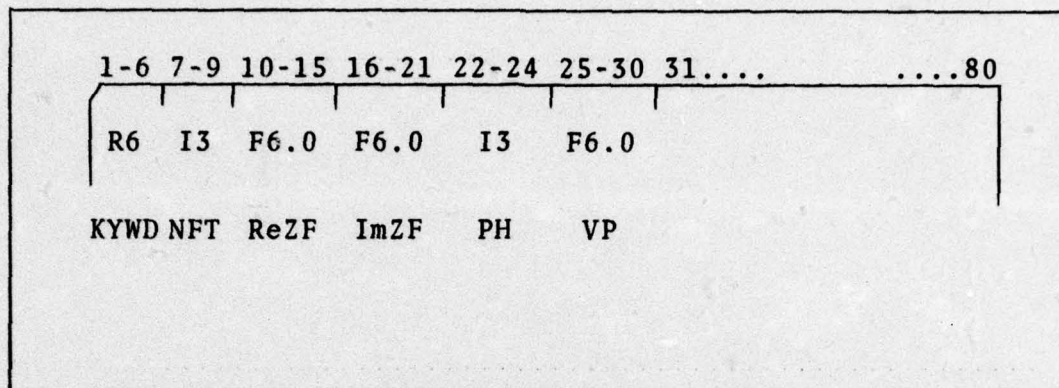


Fig. A-14. Fault Impedance Control Card

Table A-XVI

Fault Impedance Control Card Variables

Variables (Card Column)	Definition/Use
KYWD (1-6)	Key word, use NOFALT.
NFT (7-9)	Number of buses to which fault impedance applies. If none, leave blank. Last NOFALT card must have NFT=0.
ZF (10-15) (16-21)	Fault impedance in ohms, complex. If any, usually all real.
PH (22-24)	Phase. Choose 1 or 3. All buses in group must have same phase.
VP (25-30)	Voltage (KV). Line-to-line if 3-Ph. Line-to-neutral if single phase. All buses in group must have the same voltage.

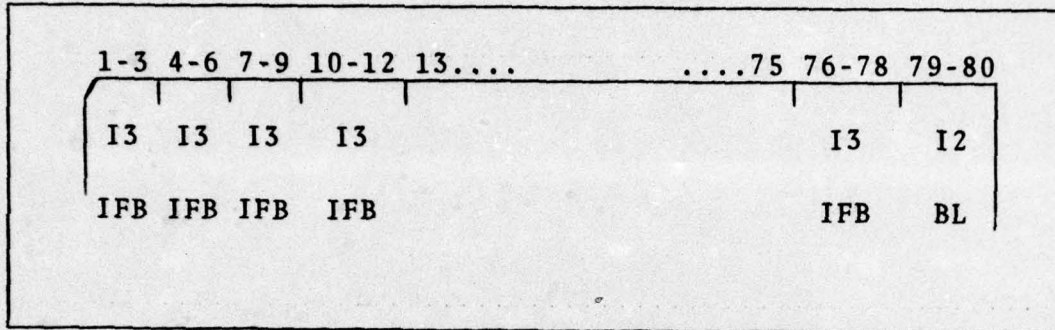


Fig. A-15. Bus List Format for Fault Impedance and Subsystem Buses.

Table A-XVII

Bus List Card Variables

Variables (Card Column)	Definition
IFB (1-3) (4-6) etc.	Bus numbers to which fault impedance or subsystem applies. Numbers should be in increasing order. Skip numbers to which does not apply.
BL (79-80)	Leave blank. If more than 26 buses in list, continue on a following card.

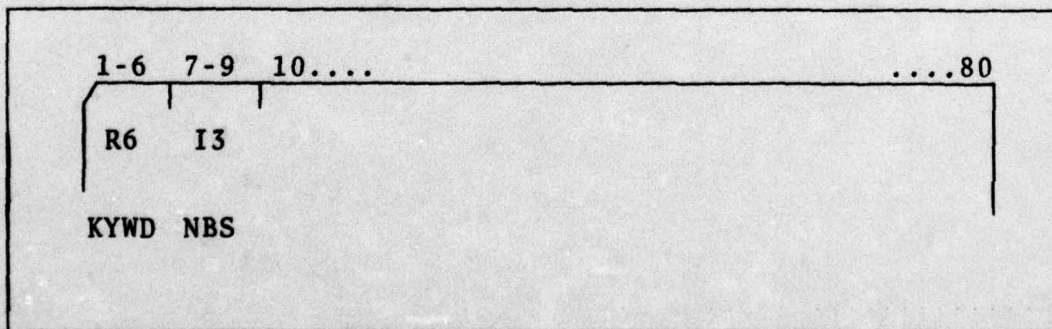


Fig. A-16. Subsystem Control Card

Table A-XVIII
Subsystem Control Card Variables

Variables (Card Column)	Definition/Use
KYWD (1-6)	Key word, use NOBSYS.
NBS (7-9)	Number of buses in subsystem list. Choose number between 1 and 50. List bus numbers on following card.

of interest will be in the same group if the system is large. For the line current to be printed, both the start (SB) and end bus (EB) of the line must be in the same group.

Limitations

There is no way to control to what phase of a three phase system a single phase branch circuit will be attached. A single phase circuit is always assumed to be attached to Ph-A. For better accuracy in short circuit studies, it may be beneficial to simulate the three-phase and single phase parts of the system separately.

Output

The output of the Short Circuit routine is controlled by the OUT variable on the Input-Output Control card and by the SCOP variable on the Short Circuit Control card. When both OUT and SCOP are 0, maximum information will be printed.

The Tape 1 printout shows the Short Circuit Input Data.

This includes the current sources, fault impedance and mutual coupling information.

Tape 2 printout shows the Fault Summary for each bus in the selected subsystem. The complete summary includes the three-phase, phase-ground, phase-phase, and phase-phase-ground fault currents and X/R ratios. An R for the X/R ratio indicates that no ratio could be formed as one value was 0. If any value appears as ****, this indicates the calculated value has exceeded the allowed space for the number. For the current magnitudes and X/R ratios, 999 per-unit is the limit.

Also as part of the Fault Summary are printed the phase voltages for the various types of faults. For faults between phases, the routine assumes the fault is between phases B and C. For the three-phase and phase-to-ground faults, the bus voltages for all buses in the subsystem will be listed. Likewise, line currents for three-phase and phase-to-ground will be listed for any lines connecting buses in the subsystem. If the fault current on a specific line is desired, both the start and end buses must be included in the subsystem list.

The magnitude of the fault current will be greatly influenced by the location and size of current sources, transformer connections, and impedance values for conductors. A small error in any one of the foregoing, could make a significant difference in the fault currents.

When the faulted bus is single phase, even if part of a

three-phase system, only the phase-to-ground information will be printed. Again, note that all values are in per-unit and should be converted to amps, ohms or volts by using the appropriate base values.

6. Trouble Shooting

The PDSAP program has numerous routines designed to detect errors with input data and errors that occur with calculations. When there is an error, usually it will appear as either a program error or a FORTRAN error.

For program errors, a statement will appear with the output data stating that the program has terminated for a particular reason. Usually there is enough information given to locate the input data card causing the problem. Program errors can usually be grouped into those caused by data cards and those caused by program control cards. When trouble develops, first check the deck structure as shown in Fig. A-1 for proper sequence of control cards and data cards.

Errors on data cards are usually caused by data not in the proper columns or improper coding. Both are very common errors and can be held to a minimum by careful checking of the data after the cards have been punched. Sometimes values entered for impedance, wire size or other parameters are non-standard and may cause problems. When this type of error occurs, the best solution is to change the variable causing the problem to a standard size or value. Also, impedance values of several hundred ohms may be too large for some operations. This can be corrected by adding additional buses to lower the values.

Other program errors can result from insufficient data being supplied. In some cases, values will be assumed for sizes, impedance or other variables when the input is omitted.

However, in other cases no entry results in a zero value which will halt the program. The error messages should help to identify the variable causing the problem.

A third situation that may cause an error is improper format being used. This is the result of a decimal point being omitted or an error in placing the data in the proper columns. This type of error may or may not halt the program. It is advisable to at least once check the printout of the assembled input line data to verify that no mistakes have occurred even if the program has run successfully.

When the error is a FORTRAN error, no output is produced except the day file which contains the FORTRAN control card information and the FORTRAN error. FORTRAN errors are not as easy to detect or correct as program errors.

Two common mistakes are to omit necessary data cards or to mis-count data cards. In several instances the number of data cards must match a variable designated on a preceding control card. This is true with the Load Change Control cards, BUSCHG, and the Current Source Control cards, CURSOR. In the Load Flow routine, the bus data cards must be the same number as the number of buses used in the line element program. When the number of data cards is less than required, or data cards are missing, the most frequent result is the FORTRAN error, "Illegal data in the field." When this type of error occurs, it means that the card with the data listed above the error statement is not compatible with the input format required. For this type of error, first check the

card for proper format and location. Perhaps a line element card got in the bus card group. Also check for missing control cards or missing data cards.

Another method of finding the cause of the FORTRAN error, "Illegal data in the field" is to look for the line numbers of the routines that were being used when the error occurred. This information will be just below the error statement. Fig. A-17 has an example of this type of error. By checking the line number with the same line number in a listing of the program, it is often easy to see why the error occurred.

One FORTRAN error that is almost impossible to locate is a mode error. A mode error usually occurs when division by zero is attempted in the program. For resolution of mode errors or other FORTRAN errors, one of the following methods may prove successful.

Try each function separately. Start by running input data only with either the LINEZ or LINDATA routines. Then add either the Load Flow or Short Circuit functions. This should determine which program function is causing the error. Carefully review the input data that is used by the program and compare it with the formats in the preceding chapters.

A second method that may prove useful if the system is quite large is to divide the system into smaller parts. If the program will work with 10 buses, add more until the problem is encountered. Look for abnormal sizes or mistakes in

```

S
Q
1      $$$$$$  $$  $$  $$$$$$  $$$$$$$  $$$$$$$  $$$$$$
      $$  $$  $$  $$  $$  $$  $$  $$  $$  $$  $$  $$
      $$  $$  $$  $$  $$  $$  $$  $$  $$  $$  $$  $$
      $$  $$  $$  $$$$$$  $$$$$$$  $$$$$$$  $$$$$$
      $$  $$  $$  $$  $$  $$  $$  $$  $$  $$  $$  $$
      $$  $$  $$  $$  $$  $$  $$  $$  $$  $$  $$  $$
      $$  $$  $$  $$  $$  $$  $$  $$  $$  $$  $$  $$
      $$  $$  $$  $$  $$  $$  $$  $$  $$  $$  $$  $$
      $$  $$$$$$  $$$$$$  $$$$$$$  $$$$$$$  $$$$$$
0 RECORD NO. 17      SHTCKT
*..... 123456789012345678901234567890123456789
0* ERROR DATA INPUT * ILLEGAL DATA IN FIELD *
ERROR NUMBER 78 DETECTED BY INCOM= AT ADDRESS 000202
CALLED FROM KPAKER= AT ADDRESS 000345
CALLED FROM INPC= AT ADDRESS 000075
CALLED FROM LINEZ AT LINE 58
CALLED FROM INPUT AT LINE 24
CALLED FROM EXEC AT LINE 93

1 CSB NOS/RE L414E CYBR CMR3 R.0 22.33
22.42.13.TU63554 FROM /B3
22.42.13.IP 00000320 WORDS - FILE INPUT , DC 00
22.42.13.TU6,T30,I020,CM165000,STCSB.7750159,UNDE
22.42.13.RW000,4014
22.42.17.ATTACH,LDFLOW,LDF.OW5,CY=5,IO=T760162.
22.42.17.LDF.OW.
22.44.40.FTN - FATAL ERROR 78
22.44.40. .155 CP SECONDS EXECUTION TIME
22.44.40.OP 00000128 WORDS - FILE OUTPUT , DC 40
22.44.40.MS 82432 WORDS ( 82432 MAX USED)
22.44.40.SCM 164000 WORDS MAXIMUM
22.44.40.CPA 4.317 SEC. 1.553 ADJ.
22.44.40.IO 6.909 SEC. 3.743 ADJ.
22.44.40.CH 534.982 KMS. 7.627 ADJ.
22.44.40.CRUS 12.924
22.44.40.COST $ .77
22.44.40.PP 12.486 SEC. DATE 11/22/76
22.44.40.EJ END OF JOB, 99 T750159.
0
***** TU6BBSH //// END OF LIST ////
0

```

Fig. A-17. FORTRAN Error Example.

coding that may have occurred.

Sometimes the use of the tape dump routine may help find a mode error. By use of additional FORTRAN control cards, the content of the output tapes, Tape 1 and Tape 2, will be printed even if the mode error occurs. These cards are placed just before the first EOR card and should be after the DISPOSE, TAPE2 card. The cards are as follows:

1. EXIT.
2. DISPOSE,TAPE1,PR=IBB.
3. DISPOSE,TAPE2,PR=IBB.

The trouble is usually beyond the last information printed as the program execution is ahead of the printer.

A fourth method to find a mode error is the mode override method. By placing a FORTRAN control card, MODE;. immediately following the ATTACH,PDSAP,ID=Txxxxxx,CY=XX card, the program will continue even when a mode error is encountered. The results from the program should not be trusted under this condition. The reasons for the mode error appears on a separate annotation in the dayfile listing.

In an effort to help the user locate some of the more frequent errors, Table A-XIX is a list of some of the error messages and their cause. Other error messages may appear but there should be enough information given to locate the problem. It would be most helpful to have a listing of the program to trace some of the errors when the messages do not specify the exact card causing the problem.

In summary, if there is no success in resolving the

Table A-XIX
Error Messages and Causes

Error Message	Cause
Control card not in proper format or location.	Mis-spelling of keyword. Control cards omitted or out of order.
Illegal Data in the Field (FORTRAN ERROR).	(1) SB=0,EB=0 card not last card in line element list. (2) Number of bus data cards not equal to highest bus number in line element list. (3) Number of bus data cards not equal value on preceding control card.
Use of CON code () not allowed.	Improper CON code, check table of values again.
Too many slack buses.	Two or more Type 3 buses in bus data cards for Load Flow routine. Only one allowed.
Bus () should be connected to line bus of same number.	Bus numbers on bus data cards for Load Flow must match line bus numbers.
Error with ID or C value, line card ().	Improper ID or C value on line element data card. Check proper columns and values.
Error with input line card ().	Error with wire size or number of ground wires. Check if standard size and in proper columns.
Positive and zero sequence impedance values are zero.	No impedance specified on line element data card for LINEZ routine.
IDB can't be greater than 1 in Load Change routine.	Bus data card in Load Change routine is in error. Check format. No Type 2 or 3 buses allowed.

Table A-XIX (Cont'd)

Error Message	Cause
Error in transformer connection code.	Wrong connection code for transformer or format error. Check with table of codes.
Fault source impedance routine has detected a zero input for 3-ph fault	No 3-ph fault current specified for current source (FA3-0). FA3 can not be zero.
Bus 0 cannot be in sub-system list.	Bus number specified on NOBSYS not equal to number of buses on card following.

problem by using the foregoing information, the advice of a qualified programmer would be helpful. Although extensive use of the program has been made by simulating various actual and theoretical systems, unique situations may be encountered that cannot be simulated with the program without some modification.

AD-A035 292

AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 10/2
A USER-ORIENTED POWER DISTRIBUTION SYSTEM ANALYSIS PROGRAM.(U)

UNCLASSIFIED

6E/EE/76-43

NL

2 OF 3
AD
A035292



A microfiche card containing a grid of 120 frames (10 rows by 12 columns). The frames contain various technical diagrams, tables, and text. The first frame in the fifth row from the top is labeled "BLANK PAGE".

7. Example Problems

Example problems contained in this chapter were selected to illustrate the capability and validity of the PDSAP program. To validate the program results, comparison was made with published results for the same problems.

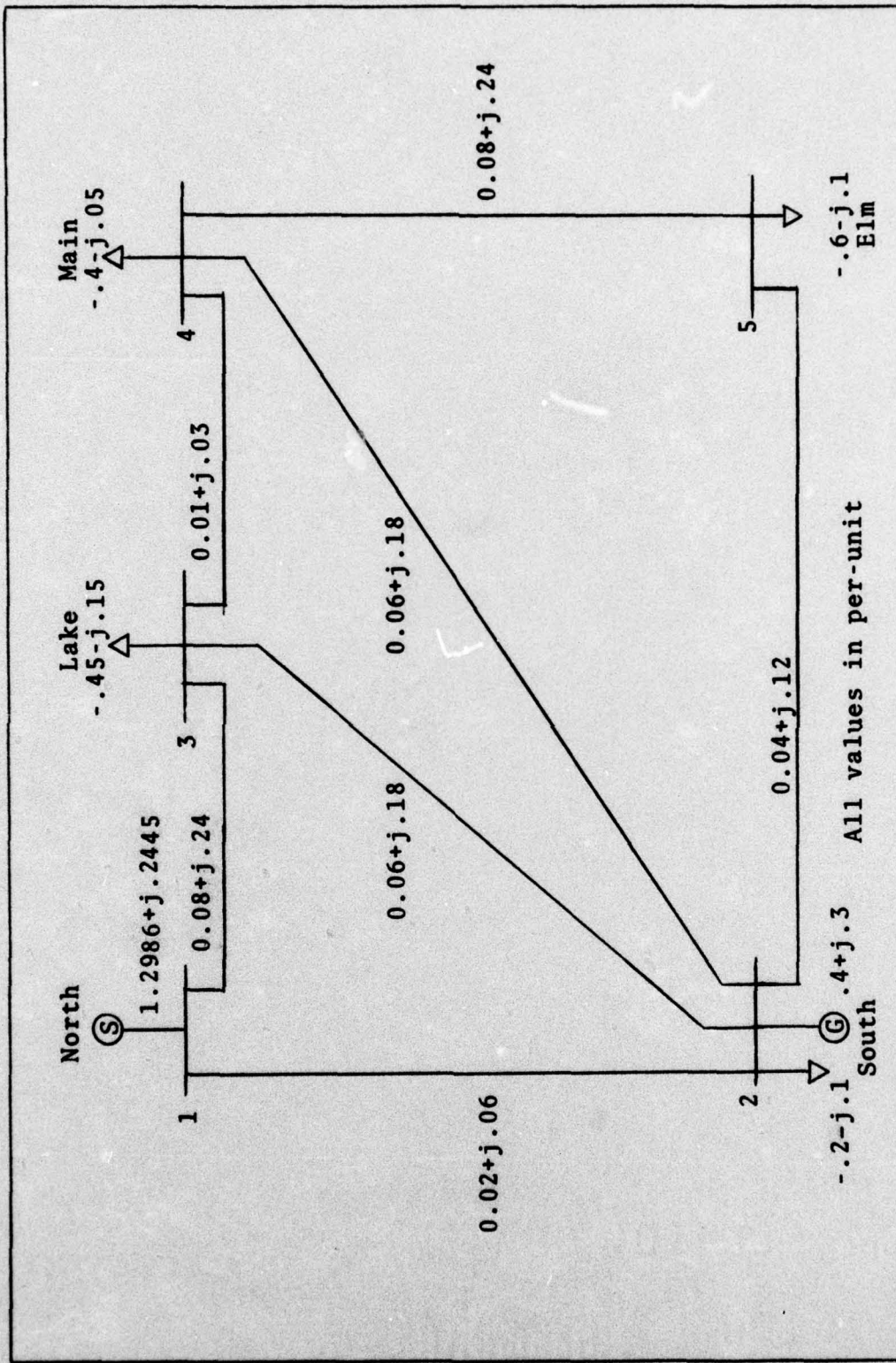
Problems were chosen for a specific purpose as noted by the introductory paragraph for each problem. Each example has a one-line diagram from which the computer input data was prepared. An illustration for each problem shows the arrangement and format of the data cards for input to the program. In order to conserve space, the FORTRAN cards are not shown. The order and format of the FORTRAN cards is found in Chapter 2.

The computer printout for each problem follows the illustration of the data cards. By choice of the various OUT codes, information not required can be omitted from the printouts.

The user is encouraged to take the one-line diagrams for each problem and duplicate the results to gain familiarity with the program. The introduction to each problem indicates which routines were used and other necessary information.

Example Problem 1.

This example illustrates the use of the LINEZ input routine and the Load Flow routine. It was adapted for the PDSAP program to validate the accuracy of the Load Flow routine (Ref 5:284). The system base is 100 MVA. The negative values indicate power out of the system and the positive values indicate power into the system. All values are in per-unit. Fig. A-18 is the one-line diagram of the system. Fig. A-19 is the data card format and order.



(Ref 5:284)

Fig. A-18. Example 1 System Diagram

1	4	MAIN	1.0	0.	-40000	-5000	
1	3	LAKE	1.0	0.	-45000	-15000	
BUSCHG 2							
1	4	RED	1.0	0.	-30000	-10000	
1	3	BLUE	1.0	0.	-45000	-15000	
BUSCHG 2							
1	5	ELM	1.0	0.	-60000	-10000	
1	4	MAIN	1.0	0.	-40000	-5000	
1	3	LAKE	1.0	0.	-45000	-15000	
1	2	SOUTH	1.0	0.	20000	20000	
3	1	NORTH	1.05	0.			
LDFLOW 25 25 .0001 .0001 2							
0	0						1
4	5	1.0 13	1.0	.08	.24		1
3	4	1.0 13	1.0	.01	.03		1
2	5	1.0 13	1.0	.04	.12		1
2	4	1.0 13	1.0	.06	.18		1
2	3	1.0 13	1.0	.06	.18		1
1	3	1.0 13	1.0	.08	.24		1
1	2	1.0 13	1.0	.02	.06		1
SYSPAR							
PGMCN 2 1 9							
EXAMPLE PROBLEM #1							
LOAD FLOW ANALYSIS(STAGG)							

FORTRAN Control Cards

Fig. A-19. Data Card Format, Example 1

```

*****
*
* POWER DISTRIBUTION SYSTEM ANALYSIS PROGRAM (PDSAP)
* TAPE 1 PRINTOUT
*
*****
    
```

```

*****
*
* EXAMPLE PROBLEM #1
* LOAD FLOW ANALYSIS (STAGG)
*
*****
    
```

**** PROGRAM CONTROL CONSTANTS ****

```

CON= 2
INP= 1
OUT= 0
CHG= -0
    
```

*** PROGRAM PARAMETER CONSTANTS ***

```

BASE KVA      FREQUENCY    TEMPERATURE    EARTH RESISTIVITY
100000. KVA   50. HZ         25. DEG. C     100. METER-OHM
    
```

```

*****
*
* LINEZ SUBROUTINE
* ASSEMBLED INPUT LINE DATA (PER-UNIT)
*
*****
    
```

CONDUCTOR NO.	1				
FROM - TO	1 2	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
		.0200	.0600	.0700	.2100
CONDUCTOR NO.	2				
FROM - TO	1 3	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
		.0900	.2400	.2800	.8400
CONDUCTOR NO.	3				
FROM - TO	2 3	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
		.0500	.1800	.2100	.6300
CONDUCTOR NO.	4				
FROM - TO	2 4	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
		.0500	.1800	.2100	.6300

CONDUCTOR NO.		5			
FROM - TO	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)	
2 3	.0490	.1200	.1400	.4200	
CONDUCTOR NO.		6			
FROM - TO	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)	
3 4	.0100	.0300	.0350	.1050	
CONDUCTOR NO.		7			
FROM - TO	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)	
4 5	.0930	.2400	.2900	.8400	

SUMMARIZED INPUT BUS DATA PERUNIT
LISTED BY ASCENDING BUS NUMBERS

NO.	TYPE	V(MAG)	V(ANG-DEG)	POWER		P(MW)	Q(MVAR)
				REAL	REACTIVE		
1	3	1.060	0.0000	0.00000	0.00300	0.0000	0.0000
2	1	1.000	0.0000	.20000	.20000	0.0000	0.0000
3	1	1.000	0.0000	-.45000	-.15000	0.0000	0.0000
4	1	1.000	0.0000	-.40000	-.05000	0.0000	0.0000
5	1	1.000	0.0000	-.60000	-.10000	0.0000	0.0000

REORDERED BUSLIST RETURNED BY SUBROUTINE ORDER

1 3 2 4 5

 *
 * SORTED LINE INPUT DATA *
 * LISTED BY ASCENDING BUS NUMBERS *
 *

S9	E9	G	B	RE(Z0)	IM(Z0)
1	2	.50000E+01	-.15000E+02	.70000E-01	.21000E+00
1	3	.12500E+01	-.37500E+01	.28000E+00	.84000E+00
2	1	.50000E+01	-.15000E+02	.70000E-01	.21000E+00
2	3	.16667E+01	-.50000E+01	.21000E+00	.63000E+00
2	4	.16667E+01	-.50000E+01	.21000E+00	.63000E+00
2	5	.25000E+01	-.75000E+01	.14000E+00	.42000E+00
3	1	.12500E+01	-.37500E+01	.28000E+00	.84000E+00
3	2	.16667E+01	-.50000E+01	.21000E+00	.63000E+00
3	4	.10000E+02	-.30000E+02	.35000E-01	.10500E+00
4	2	.16667E+01	-.50000E+01	.21000E+00	.63000E+00
4	3	.10000E+02	-.30000E+02	.35000E-01	.10500E+00
4	5	.12500E+01	-.37500E+01	.28000E+00	.84000E+00
5	2	.25000E+01	-.75000E+01	.14000E+00	.42000E+00
5	4	.12500E+01	-.37500E+01	.28000E+00	.84000E+00

```
*****  
* POWER DISTRIBUTION SYSTEM ANALYSIS PROGRAM (PDSAP) *  
* TAP 2 PRINTOUT *  
*****
```

```
*****  
* EXAMPLE PROBLEM #1 *  
* LOAD FLOW ANALYSIS(STAGG) *  
*****
```

**** PROGRAM CONTROL CONSTANTS ****

CON= 2
INP= 1
OUT= 0
CHG= -0

*** PROGRAM PARAMETER CONSTANTS ***

BASE KVA	FREQUENCY	TEMPERATURE	EARTH RESISTIVITY
100000. KVA	60. HZ	25. DEG. C	100. METER-OM

```
*****  
* RESULTS OF FAST DECOUPLED LOAD FLOW ANALYSIS *  
* ALL MAGNITUDE VALUES ARE PER-UNIT *  
* SYSTEM HAS 5 BUSES; 0 ARE TYPE 2. *  
* NUMBER OF TIMES LOAD BUSES WILL BE CHANGED(NLC)= 2. *  
* CONVERGENCE TOLERANCES: *  
* PTOL= .00010 *  
* QTOL= .00010 *  
*****
```

```

*****
*
*          SYSTEM SUMMARY
*
*  CONVERGENCE OBTAINED IN:
*    8 DELTA THETA AND
*    6 DELTA V ITERATIONS.
*
*****

```

```

***** SLACK BUS POWER *****
BUS NO.      REAL      REACTIVE      MAG.
-----
1            1.29822    .24445        1.32103

```

```

*****
*
*          CALCULATED LINE FLOWS
*  (THE LINE FLOWS ARE DEFINED POSITIVE WHEN FLOWING OUT FROM THE BUS)*
*
*****

```

LINE		POWER		LINE		POWER	
FROM	TO	REAL	REACTIVE	FROM	TO	REAL	REACTIVE
1	2	.88955	.13865	2	1	-.87512	-.09537
1	3	.40857	.10580	3	1	-.39598	-.05774
2	3	.24688	.09146	3	2	-.24311	-.07014
2	4	.27932	.08061	4	2	-.27450	-.05545
2	5	.54891	.13330	5	2	-.53703	-.09766
3	4	.18909	-.01213	4	3	-.18874	.01319
4	5	.06334	.00326	5	4	-.06302	-.00231

```

*****
*
*          OUTPUT BUS DATA
*
*****

```

NO.	TYPE	NAME	V(MAG)	V(ANG-DEG)	POWER	
					REAL	REACTIVE
1	3	NORTH	1.0600	0.0000	1.29822	.24445
2	1	SOUTH	1.0355	-2.6397	.20000	.20000
3	1	LAKE	1.0088	-4.8077	-.45000	-.15000
4	1	MAIN	1.0073	-5.1344	-.40000	-.05000
5	1	FLM	1.0016	-5.9830	-.50000	-.10000

```

*****
*                               *
*   LOAD FLOW BUS CHANGE       *
*   CHANGE NUMBER 1           *
*   NUMBER OF BUSES CHANGED 2 *
*                               *
*****
    
```

```

*****
*                               *
*   BUS CHANGE DATA          *
*                               *
*****
    
```

NO.	TYPE	NAME	V(MAG)	V(ANG-DEG)	POWER	
					REAL	REACTIVE
3	1	BLUE	1.000	0.000	-.45000	-.15000
4	1	RED	1.000	0.000	-.30000	-.10000

```

*****
*                               *
*   SYSTEM SUMMARY           *
*                               *
*   CONVERGENCE OBTAINED IN: *
*   14 DELTA THETA AND      *
*   14 DELTA V ITERATIONS.  *
*                               *
*****
    
```

BUS NO.	***** SLACK BUS POWER *****		MAG.
	REAL	REACTIVE	
1	1.19249	.27750	1.22437

```

*****
*                               *
*   CALCULATED LINE FLOWS   *
*   (THE LINE FLOWS ARE DEFINED POSITIVE WHEN FLOWING OUT FROM THE BUS)*
*                               *
*****
    
```

LINE		POWER		LINE		POWER	
FROM	TO	REAL	REACTIVE	FROM	TO	REAL	REACTIVE
1	2	.81844	.16079	2	1	-.80605	-.12364
1	3	.37406	.11682	3	1	-.35311	-.08402
2	3	.22465	.09057	3	2	-.22157	-.09072
2	4	.24783	.09387	4	2	-.24391	-.09210
2	5	.53340	.13920	5	2	-.52208	-.10525
3	4	.13458	.01474	4	3	-.13450	-.01419
4	5	.07440	-.00371	5	4	-.07792	.00517

```

*****
*                                     *
*                               OUTPUT BUS DATA                               *
*                                     *
*****
    
```

NO.	TYPE	NAME	V (MAG)	V (ANG-DEG)	POWER	
					REAL	REACTIVE
1	3	NORTH	1.0500	0.0000	1.13249	.27760
2	1	SOUTH	1.0364	-2.3742	.20000	.20000
3	1	BLUE	1.0082	-4.3150	-.45000	-.15000
4	1	PEP	1.0064	-4.5353	-.30000	-.10000
5	1	FLM	1.0012	-5.6227	-.50000	-.10000

```

*****
*                                     *
*                               LOAD FLOW BUS CHANGE                               *
*                               CHANGE NUMBER 2                               *
*                               NUMBER OF BUSES CHANGED 2                       *
*                                     *
*****
    
```

```

*****
*                                     *
*                               BUS CHANGE DATA                               *
*                                     *
*****
    
```

NO.	TYPE	NAME	V (MAG)	V (ANG-DEG)	POWER	
					REAL	REACTIVE
4	1	MAIN	1.000	0.000	-.40000	-.05000
3	1	LAKE	1.000	0.000	-.45000	-.15000

```

*****
*                                     *
*                               SYSTEM SUMMARY                               *
*                                     *
*                               CONVERGENCE OBTAINED IN:                       *
*                               19 DELTA THETA AND                             *
*                               18 DELTA V ITERATIONS.                         *
*                                     *
*****
    
```

```

***** SLACK BUS POWER *****
BUS NO.    REAL    REACTIVE    MAG.
    1      1.29815    .24453      1.32098
    
```

 *
 * CALCULATED LINE FLOWS *
 * (THE LINE FLOWS ARE DEFINED POSITIVE WHEN FLOWING OUT FROM THE BUS)*
 *

POWER				POWER			
LINE FROM	TO	REAL	REACTIVE	LINE FROM	TO	REAL	REACTIVE
1	2	.88949	.13871	2	1	-.87505	-.09544
1	3	.40866	.10592	3	1	-.39597	-.06775
2	3	.24688	.08146	3	2	-.24311	-.07014
2	4	.27972	.08062	4	2	-.27460	-.06645
2	5	.54930	.13336	5	2	-.53702	-.09771
3	4	.18909	-.01211	4	3	-.18374	.01317
4	5	.06333	.00329	5	4	-.06302	-.00234

 *
 * OUTPUT BUS DATA *
 *

NO.	TYPE	NAME	V (MAG)	V (ANG-DEG)	POWER	
					REAL	REACTIVE
1	3	NORTH	1.0500	0.0000	1.29815	.24453
2	1	SOUTH	1.0365	-2.5395	.20000	.20000
3	1	LAKE	1.0037	-4.8075	-.45000	-.15000
4	1	MAIN	1.0072	-5.1342	-.40000	-.05000
5	1	FL4	1.0015	-5.9325	-.50000	-.10000

Example Problem 2

This example illustrates the use of the LINDATA input routine and the Short Circuit routine. It is adapted from a text to show the validity of the PDSAP program in calculation of fault currents (Ref 3:11). Fig. A-20 is the one-line diagram and Fig. A-21 illustrates the format and order of the data cards.

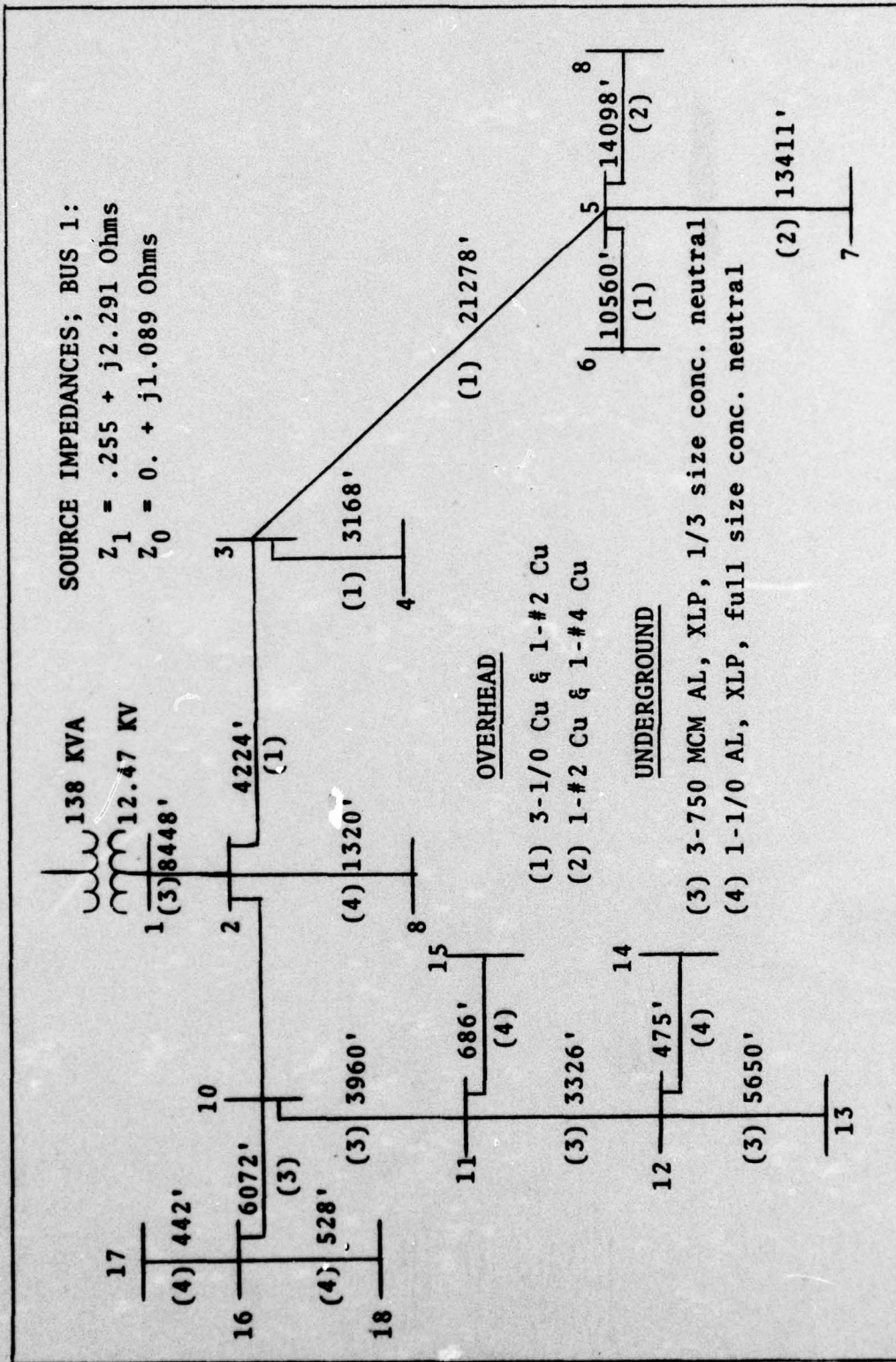


Fig. A-20. System Diagram, Example Problem 2 (Ref 3:40)

1	3	5	6	
NOBSYS	4			
NOFALT	7			
NOFALT	1	20.	1	7.2
	3	4	5	6
NOFALT	5	20.	3	12.47
NOMUTL				
	1	7.1995579	0.0	345.44 -3104.09 339.84 -3778.06
CURSOR	1			
SHTCKT	1			
	0	0		
	16	18	7.20	3 529 1 105500
	16	17	7.20	3 422 1 105500
	10	16	12.47	3 6072 3 750000
	11	15	7.20	1 686 1 105500
	6	13	12.47	3 5650 3 750000
	12	13	12.47	3 5650 3 750000
	12	14	7.20	3 475 1 105500
	11	12	12.47	3 3325 3 750000
	10	11	12.47	3 3960 3 750000
	2	10	12.47	3 4699 3 750000
	2	9	7.20	3 1320 1 105500
	5	9	7.20	1 14098 1 66370
	5	7	7.20	1 13411 1 66370
	5	6	12.47	1 10550 3 105500
	3	5	12.47	1 21279 3 105500
	3	4	12.47	1 3168 3 105500
	2	3	12.47	1 4224 3 105500
	1	2	12.47	3 8448 3 750000
SYSPAR				
PGMCUN	3	0	4	
EXAMPLE PROBLEM	42			

SHORT CIRCUIT ANALYSIS
FORTRAN Control Cards

Fig. A-21. Data Card Format, Example 2

POWER DISTRIBUTION SYSTEM ANALYSIS PROGRAM (POSAP)
TAPE 1 PRINTOUT

EXAMPLE PROBLEM #2
SHORT CIRCUIT ANALYSIS

**** PROGRAM CONTROL CONSTANTS ****

CON= 3
INP= 0
OUT= 4
CHG= -0

*** PROGRAM PARAMETER CONSTANTS ***

BASE KVA FREQUENCY TEMPERATURE EARTH RESISTIVITY
100000. KVA 60. HZ 25. DEG. C 100. METER-OHM

LI NDATA SUBROUTINE
ASSEMBLED INPUT LINE DATA (PER-UNIT)

CONDUCTOR NO.	FROM - TO	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
1	1 2	.2977	.3167	.6454	.1695
2	2 3	.2856	.3773	.6070	1.0602
3	3 4	.2142	.2830	.4553	.7951
4	3 5	1.4367	1.9009	3.0579	5.3404
5	5 5	.7140	.9433	1.5175	2.6504

CONDUCTOR NO.	5				
FROM - TO	RE(Z)	IM(Z)	RE(Z)	IM(Z)	
5 - 7	1.4404	1.2457	2.6315	3.7903	
CONDUCTOR NO.	7				
FROM - TO	RE(Z)	IM(Z)	RE(Z)	IM(Z)	
5 - 8	1.5142	1.3095	2.7663	3.9444	
CONDUCTOR NO.	8				
FROM - TO	RE(Z)	IM(Z)	RE(Z)	IM(Z)	
2 - 9	.1740	.0461	.4837	.2210	
CONDUCTOR NO.	9				
FROM - TO	RE(Z)	IM(Z)	RE(Z)	IM(Z)	
2 - 10	.1656	.1762	.3530	.0943	
CONDUCTOR NO.	10				
FROM - TO	RE(Z)	IM(Z)	RE(Z)	IM(Z)	
10 - 11	.1395	.1485	.3025	.0735	
CONDUCTOR NO.	11				
FROM - TO	RE(Z)	IM(Z)	RE(Z)	IM(Z)	
11 - 12	.1172	.1247	.2541	.0667	
CONDUCTOR NO.	12				
FROM - TO	RE(Z)	IM(Z)	RE(Z)	IM(Z)	
12 - 14	.0526	.0166	.1752	.0795	
CONDUCTOR NO.	13				
FROM - TO	RE(Z)	IM(Z)	RE(Z)	IM(Z)	
12 - 13	.1991	.2118	.4317	.1134	
CONDUCTOR NO.	14				
FROM - TO	RE(Z)	IM(Z)	RE(Z)	IM(Z)	
11 - 15	.0464	.0613	.0703	.2555	
CONDUCTOR NO.	15				
FROM - TO	RE(Z)	IM(Z)	RE(Z)	IM(Z)	
10 - 15	.2140	.2276	.4633	.1218	
CONDUCTOR NO.	16				
FROM - TO	RE(Z)	IM(Z)	RE(Z)	IM(Z)	
16 - 17	.0556	.0147	.1555	.0707	
CONDUCTOR NO.	17				
FROM - TO	RE(Z)	IM(Z)	RE(Z)	IM(Z)	
16 - 18	.0696	.0184	.1953	.0384	

*
* SHORT CIRCUIT INPUT DATA *
*

SOURCE IMPEDANCE BUS NO. 1 VOLTS L-V(KV) 7.20 0.00
3-PH FAULT CURRENT (AMPS) 345.44 -3104.09
PH-GND FAULT CURRENT (AMPS) 339.94 -3778.06
FAULT Z(OHMS) -0.0 -0.0; NEUT Z(OHMS) -0.0 -0.0

FAULT IMPEDANCE(ZF) (OHMS)=20.000 -0.000
PHASE= 3 BUS VOLTAGE (KV)= 12.47
BUS 3
BUS 4
BUS 5
BUS 6
BUS 8

FAULT IMPEDANCE(ZF) (OHMS)=20.000 -0.000
PHASE= 1 BUS VOLTAGE (KV)= 7.20
BUS 7

```
*****  
* POWER DISTRIBUTION SYSTEM ANALYSIS PROGRAM (PDSAP) *  
* TAPE 2 PRINTOUT *  
*****
```

```
*****  
* EXAMPLE PROBLEM #2 *  
* SHORT CIRCUIT ANALYSIS *  
*****
```

**** PROGRAM CONTROL CONSTANTS ****

CON= 3
INP= 0
OUT= 4
CHG= -0

*** PROGRAM PARAMETER CONSTANTS ***

BASE KVA	FREQUENCY	TEMPERATURE	EARTH RESISTIVITY
100000. KVA	60. HZ	25. DEG. C	100. METER-OHM

```
*****  
* RESULTS OF SHORT CIRCUIT ANALYSIS *  
* ALL VALUES ARE PER-UNIT *  
* SYSTEM HAS 19 BUSES. FAULT CODE (SCOP) IS -0 *  
* THERE ARE 1 SUBSYSTEM STUDIES (ISYS). *  
*****
```

```
*****  
* SUBSYSTEM STUDY NO. 1. *  
* NUMBER OF BUSES IN THIS SYSTEM IS: 4. *  
*****
```

 * FAULT SUMMARY FOR BUS 3 *
 * ZF= (12.86, 0.00) P.U. *
 * ZG= (0.00, 0.00) P.U. *

THREE-PHASE

 IF(MAG)= .0726
 X/R= .159

PHASE-GROUND

 IF(MAG)= .0718
 X/R= .152

PHASE-PHASE

 IF(MAG)= .1155
 X/R= .302
 EF(A)= 1.0000
 EF(B)= 1.0081
 EF(C)= .7551

P4-PH-GROUND

 IF(MAG)= .0372
 X/R= .075
 EF(A)= 1.0060
 EF(B)= .4784
 EF(C)= .4784
 IF(B)= .3949
 X/R(B)= 3.035
 IF(C)= .3607
 X/R(C)= 2.713

BUS VOLTAGES
 BUS V(MAG)
 1 .9770
 5 .9733
 3 .9733
 5 .9733

PHASE VOLTAGES
 A B C
 .98 .98 1.01
 .92 1.00 1.01
 .92 1.00 1.01
 .92 1.00 1.01

LINE CURRENTS
 LINE FAULT(I)
 6 5 0.0000
 5 3 0.0000

LINE CURRENTS
 FAULT(I) PH-A
 0.0000
 0.0000

1

 * FAULT SUMMARY FOR BUS 5 *
 * ZF= (12.86, 0.00) P.U. *
 * ZG= (0.00, 0.00) P.U. *

THREE-PHASE

 IF(MAG)= .0542
 X/R= .270

PHASE-GROUND

 IF(MAG)= .0603
 X/R= .326

PHASE-PHASE

 IF(MAG)= .0903
 X/R= .472
 EF(A)= 1.0000
 EF(B)= .9171
 EF(C)= .5849

P4-PH-GROUND

 IF(MAG)= .0334
 X/R= .212
 EF(A)= 1.0309
 EF(B)= .4291
 EF(C)= .4291
 IF(B)= .2005
 X/R(B)= 2.129
 IF(C)= .1751
 X/R(C)= 1.614

BUS VOLTAGES
 BUS V(MAG)
 1 .9532
 6 .8251
 3 .9259
 5 .8251

PHASE VOLTAGES
 A B C
 .97 .97 1.01
 .78 1.07 .99
 .91 1.00 1.01
 .78 1.07 .99

LINE CURRENTS
 LINE FAULT(I)
 6 5 0.0000
 5 3 .0542

LINE CURRENTS
 FAULT(I) PH-A
 0.0000
 .0503

0

1

 * FAULT SUMMARY FOR BUS 1 *
 * ZF= (0.00, 0.00) P.U. *
 * ZG= (0.00, 0.00) P.U. *

THREE-PHASE

 IF(MAG)= .6745
 X/R= 8.985

PHASE-GROUND

 IF(MAG)= .8193
 X/R= 11.117

PHASE-PHASE

 IF(MAG)= .5942
 X/R= 8.985
 EF(A)= 1.0000
 EF(B)= .5000
 EF(C)= .5000

PH-PH-GROUND

 IF(MAG)= 1.0422
 X/R= 17.501
 EF(A)= .7293
 EF(B)= 0.0000
 EF(C)= 0.0000
 IF(3)= .7516
 X/R(B)= 1.340
 IF(C)= .8034
 X/R(C)= .942

BUS VOLTAGES
 BUS V(MAG)
 1 0.0000
 6 .0000
 3 .0000
 5 .0000

PHASE VOLTAGES
 A B C
 0.00 .94 .89
 0.00 .94 .99
 0.00 .94 .89
 0.00 .94 .89

LINE CURRENTS
 LINE FAULT(I)
 6 5 0.0000
 5 3 0.0000

LINE CURRENTS
 FAULT(I) PH-A
 0.0000
 0.0000

1

 * FAULT SUMMARY FOR BUS 6 *
 * ZF= (12.86, 0.00) P.U. *
 * ZG= (0.00, 0.00) P.U. *

THREE-PHASE

 IF(MAG)= .0505
 X/R= .718

PHASE-GROUND

 IF(MAG)= .0555
 X/R= .397

PHASE-PHASE

 IF(MAG)= .0913
 X/R= .537
 EF(A)= 1.0000
 EF(B)= .3805
 EF(C)= .5270

PH-PH-GROUND

 IF(MAG)= .0315
 X/R= .271
 EF(A)= 1.0441
 EF(B)= .4060
 EF(C)= .4060
 IF(3)= .1611
 X/R(B)= 2.044
 IF(C)= .1389
 X/R(C)= 1.445

BUS VOLTAGES
 BUS V(MAG)
 1 .9670
 6 .7776
 3 .9239
 5 .8228

PHASE VOLTAGES
 A B C
 .97 .93 1.01
 .71 1.10 .99
 .91 1.00 1.01
 .77 1.05 1.00

LINE CURRENTS
 LINE FAULT(I)
 6 5 .0505
 5 3 .0605

LINE CURRENTS
 FAULT(I) PH-A
 .0555
 .0555

1

Example Problem 3

This example illustrates the use of the LINDATA input routine and both the Load Flow and Short Circuit routines. Fig. A-22 is the one-line diagram of a system specifically designed to test the total PDSAP program (Ref 2:258). This example highlights the capability of the PDSAP program to utilize information other than impedance values to make a system analysis.

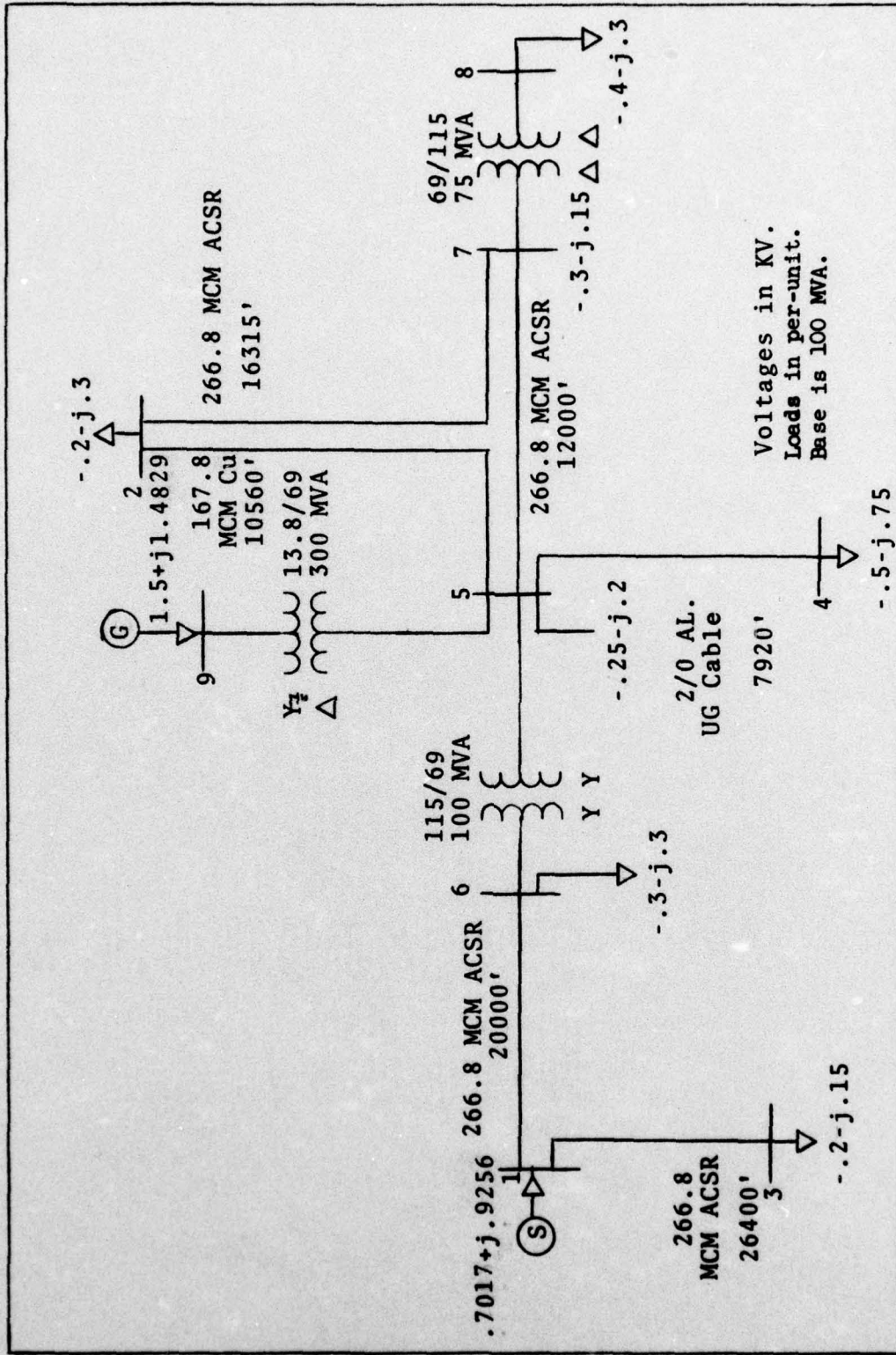


Fig. A-22. System Diagram, Example 3

(Ref 2:258)


```

*****
*
*      POWER DISTRIBUTION SYSTEM ANALYSIS PROGRAM (PDSAP)
*      TAPE 1 PRINTOUT
*
*****
    
```

```

*****
*
*      EXAMPLE PROBLEM #3
*      LOAD FLOW AND SHORT CKT ANALYSIS
*
*****
    
```

**** PROGRAM CONTROL CONSTANTS ****

```

CON= 5
INP= 0
OUT= 12
CHG= -0
    
```

*** PROGRAM PARAMETER CONSTANTS ***

```

BASE KVA      FREQUENCY    TEMPERATURE    EARTH RESISTIVITY
100010. KVA   60. 42         25. 023. C     100. 4ETER-344
    
```

```

*****
*
*      LINDATA SUBROUTINE
*      ASSEMBLED INPUT LINE DATA (PER-UNIT)
*
*****
    
```

```

CONDUCTOR NO. 1
FROM - TO    RE(Z)      IM(Z)      RE(Z0)     IM(Z0)
1 3          .0132     .0293     .0243     .1036

CONDUCTOR NO. 2
FROM - TO    RE(Z)      IM(Z)      RE(Z0)     IM(Z0)
1 6          .0100     .0222     .0192     .0765

TRANSFORMER, LTC NO. 1
FROM - TO    RE(Z)      IM(Z)      RE(Z0)     IM(Z0)
5 9          0.0000   .0293     0.0000     .0293
TAP          T4N      T4X      CONEC CODE
1.00000     .90000   1.10000   5

TRANSFORMER, AUTO. NO. 1
FROM - TO    RE(Z)      IM(Z)      RE(Z0)     IM(Z0)
5 5          0.0000   .0464     0.0000     0.0000
TAP          CONEC CODE
1.000        1
    
```

CONDUCTOR NO.	3				
FROM - TO	4 - 5	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
		.0295	.0154	.0791	.0412
CONDUCTOR NO.	4				
FROM - TO	2 - 5	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
		.0147	.0344	.0257	.1170
CONDUCTOR NO.	5				
FROM - TO	2 - 7	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
		.0227	.0502	.0413	.1779
CONDUCTOR NO.	6				
FROM - TO	5 - 7	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
		.0157	.0370	.0304	.1309
TRANSFORMER, FIXED NO.	1				
FROM - TO	7 - 8	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
		0.0000	.1486	0.0000	.1485
TAP	CONEC CODE				
1.00000	2				

 *
 * SUMMARIZED INPUT BUS DATA: PER-UNIT *
 * LISTED BY ASCENDING BUS NUMBERS *
 *

NO.	TYPE	V(MAG)	V(ANG-DEG)	POWER		P(MIN)	Q(MAX)
				REAL	REACTIVE		
1	3	1.000	0.0000	0.00000	0.00000	0.0000	0.0000
2	1	1.000	0.0000	-.20000	-.30000	0.0000	0.0000
3	1	1.000	0.0000	-.20000	-.15000	0.0000	0.0000
4	1	1.000	0.0000	-.50000	-.75000	0.0000	0.0000
5	1	1.000	0.0000	-.25000	-.20000	0.0000	0.0000
6	1	1.000	0.0000	-.30000	-.30000	0.0000	0.0000
7	1	1.000	0.0000	-.30000	-.15000	0.0000	0.0000
8	1	1.000	0.0000	-.40000	-.30000	0.0000	0.0000
9	2	1.000	0.0000	1.50000	0.00000	0.0000	1.5000

 *
 * REORDERED BUSLIST RETURNED BY SUBROUTINE ORDER *
 *

3 1 4 6 8 9 2 5 7

 *
 * SHORT CIRCUIT INPUT DATA *
 *

SOURCE IMPEDANCE BUS NO. 1 VOLTS L-N(KV) 55.40 0.00
 3-PH FAULT CURRENT (AMPS) 575.73 -1078.32
 PH-GND FAULT CURRENT (AMPS) 434.94 -754.27
 FAULT Z(OHMS) -0.0 -0.0; NEUT Z(OHMS) -0.0 -0.0

 *
 * SORTED LINE INPUT DATA *
 * LISTED BY ASCENDING BUS NUMBERS *
 *

SB	EB	G	B	RE(Z0)	IM(Z0)
0	1	.11459E+01	-.21491E+01	.47757E+00	.77415E+00
0	9	0.	0.	0.	.29263E-01
1	0	.11469E+01	-.21491E+01	.47757E+00	.77415E+00
1	3	.12830E+02	-.28359E+02	.24049E-01	.10361E+00
1	6	.15935E+02	-.37434E+02	.18219E-01	.79496E-01
2	5	.10509E+02	-.24599E+02	.26592E-01	.11598E+00
2	7	.74735E+01	-.16520E+02	.41283E-01	.17787E+00
3	1	.12830E+02	-.28359E+02	.24049E-01	.10361E+00
4	5	.26579E+02	-.13827E+02	.78052E-01	.41155E-01
5	2	.10509E+02	-.24599E+02	.26592E-01	.11598E+00
5	4	.26579E+02	-.13827E+02	.78052E-01	.41155E-01
5	6	0.	-.21532E+02	0.	0.
5	7	.10161E+02	-.22461E+02	.30365E-01	.13083E+00
5	9	0.	-.34173E+02	0.	.29263E-01
6	1	.15935E+02	-.37434E+02	.18219E-01	.79496E-01
6	5	0.	-.21532E+02	.10000E+11	0.
7	2	.74735E+01	-.16520E+02	.41283E-01	.17787E+00
7	5	.10151E+02	-.22451E+02	.30365E-01	.13083E+00
7	8	0.	-.67293E+01	.10000E+11	0.
8	7	0.	-.67293E+01	.10000E+11	0.
9	0	0.	0.	0.	.29263E-01
9	5	0.	-.34173E+02	.10000E+11	0.

```
*****  
* POWER DISTRIBUTION SYSTEM ANALYSIS PROGRAM (PDSAP) *  
* TAPE 2 PRINTOUT *  
*****
```

```
*****  
* EXAMPLE PROBLEM #3 *  
* LOAD FLOW AND SHORT CKT ANALYSIS *  
*****
```

**** PROGRAM CONTROL CONSTANTS ****

CON= 5
INP= 0
OUT= 12
CHG= -0

*** PROGRAM PARAMETER CONSTANTS ***

BASE KVA	FREQUENCY	TEMPERATURE	EARTH RESISTIVITY
100000. KVA	60. HZ	25. DEG. C	100. METER-OHM

```
*****  
* RESULTS OF FAST DECOUPLED LOAD FLOW ANALYSIS *  
* ALL MAGNITUDE VALUES ARE PER-UNIT *  
* SYSTEM HAS 9 BUSES; 1 ARE TYPE 2. *  
* NUMBER OF TIMES LOAD BUSES WILL BE CHANGED(NLC)= -0. *  
* CONVERGENCE TOLERANCES: *  
* PTOL= .00100 *  
* QTOL= .00100 *  
*****
```

**** BUS NO. 9 EXCEEDED ITS MAXIMUM Q LIMIT
Q SPECIFIED: 1.5000 PER-UNIT.
Q CALCULATED: 1.5127 PER-UNIT.
AMOUNT EXCEEDED: .95 PERCENT.

```

*****
*
*                               SYSTEM SUMMARY
*
*   CONVERGENCE OBTAINED IN:
*   5 DELTA THETA AND
*   5 DELTA V ITERATIONS.
*
*****
    
```

```

***** SLACK BUS POWER *****
BUS NO.      REAL      REACTIVE      MAG.
    1         .70173      .92561         1.15154
    
```

```

*****
*
*   CALCULATED LINE FLOWS
* (THE LINE FLOWS ARE DEFINED POSITIVE WHEN FLOWING OUT FROM THE BUS)
*
*****
    
```

LINE		POWER		LINE		POWER	
FROM	TO	REAL	REACTIVE	FROM	TO	REAL	REACTIVE
1	3	.20084	.15186	3	1	-.20000	-.15000
1	6	.50089	.77375	5	1	-.49235	-.75491
2	5	-.35554	-.37255	5	2	.35007	.39294
2	7	.15555	.07255	7	2	-.15489	-.07086
4	5	-.49990	-.74964	5	4	.52773	.75417
5	6	-.19236	-.44306	6	5	.19236	.45491
5	7	.55444	.44810	7	5	-.54514	-.42756
5	9	-1.50000	-1.35233	9	5	1.50000	1.48286
7	8	.40000	.34842	8	7	-.40000	-.30000

```

*****
*
*                               OUTPUT BUS DATA
*
*****
    
```

NO.	TYPE	NAME	V (MAG)	V (ANG-DEG)	POWER	
					REAL	REACTIVE
1	3	ALP48	1.0000	0.0000	.70173	.92561
2	1	FOXTROT	.9370	-1.1754	-.20000	-.30000
3	1	BRAVO	.9929	-.2232	-.21000	-.15000
4	1	ECHO	.9277	.1930	-.50000	-.75000
5	1	DELTA	.9563	-.7434	-.25000	-.20000
6	1	CHARLIE	.9778	-.1950	-.30000	-.30000
7	1	GOLF	.9233	-1.5515	-.30000	-.15000
8	1	HOTEL	.8750	-5.7583	-.40000	-.30000
9	2	INDIA	.9957	1.5910	1.50000	1.48286

 *
 * OUTPUT TRANSFORMER DATA *
 *

SB	EB	TAP	TAP(MIN)	TAP(MAX)
5	5	1.00000	-.00000	-.00000
5	9	1.00000	.90000	1.10000
7	8	1.00000	-.00000	-.00100

 * RESULTS OF SHORT CIRCUIT ANALYSIS *
 * ALL VALUES ARE PER-UNIT *
 * SYSTEM HAS 9 BUSES, FAULT CODE (SCOP) IS -0. *
 * THERE ARE 1 SUBSYSTEM STUDIES (ISYS). *

 * SUBSYSTEM STUDY NO. 1. *
 * NUMBER OF BUSES IN THIS SYSTEM IS: 2. *

1

 * FAULT SUMMARY FOR BUS 1 *
 * ZF= (0.00, 0.00) P.U. *
 * ZS= (0.00, 0.00) P.U. *

THREE-PHASE	PHASE-GROUND	PHASE-PHASE	PH-PH-GROUND
IF(MAG)= 2.4359	IF(MAG)= 1.7343	IF(MAG)= 2.1095	IF(MAG)= 1.3459
X/R= 1.874	X/R= 1.734	X/R= 1.874	X/R= 1.364
*****	*****	EF(A)= 1.0000	EF(A)= 1.2242
		EF(B)= .5000	EF(B)= 0.0000
		EF(C)= .5000	EF(C)= 0.0000
		*****	IF(B)= 2.2467
			X/R(B)= 5.302
			IF(C)= 2.1812
			X/R(C)= .955

1

 * FAULT SUMMARY FOR BUS 9 *
 * ZF= (0.00, 0.00) P.U. *
 * ZS= (0.00, 0.00) P.U. *

THREE-PHASE	PHASE-GROUND	PHASE-PHASE	PH-PH-GROUND
IF(MAG)= 1.9991	IF(MAG)= 2.9047	IF(MAG)= 1.7217	IF(MAG)= 5.3359
X/R= 2.263	X/R= 2.335	X/R= 2.263	X/R= 2.551
*****	*****	EF(A)= 1.0000	EF(A)= .1576
		EF(B)= .5000	EF(B)= 0.0000
		EF(C)= .5000	EF(C)= 0.0000
		*****	IF(B)= 3.1339
			X/R(B)= 1.412
			IF(C)= 3.2571
			X/R(C)= .185

Example Problem 4

This example illustrates the use of the LINEZ input routine and the Short Circuit routine. This example was adapted for use with the PDSAP program to show the capability of the program to calculate line currents (Ref 7:2-34). Fig. A-24 is the one-line diagram of the system. System base for this example is 50 MVA. Note that the example has a three-winding transformer and the program equivalent is shown in Detail A. Fig. A-25 illustrates the format and order of the data cards.

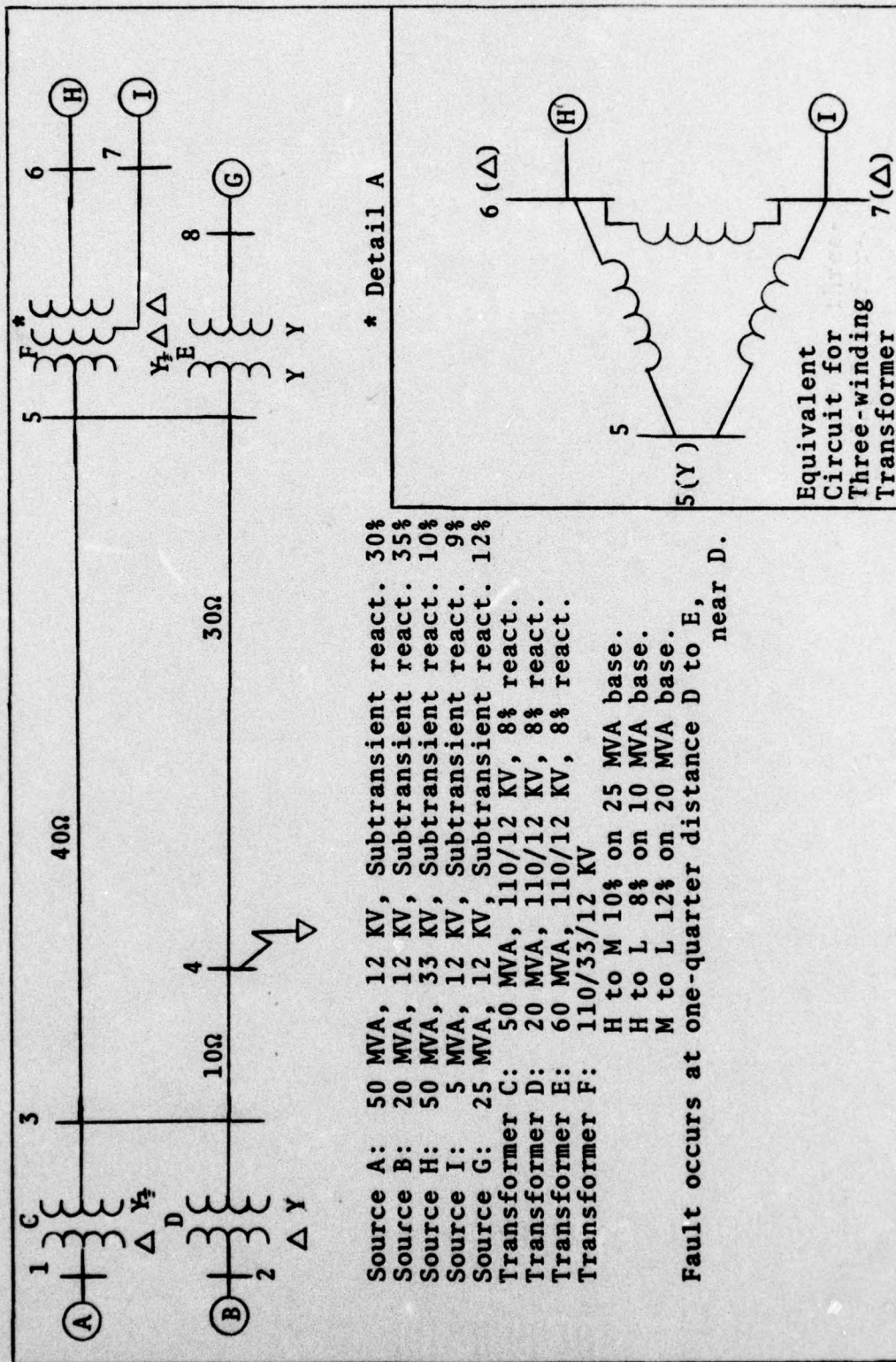


Fig. A-24. System Diagram, Example 4

NO	ADFALT	NO	ADFALT	NO	ADFALT	NO	ADFALT
8	6.928	0.0	0.0	-10023.4	0.0	-10023.4	
7	6.928	0.0	0.0	-2672.9	0.0	-2672.9	
6	19.05	0.0	0.0	-8751.08	0.0	-8751.08	
2	6.928	0.0	0.0	-2749.3	0.0	-2749.3	
1	6.928	0.0	0.0	-8018.6	0.0	-8018.6	

CUPSDP 5							

SHTCKT							
0	0						
5	6	110	53	50000	55.63	55.63	1.
6	7	110	53	50000	0.9231	0.9231	1.
5	7	110	53	50000	0.28001	0.28001	1.
5	8	110	53	60000	0.	0.	1.
2	3	110	53	20000	0.	0.	1.
3	1	110	53	50000	0.	0.	1.
3	5	110	12	1	0.	0.	1.
2	4	110	12	1	0.	0.	1.
4	5	110	12	1	0.	0.	1.

SYSPAR 50000.							

PGMCEN 3 1 4							

EXAMPLE PROBLEM #4							

SHORT CIRCUIT ANALYSIS (QUESTINGHOUSE)							

FORTRAN Control Cards

Fig. A-25. Data Card Format, Example 4

```

.....
*
* POWER DISTRIBUTION SYSTEM ANALYSIS PROGRAM (PDSAP)
* TAP 1 PRINTOUT
*
*
.....
    
```

```

.....
*
* EXAMPLE PROBLEM #4
* SHORT CIRCUIT ANALYSIS (WESTINGHOUSE)
*
*
.....
    
```

**** PROGRAM CONTROL CONSTANTS ****

```

CON= 3
INP= 1
OUT= 4
CHG= -0
    
```

*** PROGRAM PARAMETER CONSTANTS ***

```

BASE KVA      FREQUENCY    TEMPERATURE    EARTH RESISTIVITY
50000. KVA    50. HZ         25. DEG. C     100. METER-OHM
    
```

```

.....
*
* LINE7 SUBROUTINE
* ASSEMBLED INPUT LINE DATA (PEP-UNIT)
*
*
.....
    
```

CONDUCTOR NO.	1				
FROM - TO	4 - 5	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
		0.0000	.1240	0.0000	.4333
CONDUCTOR NO.	2				
FROM - TO	3 - 4	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
		0.0000	.0413	0.0000	.1446
CONDUCTOR NO.	3				
FROM - TO	3 - 5	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
		0.0000	.1653	0.0000	.5795
TRANSFORMER, FIXED NO.	1				
FROM - TO	3 - 1	RE(Z)	IM(Z)	RE(Z0)	IM(Z0)
		-0.0000	.0798	0.0000	.0798
TAP	CONEC CODE				
1.00000	4				

TRANSFORMER, FIXED NO.	2			
FROM - TO	RE(7)	IM(7)	RE(70)	IM(70)
2 7	0.0000	.2000	0.0000	.2000
TAP	CONEC CODE			
1.00000	1			
TRANSFORMER, FIXED NO.	3			
FROM - TO	RE(7)	IM(7)	RE(70)	IM(70)
5 8	0.0000	.0667	0.0000	.0667
TAP	CONEC CODE			
1.00000	1			
TRANSFORMER, FIXED NO.	4			
FROM - TO	RE(7)	IM(7)	RE(70)	IM(70)
5 7	0.0000	1.1571	0.0000	1.1571
TAP	CONEC CODE			
1.00000	4			
TRANSFORMER, FIXED NO.	5			
FROM - TO	RE(7)	IM(7)	RE(70)	IM(70)
6 7	0.0000	.3814	0.0000	.3814
TAP	CONEC CODE			
1.00000	2			
TRANSFORMER, FIXED NO.	6			
FROM - TO	RE(7)	IM(7)	RE(70)	IM(70)
5 5	0.0000	.2299	0.0000	.2299
TAP	CONEC CODE			
1.00000	4			

.....
 *
 * S40PT CIRCUIT INPUT DATA *
 *
 *

SOURCE IMPEDANCE BUS NO.	1	VOLTS L-N(KV)	5.93	0.00
3-PH FAULT CURRENT (AMPS)	0.00	-8019.50		
PH-GND FAULT CURRENT (AMPS)	0.00	-8019.50		
FAULT Z(OHMS)	-0.0	-0.01	NEUT Z(OHMS)	-0.0 -0.0
SOURCE IMPEDANCE BUS NO.	2	VOLTS L-N(KV)	5.93	0.00
3-PH FAULT CURRENT (AMPS)	0.00	-2749.30		
PH-GND FAULT CURRENT (AMPS)	0.00	-2749.30		
FAULT Z(OHMS)	-0.0	-0.01	NEUT Z(OHMS)	-0.0 -0.0
SOURCE IMPEDANCE BUS NO.	5	VOLTS L-N(KV)	19.95	0.00
3-PH FAULT CURRENT (AMPS)	0.00	-3751.08		
PH-GND FAULT CURRENT (AMPS)	0.00	-3751.08		
FAULT Z(OHMS)	-0.0	-0.01	NEUT Z(OHMS)	-0.0 -0.0
SOURCE IMPEDANCE BUS NO.	7	VOLTS L-N(KV)	5.93	0.00
3-PH FAULT CURRENT (AMPS)	0.00	-2672.30		
PH-GND FAULT CURRENT (AMPS)	0.00	-2672.30		
FAULT Z(OHMS)	-0.0	-0.01	NEUT Z(OHMS)	-0.0 -0.0
SOURCE IMPEDANCE BUS NO.	8	VOLTS L-N(KV)	5.93	0.00
3-PH FAULT CURRENT (AMPS)	0.00	-10023.40		
PH-GND FAULT CURRENT (AMPS)	0.00	-10023.40		
FAULT Z(OHMS)	-0.0	-0.01	NEUT Z(OHMS)	-0.0 -0.0

* POWER DISTRIBUTION SYSTEM ANALYSIS PROGRAM (POSAP) *
* TAP 2 PRINTOUT *

* EXAMPLE PROBLEM #4 *
* SHORT CIRCUIT ANALYSIS (WESTINGHOUSE) *

**** PROGRAM CONTROL CONSTANTS ****

CON= 3
INP= 1
OUT= 4
CHG= -0

*** PROGRAM PARAMETER CONSTANTS ***

BASE KVA	FREQUENCY	TEMPERATURE	EARTH RESISTIVITY
50000. KVA	60. HZ	25. DEG. C	100. METER-OHM

* RESULTS OF SHORT CIRCUIT ANALYSIS *
* ALL VALUES ARE PER-JUNIT *
* SYSTEM HAS 8 BUSES. FAULT CODE(SCOP) IS -0. *
* THERE ARE -0 SUBSYSTEM STUDIES(ISYS). *

* AUTOMATIC SHORT-CIRCUIT STUDY: *
* ENTIRE NETWORK STUDY WILL BE COMPLETED *
* IN 1 PASSES OF SHORT-CIRCUIT PROGRAM. *

1

```
*****
* FAULT SUMMARY FOR BUS 1 *
* IF = ( 0.00, 0.00) P.U. *
* IG = ( 0.00, 0.00) P.U. *
*****
```

```
THREE-PHASE
*****
IF(MAG)= 7.0437
X/R= 0
*****
```

```
PHASE-GROUND
*****
IF(MAG)= 5.1393
X/R=*****
*****
```

```
PHASE-PHASE
*****
IF(MAG)= 6.1047
X/R= R
EF(A)= 1.0000
EF(B)= .5000
EF(C)= .5000
*****
```

```
P4-PH-GROUND
*****
IF(MAG)= 4.0437
X/R=*****
EF(A)= 1.2132
EF(B)= 0.0000
EF(C)= 0.0000
IF(B)= 6.4307
X/R(B)= 3.013
IF(C)= 5.4307
X/R(C)= 3.013
*****
```

```
BUS VOLTAGES
BUS V(MAG)
1 0.0000
3 .2955
2 .4274
4 .3593
5 .5496
6 .8591
7 .8749
8 .6475
```

```
PHASE VOLTAGES
A B C
0.00 1.15 1.16
.65 .93 .93
.72 .94 .94
.69 .93 .93
.78 .95 .95
.93 .98 .98
.92 .93 .98
.83 .95 .96
```

```
LINE CURRENTS
LINE FAULT(I)
3 1 3.7151
3 2 .5544
4 3 1.5309
5 3 1.5309
5 4 1.5309
6 5 1.3454
7 5 .2466
7 6 .0332
8 5 1.5537
```

```
LINE CURRENTS
FAULT(I) PH-A
1.9062
.3180
.7441
.7441
.7441
.6544
.1199
.0307
.7139
```

1

 * FAULT SUMMARY FOR BUS 2 *
 * ZF= (0.00, 0.00) P.U. *
 * ZS= (0.00, 0.00) P.U. *

THREE-PHASE

 IF(MAG)= 4.0559
 X/R= P

PHASE-GROUND

 IF(MAG)= 2.1330
 X/R=*****

PHASE-PHASE

 IF(MAG)= 3.5173
 X/R= R
 EF(A)= 1.0000
 EF(B)= .5000
 EF(C)= .5000

P4-PH-GROUND

 IF(MAG)= 1.5025
 X/R=*****
 EF(A)= 1.3148
 EF(B)= 0.0000
 EF(C)= 0.0000
 IF(B)= 3.5927
 X/R(B)= 4.677
 IF(C)= 3.5927
 X/R(C)= 4.677

BUS VOLTAGES
 BUS V(MAG)
 1 .6705
 3 .5879
 2 0.0000
 4 .6203
 5 .7323
 6 .9164
 7 .9021
 8 .7009

PHASE VOLTAGES
 A B C
 .88 .97 .97
 .85 .95 .96
 0.00 1.23 1.29
 .85 .97 .97
 .90 .98 .98
 .97 .99 .99
 .95 .99 .99
 .92 .93 .98

LINE CURRENTS
 LINE FAULT(I)
 3 1 1.0334
 3 2 2.9141
 4 3 .0079
 5 3 .0079
 5 4 .0079
 6 5 .7034
 7 5 .1453
 7 6 .0775
 8 5 .8710

LINE CURRENTS
 FAULT(I) PH-A
 .3958
 1.0501
 .3272
 .3272
 .3272
 .2877
 .0527
 .0135
 .3139

1

 * FAULT SUMMARY FOR BUS 3 *
 * ZF= (0.00, 0.00) P.U. *
 * ZG= (0.00, 0.00) P.U. *

THREE-PHASE

 IF(MAG)= 7.9155
 X/R= R

PHASE-GROUND

 IF(MAG)= 9.3424
 X/R=*****

PHASE-PHASE

 IF(MAG)= 6.8543
 X/R= R
 EF(A)= 1.0000
 EF(B)= .5000
 EF(C)= .5000

PH-PH-GROUND

 IF(MAG)= 11.3971
 X/R=*****
 EF(A)= .7901
 EF(B)= 0.0000
 EF(C)= 0.0000
 IF(B)= 9.9139
 X/R(B)= 1.203
 IF(C)= 9.9139
 X/R(C)= 1.203

BUS VOLTAGES
 BUS V(MAG)
 1 .2101
 3 0.0000
 2 .1850
 4 .0993
 5 .3797
 6 .7997
 7 .7554
 8 .4989

PHASE VOLTAGES
 A B C
 1 .38 .69 .49
 3 0.00 .92 .92
 2 .36 .84 .98
 4 .10 .92 .92
 5 .41 .97 .93
 6 .84 .95 .96
 7 .82 .95 .96
 8 .61 .97 .92

LINE CURRENTS
 LINE FAULT (T)
 3 1 2.5329
 3 2 .9302
 4 3 2.1752
 5 3 2.1752
 5 4 2.1752
 6 5 1.9139
 7 5 .3506
 7 6 .0839
 8 5 2.0979

LINE CURRENTS
 FAULT (I) PH-A
 2.0717
 .7319
 1.9339
 1.3339
 1.3339
 1.5060
 .2759
 .2707
 1.3339

 * FAULT SUMMARY FOR BUS 4 *
 * ZF= (0.00, 0.00) P.U. *
 * ZG= (0.00, 0.00) P.U. *

THREE-PHASE

 IF(MAG)= 7.1777
 X/R= P

PHASE-GROUND

 IF(MAG)= 6.7726
 X/R=*****

PHASE-PHASE

 IF(MAG)= 6.2732
 X/R= R
 EF(A)= 1.0000
 EF(B)= .5000
 EF(C)= .5000

PH-PH-GROUND

 IF(MAG)= 5.3950
 X/R=*****
 EF(A)= 1.0554
 EF(B)= 0.0000
 EF(C)= 0.0000
 IF(1)= 7.0054
 X/R(B)= 1.950
 IF(2)= 7.0054
 X/R(C)= 1.950

BUS	V(MAG)
1	.3463
3	.1725
2	.7254
4	0.0000
5	.3747
6	.8044
7	.7709
8	.5103

PHASE VOLTAGES			
	A	B	C
1	.53	.91	.91
3	.35	.94	.94
2	.58	.91	.91
4	0.00	1.03	1.03
5	.49	.95	.96
6	.88	.97	.97
7	.86	.97	.97
8	.69	.93	.93

LINE	FAULT(I)
3 1	2.1798
3 2	.7638
4 3	4.1734
5 3	1.2248
5 4	3.0242
6 5	1.8695
7 5	.3423
7 6	.0374
8 5	2.0393

LINE	FAULT(I) PH-A
3 1	1.3668
3 2	.4829
4 3	4.2872
5 3	.7923
5 4	2.4854
6 5	1.1721
7 5	.2147
7 6	.0551
8 5	1.2786

 * FAULT SUMMARY FOR BUS 5 *
 * ZF= (0.00, 0.00) P.U. *
 * ZG= (0.00, 0.00) P.U. *

THREE-PHASE

 IF(MAG)= 9.5500
 X/R= P

PHASE-GROUND

 IF(MAG)= 8.9386
 X/R=*****

PHASE-PHASE

 IF(MAG)= 9.2703
 X/R= R
 EF(A)= 1.0000
 EF(B)= .5000
 EF(C)= .5000

PH-PH-GROUND

 IF(MAG)= 9.4103
 X/R=*****
 EF(A)= 1.0502
 EF(B)= 0.0000
 EF(C)= 0.0000
 IF(B)= 9.2757
 X/R(B)= 1.953
 IF(C)= 9.2757
 X/R(C)= 1.953

BUS	V(MAG)
1	.3809
3	.2275
2	.3712
4	.1705
5	0.0000
6	.6071
7	.6335
8	.2174

PHASE VOLTAGES			
	A	B	C
1	.62	.92	.92
3	.44	.33	.93
2	.61	.92	.92
4	.33	.95	.95
5	0.00	1.03	1.03
6	.80	.35	.95
7	.77	.95	.95
8	.51	.90	.90

LINE CURRENTS		
LINE	FAULT (I)	
3	1	2.0339
3	2	.7185
4	3	1.3763
5	3	1.3763
5	4	1.3763
6	5	2.0991
7	5	.5476
7	6	.1404
8	5	3.2508

LINE CURRENTS		
LINE	FAULT (I)	PH-A
3	1	1.2591
3	2	.4484
4	3	1.3682
5	3	1.3682
5	4	1.3682
6	5	1.8652
7	5	.3417
7	6	.0876
8	5	2.0347

 * FAULT SUMMARY FOR BUS 5 *
 * ZF= (0.00, 0.00) P.U. *
 * ZG= (0.00, 0.00) P.U. *

THREE-PHASE

 IF(MAG)= 13.7595
 X/R= P

PHASE-GROUND

 IF(MAG)= 12.0153
 X/R=*****

PHASE-PHASE

 IF(MAG)= 11.5593
 X/R= R
 EF(A)= 1.0000
 EF(B)= .5000
 EF(C)= .5000

PH-PH-GROUND

 IF(MAG)= 10.9163
 X/R=*****
 EF(A)= 1.0914
 EF(B)= 0.0000
 EF(C)= 0.0000
 IF(B)=12.7922
 X/R(B)= 2.120
 IF(C)= 12.7922
 X/R(C)= 2.120

BUS VOLTAGES
 V(MAG)
 1 .7323
 3 .6619
 2 .7243
 4 .6770
 5 .5823
 6 0.0000
 7 .3474
 8 .6575

PHASE VOLTAGES
 A B C
 .84 .95 .96
 .80 .95 .95
 .83 .95 .96
 .78 .95 .95
 .74 .94 .94
 0.00 1.05 1.05
 .61 .92 .92
 .79 .95 .95

LINE CURRENTS
 LINE FAULT(I)
 3 1 .4902
 3 2 .3145
 4 3 .6024
 5 3 .6024
 5 4 .6024
 6 5 2.4452
 7 5 .1457
 7 5 .9108
 8 5 1.4272

LINE CURRENTS
 FAULT(I) PH-A
 .5338
 .1886
 .3612
 .3612
 .3612
 1.4667
 .1114
 .5461
 .8557

 * FAULT SUMMARY FOR BUS 7 *
 * 7F= (0.00, 0.00) P.U. *
 * 7G= (0.00, 0.00) P.U. *

THREE-PHASE

 IF(MAG)= 3.0517
 X/R= R

PHASE-GROUND

 IF(MAG)= 2.1349
 X/R=*****

PHASE-2PHASE

 IF(MAG)= 3.4282
 X/R= R
 EF(A)= 1.0000
 EF(B)= .5000
 EF(C)= .5000

P1-PH-GROUND

 IF(MAG)= 1.4515
 X/R=*****
 EF(A)= 1.3154
 EF(B)= 0.0000
 EF(C)= 0.0000
 IF(B)= 3.5052
 X/R(B)= 4.692
 IF(C)= 3.5052
 X/R(C)= 4.692

BUS VOLTAGES
 BUS V(MAG)
 1 .9073
 3 .8927
 2 .9045
 4 .8740
 5 .8481
 6 .9065
 7 0.0000
 8 .8911

PHASE VOLTAGES
 A B C
 .97 .99 .99
 .95 .99 .99
 .97 .99 .99
 .95 .99 .99
 .95 .99 .99
 .93 .99 .98
 0.00 1.29 1.29
 .95 .99 .99

LINE CURRENTS
 LINE FAULT(I)
 3 1 .7099
 3 2 .1092
 4 3 .2091
 5 3 .2091
 5 4 .2091
 6 5 .1694
 7 5 .7330
 7 6 2.1147
 8 5 .4953

LINE CURRENTS
 FAULT(I) PH-A
 .1111
 .0392
 .0752
 .0752
 .0752
 .0649
 .2635
 .7603
 .1781

 * FAULT SUMMARY FOR BUS 8 *
 * ZF= (0.00, 0.00) P.U. *
 * ZG= (0.00, 0.00) P.U. *

THREE-PHASE

 IF(MAG)= 8.5978
 X/R= P

PHASE-GROUND

 IF(MAG)= 5.3475
 X/R=*****

PHASE-PHASE

 IF(MAG)= 7.4457
 X/R= R
 EF(A)= 1.0000
 EF(B)= .5000
 EF(C)= .5000

PH-PH-GROUND

 IF(MAG)= 5.0399
 X/R=*****
 EF(A)= 1.2074
 EF(B)= 0.0000
 EF(C)= 0.0000
 IF(B)= 7.9591
 X/R(A)= 2.950
 IF(C)= 7.9591
 X/R(C)= 2.950

BUS VOLTAGES
 BUS V(MAG)
 1 .6781
 3 .4557
 2 .5570
 4 .4156
 5 .2954
 6 .7796
 7 .7413
 8 0.0000

PHASE VOLTAGES
 A B C
 1 .79 .95 .95
 3 .73 .94 .94
 2 .78 .95 .95
 4 .71 .94 .94
 5 .65 .93 .93
 6 .89 .97 .97
 7 .87 .97 .97
 8 0.00 1.15 1.15

LINE CURRENTS
 LINE FAULT(I)
 3 1 1.4331
 3 2 .5063
 4 3 .9597
 5 3 .9597
 5 4 .9597
 6 5 2.1051
 7 5 .3353
 7 6 .0989
 8 5 4.4313

LINE CURRENTS
 FAULT(I) PH-A
 .7053
 .2492
 .4773
 .4773
 .4773
 1.0366
 .1899
 .0487
 2.1810

Appendix B. Program Flow Charts

This appendix, updated from the original (Ref 2:56-84), is included to provide a more detailed examination of the program flow and execution. The flow charts do not provide minute details of every program element, but do provide the basic functions performed by major programs and subroutines. The format is also non-standard, and was devised to fit the purpose of this appendix.

A few points about the format are included to aid understanding the figures that follow. Arrows (+ →) indicate program flow and branch entry locations. The capped line (⌈) indicates a statement or a group of statements in the program that perform a specific or related function. The text following the capped line summarizes the particular function. Numbers indicate actual program statement numbers. Figures B-1 through B-10 on the following pages show the major program element flow charts.

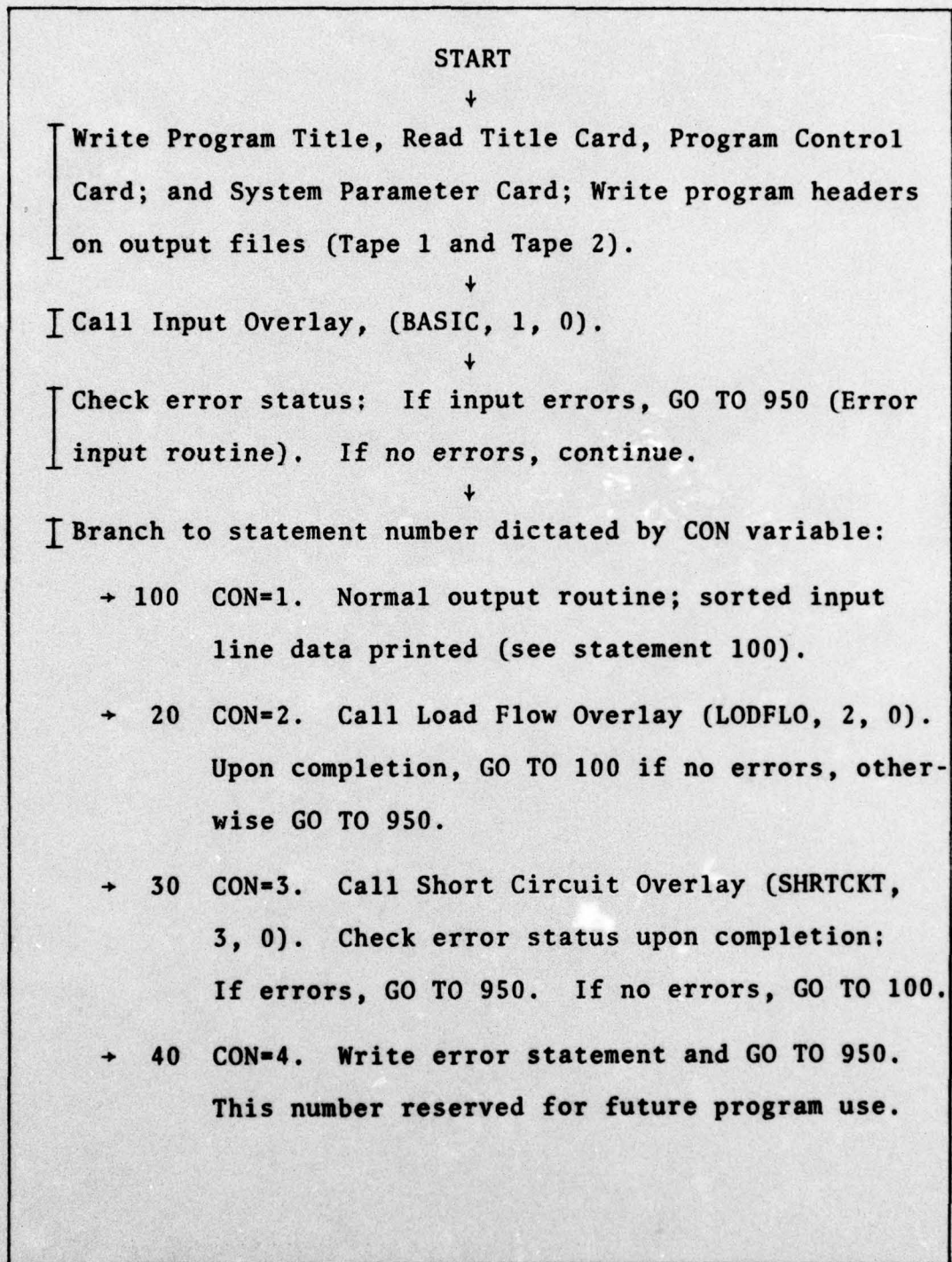


Fig. B-1. EXECUTIVE Routine Flow Chart

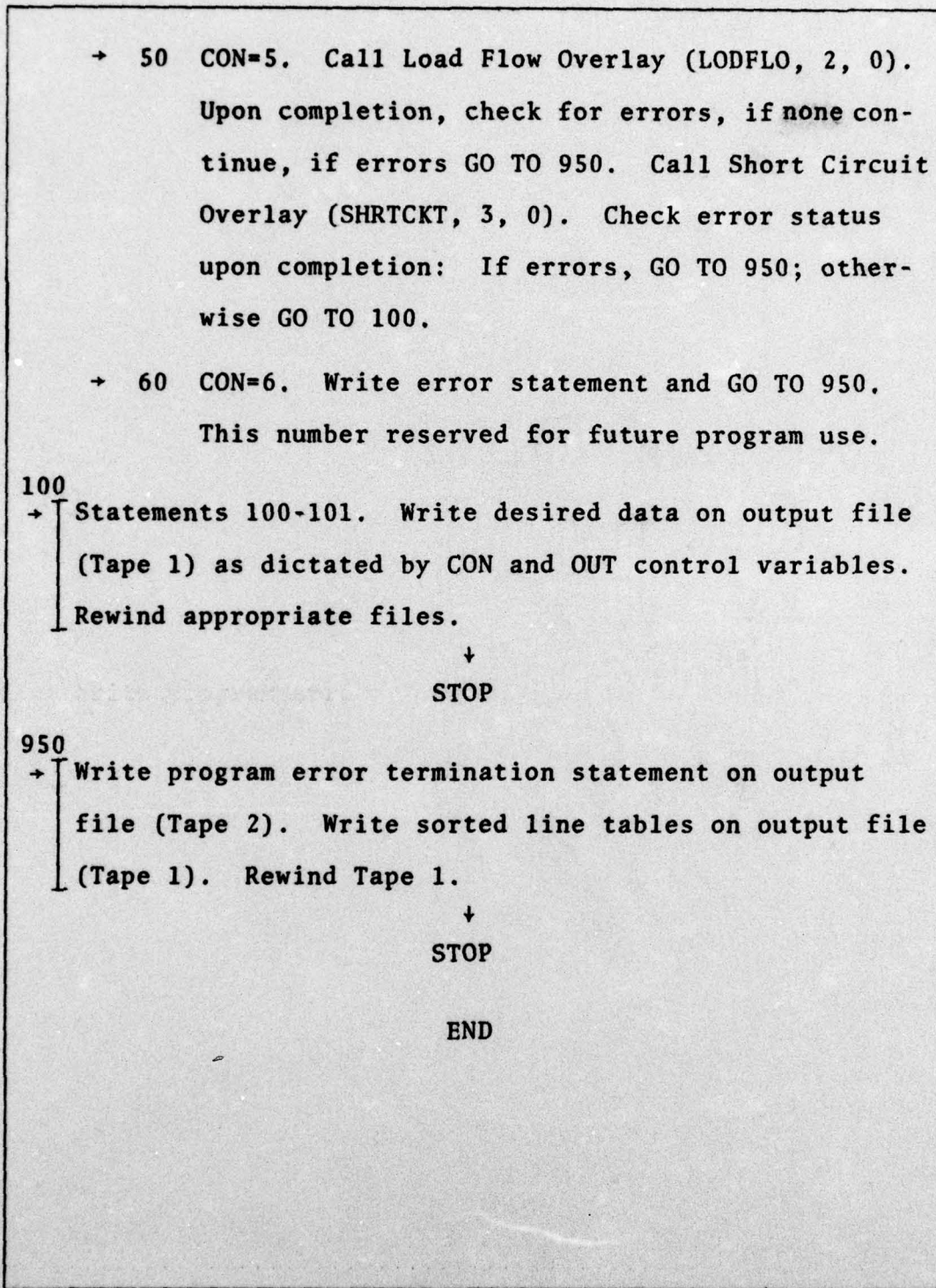


Fig. B-1. (Cont'd)

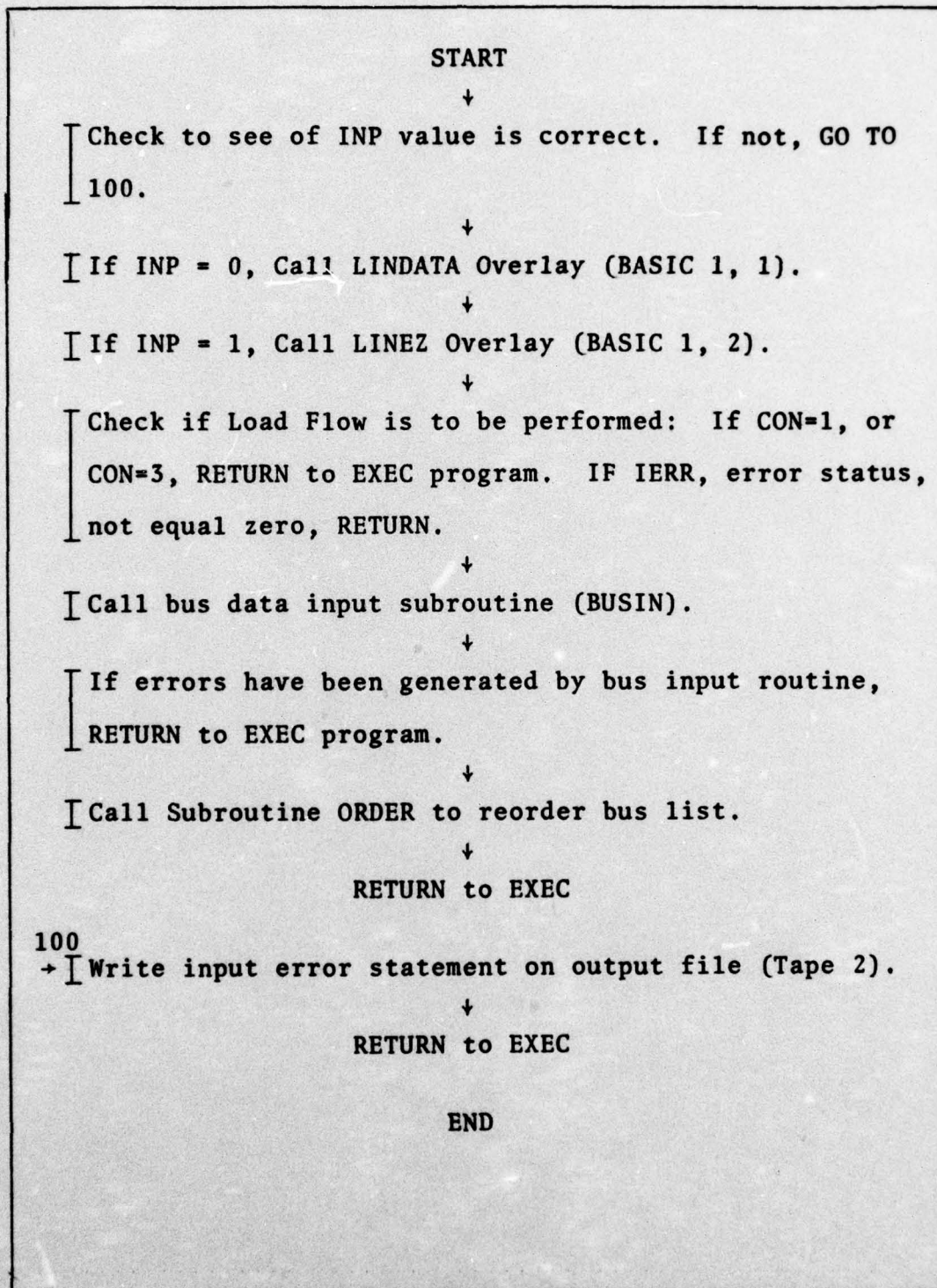


Fig. B-2. INPUT Routine Flow Chart

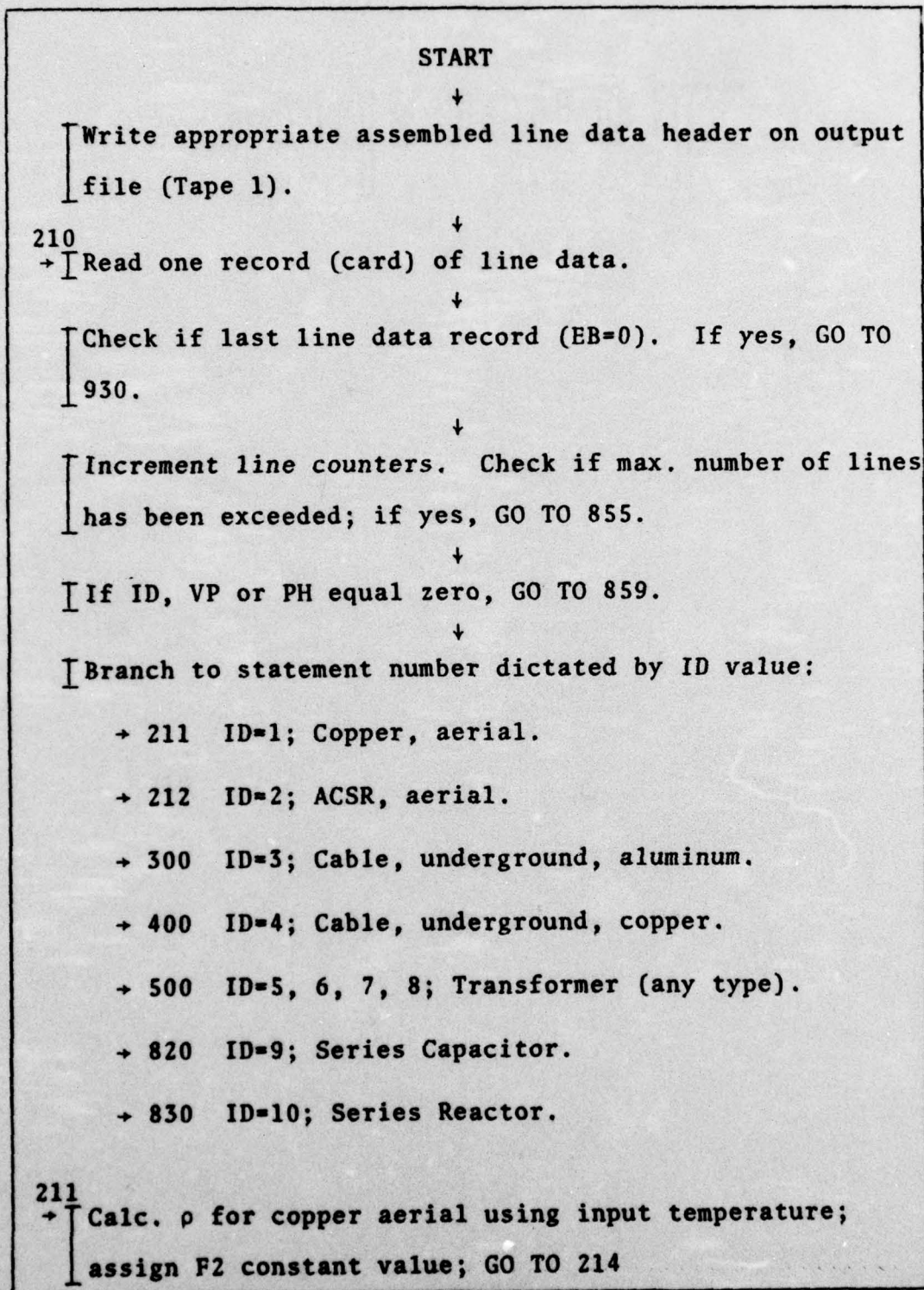


Fig. B-3. LINDATA Routine Flow Chart

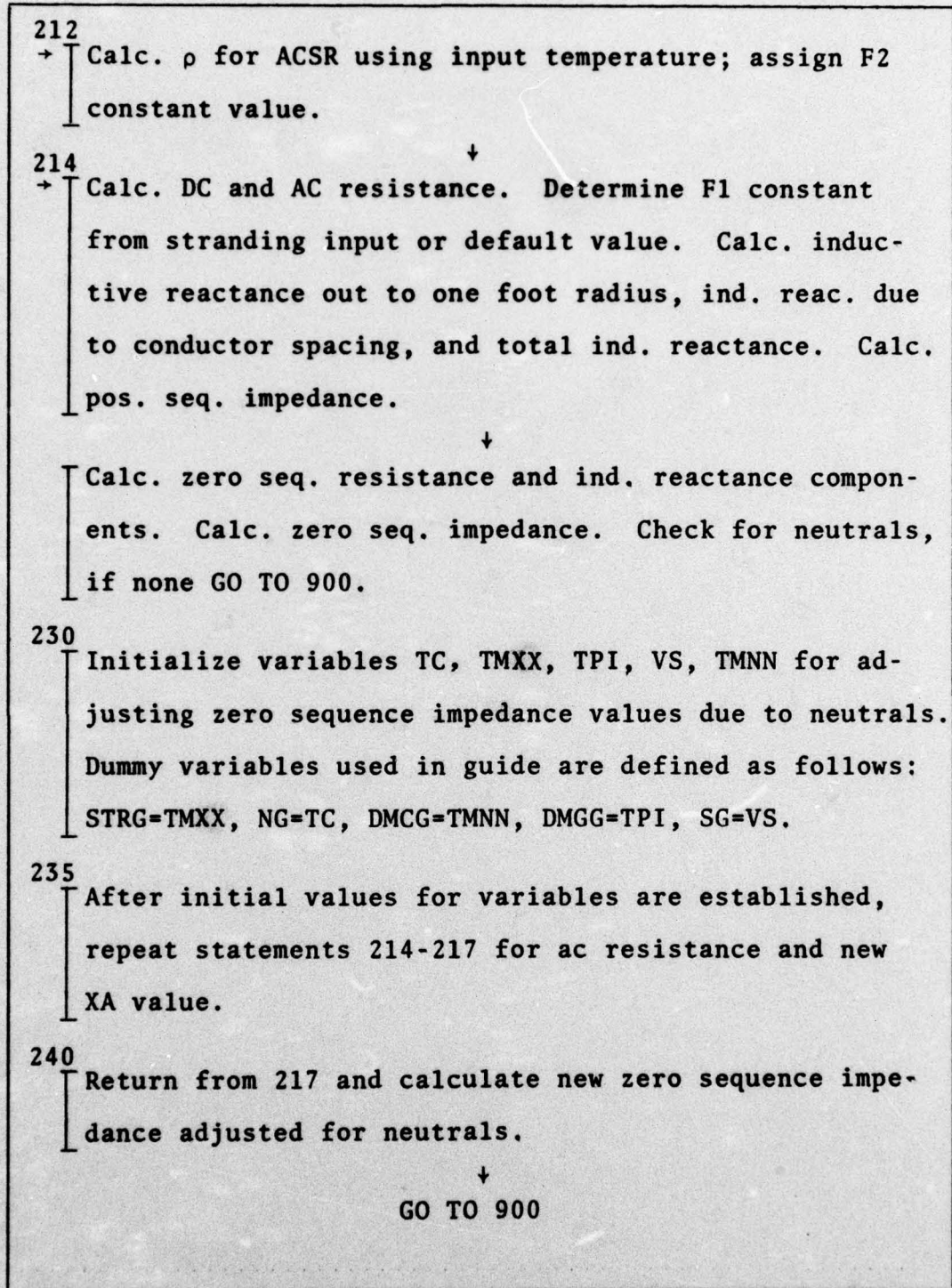


Fig. B-3. (Cont'd)

[Using table look-up (based on cable size), assign pos. and zero seq. impedances for aluminum underground cables.

↓

[Statement numbers 302-339; table of alum. UG cable impedances.

↓

GO TO 900

[Using table look-up (based on cable size), assign pos. and zero seq. impedances for copper cable.

↓

[Statement numbers 402-440; table of copper UG cable impedances.

↓

GO TO 900

[Increment transformer counter (NOTR). Check if max. number transformers has been exceeded; if yes, GO TO 857.

↓

[Branch to statement number dictated by ID (type of transformer).

→ 858 ID=1, 2, 3, 4, 9, 10; error.

→ 860 ID=5, 6, 7 and C greater than 5; error.

→ 518 ID=5; fixed transformer.

Fig. B-3. (Cont'd)

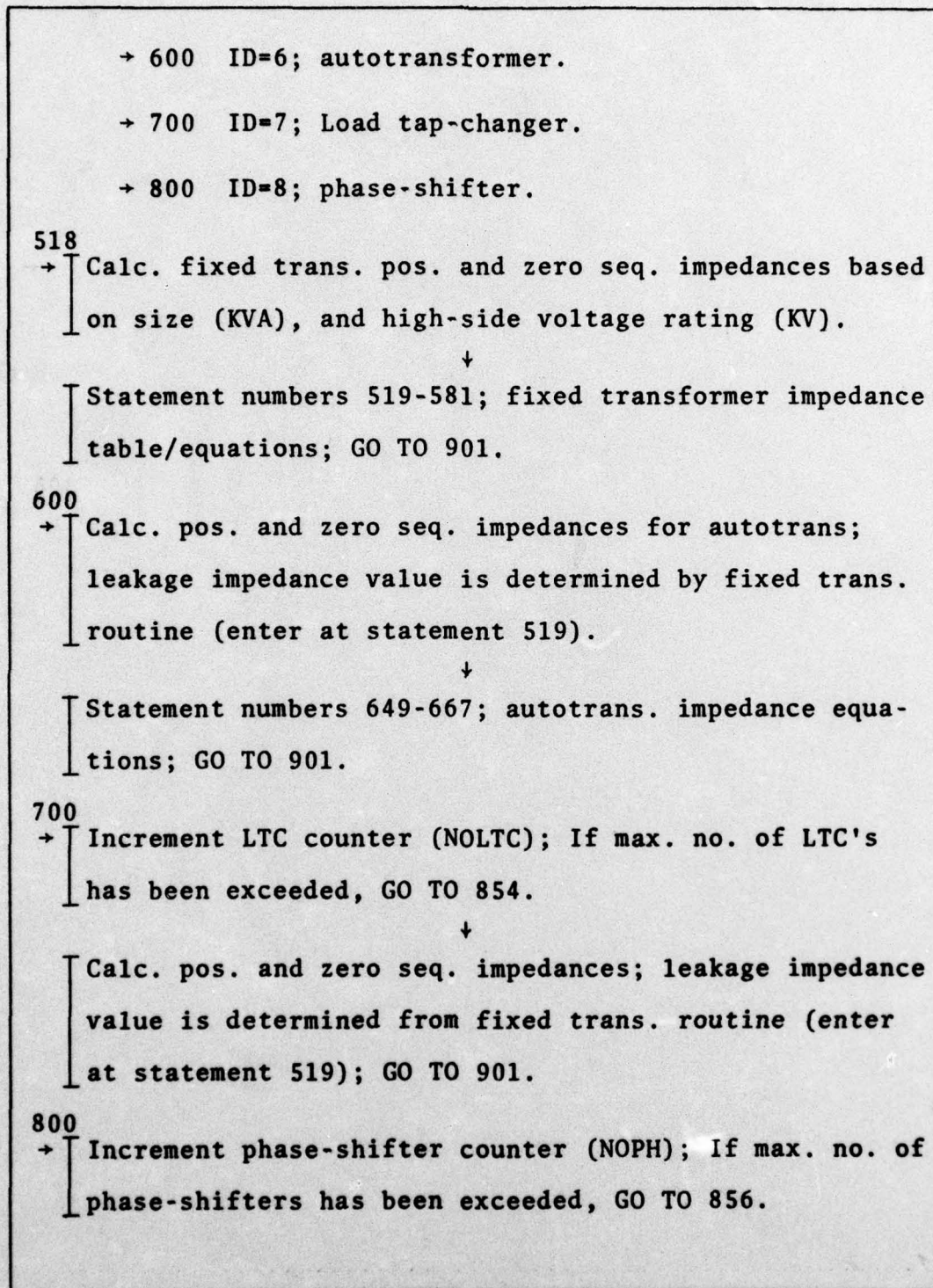


Fig. B-3. (Cont'd)

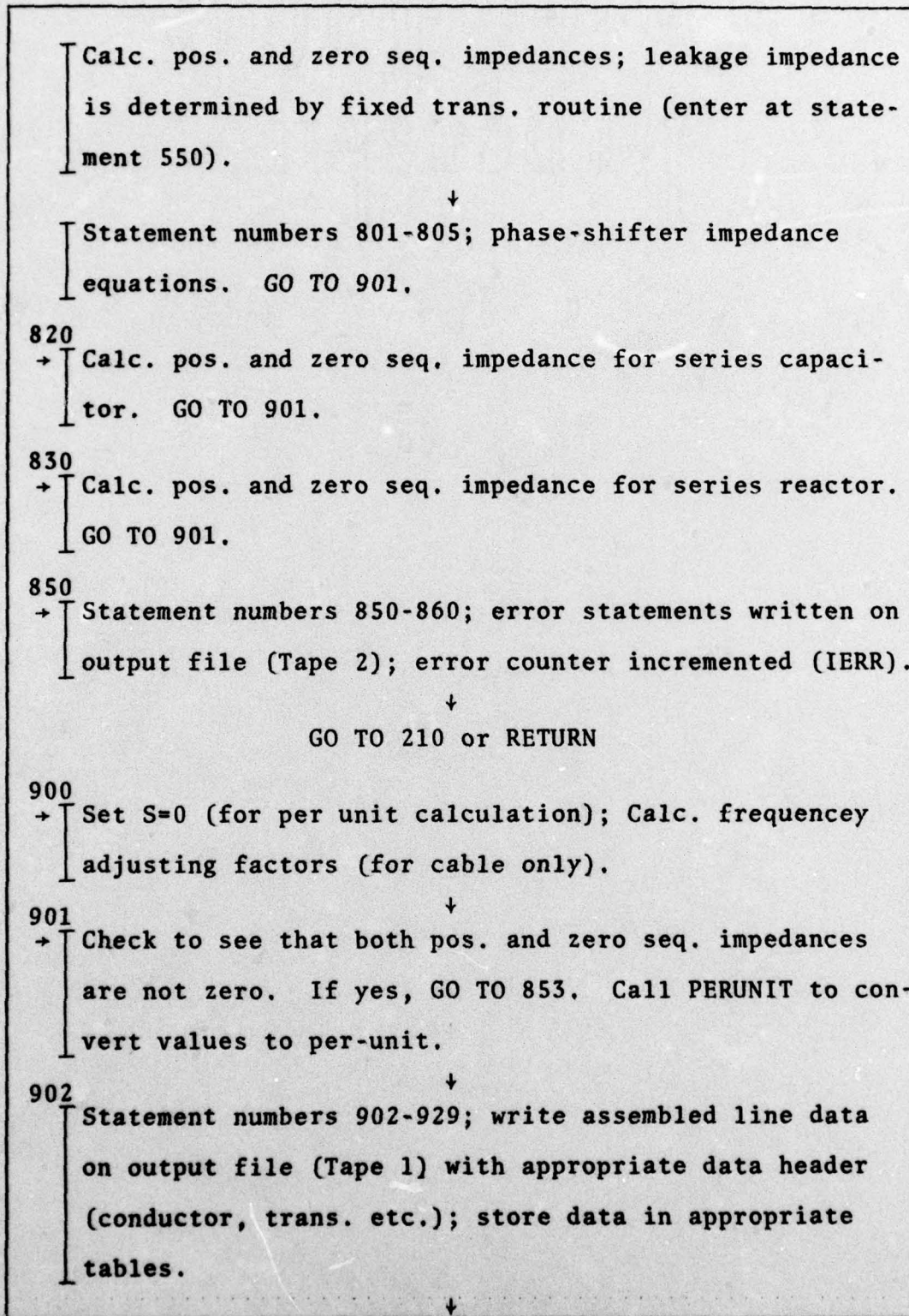


Fig. B-3. (Cont'd)

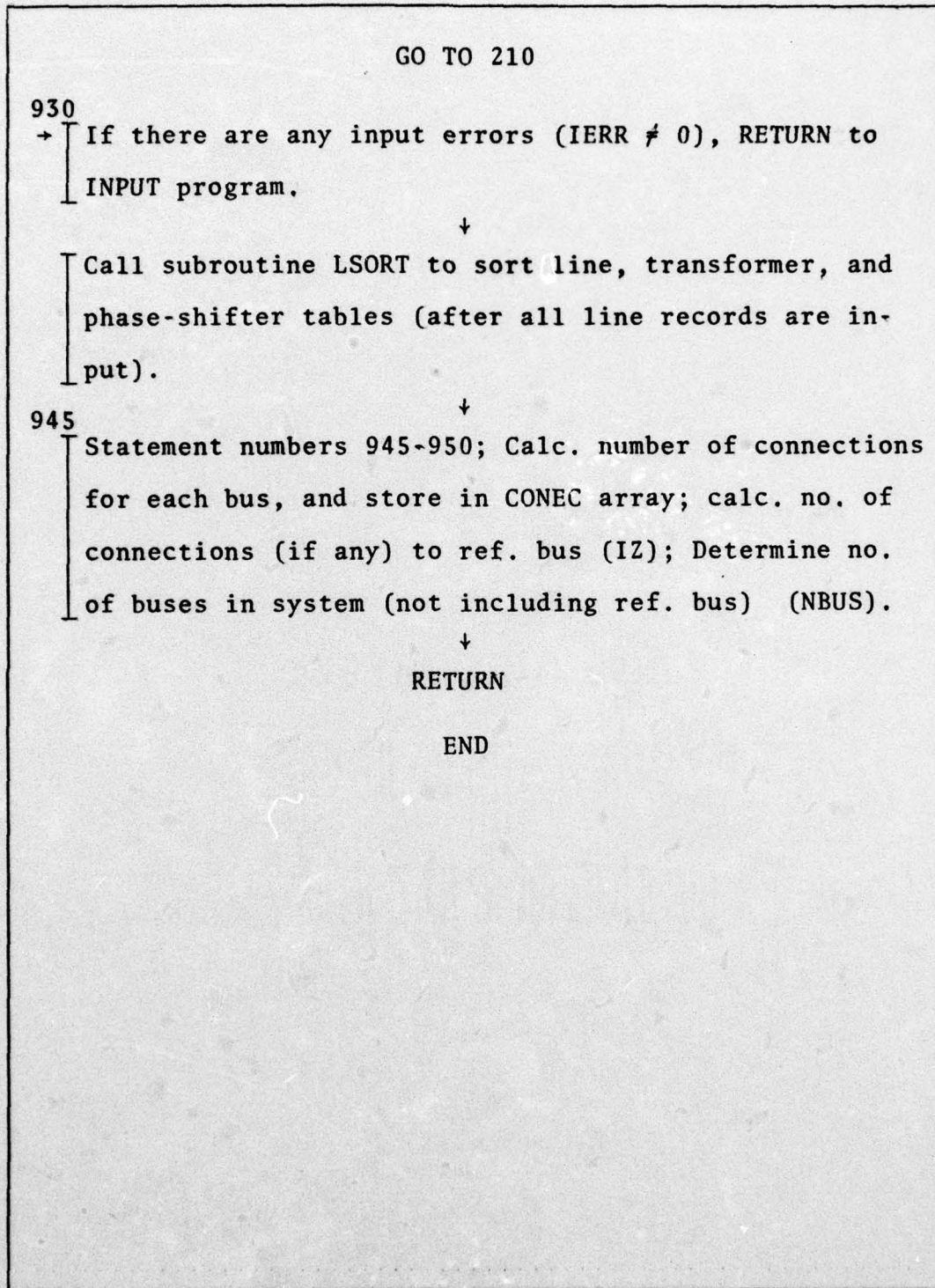


Fig. B-3. (Cont'd)

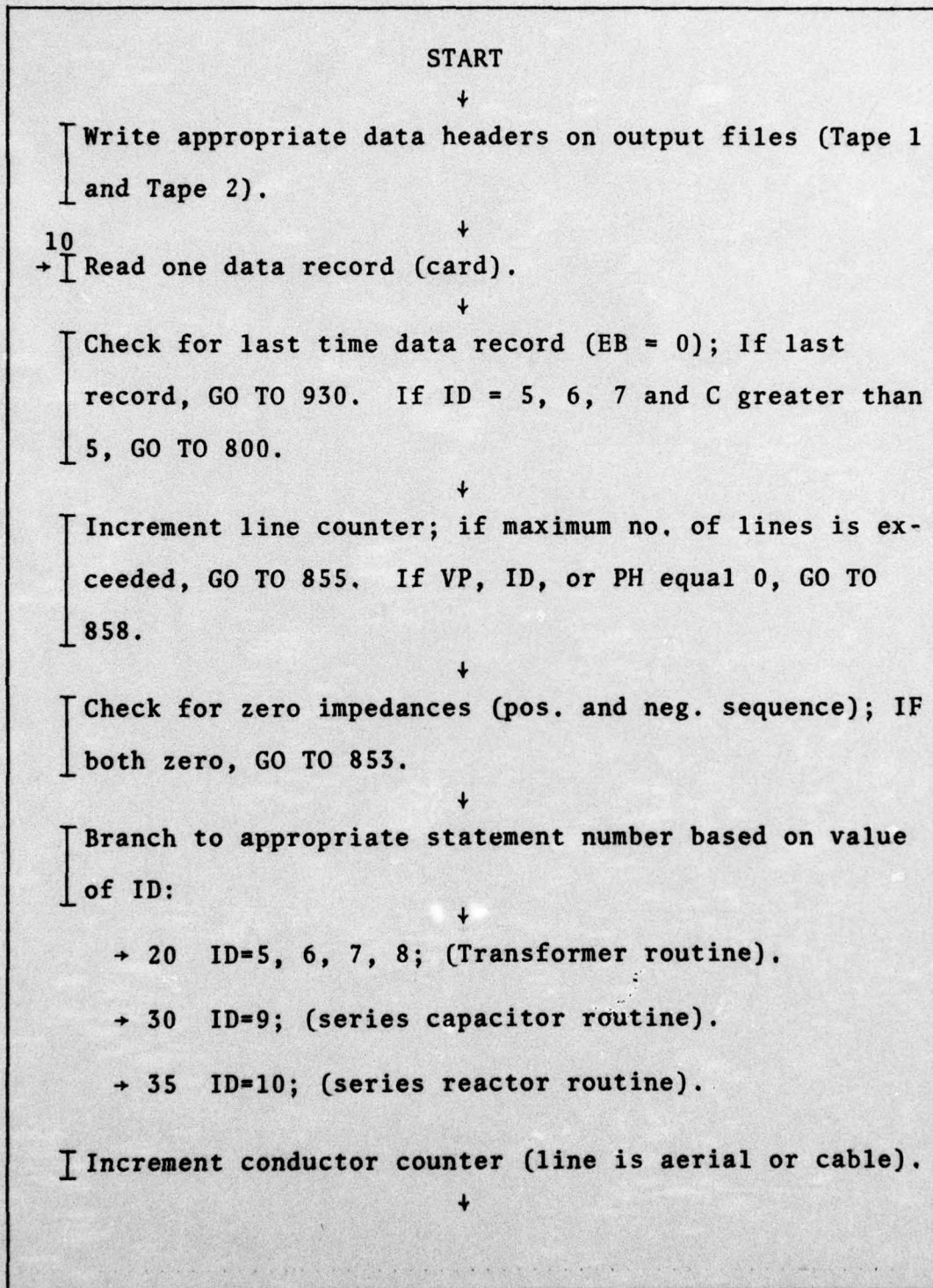


Fig. B-4. LINEZ Routine Flow Chart

Assign pos. and zero seq. impedances and multiply by line length. If zero seq. not specified, zero seq. equal to 2.7 or 3.5 times pos. sequence determined by presence or absence of neutrals.

+

GO TO 900

20

→ Increment transformer counter; if max. no. of transformers exceeded, GO TO 857.

+

Branch to statement number based on ID:

→ 800 ID=1, 2, 3, 4, 9. or 10 (error).

→ 21 ID=5; fixed transformer; increment fixed transformer counter and assign Z and Z_0 ; GO TO 900.

→ 23 ID=7; LTC; increment LTC counter; if max. no. of LTC's exceeded, GO TO 854; assign Z and Z_0 . GO TO 900.

→ 24 ID=8; Phase-shifter; increment phase-shifter counter; if max. no. of phase-shifters exceeded, GO TO 856; assign Z and Z_0 . GO TO 900.

30

→ Increment capacitor counter; assign Z and Z_0 ; GO TO 900.

35

→ Increment reactor counter; assign Z and Z_0 ; GO TO 900.

Fig. B-4. (Cont'd)

```
800
→ | Error statements.
900
→ | Statements 900-920. Write assembled data on output
   | file (Tape 1) with appropriate header, in ohms or per-
   | unit values; convert data to per-unit values if neces-
   | sary.
   |
   |           ↓
   |         GO TO 10
930
→ | Statements 930-960. If there are input errors (IERR
   | ≠ 0) RETURN. Sort line, transformer, and phase-shif-
   | ter tables into order by ascending bus numbers (call
   | subroutine LSORT). Do 950 Loop: Calculate no. of con-
   | nections for each bus and store in CONEC array; count
   | no. of connections to ref. bus (if any) and save as
   | IZ: Determine no. of buses in system (not including
   | ref. bus) and save as NBUS.
   |
   |           ↓
   |         RETURN
   |
   |         END
```

Fig. B-4. (Cont'd)

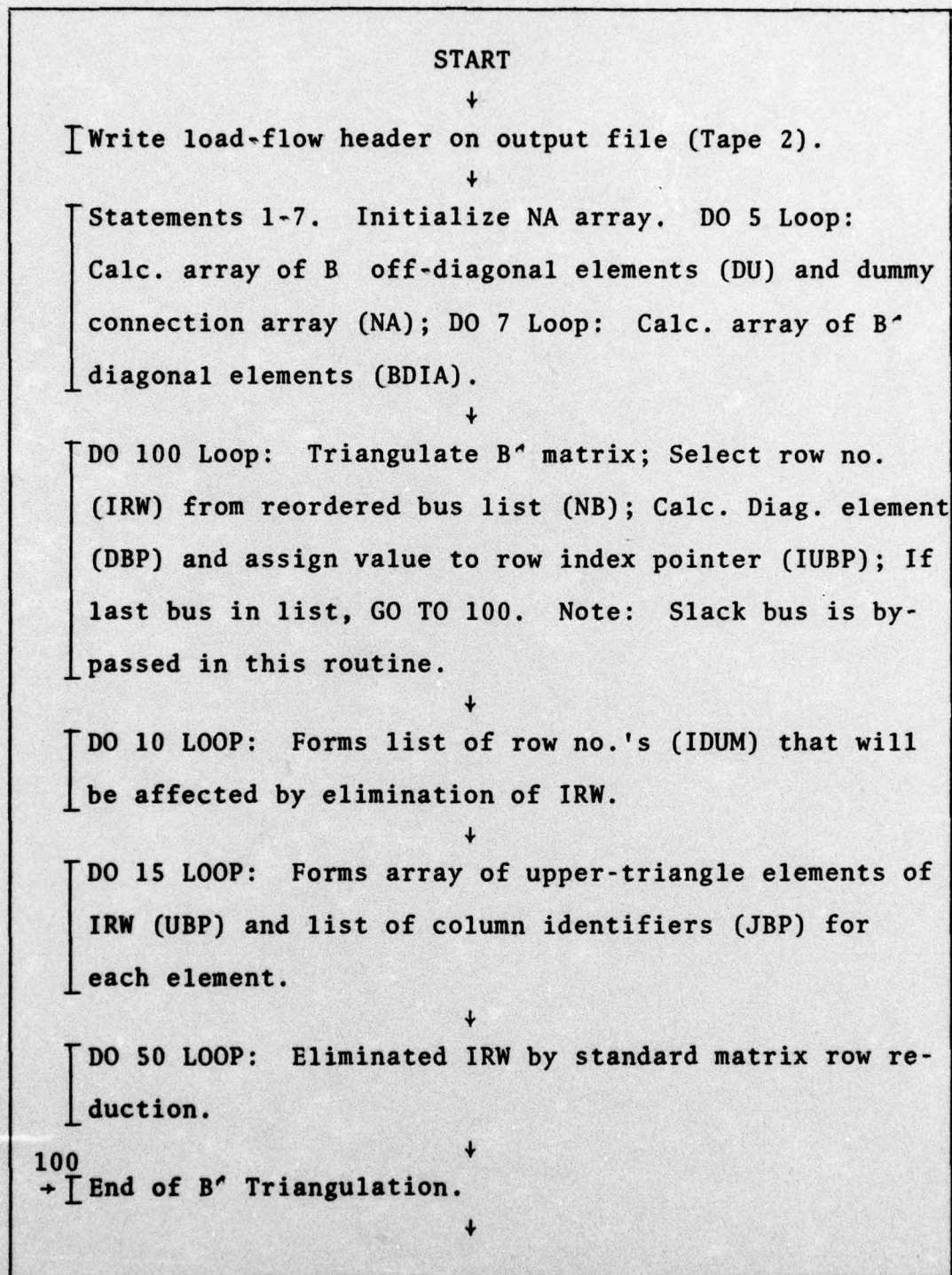


Fig. B-5. Load Flow (FDLFLOW)

```

DO 105 Loop: Calc. array of B'' off-diag. elements
(DU) and dummy connection array (NA); DO 107 Loop:
Calc. array of B'' diag. elements (BDIA).
↓
DO 200 LOOP: Triangulate B'' matrix; Select row no.
(IRW) from reordered bus list (NB); Calc. Diag. ele-
ment (DBPP) and assign value to row index pointer
(IUBPP); if last bus in list, GO TO 100. Note 1:
Slack bus and PV buses are bypassed in this routine.
Note 2: interior loops perform same functions as in
DO 100 Loop above.
↓
200 → End of B'' Triangulation.
↓
Restore dummy NA array; initialize LIST array.
↓
211 → Entry point for ΔP - Δθ solution routine.
↓
DO 250 Loop: Calc. ΔP/V array (DLP) for each bus in
B'; If |ΔP/V| > tolerance, set KP = 1.
↓
If KP = 0 (Converged) GO TO 400. If max. no. of itera-
tions has been exceeded, GO TO 500.
↓
DO 270 Loop, DO 275 Loop and DO 285 Loop: Direct soln.
of [ΔP/V] = [B']·[Δθ] equation for [Δθ]. Note: Matrix
operations performed on DLP array transform it into the
Δθ solution array.

```

Fig. B-5. (Cont'd)

```

[ DO 290 Loop: Updates bus angle array (ANG) by adding
  Δθ for each bus.
      ↓
[ Increment ΔP-Δθ iteration counter.
      ↓
291 → [ Entry point for ΔQ-ΔV solution routine.
      ↓
[ DO 350 loop: Calc. ΔQ/V array (DLQ) for each bus in
  B''; If |ΔQ/V| > tolerance KQ = 1.
      ↓
[ IF KQ = 0 (Converged) GO TO 401. If max. no. of itera-
  tions has been exceeded, GO TO 500.
      ↓
[ DO 370 loop, DO 375 loop, and DO 385 loop: Direct so-
  lution of ΔQ/V = [B''] [ΔV] equation for [ΔV]. Note:
  Matrix operations performed on DLQ array transform it
  into the ΔV solution array.
      ↓
[ DO 390 and DO 388 loops: Update bus voltage magnitude
  array (V) by adding ΔV for each bus.
      ↓
[ Increment ΔQ-ΔV iteration counter.
      ↓
[ Call subroutine LIMIT (Check for PV bus Q-limit viola-
  tions.)
      ↓
      GO TO 211

```

Fig. B-5. (Cont'd)

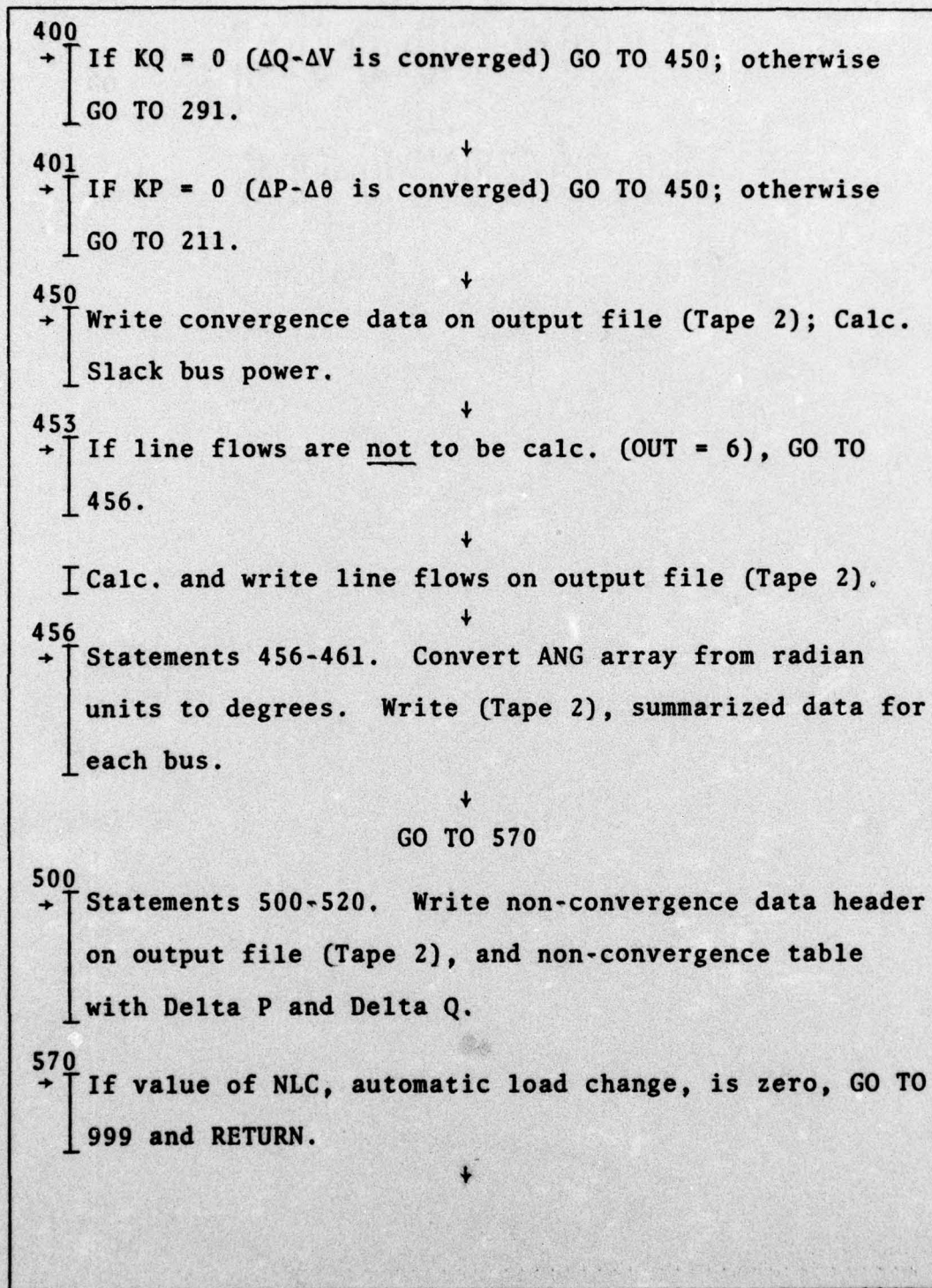


Fig. B-5. (Cont'd)

580
| Statements 580-602 contain the load change routine.
| Read data cards for load changes and write load change
| information on Tape 2. Number of bus changes is NC.
| NCC is number of times load change routine used. Data
| read in as in BUSIN routine with variables IDB, BUSNAME,
| V, ANG, P, Q, QMIN, QMAX. Arrays are adjusted with
| new information. Re-calculate load flow results by
| starting at 210 again.

900
| Statements 900-980 are error messages.

999
| RETURN.

END

Fig. B-5. (Cont'd)

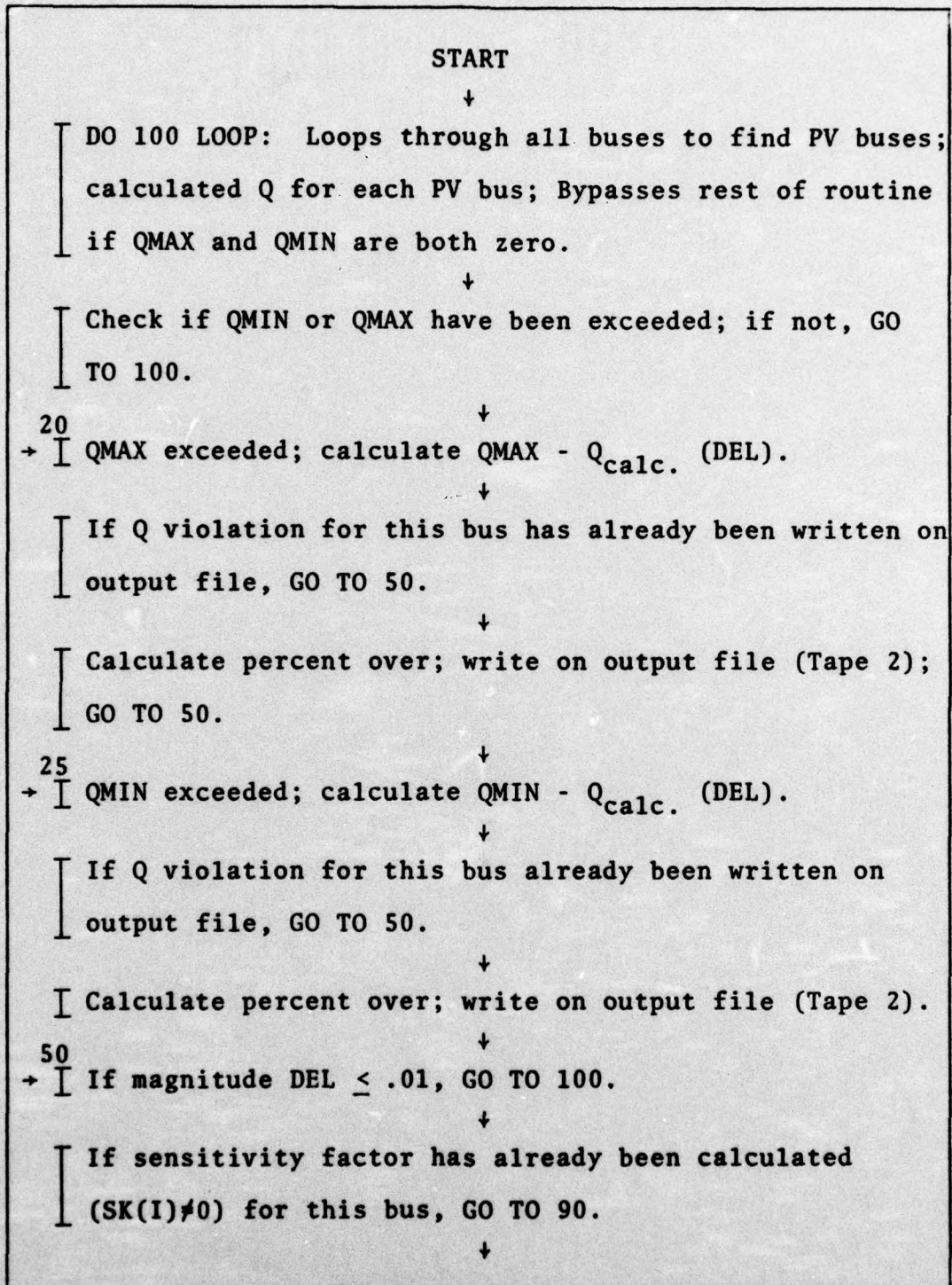


Fig. B-6. LIMIT Subroutine Flow Chart

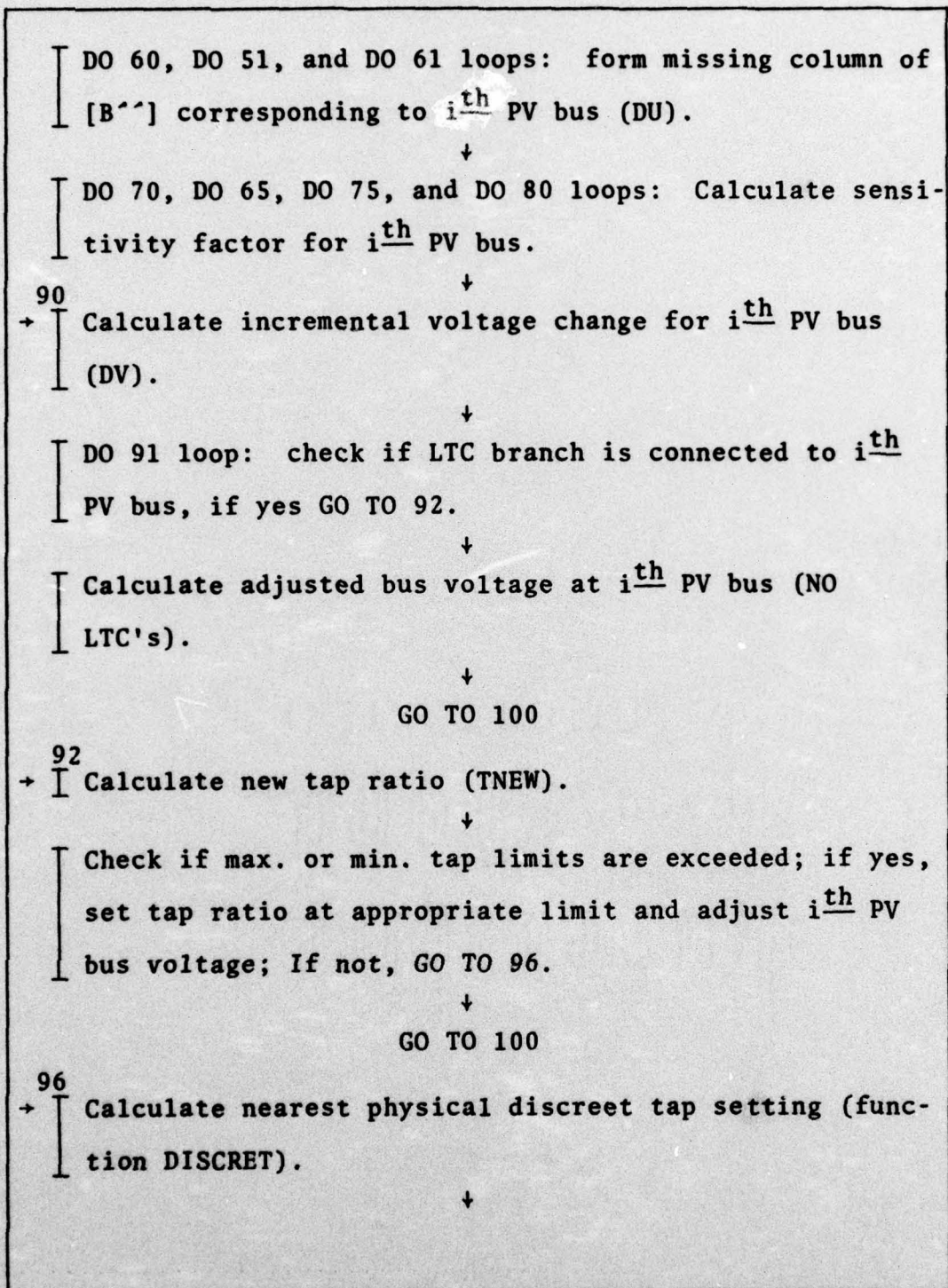


Fig. B-6. (Cont'd)

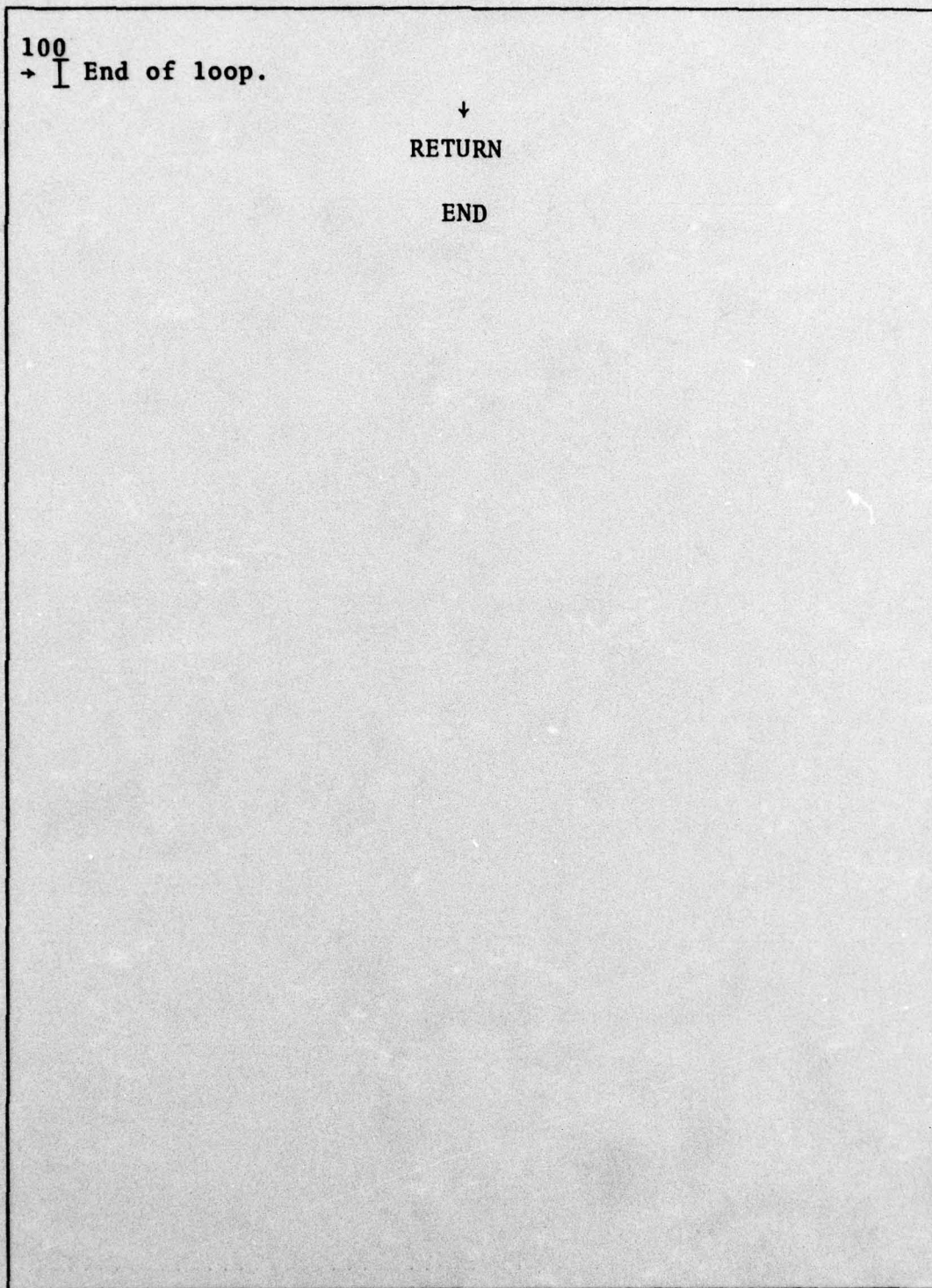


Fig. B-6. (Cont'd)

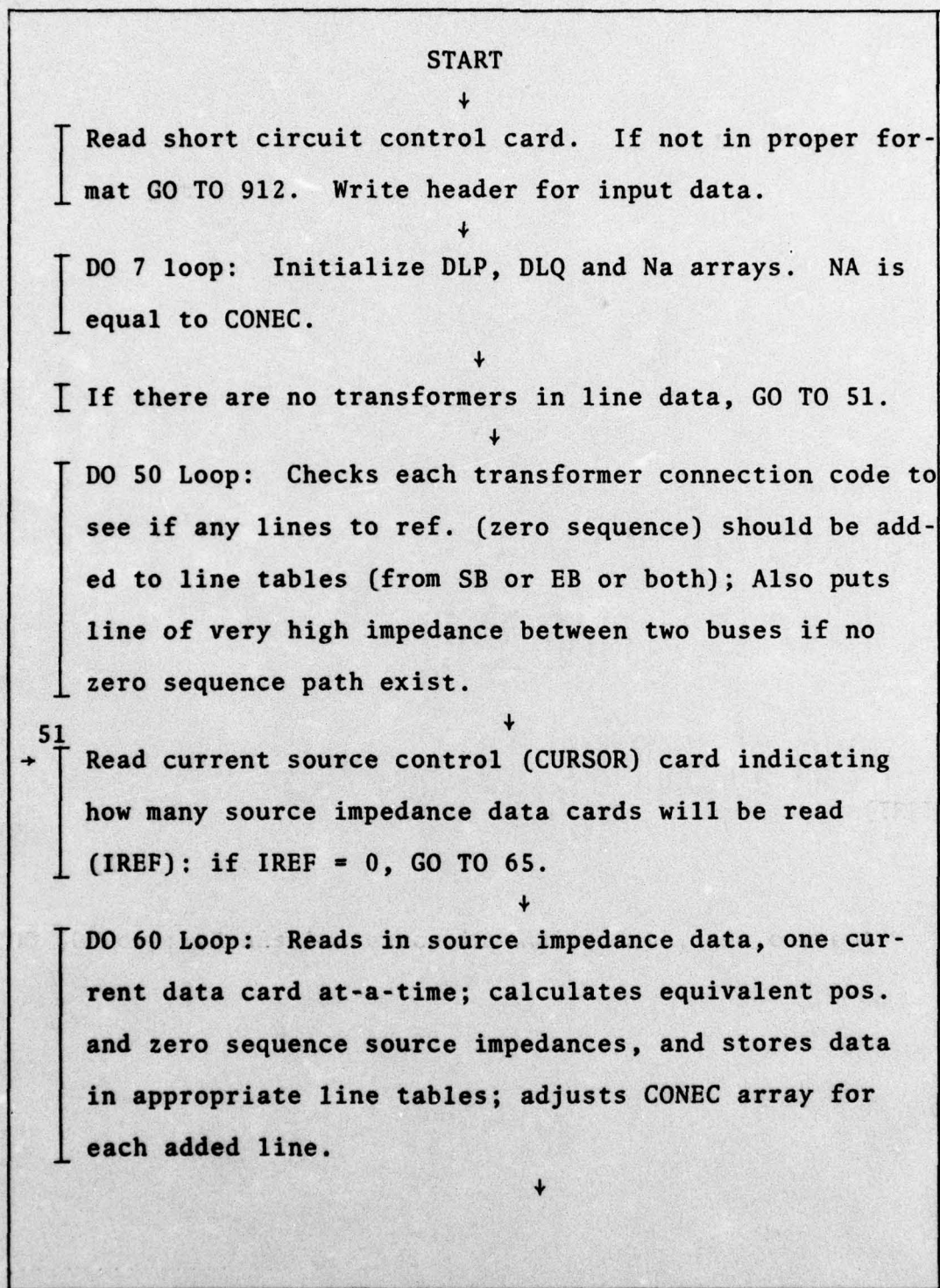


Fig. B-7. Short Circuit (FAULT) Routine

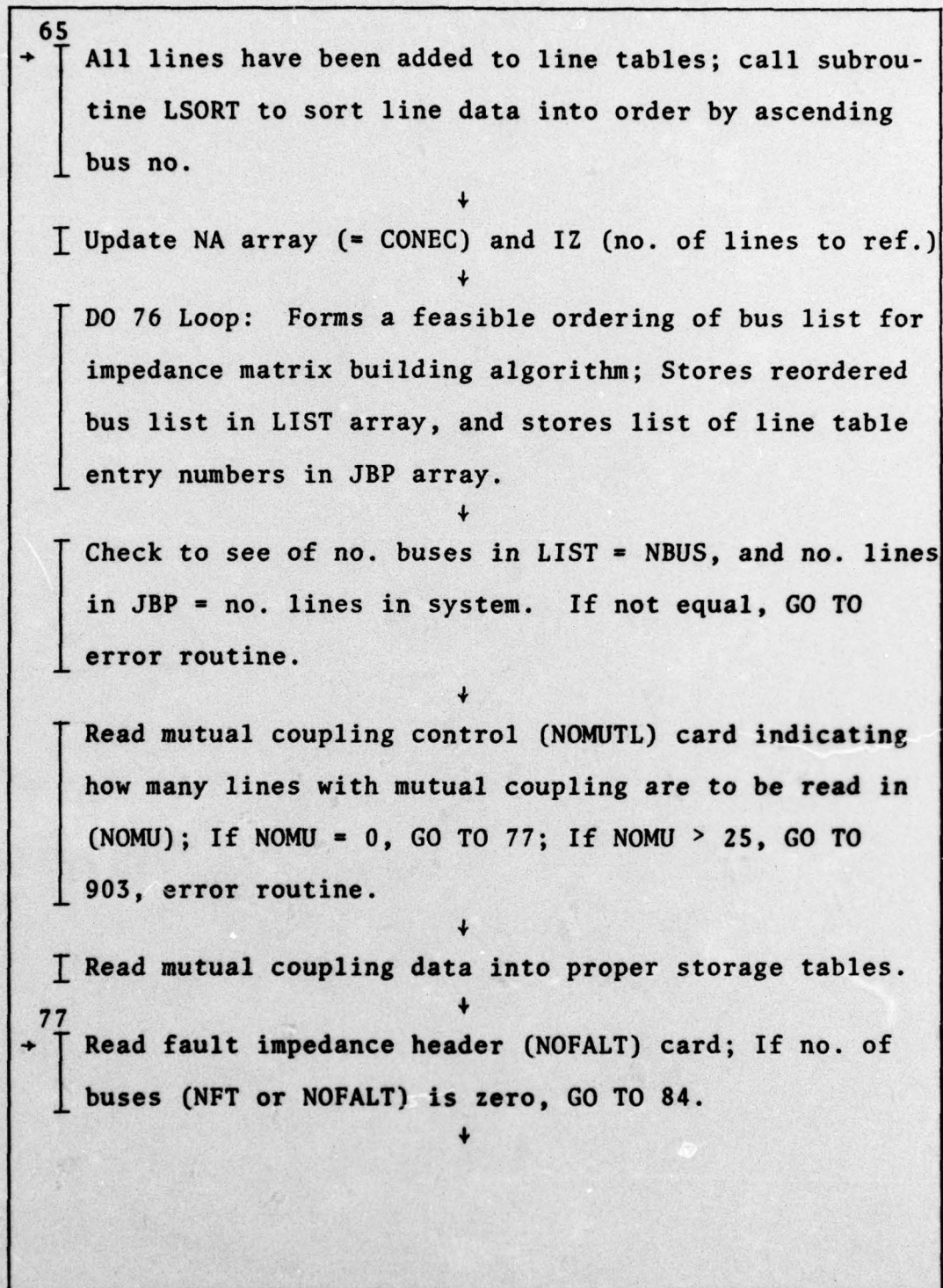


Fig. B-7. (Cont'd)

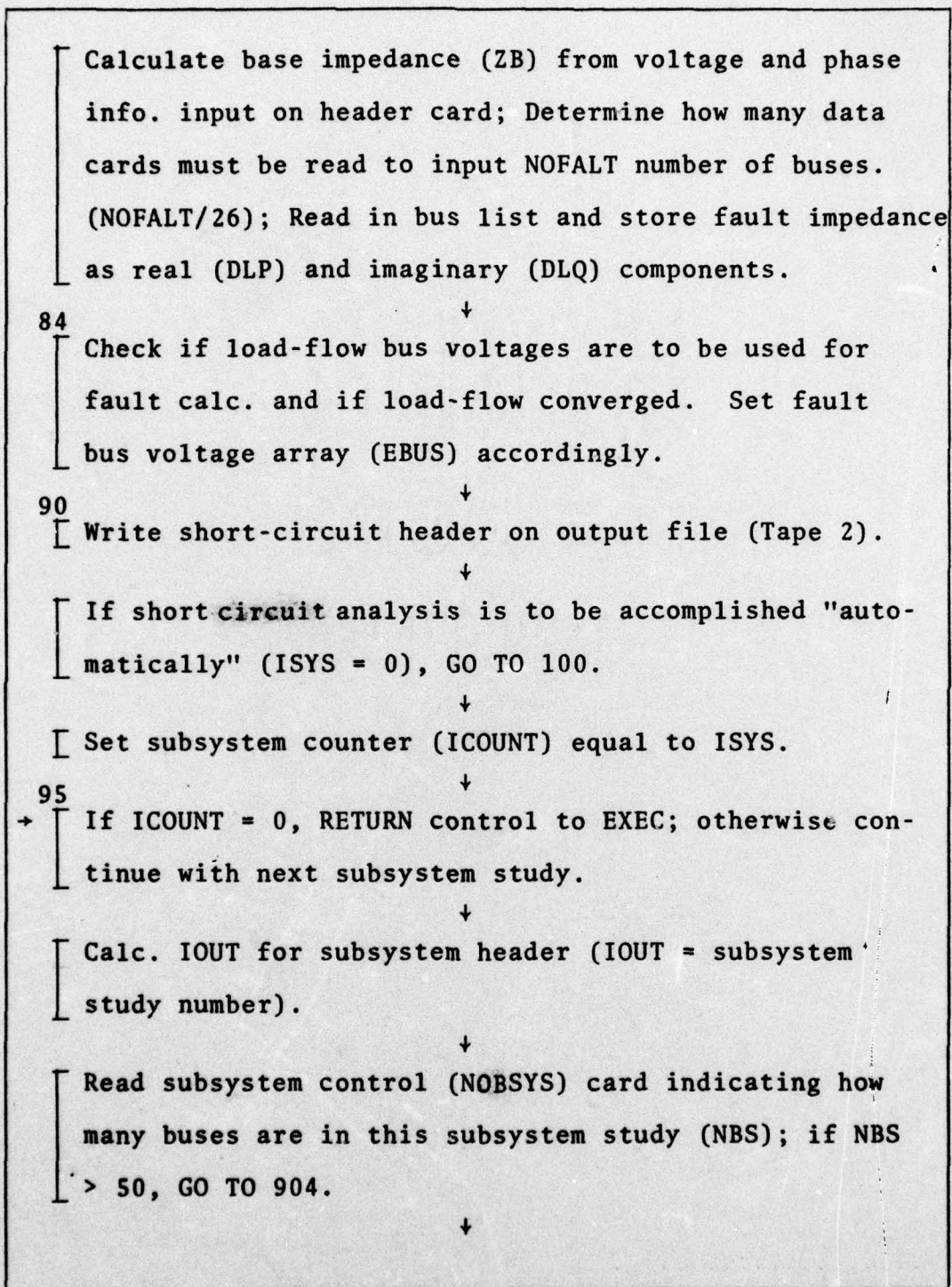


Fig. B-7. (Cont'd)

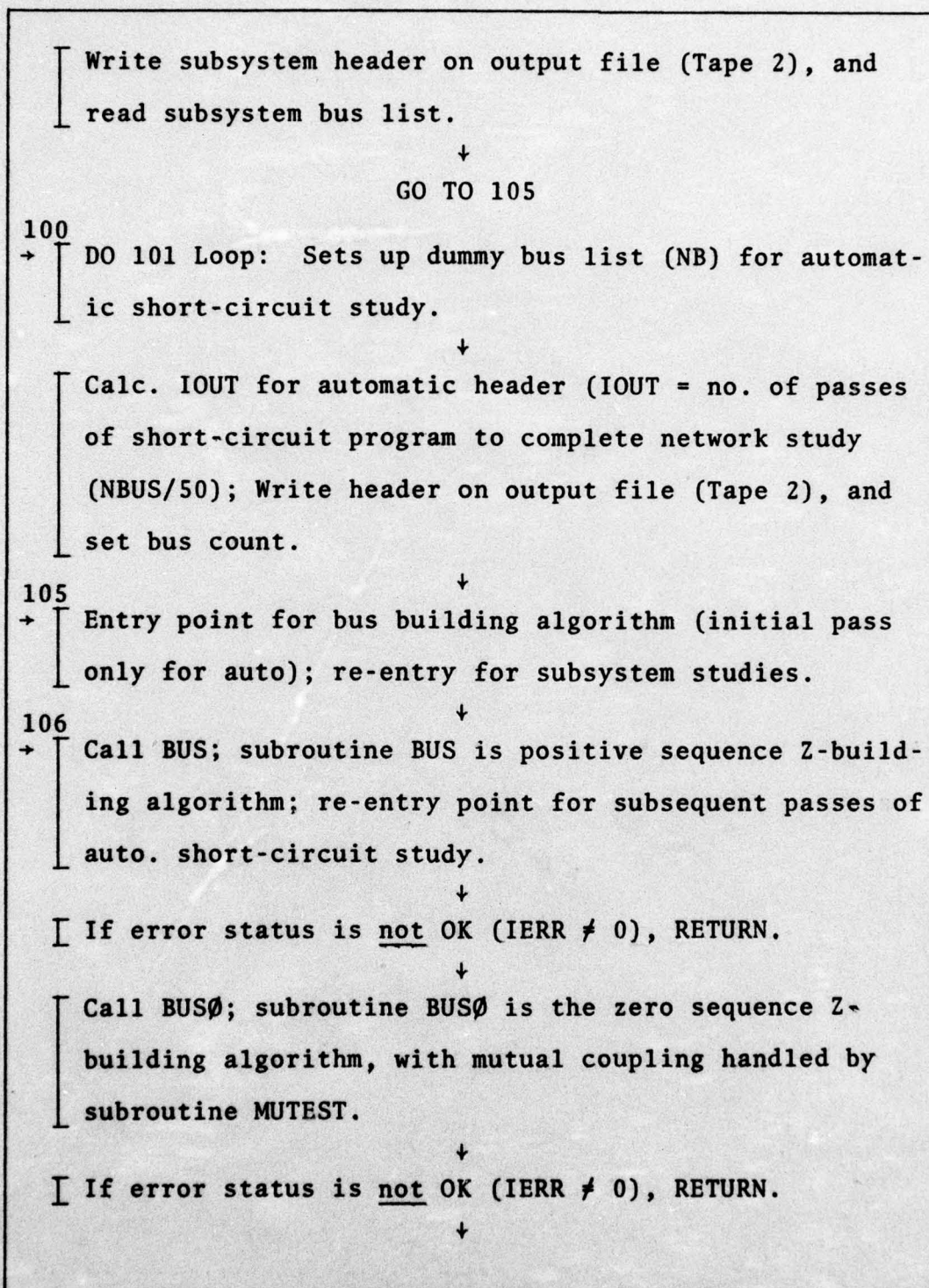


Fig. B-7. (Cont'd)

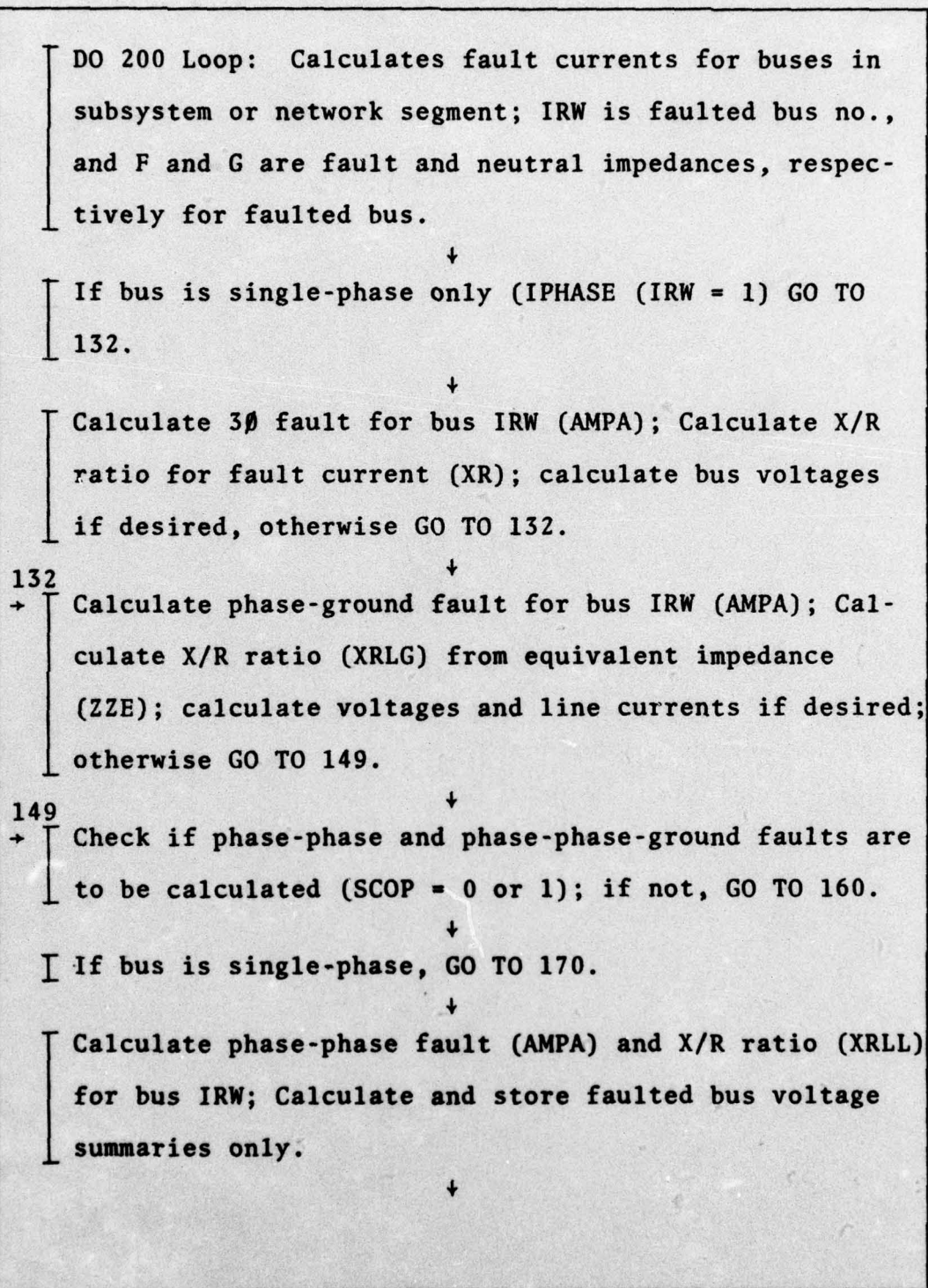


Fig. B-7. (Cont'd)

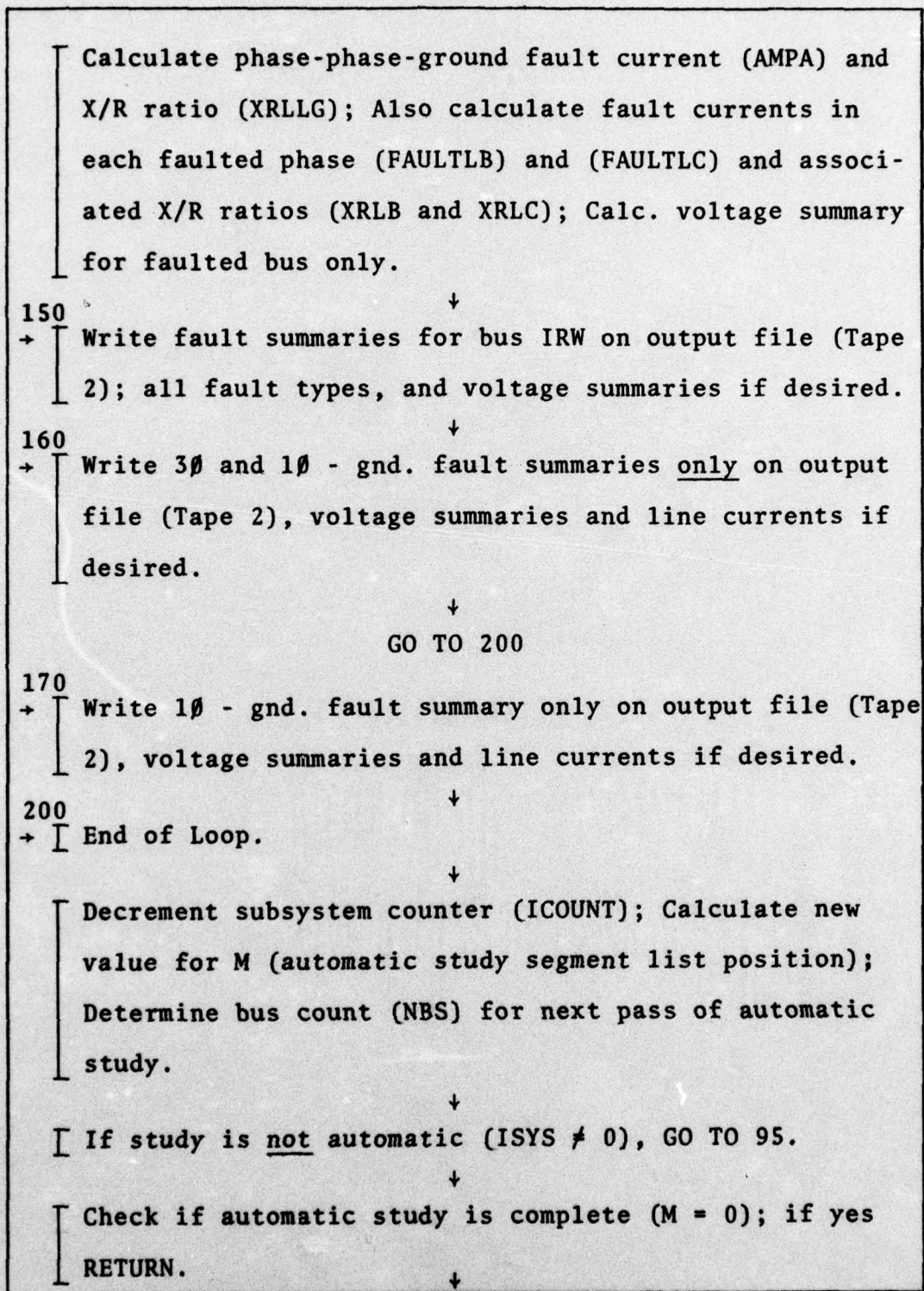


Fig. B-7. (Cont'd)

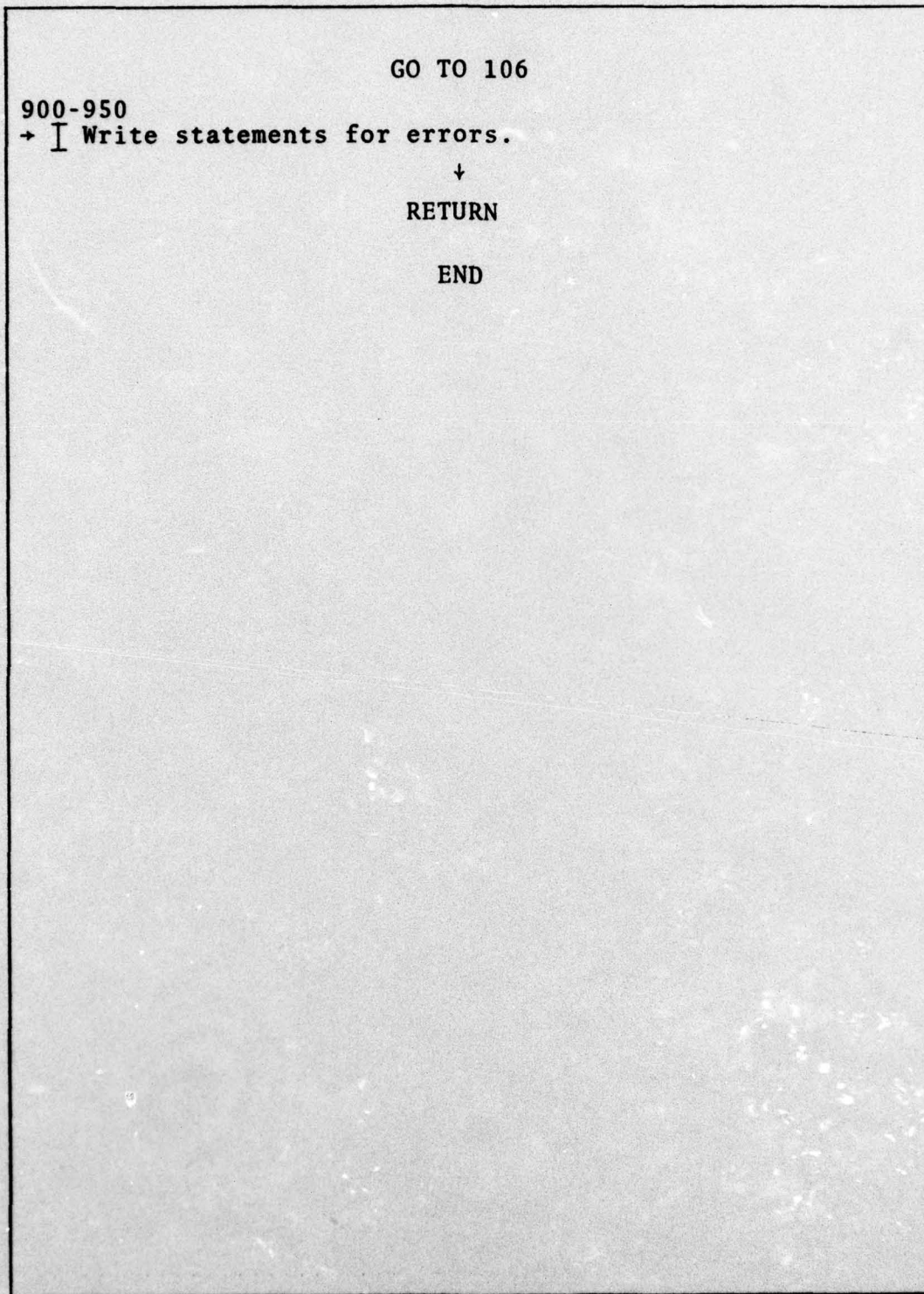


Fig. B-7. (Cont'd)

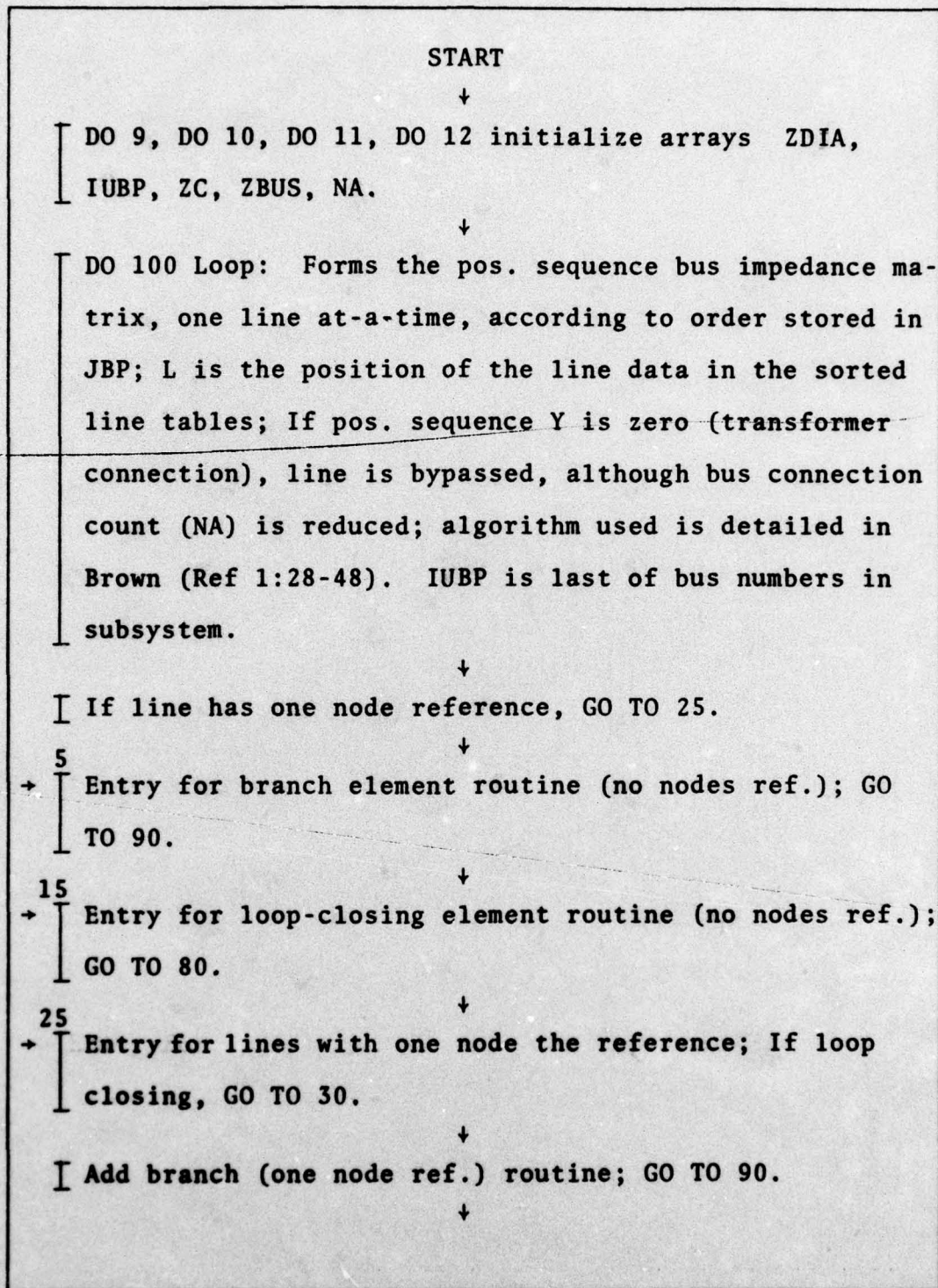
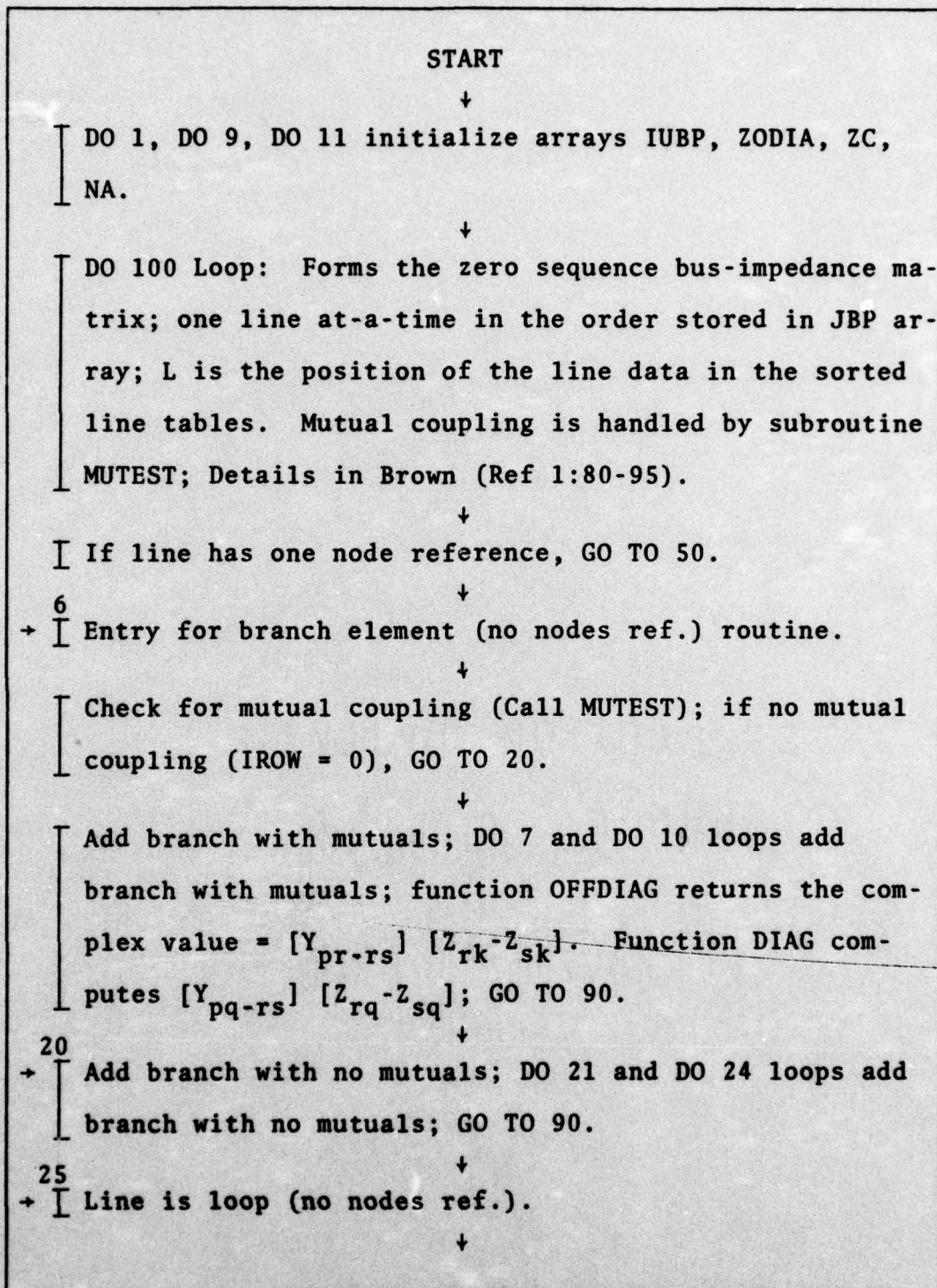


Fig. B-8. BUS Subroutine Flow Chart

Fig. B-9. BUS \emptyset Subroutine Flow Chart

```

      | Check for mutual coupling, (Call MUTEST); if no mutual
      | coupling (IROW = 0) GO TO 41.
      |
      |   ↓
      | Add loop with mutuals; DO 26 and DO 30 Loops add loop
      | with mutuals; as above, OFFDIAG computes  $[Y_{pq-rs}]$ 
      |  $[Z_{rk}-Z_{sk}]$ , and DIAG computes  $[Y_{pq-rs}] [Z_{r-loop}-Z_{s-loop}]$ ;
      | GO TO 80.
      |
      |   ↓
      | 41 → | Add loop with no mutuals; DO 42 and DO 45 loops add
      |     | loop element with no mutuals; GO TO 80.
      |     |
      |     |   ↓
      |     | 50 → | Entry for line element with one node the ref; if loop-
      |     |     | closing, GO TO 60.
      |     |     |
      |     |     |   ↓
      |     |     | | Add branch with one node ref; GO TO 90.
      |     |     |   ↓
      |     |     | 60 → | Line element is loop closing; add loop without mutuals
      |     |     |     | (one node ref.)
      |     |     |     |   ↓
      |     |     |     | 80 → | Eliminate loop axis by Kron reduction (DO 84 and DO 85
      |     |     |     |     | loops).
      |     |     |     |     |   ↓
      |     |     |     |     | 90 → | Network reduction routine; connection count for buses
      |     |     |     |     |     | on each end of line is decremented; If no more connec-
      |     |     |     |     |     | tions are to processed (NA (XX) = 0), and the bus is
      |     |     |     |     |     | not in the subsystem study area, it is eliminated by
      |     |     |     |     |     | subroutine SWAPZØ; If swapped out, the remaining net-
  
```

Fig. B-9. (Cont'd)

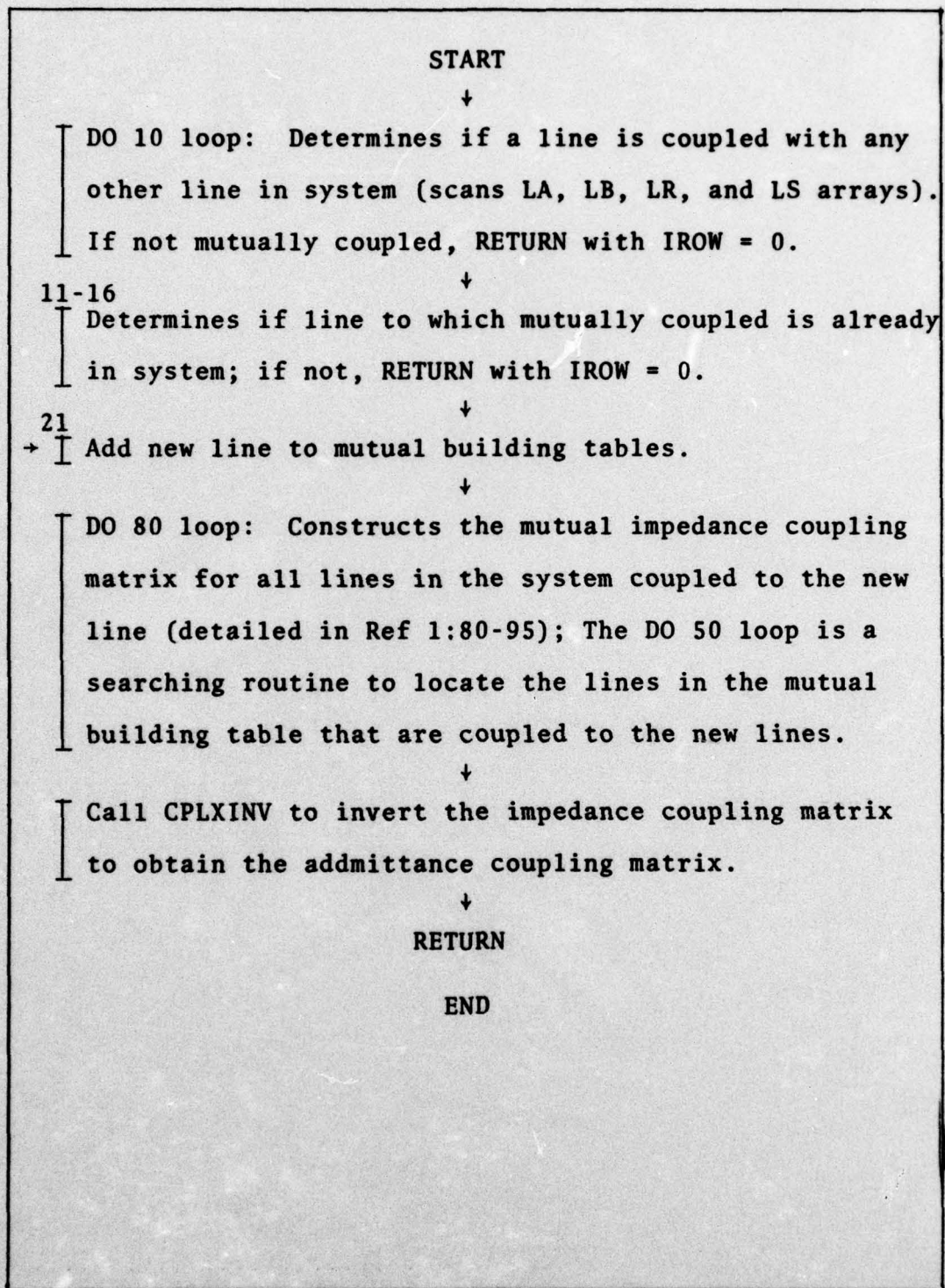


Fig. B-10. MUTEST Subroutine Flow Chart

Appendix C. Program Variables

This appendix was developed to further document the PDSAP program. All variables and arrays listed in the COMMON statements are indicated as to the overlay source and other overlays where used. Tables contain the variables or arrays listed by COMMON block. Each table lists the source overlay and a brief explanation of the significance of the variable or array. Additionally, figures depict the data flow from the source overlay to the other overlays. When COMMON variables are not used outside the source overlay, the variable is omitted in the corresponding figure. For this reason, there is no figure for the COMMON ZERO arrays or COMMON ZCONST variables as all are unique to the source overlay.

Variables and arrays not listed in this appendix are unique to each overlay and are traceable to their origin with considerably less effort than the COMMON variables.

Table C-I.

Common COMA Variables

Variable	Overlay Source	Definition/Use/Comments
BKVA	Basic,0,0	Base KVA
BKVA1	Basic,0,0	Base KVA
CHG	Basic, 0,0	Change, not used presently.
CON	Basic, 0, 0	Program control, selects program functions to be used.
F	Basic,0,0	Frequency
INP	Basic,0,0	Input, selects LINEZ or LINDATA routines
ISYS	Shrtckt,3,0	Number of subsystems in short circuit routine
LODOP	Basic,0,0	Used as index in Sub-routine LIMIT.
MAXLTC	Basic,0,0	Maximum number of Tap Changing Under Load transformers.
MAXPH	Basic,0,0	Maximum number of phase shifting transformers
MAXTR	Basic,0,0	Maximum number of transformers allowed.
NMAX	Basic,0,0	Number of lines allowed times 2.
OUT	Basic,0,0	Output, controls output printouts.
SCOP	Shrtckt,3,0	Short circuit control, selects base voltages and type faults.
T	(1)Basic,0,0 (2)Lodflo,2,0	(1)Temperature (2)Transformer Tap Setting

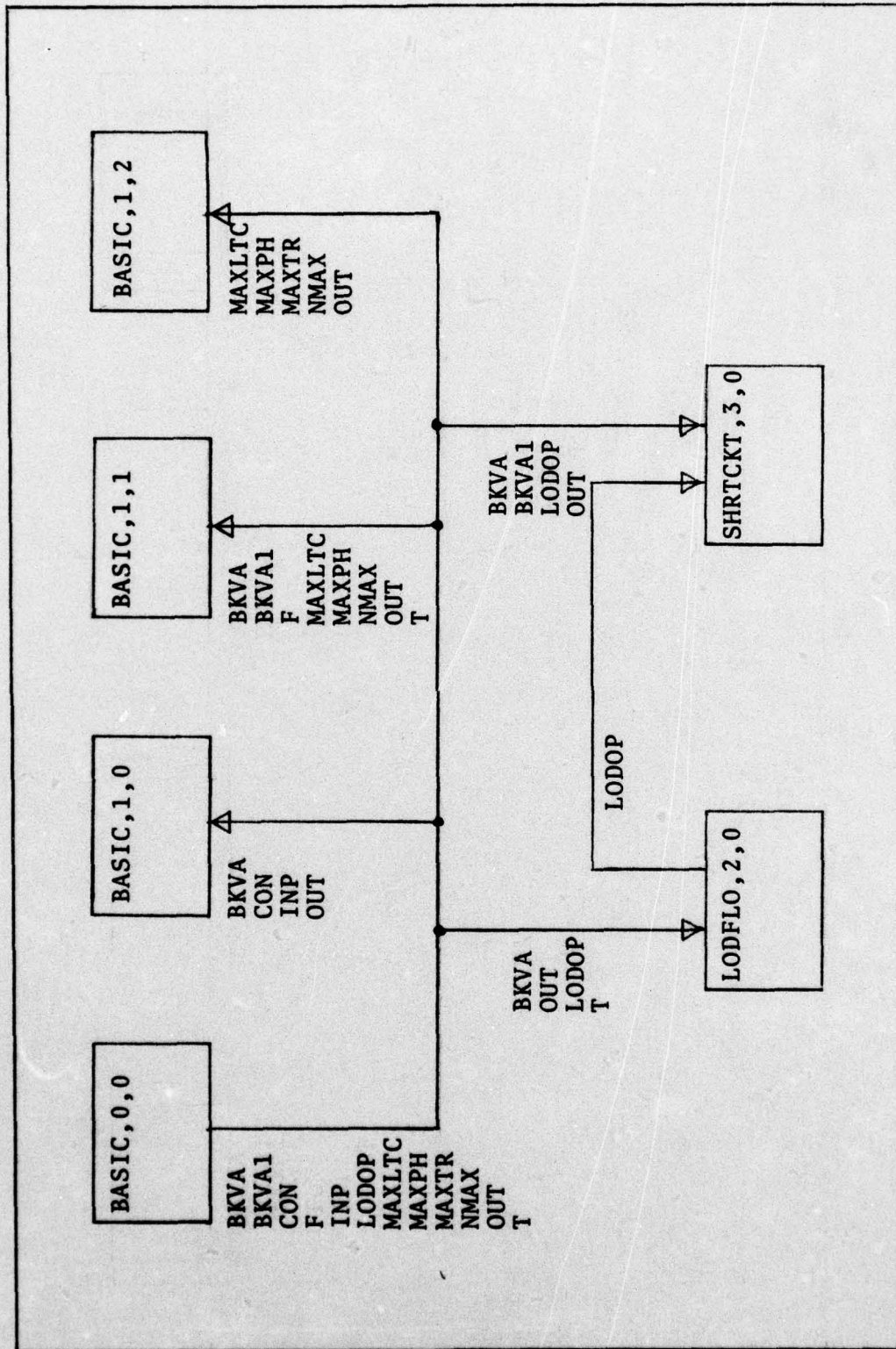


Fig. C-1. COMMON COMA Variables and Data Flow

Table C-II.

Common COMB Arrays

Array(Size)	Overlay Source	Definition/Use/Source
ANG(250)	Basic,1,0and/or Shrtckt,3,0	Bus numbers.
B(1450)	Basic,1,1 or Basic,1,2	Real part of 1/Z, line element susceptance.
BDIA(250)	Lodflo,2,0	Diagonal elements of B' matrix.
BUSNAME(250)	Basic,1,0 or Shrtckt,3,0	Alphanumeric name of bus.
CONEC(250)	Basic,1,1 or Basic1,2	Number of connections for each bus.
DBP(250)	Lodflo2,0	1/BDIA in triangulation of the B' matrix.
DBPP(250)	Lodflo2,0	1/BDIA in triangulation of the B' matrix.
DLP(250)	Lodflo,2,0 (1) Shrtckt,3,0 (2)	Delta P array. Real part of ZF in p.u.
DLQ(250)	Lodflo,2,0 (1) Shrtckt,3,0 (2)	(1) Delta Q array. (2) Img. part of ZF.
G(1450)	Basic,1,1 or Basic,1,2	Real part of 1/Z, line element admittance.
IBUS(250)	Basic,1,0 and/or Shrtckt,3,0	Bus number array.
ICC(250)	Basic,1,1 or Basic,1,2	Transformer code, C + IADD.
IPHASE(250)	Basic,1,1 or Basic1,2	Bus phase array.
IUBP(250)	Lodflo,2,0 (1) Shrtckt,3,0 (2)	(1) Index of first ele- ments of B' matrix. (2) List of bus numbers for subsystem.

Table C-II. (Cont'd)

Array(Size)	Overlay Source	Definition/Use/Source
IUBPP(250)	Lodflo,2,0	Used in triangulation of the B'' matrix.
JBP(3000)	Lodflo,2,0 (1) Shrtckt,3,0 (2)	(1) List of solumn identifiers for B' matrix. (2) List of sorted line table entry numbers.
JBPP(3000)	Lodflo,2,0	List of column identifiers for B'' matrix.
LINA(1450)	Basic,1,1 or Basic,1,2	End bus. Determined by second bus number.
LINB(1450)	Basic,1,1 or Basic,1,2	Start bus. Determined by first bus number.
LIST(250)	Basic,1,0 and/or Shrtckt,3,0	Used to form re-ordered bus lists.
LPHA(50)	Basic,1,1 or Basic,1,2	End bus for phase-shifter transformers.
LPHB(50)	Basic,1,1 or Basic,1,2	Start bus for phase-shifter transformers.
LTRA(250)	Basic,1,1 or Basic,1,2	SB of transformer, reactor or capacitor.
LTRB(250)	Basic,1,1 or Basic,1,2	End bus of transformer, reactor or capacitor.
Q(250)	Basic,1,0 or Shrtckt,3,0	Reactive power array.
QMAX(250)	Basic,1,0 or Shrtckt,3,0	QMX/BKVA or QMAXN.
QMIN(250)	Basic,1,0 or Shrtckt,3,0	QMN/BKVA or QMINN.
P(250)	Basic,1,0 or Shrtckt,3,0	Real power array.
PHANG(50)	Basic,1,1 or Basic,1,2	Phase angle for phase shifter transformers.

Table C-II. (Cont'd)

Array(Size)	Overlay Source	Definition/Use/Source
TAP(250)	Basic,1,1 or Basic,1,2	Initial setting of tap on transformers.
TMIN(250)	Basic,1,1 or Basic,1,2	Minimum transformer tap setting, in p.u.
TMX(250)	Basic,1,1 or Basic,1,2	Maximum tap setting of transformer.
UBP(3000)	Lodflo,2,0 (1) Shrtckt,3,0 (2)	(1) Upper triangular elements of B' matrix. (2) Absolute value of voltage for buses in sub-system.
UBPP(3000)	Lodflo,2,0	Upper triangular list of elements for B'' matrix.
V(250)	Basic,1,0 Shrtckt,3,0	Bus voltage array.
ZO1(1450)	Basic,1,1 or Basic,1,2	Imaginary part of zero sequence impedance.
ZOR(1450)	Basic,1,1 or Basic,1,2	Real part of zero sequence impedance.

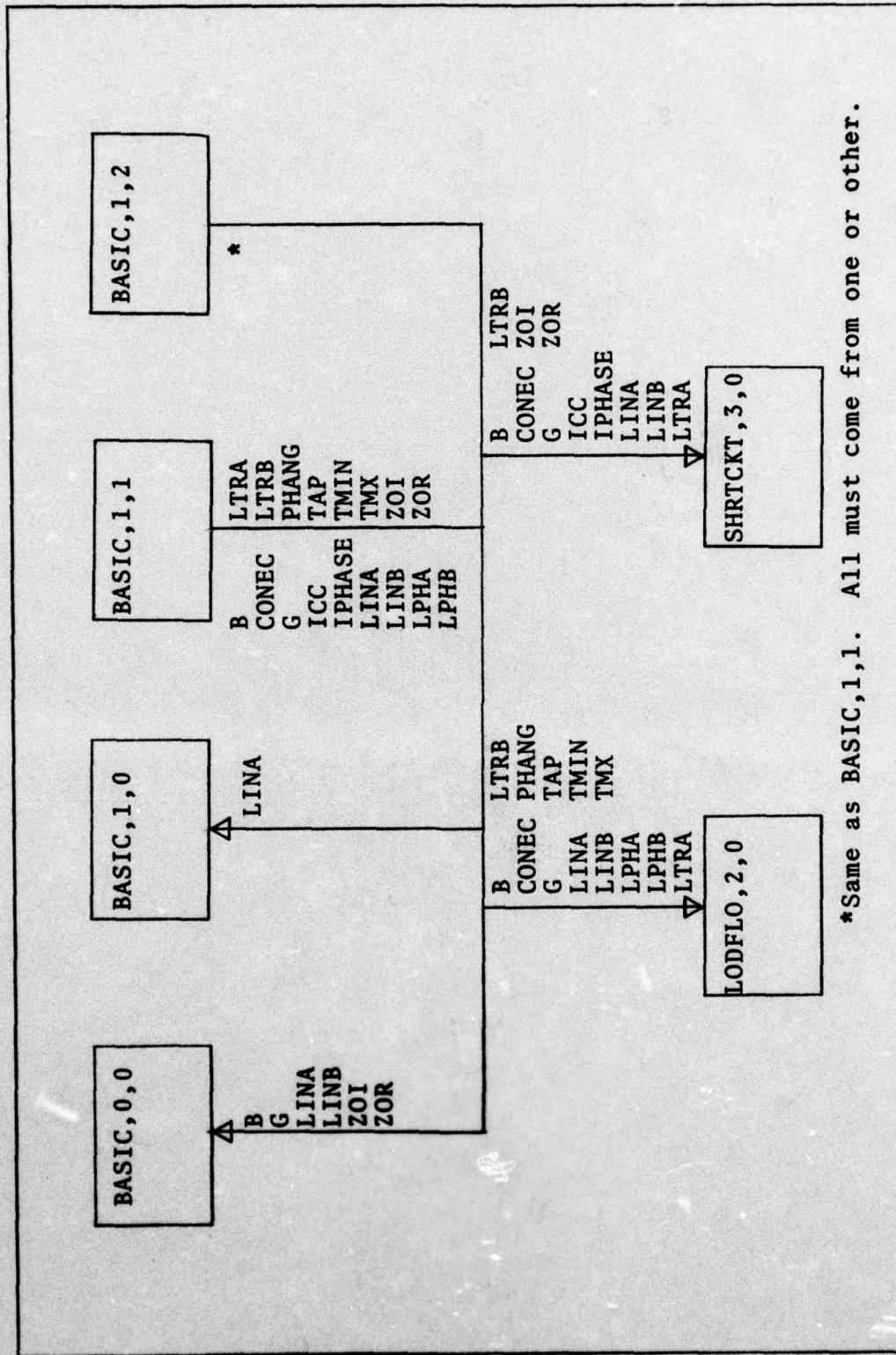


Fig. C-2. COMMON COMB Arrays and Data Flow

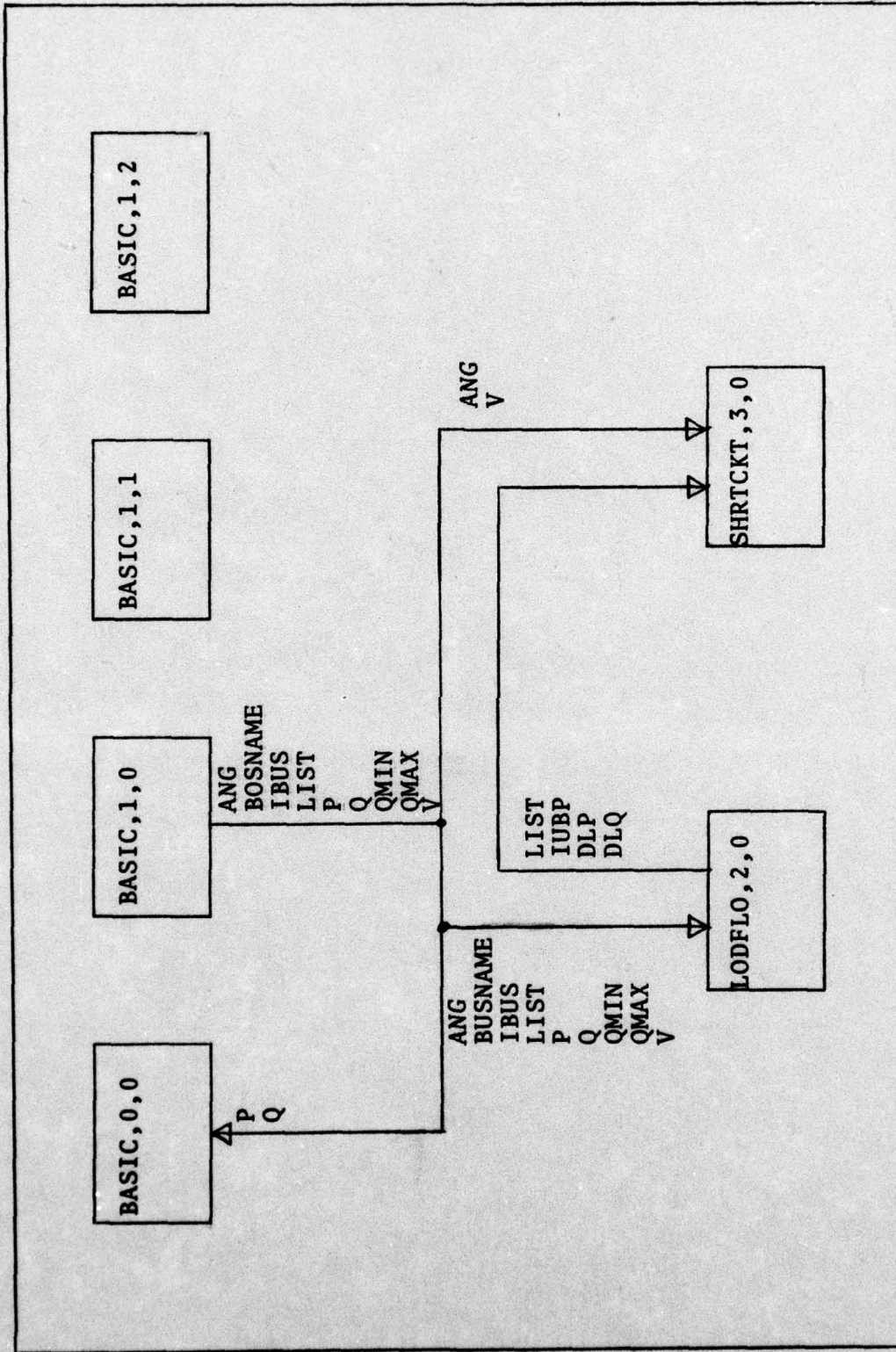


Fig. C-2. (Cont'd) COMMON COMB Arrays and Data Flow

Table C-III.
Common COMC Arrays

Array(Size)	Overlay Source	Definition/Use/Comments
DU(1000)	Lodflo,2,0 (1) Shrtckt,3,0 (2)	(1) Array of B' matrix off diagonal elements. (2) Array of bus vol- tages.
IDB(250)	Basic,1,0 and/or Lodflo,2,0 and/or Shrtckt,3,0	Bus type (ID).
JCOL(1000)	Basic,0,0 and Lodflo,2,0	List of end bus numbers.
NA(250)	Basic,0,0 and Lodflo,2,0 and/or Shrtckt,3,0	Number of connections to each bus. Used as dummy array in each routine.
NB(250)	Basic,0,0 (1) Shrtckt,3,0 (2)	(1) Used as reordered bus list. (2) List of buses in subsystem, NBS.

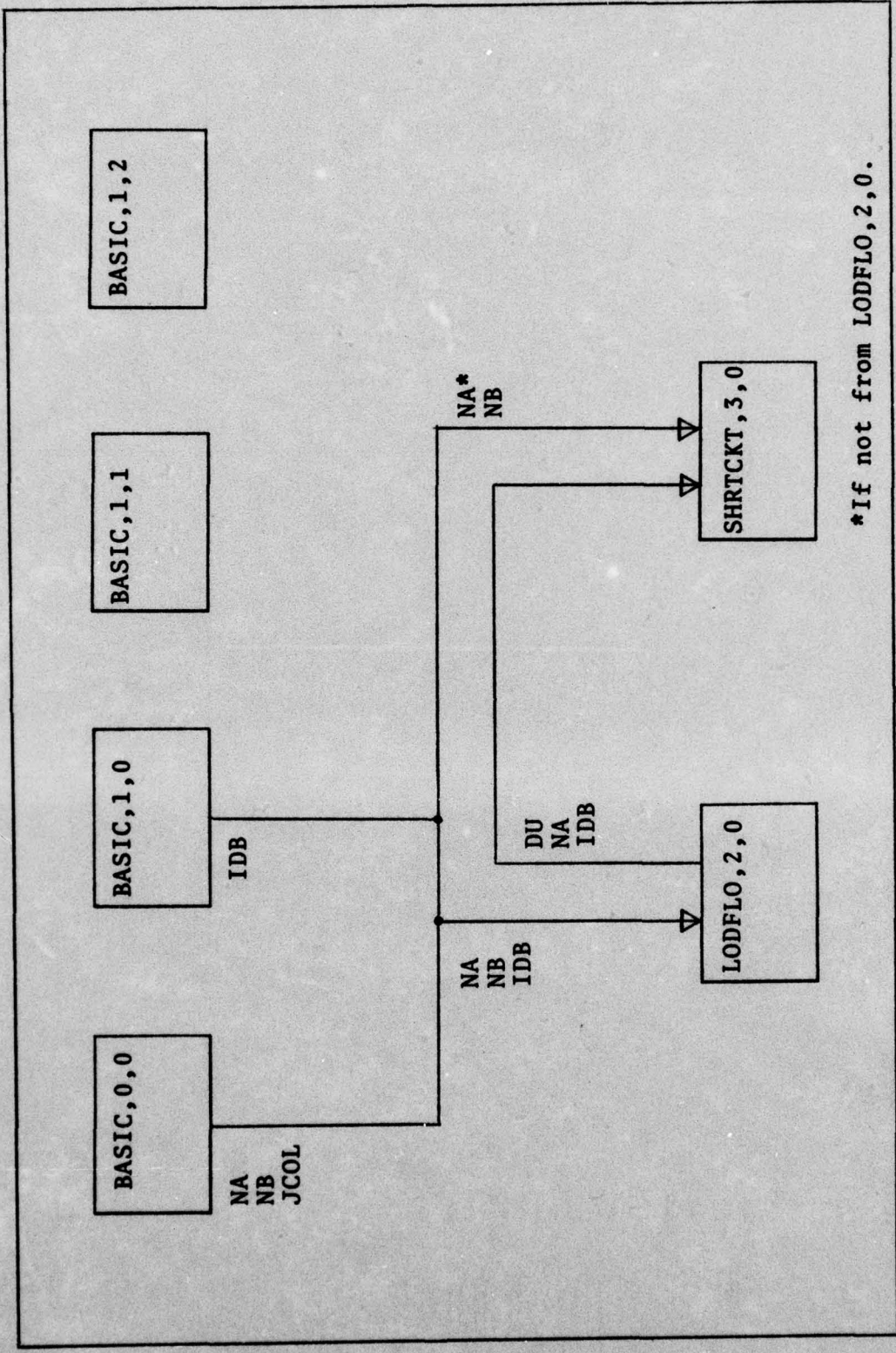


Fig. C-3. COMMON COMC Arrays and Data Flow

Table C-IV.
Common CONST Variables

Variable	Overlay Source	Definition/Use/Source
IPV	Basic,1,0	Number of Type 2 buses.
ISS	Basic,0,0 and Lodflo,2,0 (1) Shrtckt,3,0 (2)	(1) Total number of JCOL entries. (2) Number of lines in system (NL) divided by 2.
ITR1	Basic,1,0	Number of iterations for PTOL. Equal to ITRMAX1.
ITR2	Basic,1,0	Number of iterations for QTOL. Equal to ITRMAX2.
IZ	Basic,1,1 or Basic,1,2 and Shrtckt,3,0	Number of line elements to reference.
LL1	Basic,1,1 or Basic,1,2	Number of aerial and underground conductors.
LL2	Basic,1,1 or Basic,1,2	Number of fixed transformers.
LL3	Basic,1,1 or Basic,1,2	Number of autotransformers.
LL4	Basic,1,1 or Basic,1,2	Number of phase-shifter transformers.
NBUS	Basic,1,1 or Basic,1,2	Number of buses in system.
NL	Basic,1,1 or Basic,1,2 and Shrtckt,3,0	Number of lines in system times two.
NLC	Basic,1,0	Number of times loads are changed in Load Flow routine.
NOLTC	Basic,1,1 or Basic,1,2	Number of TCUL transformers.
NOTR	Basic,1,1 or Basic,1,2	Number of transformers in system.
PTOL	Basic,1,0	Tolerance for real power.
QTOL	Basic,1,0	Tolerance for reactive power in Load Flow routine.

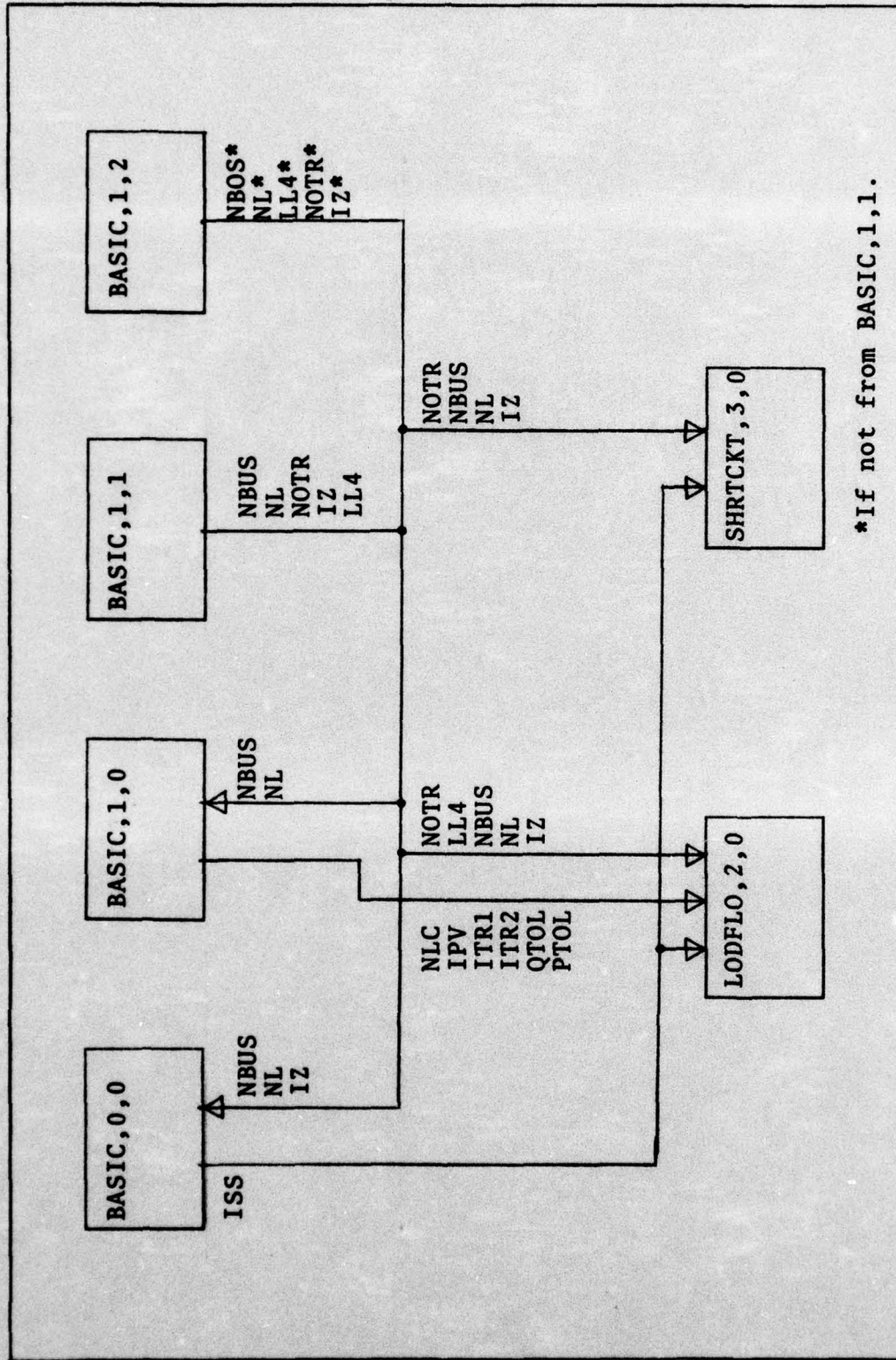


Fig. C-4. COMMON CONST Variables and Data Flow

Table C-V.

Common SAVE Variables

Variable	Overlay Source	Definition/Use/Source
IERR	Basic,0,0 or Basic,1,0 or Basic,1,1 or Basic,1,2 or Lodflo,2,0 or Shrtckt,3,0	Error counter for program.

Table C-VI.

Common ZCONST Variables

Variable	Overlay Source	Definition/Use/Source
IDUMM	Shrtckt,3,0	Second counter for number of buses in mutual impedance table.
IMUT	Shrtckt,3,0	Number of buses in mutual impedance table.
IROW	Shrtckt,3,0	Number of mutually coupled lines from subroutine MUTEST.
NBS	Shrtckt,3,0	Number of lines in a sub-system.
NOMU	Shrtckt,3,0	Number of mutually couples lines.

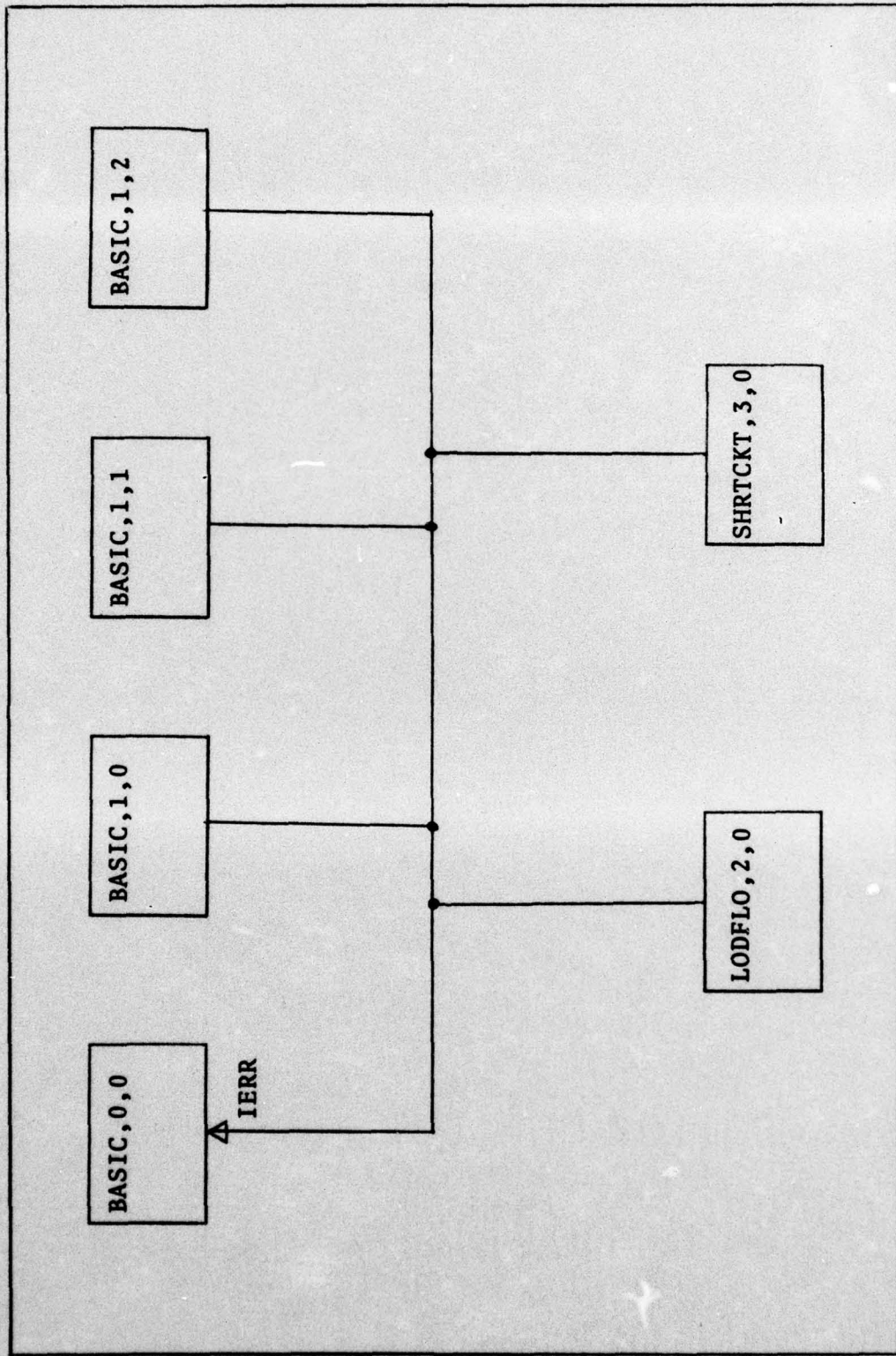


Fig. C-5. COMMON SAVE Variables and Data Flow

Table C-VII.
Common ZERO Arrays

Array(Size)	Overlay Source	Definition/Use/Comments
EBUS(250)	Shrtckt,3,0	Voltage array.
IJK(25)	Shrtckt,3,0	Bus number array of mutually coupled lines.
ISAVE(8)	Shrtckt,3,0	Bus number array in subroutine MUTEST.
ITZ(25)	Shrtckt,3,0	Additional array for mutually coupled lines.
KJI(25)	Shrtckt,3,0	Second bus number array for mutually coupled lines.
LA(25)	Shrtckt,3,0	Start bus array of mutually coupled lines.
LB(25)	Shrtckt,3,0	End bus array for mutually coupled lines.
LR(25)	Shrtckt,3,0	Array of start bus to which lines mutually coupled.
LS(25)	Shrtckt,3,0	End bus array of lines to which mutually coupled.
YCOUP(8,8)	Shrtckt,3,0	Mutual coupling matrix.
ZOBUS(2775)	Shrtckt,3,0	Zero sequence off diagonal terms of ZO matrix.
ZODIA(75)	Shrtckt,3,0	Array of diagonal zero sequence impedance terms of ZO matrix.
ZBUS(2775)	Shrtckt,3,0	Positive sequence off diagonal terms of Z matrix.
ZC(75)	Shrtckt,3,0	Impedance array used in BUS and BUSO subroutines.
ZDIA(75)	Shrtckt,3,0	Array of diagonal positive sequence terms of Z matrix.
ZM(25)	Shrtckt,3,0	Array of mutual impedance between mutually coupled lines.

Appendix D.

This appendix contains a complete listing of the PDSAP program. The numbers near the left margin are line numbers. If trying to find a particular line number in a routine, use these left margin numbers. The numbers near the right margin are sequence numbers and refer to the card numbers in the source deck. These numbers are used to insure the cards remain in proper sequence.

```

1          OVERLAY(BASIC,0,0)                                000100
          PROGRAM EXEC(INPJ,TAP5=INPUT,OUTPJ,TAP6=OUTPUT,TAP1,TAP2) 000110
C THIS PROGRAM SUPERVISES THE OVERALL EXECUTION OF THE COMPLETE PRO- 000120
C GRAM BY MEANS OF PARAMETERS READ IN ON CONTROL CARDS. THESE PARAM- 000130
9          ETERS ARE DEFINED AND CODED IN THE USER'S INSTRUCTIONS.    000140
C THE PURPOSE OF THE PROGRAM IS TO                                000150
C PERFORM VARIOUS STANDARD SYSTEM ANALYSIS ROUTINES ON A POWER DISTRI- 000160
C BUTION SYSTEM. THE PROGRAM IS SET UP TO HANDLE A 250 BUS SYSTEM WITH 000170
C 500 LINES/TRANSFORMERS. OF THE 500 LINES/TRANS., 50 MAY BE TAP- 000180
10          C CHANGING-UNDER-LOAD TYPES(TCUL'S). THE PROGRAM WILL ALSO ACCEPT 000190
C PHASE-ANGLE REGULATORS (MAX. OF 25).                          000200
C THE PROGRAM IS CAPABLE OF LOADFLOW AND SHORT-CIRCUIT ANALYSES AT THE 000210
C PRESENT. AN INPUT OPTION ALLOWS THE LINE DATA TO BE READ IN EITHER 000220
C PRE-CALCULATED OR AS "RAW" DATA, WITH THE IMPEDANCES CALCULATED BY 000230
15          C THE PROGRAM. OVERALL PROGRAM REVISION NOV 76 WITH CODE NAME POSA. 000240
          ZOMP_EX ZH,YCOU,ZCOUP,ZBUS,ZBUS,ZDIA,ZDIA,ZDIA,ZDIA,ZDIA,ZDIA,ZC
          INTEGER CON,CHG,SCOP,OUT,CONEC,A,C,D,E
          DIMENSION TITLE(8)
          ZONNOM /COMA/CON,C13,L000,ZSCOP,INP,OUT,F,T,BKVA,RH01,NMAX,MAXTR,
20          1MAXLTC,MAXPH,ZSYS,3KVA1
          ZONNOM /COMB/LINA(1450),LINA(1450),LINA(1450),B(1450),P(250),Q(250),
          L_P49(50),PHANG(50),LTRA(250),LTR3(250),TAP(250),TMM(250),V(250),
          2TRK(250),IUBPP(250),ANG(250),IRUS(250),DPP(250),UBP(3000),
          3BUSNAME(250),LPHA(50),LIST(250),IU9P(250),QMIN(250),QMAX(250),
25          4BPP(250),UAPP(3000),J3P(3000),J3PP(3000),ICC(250),DLP(250),
          5ZBI(1450),ZOR(1450),ROIA(250),CONEC(250),JLQ(250),IPHASE(250)
          ZONNOM /COMC/ NA(250),NB(250),JCOL(1000),DJ(1000),ID9(250)
          ZONNOM /CONST/ VBUS,NL,ISS,IPV,LL1,LL2,LL3,LL4,NOTR,IZ,NOLTC
30          1,ITR1,ITR2,PTOL,QTOL,NLC
          ZONNOM /SAVE/IERZ
          ZONNOM /ZERO/LA(25),LB(25),LR(25),LS(25),Z4(25),YCOUP(0,8),
          1ZCOUP(25),IJK(25),CJI(25),ITZ(25),ISAVE(8),ZDIA(75),ZDIA(75),
          2ZBUS(2775),ZBUS(2775),EBUS(250),ZC(75)
          ZONNOM /ZCONST/ NO4U,N9S,IRCH,I4JT,IOUMH
35          1000 FORMAT(R6,4I3)
          1001 FORMAT(8R10)
          1002 FORMAT(///1X,70(1H*)/1X,14*,68X,1H*/1X,1H*,13X,4R10,15X,1H*/1X,
          11H*,13X,4R10,15X,14*/1X,1H*,68X,1H*/1X,70(1H*)/)
          1004 FORMAT(R6,F7.0,3F5.0)
          1005 FORMAT(/T20,"*** PROGRAM PARAMETER CONSTANTS ***"/T6,"BASE KVA",
40          1T20,"FREQUENCY",T34,"TEMPERATURE",T31,"EARTH RESISTIVITY"/3X,F7.0,
          2" KVA",5X,F5.0," HZ",6X,F5.0," DEG. C",9X,F5.0," METER-OHM"/)
          1006 FORMAT("USE OF CONTROL PARAMETERS 4 OR 6 OR 7 IS NOT ALLOWED AT T00520
          14IS TIME.")
45          1007 FORMAT(1X,"USE OF CON CODE",I3," NOT ALLOWED.")
          1010 FORMAT(///1X,"*** THE PROGRAM HAS TERMINATED DUE TO",I5," ERRORS!"000550
          1/T20,"LISTED ABOVE IS A DESCRIPTION OF EACH"/T20,"ERROR AND RELEVAN000560
          2AT DATA FOR LOCATING"/T20,"THE ERROR SOURCE. GOOD LUCK!")
          1011 FORMAT(I3,"SB",T7,"EB",T17,"S",T31,"B",T43,"RE(Z0)",T57,"IM(Z0)"/000550
          1(2(1X,I3),3X,4(E11.5,3X))/)
          1012 FORMAT(///1X,70(1H*)/1X,1H*,58X,14*/1X,1H*,T23,"SORTED LINE INPUT 0000500
          1AT",T71,1H*/1X,1H*,T19,"LISTED BY ASCENDING BUS NUMBERS",T71,14*/000510
          21X,14*,68X,1H*/1X,70(1H*)/)
          1017 FORMAT(/T20,"*** PROGRAM CONTROL CONSTANTS ***"/T33,"CON=",
95          1I3/T33,"INP=",I3/T33,"OUT=",I3/T33,"CHG=",I3/)
          1018 FORMAT(I3,"FOLLOWING IS THE SORTED LINE TABLE AS CALCULATED "/)
          1019 FORMAT(I3,"PROGRAM CONTROL CARD NOT IN PROPER FORMAT OR LOCATION. 000650

```

AD-A035 292

AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 10/2
A USER-ORIENTED POWER DISTRIBUTION SYSTEM ANALYSIS PROGRAM.(U)
DEC 76 J A UNDERWOOD

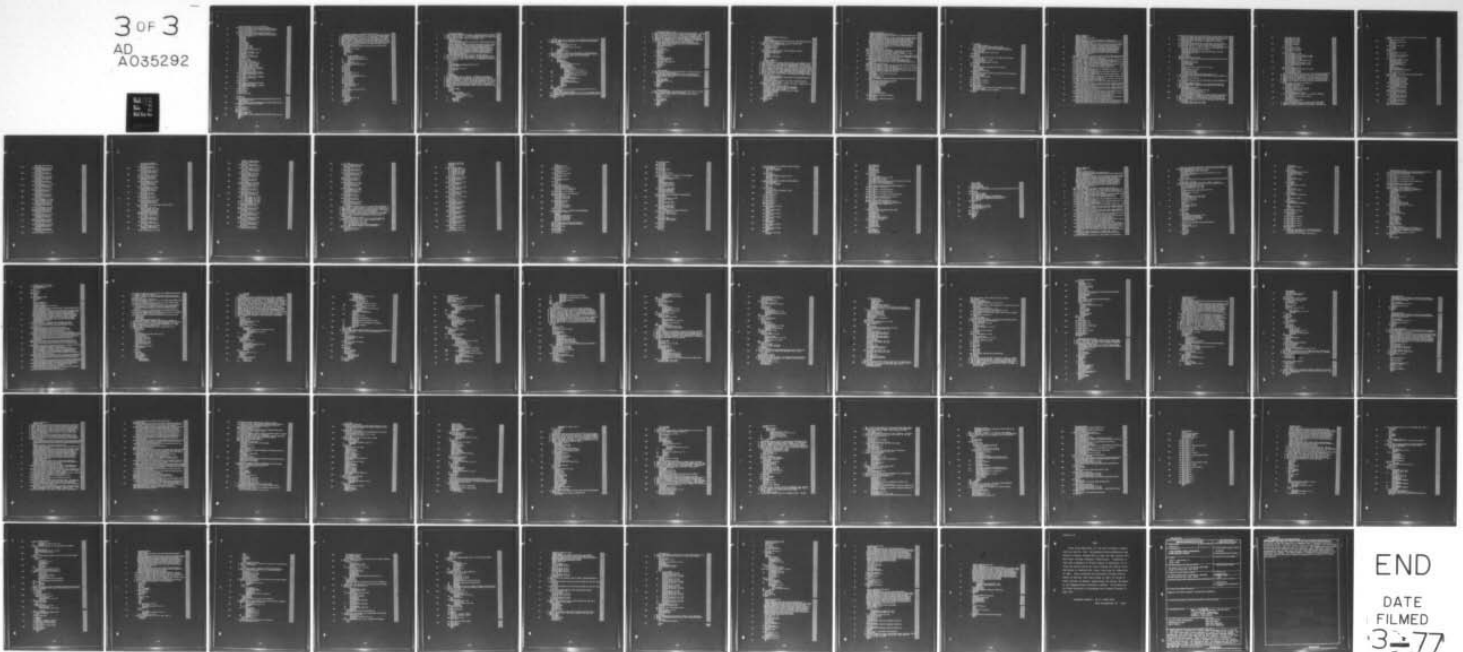
UNCLASSIFIED

6E/EE/76-43

NL

3 of 3

AD
A035292



END

DATE
FILMED

3-77

	1020	FORMAT (//1X,70(14°)/1X,1H°,58X,14°/1X,1H°,9X, 1°PJMR DISTRIBUTION SYSTEM ANALYSIS PROGRAM (POSAP)", 23X,14°/1X,1H°,26X,"TAPE 2 PRINTOJT",27X,14°/1X,1H°, 358X,14°/1X,70(1H°)/)	000670 000680 000690 000700 000710 000720 000730 000740 000750 000760 000770 000780 000790
60			
	1021	FORMAT (//1X,70(14°)/1X,1H°,58X,14°/1X,1H°,9X, 1°PJMR DISTRIBUTION SYSTEM ANALYSIS PROGRAM (POSAP)", 23X,14°/1X,1H°,26X,"TAPE 1 PRINTOJT",27X,14°/1X,1H°, 358X,1H°/1X,70(1H°)/)	000800 000810 000820 000830 000840 000850 000860 000870 000880 000890 000900 000910 000920 000930 000940 000950 000960 000970 000980 000990
65			
		4MAX=1000 4AKTR=250 4ACLTC=50 4AXPH=25	000990 001000 001010 001020 001030 001040 001050 001060 001070 001080 001090 001100 001110 001120 001130 001140 001150 001160 001170 001180 001190 001200 001210 001220 001230
70			
		IERR=LODOP=0 4RITE(2,1020) 4RITE(1,1021) READ(5,1001) TITLE WRITE(2,1002) TITLE 4RITE(1,1002) TITLE READ(5,1000) A,CON,IMP,OUT,CHG Z=52°SMCON IF(A.NE.C) GO TO 900 READ(5,1004) D,0KVA,F,T,RM01 Z=52°SYSPAR IF(D.NE.E) GO TO 931 IF(0KVA.EQ.0.) 0KVA=100000. 0K/01=0KVA	
75			
		IF(F.EQ.0) F=60. IF(T.EQ.0) T=25. IF(R401.EQ.0) RM01=100. 4RITE(2,1017) CON,IMP,OUT,CHG 4RITE(1,1017) CON,IMP,OUT,CHG 4RITE(2,1005) 0KVA,F,T,RM01 4RITE(1,1005) 0KVA,F,T,RM01 IF(CON.EQ.0.OR.CON.GT.7) GO TO 90 CALL OVERLAY(5H9ASIC,1,0,6HRECALL) IF(IERR.NE.0) GO TO 950 IF(OJT.EQ.15) GO TO 101 GO TO(100,20,30,40,50,60,70) CON 20 CALL OVERLAY(6HLOD°LO,2,0,54RECALL) IF(IERR.NE.0) GO TO 950 GO TO 100	
80			
		30 CALL OVERLAY(7HSHRTCKT,3,0,54RECALL) IF(IERR.NE.0) GO TO 950 GO TO 100 40 4RITE(2,1006) GO TO 90	
85			
		50 CALL OVERLAY(6HLODFLO,2,0,54RECALL) IF(IERR.NE.0) GO TO 950 CALL OVERLAY(7HSHRTCKT,3,0,54RECALL) IF(IERR.NE.0) GO TO 950 GO TO 100 60 4RITE(2,1006) GO TO 90 70 4RITE(2,1006) GO TO 90 80 4RITE(2,1007) CON	
90			
		90 IERR=1 GO TO 950 100 IF(OJT.EQ.2.OR.OUT.EQ.4.OR.OJT.EQ.5.OR.OJT.EQ.10) GO TO 101 IF(OUT.EQ.8) GO TO 101 4RITE(1,1012) 4RITE(1,1011) (LIND(I),LIND(I),6(I),0(I),ZOR(I),ZOR(I),I=1,NL) 101 IF(OUT.NE.2.AND.OUT.NE.10) REWIND 1 REWIND 2 STOP	001240 001250 001260 001270 001280 001290 001300 001310 001320 001330 001340 001350 001360 001370 001380 001390 001400 001410 001420
95			
		900 4RITE(2,1019) 0 GO TO 900 901 4RITE(2,1019) 0 909 IERR=1 950 4RITE(2,1010) IERR WRITE(1,1010) 4RITE(1,1011) (LIND(I),LIND(I),6(I),0(I),ZOR(I),ZOR(I),I=1,NL) REWIND 1 STOP END	
100			
105			
110			
115			
120			
125			
130			

```

1      SUBROUTINE LSORT(NL,LA,LB,LSIZE,LTAB,A,B,C,D,IE) 001430
C THIS SUBROUTINE SORTS TWO TABLES (LA & LB) INTO INCREASING SEQUENCE 001440
C IN DOUBLE-PRECISION FASHION, I.E. LA IS SORTED AS THE MOST SIGNIFI- 001450
C CANT FIGURE, AND THEN LB IF VALUES OF LA ARE EQUAL. IN ADDITION, 001460
5      C UP TO FIVE OTHER TABLES (A,B,C,D, & IE) CAN BE SORTED IN A CORRES- 001470
C PONDING MANNER. LTAB INDICATES HOW MANY ADDITIONAL TABLES ARE IN- 001480
C VOLVED, WHILE LSIZE INDICATES THE NUMBER OF ENTRIES PER TABLE (FOR 001490
C ALL TABLES). NL INDICATES WHETHER LA & LB WILL BE SORTED IN DOUBLE- 001500
C PRECISION FASHION (NL=2), OR SORTED IN SINGLE-PRECISION FASHION 001510
10     C (NL=1) USING ONLY TABLE LA. 001520
      DIMENSION LA(1450),LB(1450),A(1450),B(1450),C(1450),D(1450), 001530
      IIE(1450) 001540
      IF(LSIZE.LE.1) RETURN 001550
      N=LSIZE-1 001560
      NTAB=LTAB+1 001570
      DO 70 J=1,N 001580
      IM=J 001590
      --J+1 001600
      DO 30 I=L,LSIZE 001610
      IF(LA(I)-LA(IM)) 25,19,30 001620
      10 IF(NL.EQ.1) GO TO 30 001630
      IF(LB(I).GE.LB(IM)) GO TO 33 001640
      25 IM=I 001650
      30 CONTINUE 001660
      25 IF(IM.LE.J) GO TO 70 001670
      DO T(45,44,43,42,41,40) NTAB 001680
      40 ITE=IE(IM) 001690
      41 ID=D(IM) 001700
      42 IC=C(IM) 001710
      43 IB=B(IM) 001720
      44 IA=A(IM) 001730
      45 LTA=LA(IM) 001740
      IF(NL.EQ.1) GO TO 48 001750
      LTB=LB(IM) 001760
      35 40 IF(IM.LE.J) GO TO 53 001770
      DO T(55,54,53,52,51,50) NTAB 001780
      50 IE(IM)=IE(IM-1) 001790
      51 ID(IM)=ID(IM-1) 001800
      52 IC(IM)=IC(IM-1) 001810
      53 IB(IM)=IB(IM-1) 001820
      54 IA(IM)=IA(IM-1) 001830
      55 LA(IM)=LA(IM-1) 001840
      IF(NL.EQ.1) GO TO 57 001850
      LB(IM)=LB(IM-1) 001860
      45 57 IM=IM-1 001870
      DO T(40 001880
      59 DO T(65,64,63,62,61,60) NTAB 001890
      60 IE(J)=ITE 001900
      61 ID(J)=ID 001910
      62 IC(J)=IC 001920
      63 IB(J)=IB 001930
      64 IA(J)=IA 001940
      65 LA(J)=LTA 001950
      IF(NL.EQ.1) GO TO 70 001960
      LB(J)=LTB 001970
      95 70 CONTINUE 001980
      80 RETURN 001990

      END 002000

```

```

1      SUBROUTINE ORDER(IJ0)                                002010
C      THIS SUBROUTINE COMPUTES THE ORDER OF ELIMINATION FOR THE TRIANGULARIZATION 002020
C      TION OF THE "R" MATRIX. THE ORDERING SCHEME IS REFERRED TO AS THE 002030
C      DYNAMIC ORDERING SCHEME. IT SELECTS THE NEXT ROW TO BE ELIMINATED AS 002040
9      C THE ONE WITH THE FEWEST TERMS IN THE REDUCED MATRIX. 002050
      INTEGER OUT 002060
      DIMENSION IOUN(50) 002070
      COMMON /COMA/CMN,CM5,LODOP,SCOP,IMP,OUT,F,T,BKVA,RM01,NMAX,MAXTR, 002080
      1MAXLTC,MAXPH,ISYS,3KVA1 002090
10     COMMON /COMB/LINA(1450),LINA(1450),S(1450),B(1450),P(250),Q(250), 002100
      1LPH9(50),PHANG(50),LTRA(250),LTRA(250),TAP(250),THN(250),V(250), 002110
      2THX(250),IURPP(250),ANG(250),IBUS(250),OBP(250),UBP(3000), 002120
      3BUSNAME(250),LPHA(30),LIST(250),IURP(250),QMIN(250),QMAX(250), 002130
      4JBP(250),URPP(3000),JRP(3000),JBP(3000),ICC(250),OLP(250), 002140
      5ZBI(1450),ZOR(1450),9DIA(250),CONEC(250),JLQ(250),IPHASE(250) 002150
15     COMMON /COMC/ NA(250),NB(250),JCO.(1000),DJ(1000),ID9(250) 002160
      COMMON /CONST/ NBUS,NL,ISS,IPV,L.1,L.2,L.3,LL6,NOTR,IZ,MOLTC 002170
      1,ITR1,ITR2,PTOL,ITOL,NLC 002180
C      THE FOLLOWING LOOP FORMS THE JCOL AND NA VECTORS FROM THE 002190
C      LINE DATA READ IN BY THE INPUT PROGRAM. THE JCOL VECTOR CONTAINS A 002200
C      LIST OF ALL THE J TERMS FROM EACH ROW OF THE BUS ADMITTANCE MATRIX. 002210
C      THE NA VECTOR IS A LIST OF THE TOTAL NO. OF CONNECTIONS TO EACH BUS. 002220
      DO 5 I=1,250 002230
      5 NA(I)=0 002240
      J=1 002250
      DO 10 I=1,NL 002260
      IF(LINA(I).EQ.0.OR.LINA(I).EQ.0) GO TO 10 002270
      JCOL(N)=LINA(I) 002280
      N=N+1 002290
20     7 IF(LINA(I).NE.J) GO TO 8 002300
      NA(J)=NA(J)+1 002310
      DO 7 J=1 002320
      8 J=J+1 002330
      DO 7 J=1 002340
35     10 CONTINUE 002350
      ISS=NL-2*IZ 002360
C      IN THE FOLLOWING LOOP, THE BUS LIST IS SCANNED AND THE NUMBER 002370
C      OF CONNECTIONS TO EACH IS COMPARED. THE BUS WITH THE FEWEST CON- 002380
C      NECTIONS IS SELECTED AS THE NEXT BUS TO BE ELIMINATED IN THE TRI- 002390
C      ANGULARIZATION ROUTINE. THE ACTUAL ELIMINATION IS SIMULATED ON THE JCOL 002400
C      VECTOR, WITH "J" TERMS ADDED OR DELETED AS APPROPRIATE. THE NUMBER 002410
C      OF CONNECTIONS TO THE BUS IS ALTERED ACCORDINGLY. THE LOOP THEN 002420
C      REPEATS, AND THE REMAINING BUSES SCANNED TO FIND THE BUS WITH THE 002430
C      FEWEST CONNECTIONS IN THE REDUCED MATRIX, UNTIL ALL BUSES HAVE BEEN 002440
45     C REORDERED. 002450
      NB(1)=0 002460
      DO 100 I=1,NBUS 002470
      [CM]=25 002480
      IZZ=I-1 002490
50     DO 12 L=1,NBUS 002500
      IF(NA(L).GE.[CM]) GO TO 12 002510
      DO 11 LL=1,IZZ 002520
      IF(L.EQ.NB(LL)) GO TO 12 002530
11     CONTINUE 002540
95     [CM]=NA(L) 002550
      NB(I)=IRW=L 002560
12     CONTINUE 002570

```

			002500
		KKK=0	002590
60	C	THIS LOOP SCANS JC0L TO FIND ROWS WITH ELEMENTS IN THE I*TH COLUMN	002600
	C	AND THEN STORES THE BJS ROW NUMBER IF IT HAS NOT BEEN PREVIOUSLY	002610
	C	OPERATED UPON.	002620
		DO 20 M=1,ISS	002630
		IF(JCOL(M),NE,IRN) GO TO 20	002640
65		DO 15 L=1,I	002650
		IF(IRN(M),EQ,NB(L)) GO TO 20	002660
	15	CONTINUE	002670
		KKK=KKK+1	002680
		IDUM(KKK)=IRN(M)	002690
70	20	CONTINUE	002700
	C	THE FOLLOWING LOOP PERFORMS THE SIMULATED ROW ELIMINATION ON ALL	002710
	C	ROWS BELOW THE I*TH ROW. SUBROUTINE ADDRDEL IS CALLED TO ADD OR	002720
	C	DELETE AN ELEMENT FROM THE JC0L VECTOR IF NECESSARY.	002730
		IF(KKK,EQ,0) GO TO 100	002740
		DO 50 N=1,KKK	002750
75		IJJ=0	002760
	25	KI=JADD(IRN)	002770
		KF=KI+NA(IRN)-1	002780
		DO 40 J=KI,KF	002790
80		IF(JCOL(J),EQ,IRN) GO TO 40	002800
		K1=JDD(IDUM(N))	002810
		K2=K1+NA(IDUM(N))-1	002820
		IF(IJJ,EQ,1) GO TO 32	002830
		DO 30 L=K1,K2	002840
85		IF(JCOL(L),EQ,IRN) GO TO 35	002850
	30	CONTINUE	002860
	32	IF(IRN(K1),EQ,JCOL(J)) GO TO 40	002870
		DO 31 L=K1,K2	002880
		IF(JCOL(L),EQ,JCOL(J)) GO TO 40	002890
90	31	CONTINUE	002900
		CALL ADDRDEL(L,-1,1)	002910
		NA(IDUM(N))=NA(IJUM(N))+1	002920
		GO TO 40	002930
	35	CALL ADDRDEL(L,-1,0)	002940
		NA(IDUM(N))=NA(IJUM(N))-1	002950
95		IJJ=1	002960
		GO TO 25	002970
	40	CONTINUE	002980
	50	CONTINUE	002990
100	100	CONTINUE	003000
		IF(IJT,EQ,2,OR,IJT,EQ,7,OR,IJT,EQ,8,OR,IJT,EQ,10) GO TO 101	003010
		WRITE(1,1000)	003020
		WRITE(1,1001) (N9(I),I=1,N9US)	003030
1000		FORMAT(///1X,70(1M)/1X,14*,50X,14*/1X,14*,T16,"REORDERED RUSLIST	003040
		RETURNED BY SUBROUTINE ORDER",771,14*/1X,14*,50X,14*/1X,70(1M)//)	003050
1001		FORMAT(3X,1319)	003060
101		RETURN	003070
		END	

```

1      SUBROUTINE ADDDEL(L,IC,J)                                003000
C THIS SUBROUTINE ADDS OR DELETES AN ENTRY IN THE JCOL TABLE. IC IS A 003090
C PARAMETER WHICH DETERMINES IF AN ENTRY IS ADDED OR DELETED. IC=1 RE-003100
C SULTS IN AN ADDED ENTRY, AND IC=-1 RESULTS IN A DELETION. OTHER 003110
9      C TABLE ENTRIES ARE MOVED TO ALLOW FOR THE ADDED OR DELETED ENTPY. THE 003120
C PARAMETER ISS (TOTAL NUMBER OF TABLE ENTRIES) IS INCREMENTED OR DE 003130
C REMENTED AS APPROPRIATE. THE L PARAMETER DETERMINES THE POSITION OF 003140
C THE ADDED OR DELETED ENTRY, AND J IS THE PARAMETER THAT DETERMINES 003150
C THE VALUE OF THE ADDED TERM (NOT USED FOR DELETIONS). 003160
10     COMMON/COMC/ NA(250),NB(250),JCOL(1000),DJ(1000),IOB(250) 003170
      COMMON /CONST/ NSUS,NL,ISS,IPV,LL1,LL2,LL3,LL4,NOTR,IZ,NOLTC 003180
      I,ITR1,ITR2,PTOL,QTOL,MLC 003190
      IF(I.EQ.1) GO TO 20 003200
      .S=ISS-1 003210
15     DO 10 I=L,LS 003220
      JCOL(I)=JCOL(I+1) 003230
10     CONTINUE 003240
      JCOL(ISS)=0 003250
      ISS=ISS-1 003260
20     RETURN 003270
      .S=ISS-L-2 003280
      JC=JCOL(J) 003290
      JCOL(ISS+1)=JCOL(ISS) 003300
      JCOL(ISS)=JCOL(ISS-1) 003310
25     DO 30 I=1,LS 003320
      JCOL(ISS-I)=JCOL(ISS-I-1) 003330
30     CONTINUE 003340
      JCOL(L+1)=JC 003350
      ISS=ISS+1 003360
30     RETURN 003370
      END 003380

1      *FUNCTION IRN(M) 003390
C THIS FUNCTION CALCULATES THE ROW NUMBER OF AN ENTRY IN THE LINE 003400
C TABLE FROM THE GIVEN POSITION, L. 003410
9      COMMON/COMC/ NA(250),NB(250),JCOL(1000),DJ(1000),IOB(250) 003420
      COMMON /CONST/ NSUS,NL,ISS,IPV,LL1,LL2,LL3,LL4,NOTR,IZ,NOLTC 003430
      I,ITR1,ITR2,PTOL,QTOL,MLC 003440
      ISJM=0 003450
10     DO 10 II=1,NSUS 003460
      ISUM=ISUM+NA(II) 003470
      IF(ISUM.GE.M) GO TO 11 003480
10     CONTINUE 003490
      II=II-1 003500
11     IRN=II 003510
15     RETURN 003520
      END 003530

1      *FUNCTION JADD(I) 003540
C THIS FUNCTION CALCULATES THE POSITION OF THE FIRST ENTRY OF A GIVEN 003550
C ROW IN THE LINE TABLE, GIVEN THE ROW NUMBER. 003560
9      COMMON/COMC/ NA(250),NB(250),JCOL(1000),DJ(1000),IOB(250) 003570
      ISJM=0 003580
      IF(I.EQ.1) GO TO 11 003590
      MJI=I-1 003600
10     DO 10 J=1,MJI 003610
      ISJM=ISJM+NA(J) 003620
10     JADD=ISJM+1 003630
      RETURN 003640
11     JADD=1 003650
      RETURN 003660
      END 003670

```

```

1      SUBROUTINE PERUNIT(Z,Z0,PH,VP,S)                                003600
      INTEGER PH                                                    003690
      COMPLEX Z,Z0                                                  003700
      COMMON /COMA/CON,CHS,LODOP,SCOP,INP,OUT,F,T,BKVA,RHO1,NMAX,MAXTR, 003710
      14AKLTC,MAXPH,IYS,BKVA1                                       003720
C     FIRST PART IS TO CONVERT IMPED. VALUES IN PERCENT FROM
C     BASE KVA OF S TO PERUNIT ON BKVA.                            003730
      IF(S.EQ.0.) GO TO 10                                          003740
      Z=(Z*BKVA)/(S*100.)                                           003750
      Z0=(Z0*BKVA)/(S*100.)                                         003760
10     RETURN                                                       003770
      RETURN                                                         003780
C     THIS SECTION CONVERTS VALUE IN OHMS TO PERUNIT ON BKVA.    003790
      10 IF(PH.EQ.1) BKVA=BKVA/3.                                    003800
      Z0=(1000.*VP*VP)/9KVA                                         003810
      Z=Z/Z0                                                         003820
15     Z0=Z0/Z0                                                      003830
      IF(PH.EQ.1) BKVA=BKVA1                                         003840
      RETURN                                                         003850
      END                                                            003860

1      OVERLAY (BASIC,1,0)                                          003870
      PROGRAM INPUT                                                003880
C     THIS PROGRAM READS IN THE BASE CASE LINE DATA. TWO OPTIONS ARE PRJ- 003890
C     VIDED FOR INPUT: 1) LINE DATA INPUT AS IMPEDANCES PER UNIT LENGTH 003900
C     (WIRE/CABLE) OR PER CENT IMPEDANCE (TRANSFORMERS), WITH THE CONVER- 003910
C     SION TO PER UNIT ACCOMPLISHED BY THE COMPUTER; OR 2) LINE DATA INPUT 003920
C     AS "RAW" DATA, I.E. WIRE SIZE, WIRE TYPE, LENGTH OF RUN, TRANSFORMER 003930
C     SIZE, TRANSFORMER CONNECTION, LINE-LINE VOLTAGE, AND NUMBER OF PHASES 003940
C     IN THE CIRCUIT. THIS IS THE MINIMUM DATA REQUIRED FOR THE PROGRAM 003950
C     RUN UNDER OPTION 2. IF OTHER INFORMATION, SUCH AS STRANDING, EQUIV. 003960
C     SPACING, OR TRANSFORMER IMPEDANCES (PER CENT) ARE AVAILABLE, THESE 003970
C     MAY BE INPUT AS WELL UNDER OPTION 2, WITH RESULTING IMPROVED ACCURACY 003980
C     OF THE RESULTS.                                             003990
      INTEGER CON,CHG,SCOP,OUT                                       004000
      COMMON /COMA/CON,CHG,LODOP,SCOP,INP,OUT,F,T,BKVA,RHO1,NMAX,MAXTR, 004010
      14AKLTC,MAXPH,IYS,BKVA1                                       004020
      COMMON /SAVE/IERR                                             004030
      COMMON /CONST/ NSJS,NL,ISS,I*V,LL1,LL2,LL3,LL4,NOTR,IZ,NOLTC 004040
      1,ITR1,ITR2,PTOL,ITOL,NLC                                       004050
20     1000 FORMAT(1X,"ERROR IN INPUT PARAMETER INP CODE. PLEASE REFER TO P 004060
      1 PROGRAM INSTRUCTIONS. ")                                       004070
      IF(INP.GT.1.OR.INP.LT.0) GO TO 100                             004080
      IF(INP.EQ.0) CALL OVERLAY(54BASIC,1,1,6HRECALL)                004090
      IF(INP.EQ.1) CALL OVERLAY(54BASIC,1,2,6HRECALL)                004100
25     IF(CON.EQ.1.OR.CON.EQ.3) RETURN                                004110
      IF(CON.EQ.4.AND.(SCOP.EQ.0.OR.SCOP.EQ.2)) RETURN              004120
      IF(OUT.EQ.15) RETURN                                           004130
      IF(IERR.NE.0) RETURN                                           004140
30     CALL BUSINDUM                                                004150
      IF(IERR.NE.0) RETURN                                           004160
      CALL ORDER(0)                                                 004170
      RETURN                                                         004180
      100 WRITE(1,1000)                                             004190
      RETURN                                                         004200
35     END                                                            004210

```

```

1      SUBROUTINE BUSIN(0J4)                                004220
      INTEGER CON,CHG,SCOP,OUT,CONEC,A,C                    004230
      COMMON /COMA/CON,CHG,LODOP,SCOP,INP,OUT,F,T,BKVA,RHO1,NMAX,MAXTR, 004240
      14AKLYC,MAXPH,ISVS,3KVA1
5      COMMON /COMB/LINA(1450),LINA(1450),G(1450),B(1450),P(250),Q(250), 004250
      1,P49(50),PHANG(50),LTRA(250),LTRA(250),TAP(250),TMM(250),V(250), 004260
      2TRX(250),IUBPP(250),ANG(250),IBUS(250),DBP(250),USP(3000), 004270
      3BUSNAME(250),LPHA(50),LIST(250),IUMP(250),2MIN(250),QMAX(250), 004280
      4)BPP(250),UBPP(3000),JBP(3000),JBP(3000),ICC(250),DLP(250), 004290
      52BI(1450),ZOR(1450),BDIA(250),CONEC(250),DLQ(250),IPHASE(250) 004300
10     COMMON /COMC/ NA(250),NB(250),JCO.(1000),DJ(1000),IOB(250) 004310
      COMMON /CONST/ NBUS,NL,ISS,IPV,LL1,LL2,LL3,LL4,NOTR,IZ,NOLTC 004320
      1,ITR1,ITR2,PTOL,QTOL,MLC
      COMMON /SAVE/ IERR 004330
15     1000 FORMAT(I2,I3,A10,2F5.0,4F10.0) 004340
      1001 FORMAT(1X,"TOO MANY SLACK BUSES - ADDITIONAL ONE DECLARED IS ", 004350
      1410/1X," BUS NO. ",I3," BUS DATA CARD NO. ",I3/) 004360
      1002 FORMAT(1X,"TOO MANY PV BUSES - LAST ONE DECLARED IS ",A10, 004370
      1" BUS NO. ",I3," BUS DATA CARD NO. ",I3/) 004380
20     1003 FORMAT(///1X,70(1H"/1X,1H",50X,1H"/1X,1H",16X,"SUMMARIZED INPUT 9004410
      1JS DATA: PER-UNIT",T71,1H"/1X,14",T21,"LISTED BY ASCENDING BUS NUM004420
      29EQS",T71,1H"/1X,14",68X,14"/1X,70(1H"/1X) 004430
      1004 FORMAT(T30,"POWER"/T3,"NO.",T7,"TYPE",T12,"V(4AS)",T20, 004440
      1"V(ANG-DEG)",T34,"REAL",T42,"REACTIVE",T54,"Q(MIN)",T64, 004450
25     2"Q(4X)",T74,(1X,I3,3X,I1,3X,F5.1,3X,F9.4,2X,F9.5,1X,F9.5,2X,F8.4, 004460
      32X,F9.4)) 004470
      1005 FORMAT(R6,I4,I4,2F7.5,I3) 004480
      1006 FORMAT(T3,"PROGRAM CONTROL CARD NOT IN PROPER FORMAT OR LOCATION. 004490
      1CARD WITH KEYWORD ",R6," IS REQUIRED.") 004500
30     1014 FORMAT(1X,"BUS",I3," SHOULD BE CONNECTED TO A LINE BUS OF THISAGE. 004510
      1CHECK INPUT LINE LIST FOR THIS BUS NUMBER.") 004520
      1015 FORMAT(1X,"BUS LIST SHOULD HAVE BUS",I3,". CHECK BUS INPUT DATA F004530
      1R POSSIBLE MISSING OR EXTRA BUS CARDS.") 004540
      IXX=IPV=0 004550
35     READ(9,1005) A,ITR1,ITR2,PTOL,QTOL,VLG 004560
      C=5RLDFLOW 004570
      IF(A.NE.C) GO TO 903 004580
      DO 20 I=1,NBUS 004590
40     READ(9,1000)IOB(I),IBUS(I),BUSNAME(I),V(I),ANG(I),P(I),Q(I),2MN, 004600
      12MC 004610
      IF(I70(I).EQ.3.AND.IXX.NE.0) GO TO 100 004620
      IF(I70(I).EQ.3.AND.IXX.EQ.0) IXX=1 004630
      IF(I70(I).EQ.2) IPV=IPV+1 004640
      IF(IPV.GT.50) GO TO 101 004650
45     IF(V(I).EQ.0) V(I)=1.0 004660
      P(I)=PTI/BKVA 004670
      Q(I)=QTI/BKVA 004680
      2MIN(I)=QMN/BKVA 004690
      2MAX(I)=QMX/BKVA 004700
50     LIST(I)=IBUS(I) 004710
      ANG(I)=.01745329*AVG(I) 004720
      DO TO 20 004730
      100 WRITE(1,1001) BUSNAME(I),IBUS(I),I 004740
      IERR=IERR+1 004750
      GO TO 20 004760
95     101 WRITE(1,1002) BUSNAME(I),IBUS(I),I 004770
      IERR=IERR+1 004780

```

	20	CONTINUE	004790
		IF(IERR.NE.0) RETURN	004800
60		CALL LSORT(1,IBUS,0,NBUS,5,V,ANG,P,2,IN9)	004810
		CALL LSORT(1,LIST,0,NBUS,3,BUSNAME,IMIN,QMAX,0,0)	004820
	C	LOOPS 201 AND 202 CHECK THAT EACH LINE BUS HAS LOAD BUS.	004830
		DO 201 II=1,NBUS	004840
		DO 202 JJ=1,NL	004850
65		IF(LINA(JJ).EQ.IBUS(II)) GO TO 201	004860
	202	CONTINUE	004870
		GO TO 901	004880
	201	CONTINUE	004890
	C	LOOPS 204 AND 205 CHECK THAT EACH LOAD BUS HAS A LINE BUS.	004900
70		DO 204 IA=1,NL	004910
		IF(LINA(IA).EQ.0) GO TO 204	004920
		DO 205 JB=1,NBUS	004930
		IF(LINA(IA).EQ.IBUS(JB)) GO TO 204	004940
	205	CONTINUE	004950
75		GO TO 902	004960
	204	CONTINUE	004970
		IF(DUT.EQ.2.0R.0JT.EQ.6.0R.0JT.E2.10) RETJRN	004980
		WRITE(1,1003)	004990
		DO 200 I=1,NBUS	005000
80		ANG(I)=ANG(I)*57.29578	005010
	200	CONTINUE	005020
		WRITE(1,1004) (IBUS(I),JOB(I),V(I),ANG(I),P(I),2(I),QMIN(I),	005030
		12MAX(I),I=1,NBUS)	005040
		DO 209 I=1,NBUS	005050
85		ANG(I)=ANG(I)*.01745329	005060
	209	CONTINUE	005070
		GO TO 950	005080
	901	WRITE(2,1014) IBUS(II)	005090
		GO TO 940	005100
90	902	WRITE(2,1015) LINA(IA)	005110
		GO TO 940	005120
	903	WRITE(2,1006) C	005130
	940	IERR=IERR+1	005140
		WRITE(1,1004) (IBUS(I),JOB(I),V(I),ANG(I),P(I),2(I),QMIN(I),	005150
95		12MAX(I),I=1,NBUS)	005160
		WRITE(2,*) "ANGLE VALUES IN RADIAN"	005170
	950	RETURN	005180
		END	005190

```

1      OVERLAY (BASIC,1,1)                                005200
      PROGRAM LINDATA                                     005210
      COMP. EX Z,ZO,ZN,Y,ZOG,ZOAG                       005220
      REAL NN,L,L1,L2                                    005230
5      INTEGER PH,C,SB,ZB,STR,CON,CHG,SCOP,OUT,C4,CONEC,TC,IO 005240
      COMMON /COMA/CON,C4S,LDDOP,SCOP,INP,OUT,F,T,BKVA,RHO1,NMAX,MAXTR, 005250
      14XLTG,MAXPH,ISYS,3KVA1                            005260
      COMMON /COMB/LINA(1450),LIN9(1450),S(1450),9(1450),P(250),Q(250), 005270
10     L_PNS(50),PHANG(50),LTRA(250),LTR9(250),TAP(250),TMN(250),V(250), 005280
      2THX(250),IURPP(250),ANG(250),TRUS(250),DBP(250),URP(3000), 005290
      33USNAME(250),LPHA(50),LIST(250),IURP(250),QMIN(250),QMAX(250), 005300
      4J8PP(250),UBPP(3000),J9P(3000),J3PP(3000),ICC(250),OLP(250), 005310
      9ZBI(1450),ZOR(1450),9DIA(250),CONEC(250),JLQ(250),IPHASE(250) 005320
      COMMON/COMC/ NA(250),N9(250),JCO.(1000),DJ(1000),IN9(250) 005330
15     COMMON /CONST/ NBUS,NL,ISS,IPV,LL1,LL2,LL3,LL4,NOTR,IZ,NOLTC 005340
      1,I1R1,I1R2,PTOL,DTOL,NLC                          005350
      COMMON /SAVE/IERR                                   005360
200   FORMAT(2I3,F5.0,I2,F6.0,I2,I1,F7.0,I2,F4.0,F6.0,5F5.0,I1,2F6.0,I1) 005370
1000  FORMAT(1X,"ERROR IN WIRE SIZE INPUT! LINE DATA CARD NUMBER ",I5, 005380
      1" PLEASE REFER TO PROGRAM INSTRUCTIONS."/)        005390
1001  FORMAT(1X,"INVALID TRANSFORMER SIZE/VOLTAGE RATING! LINE DATA CAR 005400
      10 NUMBER ",I5," PLEASE REFER TO PROGRAM INSTRUCTIONS."/) 005410
1002  FORMAT(1X,"INVALID TRANSFORMER VOLTAGE INPUT! LINE DATA CARD N 005420
      1ER",I5," PLEASE REFER TO PROGRAM INSTRUCTIONS."/) 005430
25   1003  FORMAT(1X,"ERROR HAS RESULTED IN POSITIVE AND ZERO SEQUENCE IMPEDA 005440
      1VCS= ZERO. CHECK DATA ENTRIES ON CARD NUMBER",I5/) 005450
1004  FORMAT(1X,"NUMBER OF LTC'S INPUT EXCEEDS 14XLTG=",I5, "LINE 005460
      1)DATA CARD NUMBER ",I5/)                          005470
30   1005  FORMAT(1X,"NUMBER OF LINES INPJT HAS EXCEEDED NMAX=",I5,". LINE I 005480
      1)NPJT DATA CARD NUMBER=",I5/)                    005490
1006  FORMAT(1X,"NUMBER OF PHASE-SHIFTERS HAS EXCEEDED MAXPH=",I5,". LI 005500
      1)NE INPUT DATA CARD NUMBER=",I5/)                005510
1007  FORMAT(1X,"NUMBER OF TRANSFORMERS HAS EXCEEDED MAXTR=",I5,". LINE 005520
      1)DATA CARD NUMBER=",I5/)                          005530
35   1008  FORMAT(1X,"PROGRAM ERROR! STATEMENT NO. 500! SHOULD'NT REACH THIS 005540
      1)POINT IF ID IS LESS THAN 5. DEBJG11"/)          005550
1010  FORMAT(1X,"ERROR ON INPUT LINE CARD ",I3," WITH SB ",I3," AND E9 " 005560
      1,I3,". CHECK CARD FORMAT AND VALUES."/)        005570
40   1011  FORMAT(1X,"ERROR WITH ID OR C VALUE. LINE CARD",I3," WITH SB= ", 005580
      1I3," AND E9= ",I3," ."/)                          005590
1019  FORMAT(///1X,70(1H)/1X,1H*,68X,1H*/1X,1H*,T29,"LINDATA SUBROUTINE 005600
      1",T71,1H*/1X,1H*,T20,"ASSEMBLED INPJT LINE DATA (OHMS)",T71,14* 005610
      21X,1H*,68X,1H*/1X,70(1H)///)                    005620
45   1020  FORMAT(///1X,70(1H)/1X,1H*,69X,1H*/1X,1H*,T29,"LINDATA SUBROUTINE 005630
      1",T71,1H*/1X,1H*,T19,"ASSEMBLED INPJT LINE DATA (PER-UNIT)",T71, 005640
      21H*/1X,1H*,68X,1H*/1X,70(1H)///)                005650
1021  FORMAT(5X,"CONDUCTOR NO. ",I3/5X,"FROM - TO",5X,"RE(Z)",8X,"IM(Z) 005660
      1,8X,"RE(Z)",7X,"I4(Z)"//6X,I3,2X,I3,3X,4(F10.4,3X)/) 005670
50   1022  FORMAT(5X,"TRANSFORMER, FIXED NO. ",I3/5X,"FROM - TO",5X,"RE(Z)", 005680
      19X,"IM(Z)",8X,"RE(Z)",7X,"I4(Z)"//5X,I3,2X,I3,3X, 005690
      24(F10.4,3X)/3X,"TAP",5X,"CONEC CODE"/2X,F7.5,5X,I3/) 005700
1023  FORMAT(5X,"TRANSFORMER, AUTO. NO. ",I3/5X,"FROM - TO",5X,"RE(Z)", 005710
      19X,"IM(Z)",8X,"RE(Z)",7X,"I4(Z)"//5X,I3,2X,I3,3X, 005720
      24(F10.4,3X)/3X,"TAP",5X,"CONEC CODE"/2X,F5.3,5X,I3/) 005730
55   1024  FORMAT(5X,"TRANSFORMER, LTC NO. ",I3/5X,"FROM - TO",5X,"RE(Z)", 005740
      19X,"IM(Z)",8X,"RE(Z)",7X,"I4(Z)"//5X,I3,2X,I3,3X,4(F10.4,3X)/5X, 005750
      2"TAP",7X,"TMN",7X,"THX",5X,"CONEC CODE"/3X,3(F7.5,3X),3X,I3/) 005760

```

```

60 1025 FORMAT(5X,"TRANSFORMER, PHASE-SHIFTER NO. ",I3/5X,"FROM - TO",5X, 005770
1"RE(Z)",8X,"IM(Z)",8X,"RE(Z)",7X,"IM(Z)"/6X,I3,2X,I3,3X,4(F10.4, 005780
23X)/5X,"TAP",7X,"PING",5X,"CODE CODE"/3X,F7.5,3X,F9.4,6X,I3/) 005790
1026 FORMAT(5X,"SERIES CAPACITOR NO. ",I3/5X,"FROM - TO",5X, 005800
1"RE(Z)",8X,"IM(Z)",8X,"RE(Z)",7X,"IM(Z)"/6X,I3,2X,I3,3X, 005810
24(F10.4,3X)/) 005820
65 1027 FORMAT(5X,"SERIES REACTOR NO. ",I3/5X,"FROM - TO",5X,"RE(Z)", 005830
13X,"IM(Z)",8X,"RE(Z)",7X,"IM(Z)"/5X,I3,2X,I3,3X,4(F10.4,3X)/) 005840
1028 FORMAT(3X,"OUT CODE 15. PRINT OF LINDATA CARDS FOLLOWS."/ 005850
11X,"SB EB VP ID L C PH S STR JM S; STRS DMCG DMGG REZN IM005860
22N NG REZ IMZ C4"/32X,"PHI VS TMN TMX TPI",11X,"TC"/) 005870
N=NL=NOPH=NOLTC=NOTR=LL1=LL2=LL3=LL4=0 005880
70 IF(DJT.EQ.1) GO TO 202 005890
IF(DJT.EQ.2.OR.OUT.EQ.3.OR.DJT.E2.5.OR.OUT.EQ.8) GO TO 209 005900
IF(DJT.EQ.10) GO TO 209 005910
WRITE(1,1020) 005920
GO TO 209 005930
75 202 WRITE(1,1019) 005940
209 IF(DJT.EQ.15) WRITE(2,1029) 005950
210 READ(5,200) SB,EB,VP,ID,L,C,PH,S,STR,DM,VS,TMNM,TMXX,TPI,ZN,TC,Z, 005960
12M 005970
IF(EB.EQ.8) GO TO 330 005980
IF(DJT.EQ.15) GO TO 921 005990
IF(S.EQ.0) S=BKVA 006000
ZB=VP*VP*1000/S 006010
N=4+1 006020
IF(VP.EQ.8.OR.ID.E2.8.OR.P4.E7.8) GO TO 859 006030
85 NL=NL+2 006040
IF(NL.GT.NMAX) GO TO 855 006050
IF(ID.LT.5) LL1=LL1+1 006060
IF(IJ.EQ.8) GO TO 860 006070
IF(ID.GT.11) GO TO 850 006080
90 GO TO(211,212,300,400,500,500,500,820,830) ID 006090
C THE FOLLOWING CALCULATES RHO FOR COPPER AERIAL, AND SETS CONSTANT F2 006100
211 F2=.0636 006110
RHO=10.06*((241.+T)/266.) 006120
GO TO 214 006130
95 C THE FOLLOWING CALCULATES RHO FOR ACSR AERIAL, AND SETS CONSTANT F2: 006140
212 F2=.036 006150
RHO=17.34*((228.+T)/253.) 006160
C THE FOLLOWING SECTION CALCULATES THE DC RESISTANCE (OHMS/MILE), 006170
C AND THE AC RESISTANCE (OHMS/MILE) BASED ON THE INPUT TEMP. AND 006180
C AND FREQUENCY. 006190
214 DCRES=(RHO/(.98*S))*5280. 006200
RMR=F2*SQRT(F/DCRES) 006210
SER=(2384.*(64.-RMR**4)+RMR**8)/147456. 006220
SEI=(576.*RMR**2-RMR**6)/2306. 006230
SERP=(-1152.*RMR**3+RMR**7)/19432. 006240
SEIP=(192.*RMR-RMR**5)/384. 006250
ACRES=(DCRES*RMR/2.)*((SER*SEIP-BEI*BERP)/(BEIP*BEIP+BERP*BERP)) 006260
100 C THE FOLLOWING SECTION DETERMINES THE F1 FACTOR USED TO CALCULATE 006270
C GMR OF THE CONDUCTOR. IF STRANDING INFO. IS NOT INPUT WITH THE 006280
C DATA, AN AVERAGE VALUE OF F1 IS ASSIGNED BASED ON CONDUCTOR SIZE. 006290
C TWO DIFFERENT ROUTINES ARE USED: ONE IS FOR COPPER, AND THE OTHER 006300
C IS FOR ACSR. 006310
IF(STR.EQ.8.AND.ID.EQ.1) GO TO 220 006320
IF(STR.EQ.8.AND.ID.EQ.2) GO TO 221 006330

```

```

115      IF(IJ.EQ.2) GO TO 216      006340
      IF(STR.EQ.37) F1=.983      006350
      IF(STR.EQ.19) F1=.869      006360
      IF(STR.EQ.12) F1=.913      006370
      IF(STR.EQ.7) F1=.822      006380
120      IF(STR.EQ.3) F1=.842      006390
      IF(STR.EQ.1) F1=.779      006400
      SO TO 217      006410
      216 IF(STR.EQ.54) F1=.99      006420
      IF(STR.EQ.30) F1=1.056      006430
125      IF(STR.EQ.26) F1=1.008      006440
      IF(STR.EQ.6) GO TO 222      006450
      SO TO 217      006460
      220 IF(S.GT.500000) F1=.883      006470
      IF(S.EQ.500000) F1=.875      006480
130      IF(S.EQ.450000) JR.S.EQ.400000) F1=.969      006490
      IF(S.LE.350000) AND.S.GT.211500) F1=.891      006500
      IF(S.EQ.211600) F1=.869      006510
      IF(S.EQ.167800) F1=.8675      006520
      IF(S.EQ.133100) JR.S.EQ.105500) F1=.922      006530
135      IF(S.EQ.83690) F1=.872      006540
      IF(S.EQ.66370) JR.S.EQ.52630) F1=.814      006550
      IF(S.LT.52630) AND.S.GT.20820) F1=.91      006560
      IF(S.LT.26250) F1=.778      006570
      SO TO 217      006580
140      221 IF(S.GE.266800) F1=.94      006590
      222 IF(S.LT.266800) AND.S.GT.41590) F1=.3757      006600
      IF(S.EQ.41690) F1=.514      006610
      IF(S.EQ.33120) F1=.549      006620
      IF(S.EQ.26230) F1=.584      006630
145      C THIS SECTION CALCULATES GMR (DS), AND THEN THE INDUCTIVE REACTANCE      006640
      C OUT TO ONE FOOT RADIUS (XA, 0.445/MILE). THE CONDUCTOR SPACING      006650
      C FACTOR (DM) IS THEN DETERMINED. IF A VALUE FOR DM IS NOT INPUT,      006660
      C A DEFAULT VALUE IS ASSIGNED BASED ON THE LINE VOLTAGE. THE IN-      006670
      C DUCTIVE REAC. DUE TO COND. SPACING (XD) IS THEN CALCULATED AND      006680
      C ADDED TO XA AS THE TOTAL AC INJ. REACTANCE. THE POS. SEQ. IM-      006690
      C PEDANCE (Z) IS FORMED, AND MULTIPLIED BY THE CONDUCTOR LENGTH IN      006700
      C MILES.      006710
      217 DS=F1*SQRT((S*7.954E-7)/3.1415926)/12.      006720
      IF(DS.EQ.0) GO TO 859      006730
155      XA=4.657E-3*F*ALOG10(1./DS)      006740
      IF(DM.EQ.1) DM=218      006750
      218 IF(DM.EQ.0) AND.VP.LE.25) DM=4.63      006760
      IF(DM.EQ.0) AND.VP.GT.25) DM=12.57      006770
      XD=4.657E-3*F*ALOG10(DM)      006780
160      ACINJR=XA+XD      006790
      Z=C4PLX(ACRES,ACINJR)*L/5280.      006800
      C THE FOLLOWING SECTION CALCULATES THE ZERO SEQ. IMPEDANCE      006810
      RE=4.764E-3*F      006820
      XE=5.985E-3*F*ALOG10(4665600./RHJ1/F)      006830
165      Z01=ACRES+RE      006840
      Z02=XA+XE+2.*XD      006850
      Z0=C4PLX(Z01,Z02)*L/5280.      006860
      IF(TJ.EQ.0) GO TO 230      006870
170      C THIS SECTION ADJUSTS ZERO SEQ IMPEDANCE FOR GROUND WIRES.      006880
      C VARIABLES LISTED IN INSTRUCTIONS ARE EQUATED AS FOLLOWS:      006890
      C STRS=TMNN; NG=TC; DMCS=TMXX; J45G=TP1; SG=VS. ROUTINE USES      006900

```

	C SECTION 214-217 TO CALCULATE XA AND ACRES FOR GND WIRES.	006910
	230 Y=1	006920
175	IF(VS.EQ.0) GO TO 232	006930
	S=VS	006940
	IF(TMNN.EQ.0) GO TO 235	006950
	STR=TMNN	006960
	GO TO 214	006970
190	232 IF(I).EQ.1) GO TO 234	006980
	S=133100.	006990
	GO TO 235	007000
	234 S=105500.	007010
	235 STR=0	007020
	GO TO 214	007030
105	240 IF(TMXX.NE.0) GO TO 241	007040
	TMXX=4.	007050
	241 XD2=4.657E-3*F*ALOG10(TMXX)	007060
	IF(TPI.NE.0) GO TO 242	007070
	TPI=15.	007080
190	242 IF(TC.LT.2) TPI=1.	007090
	XD3=4.657E-3*F*ALOG10(TPI)	007100
	Z03=ACRES*3/TC+RE	007110
	Z04=3*XA/TC-3*XD3/TC+XE	007120
	Z05=CMPLX(Z03,Z04)	007130
195	IF(Z06.EQ.0) GO TO 059	007140
	Z05=RE	007150
	Z05=XE-3*XD2	007160
	Z045=CMPLX(Z05,Z06)	007170
	Z0=Z0-Z0AG*Z0AG*L/(5280.*Z03)	007180
200	Y=0	007190
	TMXX=TC=TMNN=TPI=VS=0	007200
	GO TO 900	007210
	C THE FOLLOWING SECTION PROCESSES ALUMINUM UG TABLET	007220
	300 L1=L/1000.	007230
205	IF(P4.EQ.3) GO TO 320	007240
	IF(S.GT.133100) GO TO 302	007250
	IF(S.EQ.133100) GO TO 303	007260
	IF(S.GT.66370) GO TO 304	007270
	IF(S.EQ.66370) 305,306	007280
210	302 IF(S.EQ.167800) GO TO 307	007290
	IF(S.GT.250000) GO TO 308	007300
	IF(S.EQ.250000) 309,310	007310
	303 Z=CMPLX(.1645,.0515)*L1	007320
	Z0=CMPLX(.4977,.1722)*L1	007330
215	GO TO 900	007340
	304 IF(S.EQ.105500) 311,312	007350
	305 Z=CMPLX(.328,.0509)*L1	007360
	Z0=CMPLX(.7116,.4256)*L1	007370
	GO TO 900	007380
220	306 IF(S.NE.41740) GO TO 315	007390
	Z=CMPLX(.5225,.0562)*L1	007400
	Z0=CMPLX(.8495,.5811)*L1	007410
	GO TO 900	007420
225	307 Z=CMPLX(.1285,.049)*L1	007430
	Z0=CMPLX(.426,.1209)*L1	007440
	GO TO 900	007450
	308 IF(S.EQ.350000) 313,314	007460
	309 Z=CMPLX(.087,.046)*L1	007470

	Z0=CHPLX(.3032,.0474)*L1	007400
230	30 TO 900	007490
	310 IF(S.NE.211600) GO TO 050	007500
	Z=CHPLX(.1025,.0463)*L1	007510
	Z0=CHPLX(.359,.0702)*L1	007520
	30 TO 900	007530
235	311 Z=CHPLX(.205,.0543)*L1	007540
	Z0=CHPLX(.577,.2604)*L1	007550
	30 TO 900	007560
	312 IF(S.NE.03690) GO TO 050	007570
	Z=CHPLX(.2615,.0369)*L1	007580
240	Z0=CHPLX(.6471,.3333)*L1	007590
	30 TO 900	007600
	313 Z=CHPLX(.063,.0402)*L1	007610
	Z0=CHPLX(.2212,.0175)*L1	007620
	30 TO 900	007630
245	314 IF(S.NE.300000) GO TO 050	007640
	Z=CHPLX(.073,.0419)*L1	007650
	Z0=CHPLX(.2504,.0306)*L1	007660
	30 TO 900	007670
	315 IF(S.EQ.16510) GO TO 310	007680
250	Z=CHPLX(.027,.0711)*L1	007690
	Z0=CHPLX(1.00,.73)*L1	007700
	30 TO 900	007710
	310 IF(S.NE.16510) GO TO 050	007720
	Z=CHPLX(1.315,.0702)*L1	007730
255	Z0=CHPLX(1.10,1.07)*L1	007740
	30 TO 900	007750
	320 IF(S.EQ.250000) GO TO 321	007760
	IF(S.GT.250000) GO TO 322	007770
	IF(S.GT.133100) GO TO 323	007780
260	IF(S.EQ.133100) 324,325	007790
	321 Z=CHPLX(.1034,.0529)*L1	007800
	Z0=CHPLX(.3004,.1124)*L1	007810
	30 TO 900	007820
	322 IF(S.GT.500000) GO TO 326	007830
265	IF(S.EQ.500000) 327,328	007840
	323 IF(S.EQ.211600) 329,330	007850
	324 Z=CHPLX(.170,.0926)*L1	007860
	Z0=CHPLX(.4692,.2474)*L1	007870
	30 TO 900	007880
270	325 IF(S.NE.105500) GO TO 333	007890
	Z=CHPLX(.210,.0956)*L1	007900
	Z0=CHPLX(.521,.2916)*L1	007910
	30 TO 900	007920
	326 IF(S.EQ.1000000) 331,332	007930
275	327 Z=CHPLX(.0664,.0570)*L1	007940
	Z0=CHPLX(.1653,.0436)*L1	007950
	30 TO 900	007960
	328 IF(S.NE.350000) GO TO 339	007970
	Z=CHPLX(.0032,.0763)*L1	007980
280	Z0=CHPLX(.2314,.07)*L1	007990
	30 TO 900	008000
	329 Z=CHPLX(.1170,.0059)*L1	008010
	Z0=CHPLX(.3495,.1412)*L1	008020
	30 TO 900	008030
285	330 IF(S.NE.167000) GO TO 050	008040

	Z=CMPLX(.143,.0894)*L1	000050
	Z=CMPLX(.4046,.1035)*L1	000060
	GO TO 900	000070
290	331 Z=CMPLX(.049,.05)*L1	000080
	Z=CMPLX(.0905,.0241)*L1	000090
	GO TO 900	000100
	332 IF(S.NE.750000) GO TO 850	000110
	Z=CMPLX(.0548,.0583)*L1	000120
295	Z=CMPLX(.1186,.0312)*L1	000130
	GO TO 900	000140
	333 IF(S.EQ.41740) GO TO 334	000150
	IF(S.GT.41740) 335,336	000160
	334 Z=CMPLX(.535,.105)*L1	000170
	Z=CMPLX(.909,.424)*L1	000180
300	GO TO 900	000190
	335 IF(S.EQ.66370) GO TO 337	000200
	IF(S.NE.83690) GO TO 850	000210
	Z=CMPLX(.276,.097)*L1	000220
305	Z=CMPLX(.635,.330)*L1	000230
	GO TO 900	000240
	336 IF(S.EQ.16510) GO TO 338	000250
	IF(S.NE.26250) GO TO 850	000260
	Z=CMPLX(.047,.111)*L1	000270
310	Z=CMPLX(1.48,.466)*L1	000280
	GO TO 900	000290
	337 Z=CMPLX(.344,.1)*L1	000300
	Z=CMPLX(.705,.37)*L1	000310
	GO TO 900	000320
	338 Z=CMPLX(1.335,.110)*L1	000330
315	Z=CMPLX(2.27,.51)*L1	000340
	GO TO 900	000350
	339 IF(S.NE.380000) GO TO 850	000360
	Z=CMPLX(.093,.0797)*L1	000370
320	Z=CMPLX(.2698,.0922)*L1	000380
	GO TO 900	000390
	C THE FOLLOWING SECTION PROCESSES COPPER UG CABLE:	000400
	400 L2=L/1000.	000410
	IF(PH.EQ.3) GO TO 420	000420
325	IF(S.GT.105500) GO TO 402	000430
	IF(S.EQ.105500) GO TO 403	000440
	IF(S.66370) 404,405,405	000450
	402 IF(S.167000) 407,408,409	000460
	403 Z=CMPLX(.123,.0543)*L2	000470
330	Z=CMPLX(.4218,.1180)*L2	000480
	GO TO 900	000490
	404 IF(S.NE.41740) GO TO 410	000500
	Z=CMPLX(.318,.0662)*L2	000510
	Z=CMPLX(.7046,.4279)*L2	000520
	GO TO 900	000530
335	405 Z=CMPLX(.2,.0609)*L2	000540
	Z=CMPLX(.5749,.2392)*L2	000550
	GO TO 900	000560
	406 IF(S.NE.83690) GO TO 850	000570
	Z=CMPLX(.159,.0563)*L2	000580
340	Z=CMPLX(.4939,.1706)*L2	000590
	GO TO 900	000600
	407 IF(S.NE.133100) GO TO 850	000610

	Z=CHPLX (.099,.0515)*L2	000620
	Z0=CHPLX (.3563,.0757)*L2	000630
345	GO TO 900	000640
	400 Z=CHPLX (.079,.049)*L2	000650
	Z0=CHPLX (.2950,.0446)*L2	000660
	GO TO 900	000670
	409 IF (S.NE.211600) GO TO 412	000680
350	Z=CHPLX (.0625,.0453)*L2	000690
	Z0=CHPLX (.2386,.0212)*L2	000700
	GO TO 900	000710
	410 IF(S.EQ.26250) GO TO 411	000720
	IF(S.NE.16510) GO TO 850	000730
355	Z=CHPLX(.799,.0732)*L2	000740
	Z0=CHPLX(1.01,.735)*L2	000750
	GO TO 900	000760
	411 Z=CHPLX(.501,.0711)*L2	000770
	Z0=CHPLX(.802,.535)*L2	000780
360	GO TO 900	000790
	412 IF(S-300000) 413,414,415	000800
	413 IF(S.NE.250000) GO TO 850	000810
	Z=CHPLX(.0527,.0444)*L2	000820
	Z0=CHPLX(.2003,.0133)*L2	000830
365	GO TO 900	000840
	414 Z=CHPLX(.044,.0418)*L2	000850
	Z0=CHPLX(.1716,.0034)*L2	000860
	GO TO 900	000870
	415 IF(S.NE.350000) GO TO 850	000880
370	Z=CHPLX(.071,.0736)*L2	000890
	Z0=CHPLX(.1751,.0511)*L2	000900
	GO TO 900	000910
	420 IF (S.GT.250000) GO TO 421	000920
	IF (S.EQ.250000) GO TO 422	000930
375	IF (S.GT.133100) GO TO 423	000940
	IF (S.EQ.133100) 424,425	000950
	421 IF (S.GT.500000) GO TO 425	000960
	IF (S.EQ.500000) 427,428	000970
380	422 Z=CHPLX(.078,.0774)*L2	000980
	Z0=CHPLX(.2006,.0538)*L2	000990
	GO TO 900	001000
	423 IF (S.EQ.211600) 429,430	001010
	424 Z=CHPLX(.1177,.0311)*L2	001020
	Z0=CHPLX(.349,.144)*L2	001030
385	GO TO 900	001040
	425 IF (S.NE.105500) GO TO 433	001050
	Z=CHPLX(.1448,.0343)*L2	001060
	Z0=CHPLX(.4063,.1854)*L2	001070
	GO TO 900	001080
390	426 IF (S.EQ.1000000) 431,432	001090
	427 Z=CHPLX (.055,.058)*L2	001100
	Z0=CHPLX(.106,.0295)*L2	001110
	GO TO 900	001120
	428 IF (S.NE.350000) GO TO 440	001130
395	Z=CHPLX (.0652,.063)*L2	001140
	Z0=CHPLX(.1494,.0416)*L2	001150
	GO TO 900	001160
	429 Z=CHPLX(.0849,.0515)*L2	001170
	Z0=CHPLX(.2317,.0728)*L2	001180

400	30 TO 900	009190
	430 IF (S.NE.167000) GO TO 850	009200
	Z=CMPLX (.0984,.0858)*L2	009210
	Z0=CMPLX(.2906,.1044)*L2	009220
	30 TO 900	009230
405	431 Z=CMPLX(.0402,.0364)*L2	009240
	Z0=CMPLX(.0554,.0137)*L2	009250
	30 TO 900	009260
	432 IF (S.NE.750000) GO TO 850	009270
	Z=CMPLX(.046,.0452)*L2	009280
410	Z0=CMPLX(.0724,.0222)*L2	009290
	30 TO 900	009300
	433 IF(S-41740) 436,434,435	009310
	434 Z=CMPLX(.33,.106)*L2	009320
	Z0=CMPLX(.924,.244)*L2	009330
415	30 TO 900	009340
	435 IF(S.EQ.66370) GO TO 437	009350
	IF(S.NE.83690) GO TO 850	009360
	Z=CMPLX(.17,.097)*L2	009370
	Z0=CMPLX(.476,.203)*L2	009380
420	GO TO 900	009390
	436 IF(S.EQ.16510) GO TO 439	009400
	IF(S.NE.26250) GO TO 850	009410
	Z=CMPLX(.52,.111)*L2	009420
	Z0=CMPLX(1.456,.265)*L2	009430
425	30 TO 900	009440
	437 Z=CMPLX(.220,.1)*L2	009450
	Z0=CMPLX(.616,.22)*L2	009460
	30 TO 900	009470
	438 Z=CMPLX(.81,.118)*L2	009480
430	Z0=CMPLX(1.9,.295)*L2	009490
	GO TO 900	009500
	440 IF(S.NE.300000) GO TO 850	009510
	Z=CMPLX(.071,.0716)*L2	009520
	Z0=CMPLX(.1751,.0511)*L2	009530
435	30 TO 900	009540
	C ALL TRANSFORMER PROCESSING ENTERS AT THIS STATEMENT TO UPDATE THE	009550
	C NOTR COUNTER. A CHECK IS MADE TO DETERMINE IF THE MAXIMUM NO. OF	009560
	C TRANSFORMERS HAS BEEN EXCEEDED: IF NOT, A BRANCH IS MADE TO THE	009570
	C APPROPRIATE STATEMENT NO. TO COMPLETE PROCESSING OF THE PARTICULAR	009580
440	C TYPE OF TRANSFORMER. IMPEJ, WHEN TC=0, IS PERCENT.	009590
	C TXFR IMPEJ. CAN BE IN OHMS,PERUNIT, OR NOT SPECIFIED. TC VARIABLE	009600
	C CONTRLS. 0=NOT SPECIFIED,1=IN OHMS,2=IN PERUNIT. Z0E IS BASE	009610
	C IMPED. OF TXFR. PERUNIT IS CONVERTED TO OHMS WHEN TC=2.	009620
	500 NOTR=NOTR+1	009630
445	IF(NOTR.GT.NAXTR) GO TO 857	009640
	C FOLLOWING 3 STATEMENTS BLOCK C CODES 6 TO 10 FOR FIXED,TCUL,	009650
	C AND AUTO TXFRS. CODES 6 TO 10 ARE FOR 3 WINDING TXFRS.	009660
	IF(C.GT.5.AND.ID.E2.6) GO TO 850	009670
	IF(C.GT.5.AND.ID.E3.7) GO TO 850	009680
450	IF(C.GT.5.AND.ID.E1.5) GO TO 850	009690
	30 TO (850,850,850,850,510,500,700,900,850,850) ID	009700
	C THE FOLLOWING SECTION PROCESSES FIXED TRANSFORMERS	009710
	510 LL2=LL2+1	009720
	IF(VP.GT.230) GO TO 852	009730
455	IF(TPI.EQ.0) TPI=1.0	009740
	IADD=40	009750

	IF(TC.EQ.1) GO TO 501	009760
	IF(TC.EQ.0) GO TO 519	009770
	Z=Z0F*Z	009780
460	GO TO 501	009790
	519 IF (PM.EQ.3) S=S/3.	009800
	IF (S.GT.500) GO TO 520	009810
	IF (S.GE.100) GO TO 501	009820
465	IF (S.GE.50) GO TO 502	009830
	IF (S.GE.25) GO TO 503	009840
	IF (S.GE.10) GO TO 504	009850
	IF (S.GE.1) GO TO 505	009860
	GO TO 501	009870
470	501 IF (VP.GT.2.5) GO TO 506	009880
	Z1=.00375*(S-100)+3.3	009890
	X=.004*(S-100)+3.1	009900
	GO TO 500	009910
	502 IF (VP.GT.2.5) GO TO 507	009920
	Z1=.018*(S-50)+2.4	009930
475	X=.02*(S-50)+2.1	009940
	GO TO 500	009950
	503 IF (VP.GT.2.5) GO TO 508	009960
	Z1=.004*(S-25)+2.5	009970
	X=.004*(S-25)+2.0	009980
480	GO TO 500	009990
	504 IF (VP.GT.2.5) GO TO 509	010000
	Z1=.02*(S-10)+2.2	010010
	X=.0333*(S-10)+1.5	010020
	GO TO 500	010030
485	505 IF (VP.GT.2.5) GO TO 510	010040
	Z1=2.2	010050
	X=.044*(S-1)+1.1	010060
	GO TO 500	010070
490	506 IF (VP.GT.15) GO TO 511	010080
	Z1=.0045*(S-100)+3.2	010090
	X=.005*(S-100)+2.9	010100
	GO TO 500	010110
	507 IF (VP.GT.15) GO TO 512	010120
	Z1=.014*(S-50)+2.5	010130
495	X=.016*(S-50)+2.1	010140
	GO TO 500	010150
	508 IF (VP.GT.15) GO TO 513	010160
	Z1=.008*(S-25)+2.3	010170
	X=.016*(S-25)+1.7	010180
500	GO TO 500	010190
	509 IF (VP.GT.15) GO TO 514	010200
	Z1=.0067*(S-10)+2.4	010210
	X=.0267*(S-10)+1.3	010220
	GO TO 500	010230
505	510 IF (VP.GT.15) GO TO 515	010240
	Z1=.044*(S-1)+2.0	010250
	X=.0556*(S-1)+.8	010260
	GO TO 500	010270
510	511 IF (VP.GT.25) GO TO 515	010280
	Z1=5.2	010290
	X=.0025*(S-100)+5.0	010300
	GO TO 500	010310
	512 IF (VP.GT.25) GO TO 516	010320

	Z1=5.2	010330
515	K=.002*(S-50)+4.3	010340
	GO TO 500	010350
	513 IF(VP.GT.25) GO TO 651	010360
	Z1=5.2	010370
	K=.004*(S-25)+4.8	010380
520	GO TO 500	010390
	514 IF(VP.GT.25) GO TO 651	010400
	Z1=5.2	010410
	K=.0267*(S-10)+4.4	010420
	GO TO 500	010430
525	515 Z1=6.5	010440
	K=.00025*(S-100)+6.3	010450
	GO TO 500	010460
	516 Z1=5.5	010470
	K=5.3	010480
530	GO TO 500	010490
	520 IF(PH.EQ.3) S=3.*S	010500
	IF(C.GT.5) GO TO 550	010510
	IF(VP.GT.34.5) GO TO 521	010520
	K=SQR(VP)+2.2*(1./ALOG10(VP))	010530
535	GO TO 579	010540
	521 K=SQR(VP)+2.*(1./ALOG(VP))	010550
	GO TO 579	010560
	550 K=2.*SQRT(VP)+6.4*(1./ALOG10(VP))	010570
	579 Z=CMPLX(0.,X)	010580
540	IF(IJ.EQ.6) GO TO 550	010590
	IF(ID.EQ.7) GO TO 710	010600
	IF(ID.EQ.8) GO TO 901	010610
	GO TO 501	010620
	500 IF(PH.EQ.3) S=3.*S	010630
545	Z=SQR(Z1*Z1-X*X)	010640
	Z=CMPLX(R,X)	010650
	IF(IJ.EQ.6) GO TO 550	010660
	IF(IJ.EQ.7) GO TO 710	010670
	IF(C.GT.5) Z=1.732*Z	010680
550	501 Z0=Z+3.*Z	010690
	GO TO 901	010700
	C THE FOLLOWING SECTION PROCESSES AUTOTRANSFORMERS	010710
	600 IF(TPI.EQ.0) TPI=1.0	010720
	LL3=LL3+1	010730
555	I400=60	010740
	IF(PH.EQ.1) SKVA=9<VA/3.	010750
	Z0=(1000.*VP*VP)/SKVA	010760
	IF(PH.EQ.1) SKVA=9<VA1	010770
560	IF(TZ.EQ.1) GO TO 649	010780
	IF(TZ.EQ.0) GO TO 519	010790
	Z=Z0**Z	010800
	649 GO TO(662,661,654,561,667,654,663,656) C.	010810
	650 GO TO(652,651,654,651,657,654,653,656) C	010820
	651 Z=.4167*Z	010830
565	661 Z0=Z+300.*ZN/Z0	010840
	GO TO 901	010850
	652 Z=.4167*Z	010860
	662 Z0=CMPLX(0.,0.)	010870
	GO TO 901	010880
570	654 Z=.4167*Z	010890

	664	Z0=Z+40.*ZN/Z0	010900
		GO TO 901	010910
	665	Z=4167*Z	010920
575	665	Z0=1.87*Z	010930
		GO TO 901	010940
	666	Z=229*Z	010950
	666	Z0=Z+100.*ZN/Z0	010960
		GO TO 901	010970
	667	Z=4167*Z	010980
580	667	Z0=3.36*Z+40.*ZN/Z0	010990
		GO TO 901	011000
	C THE FOLLOWING SECTION PROCESSES LOAD TAP-CHANGERS:		
	700	NOLTC=NOLTC+1	011010
		IF(NOLTC.GT.MAXLTC) GO TO 854	011020
585		IF(TPI.EQ.0) TPI=1.0	011030
		IADD=70	011040
		NN=VP/VS	011050
		IF(TMNN.EQ.0) TMNN=.9	011060
		IF(TMXX.EQ.0) TMXX=1.1	011070
590		IF(TC.EQ.1) GO TO 710	011080
		IF(TC.EQ.0) GO TO 519	011090
		Z=Z0*Z	011100
	710	IF(P1.EQ.1) BKVA=BKVA/3.	011110
		Z0=(1000.*VP*VP)/BKVA	011120
595		Z0=Z+300.*ZN/(NN*Z0)	011130
		IF(PH.EQ.1) BKVA=BKVA1	011140
		GO TO 901	011150
	C THE FOLLOWING SECTION PROCESSES PHASE-SHIFTERS:		
	800	NOPH=NOPH+2	011160
600		LL4=NOPH/2	011170
		IF(LL4.GT.MAXPH) GO TO 856	011180
		IF(TPI.EQ.0) TPI=1.0	011190
		IADD=90	011200
605		IF(TC.EQ.0) GO TO 550	011210
		IF(TC.EQ.1) GO TO 501	011220
		Z=Z*Z0E	011230
	801	IF(PH.EQ.1) BKVA=BKVA/3	011240
		Z0=(1000.*VP*VP)/BKVA	011250
610		IF(PH.EQ.1) BKVA=BKVA1	011260
		Z=300.*ZN/Z0	011270
		GO TO (802,802,802,803,803,803,804,805) C	011280
	802	IF(C.EQ.1) Z0=2.23*Z	011290
		IF(C.EQ.2) Z0=Z*(3.+6*Y)	011300
615		IF(C.EQ.3) Z0=1.6*Z+.24*Y	011310
		Z=1.4*Z	011320
		GO TO 901	011330
	803	IF(C.EQ.4) Z0=Z	011340
		IF(C.EQ.5) Z0=1.2*Z+.27*Y	011350
		IF(C.EQ.6) Z0=.8*Z+.27*Y	011360
620		Z=1.4*Z	011370
		GO TO 901	011380
	804	Z0=2.23*Z	011390
		Z=.9*Z	011400
		GO TO 901	011410
625	805	Z0=3.9*Z	011420
		Z=.8*Z	011430
		GO TO 901	011440
			011450
			011460

	C THE FOLLOWING SECTION PROCESSES SERIES CAPACITORS:	011470
630	820 NOCAP=NOCAP+1	011480
	IF(TC.EQ.0) GO TO 822	011490
	IF(TC.EQ.2) Z=Z*Z0E	011500
	Z0=Z	011510
	GO TO 901	011520
635	822 Z1=VP*VP/S	011530
	Z=C4PLX(0.,-Z1)	011540
	KE=5.985E-3*F*ALOG10(4665600.*R4J1/F)	011550
	Z01=Z1+XE	011560
	Z0=C4PLX(0.,-Z01)	011570
	GO TO 901	011580
640	C THE FOLLOWING SECTION PROCESSES SERIES REACTORS:	011590
	830 NOREAC=NOREAC+1	011600
	IF(TC.EQ.0) GO TO 832	011610
	IF(TC.EQ.2) Z=Z*Z0E	011620
	Z0=Z	011630
645	GO TO 901	011640
	832 Z1=VP*VP/S	011650
	Z=C4PLX(0.,Z1)	011660
	KE=5.985E-3*F*ALOG10(4665600.*R4J1/F)	011670
	Z01=Z1+XE	011680
650	Z0=C4PLX(0.,Z01)	011690
	GO TO 901	011700
	850 WRITE(2,1000) N	011710
	GO TO 840	011720
655	851 WRITE(2,1001) N	011730
	GO TO 840	011740
	852 WRITE(2,1002) N	011750
	GO TO 840	011760
	853 WRITE(2,1003) N	011770
660	840 IERR=IERR+1	011780
	GO TO 210	011790
	854 WRITE(2,1004) MAXLTC,N	011800
	IERR=IERR+1	011810
	RETURN	011820
665	855 WRITE(2,1005) NMAX,N	011830
	IERR=IERR+1	011840
	RETURN	011850
	856 WRITE(2,1006) MAXP4,N	011860
	IERR=IERR+1	011870
	RETURN	011880
670	857 WRITE(2,1007) MAXTR,N	011890
	IERR=IERR+1	011900
	RETURN	011910
	858 WRITE(2,1008)	011920
	IERR=IERR+1	011930
675	GO TO 210	011940
	859 WRITE(2,1010) N,SB,EB	011950
	IERR=IERR+1	011960
	RETURN	011970
680	860 WRITE(2,1011) N,SB,EB	011980
	IERR=IERR+1	011990
	RETURN	012000
	900 S=0.	012010
	IF(ID.LT.3) GO TO 901	012020
	F1=F/60.	012030

685	Z1=F1*AIMAG(Z)	012040
	Z2=F1*AIMAG(Z0)	012050
	F2=F-60.	012060
	Z3=F2*REAL(Z)	012070
	Z4=F2*REAL(Z0)	012080
690	Z=CMPLX(Z3,Z1)	012090
	Z0=CMPLX(Z4,Z2)	012100
901	Z1=CABS(Z)	012110
	Z2=CABS(Z0)	012120
	IF(T.GE.1) S=0.	012130
695	IF(Z1.EQ.0.AND.Z2.EQ.0) GO TO 953	012140
	CALL PERUNIT(Z,Z0,P4,VP,S)	012150
	IF(OUT.EQ.10) GO TO 909	012160
	IF(OUT.EQ.2.OR.OUT.EQ.3.OR.OUT.EQ.5.OR.OUT.EQ.8) GO TO 909	012170
	IF(OUT.EQ.1) Z=Z*VP*VP*1000/9KVA	012180
700	IF(OUT.EQ.1) Z0=Z0*VP*VP*1000/9KVA	012190
	S=0.	012200
	DO T(902,902,902,902,903,904,905,906,907,908) IO	012210
902	WRITE(1,1021) LL1,SB,EB,Z,Z0	012220
	DO T 909	012230
705	903 WRITE(1,1022) L2,SB,EB,Z,Z0,TPI,C	012240
	DO T 909	012250
904	WRITE(1,1023) LL3,SB,EB,Z,Z0,TPI,C	012260
	DO T 909	012270
710	905 WRITE(1,1024) NOLTC,SB,EB,Z,Z0,TPI,THNN,THXX,C	012280
	DO T 909	012290
906	WRITE(1,1025) LL4,SB,EB,Z,Z0,TPI,04,C	012300
907	WRITE(1,1026) NOCAP,SB,EB,Z,Z0	012310
	DO T 909	012320
715	908 WRITE(1,1027) NOREAC,SB,EB,Z,Z0	012330
909	IF(OUT.EQ.1) CALL PERUNIT(Z,Z0,P4,VP,S)	012340
	IF(IPHASE(SB).NE.3.AND.SB.NE.0) IPHASE(SB)=PH	012350
	IF(IPHASE(EB).NE.3) IPHASE(EB)=P4	012360
	LINA(NL)=EB	012370
	LINB(NL)=SB	012380
720	LINA(NL-1)=SB	012390
	LINB(NL-1)=EB	012400
	C PHI=04,THIS SECTION FOR PHASE-SHIFTER TRANS.	012410
	IF(ON.EQ.0) GO TO 910	012420
	LP4A(NOPH)=EB	012430
725	LP4B(NOPH)=SB	012440
	LP4A(NOPH-1)=S3	012450
	LP4B(NOPH-1)=EB	012460
	PHANG(NOPH-1)=04*.01745329	012470
	PHANG(NOPH)=-PHANG(NOPH-1)	012480
730	910 IF(I7.LT.5) GO TO 920	012490
	LTRA(NOTR)=SB	012500
	LTRB(NOTR)=EB	012510
	TAP(NOTR)=TPI	012520
	THN(NOTR)=THNN	012530
735	THX(NOTR)=THXX	012540
	IC3(NOTR)=C+IADD	012550
	920 Y=1./Z	012560
	Z(NL)=REAL(Y)	012570
	S(NL)=AIMAG(Y)	012580
740	Z0(NL)=REAL(Z0)	012590
	S0(NL)=AIMAG(Z0)	012600

	6(NL-1)=6(NL)	012610
	9(NL-1)=9(NL)	012620
	ZOR(NL-1)=ZOR(NL)	012630
745	ZOI(NL-1)=ZOI(NL)	012640
	921 IF(OUT.NE.15) GO TO 929	012650
	WRITE(2,200) SB,IS,VP,LD,L,C,P4,S,STR,D4,VS,T4N,TMX,TPI,ZN,TZ,Z,	012660
	ICH	012670
	929 CONTINUE	012680
750	GO TO 210	012690
	930 IF(IERR.NE.0) RETURN	012700
	IF(DJT.EQ.15) RETURN	012710
	CALL LSORT(2,LINA,LINB,NL,6,9,ZOR,ZOI,0)	012720
	CALL LSORT(2,LTRA,LTRB,NOTR,5,T4S,T4N,TMX,0,ICG)	012730
755	CALL LSORT(2,LP4A,_P4B,NOP4,1,P4MS,0,0,0,0)	012740
	DO 945 I=1,250	012750
	945 CONEC(I)=0	012760
	IZ=0	012770
	J=1	012780
760	DO 950 I=1,NL	012790
	IF(LINA(I).EQ.0) GO TO 947	012800
	948 IF(LINA(I).NE.J) GO TO 949	012810
	CONEC(J)=CONEC(J)+1	012820
	GO TO 950	012830
765	949 J=J+1	012840
	GO TO 948	012850
	947 IZ=IZ+1	012860
	950 CONTINUE	012870
	NBUS=J	012880
770	RETURN	012890
	END	012900

1	OVERLAY (BASIC,1,2)	012910
	PROGRAM LINEZ	012920
	REAL L	012930
	COMPLEX Z,Z0,ZN,ZZ,ZZ0,Y	012940
5	INTEGER PH,C,SB,EB,CON,CHS,SCOP,OUT,CH,CONEC	012950
	COMMON /COMA/CON,CMG,LODOP,SCOP,IMP,OUT,F,T,8KVA,RMO1,NMAX,MAXTR,	012960
	14ALTC,MAXPH,ISYS,3KVA1	012970
	COMMON /COMB/LINA(1450),LINB(1450),J(1450),B(1450),P(250),Q(250),	012980
	1.P49(50),PHANG(50),LTRA(250),LTRB(250),TAP(250),TMM(250),V(250),	012990
10	ZFMX(250),IUBPP(250),ANG(250),IPUS(250),O9P(250),U9P(3000),	013000
	39USPHE(250),LPHA(50),LIST(250),IUBP(250),ZMIY(250),QMAX(250),	013010
	4J8PP(250),UBPP(3000),J9P(3000),J9PP(3000),ICC(250),DLP(250),	013020
	SZBI(1450),ZOR(1450),80IA(250),CONEC(250),JLQ(250),IPHASE(250)	013030
	COMMON /COMC/ NA(250),NB(250),JCC(1000),OU(1000),IOB(250)	013040
15	COMMON /CONST/ N9US,NL,ISS,IPV,LL1,LL2,LL3,LL4,NOTR,IZ,NOLTC	013050
	1,ITR1,ITR2,PTOL,DTOL,NLC	013060
	COMMON /SAVE/IERR	013070
	1000 FORMAT(2I3,F5.0,I2,I1,5F6.0,"4.0,3F5.0,I2,I2,2F6.0,I1)	013080
20	1001 FORMAT(1X,"INPUT CODE ERROR! 'I' DATA CODE NOT COMPATIBLE WITH"	013090
	1" 'C' CODE. LINE CARD NO. ",I3," WITH SB=",I3," AND EB=",I3/)	013100
	1002 FORMAT(1X,"ERROR ON LINE CARD NO.",I3," WITH SB=",I3," AND EB=",I3)	013110
	1,""/1X,"VALUES VP, ID OR PA NOT VALID."/)	013120
	1003 FORMAT(1X,"ERROR HAS RESULTED IN POSITIVE AND ZERO SEQUENCE IMPEDA	013130
	1NCES= ZERO. CHECK DATA ENTRIES ON CARD NUMBER",I5/)	013140
25	1004 FORMAT(1X,"NUMBER OF LTC'S INPUT EXCEEDS MAXLTC= ",I5, "LINE	013150
	1DATA CARD NUMBER ",I5/)	013160
	1005 FORMAT(1X,"NUMBER OF LINES INPUT HAS EXCEEDED NMAX=",I5,". LINE	013170
	1INPUT DATA CARD NUMBER=",I5/)	013180
	1006 FORMAT(1X,"NUMBER OF PHASE-SHIFTERS HAS EXCEEDED MAXPH=",I5,". LI	013190
	1NE INPUT DATA CARD NUMBER=",I5/)	013200
30	1007 FORMAT(1X,"NUMBER OF TRANSFORMERS HAS EXCEEDED MAXTR=",I5,". LINE	013210
	1DATA CARD NUMBER=",I5/)	013220
	1019 FORMAT(///1X,70(140)/1X,140,68X,140/1X,140,T29,"LINEZ SUBROUTINE"	013230
	1,T71,140/1X,140,T20,"ASSEMBLED INPUT LINE DATA (OHMS)",T71,140/	013240
35	21X,140,68X,140/1X,70(140)/)	013250
	1020 FORMAT(///1X,70(140)/1X,140,68X,140/1X,140,T29,"LINEZ SUBROUTINE"	013260
	1,T71,140/1X,140,T19,"ASSEMBLED INPUT LINE DATA (PER-UNIT)",T71,	013270
	21X/1X,140,68X,140/1X,70(140)/)	013280
	1021 FORMAT(5X,"CONDUCTOR NO. ",I3/5X,"FROM - TO",5X,"RE(Z)",8X,"IM(Z)"	013290
40	1,8X,"RE(Z0)",7X,"I(1(Z0))/5X,I3,2X,I3,3X,4(F10.4,3X)/)	013300
	1022 FORMAT(5X,"TRANSFORMER, FIXED NO. ",I3/5X,"FROM - TO",5X,"RE(Z)",	013310
	19X,"IM(Z)",8X,"RE(Z0)",7X,"IM(Z0)/5X,I3,2X,I3,3X,	013320
	24(F10.4,3X)/5X,"TAP",5X,"CONEC CODE"/3X,F7.5,6X,I3/)	013330
45	1023 FORMAT(5X,"TRANSFORMER, AUTO. NO. ",I3/5X,"FROM - TO",5X,"RE(Z)",	013340
	19X,"IM(Z)",8X,"RE(Z0)",7X,"IM(Z0)/5X,I3,2X,I3,3X,	013350
	24(F10.4,3X)/3X,"TAP",5X,"CONEC CODE"/2X,F5.3,6X,I3/)	013360
	1024 FORMAT(5X,"TRANSFORMER, LTC NO. ",I3/5X,"FROM - TO",5X,"RE(Z)",	013370
	19X,"IM(Z)",8X,"RE(Z0)",7X,"IM(Z0)/5X,I3,2X,I3,3X,4(F10.4,3X)/5X,	013380
	2"TAP",7X,"TMM",7X,"FMX",5X,"CONEC CODE"/3X,3(F7.5,3X),3X,I3/)	013390
50	1025 FORMAT(5X,"TRANSFORMER, PHASE-SHIFTER NO. ",I3/5X,"FROM - TO",5X,	013400
	1"RE(Z)",8X,"IM(Z)",8X,"RE(Z0)",7X,"IM(Z0)/6X,I3,2X,I3,3X,	013410
	24(F10.4,3X)/5X,"TAP",7X,"P4A4G",5X,"CONEC CODE"/3X,F7.5,3X,F9.4,	013420
	35X,I3/)	013430
55	1026 FORMAT(5X,"SERIES CAPACITOR NO. ",I3/5X,"FROM - TO",5X,	013440
	1"RE(Z)",8X,"IM(Z)",8X,"RE(Z0)",7X,"IM(Z0)/6X,I3,2X,I3,3X,	013450
	24(F10.4,3X)/)	013460
	1027 FORMAT(5X,"SERIES REACTOR NO. ",I3/5X,"FROM - TO",5X,"RE(Z)",	013470

	19X,"I4(Z)",8X,"RE(Z0)",7X,"I4(Z0)"/5X,I3,2X,I3,3X,4(F10.4,3X)/)1	013480
	1020 FORMAT(3X,"ERROR WITH INPUT DATA. LINE",I3," TO",I3," MUST HAVE	013490
60	150ME IMPEDANCE VALUE OTHER THAN ZERO.//)	013500
	N=NL=NOTR=NOPH=NOLTC=LL1=LL2=LL3=LL4=0	013510
	IF(OJT.EQ.1) GO TO 2	013520
	IF(OJT.EQ.2.OR.OUT.EQ.3.OR.OJT.E2.5.OR.OUT.EQ.8) GO TO 10	013530
	IF(OJT.EQ.10) GO TO 10	013540
65	WRITE(1,1020)	013550
	GO TO 10	013560
	2 WRITE(1,1019)	013570
	10 READ(5,1000) SB,E4,VP,IO,P4,L,ZZ,ZZ0,PHI,TPI,TMNN,TMXX,C,IUNIT,ZN	013580
	1,C4	013590
70	IF(E3.EQ.0) GO TO 330	013600
	C THE FOLLOWING 3 STATEMENTS BLOCK TXFR 3 CODES 6 TO 10 FOR	013610
	C FIXED,TCUL AND AUTO TXFRS, CODES 5-10 ARE FOR 3 WINDING TXFRS.	013620
	IF(C.GT.5.AND.ID.EQ.5) GO TO 900	013630
	IF(C.GT.5.AND.ID.EQ.6) GO TO 900	013640
75	IF(C.GT.5.AND.ID.EQ.7) GO TO 900	013650
	V=V+1	013660
	IF(VP.EQ.0.OR.ID.EQ.0.OR.P4.EQ.0) GO TO 858	013670
	NL=NL+2	013680
	IF(NL.GT.NMAX) GO TO 855	013690
80	Z1=CABS(ZZ)	013700
	Z2=CABS(ZZ0)	013710
	IF(Z1.EQ.0.AND.Z2.EQ.0) GO TO 853	013720
	IF(IJ.GT.4.AND.IJ.LT.9) GO TO 20	013730
	IF(IJ.EQ.9) GO TO 30	013740
85	IF(ID.EQ.10) GO TO 35	013750
	LL1=LL1+1	013760
	IF(L.EQ.0) L=1.0	013770
	Z=ZZ*L	013780
	IF(ZZ0.NE.0) GO TO 15	013790
90	IF(TPI.NE.0) GO TO 14	013800
	ZZ0=3.5*ZZ	013810
	GO TO 15	013820
	14 ZZ0=2.7*ZZ	013830
	15 Z0=ZZ0*L	013840
95	=0.	013850
	GO TO 900	013860
	20 NOTR=NOTR+1	013870
	IF(L.EQ.0.AND.IUNIT.EQ.1) L=BKVA	013880
	IF(IJUNIT.EQ.2) Z2=ZZ*BKVA/L	013890
100	IF(IJUNIT.EQ.2) Z20=ZZ0*BKVA/L	013900
	IF(NOTR.GT.MAXTR) GO TO 857	013910
	IF(TPI.EQ.0) TPI=1.0	013920
	=0.	013930
	GO TO(800,800,800,900,21,22,23,24,800,800) ID	013940
105	21 Z=Z2	013950
	ZZ0=ZZ0+3*ZN	013960
	IF(ZZ0.EQ.0.) ZZ0=ZZ	013970
	Z0=ZZ0	013980
	LL2=LL2+1	013990
110	IADD=40	014000
	GO TO 900	014010
	22 LL3=LL3+1	014020
	IADD=60	014030
	Z=ZZ	014040

115	ZZ0=ZZ0+3*ZN	014050
	IF(ZZ0.EQ.0.) ZZ0=ZZ	014060
	Z0=ZZ0	014070
	GO TO 900	014080
120	23 NOLTC=NOLTC+1	014090
	IF(NOLTC.GT.MAXLTC) GO TO 854	014100
	IF(TMNN.EQ.0) TMNN=.9	014110
	IF(TMXX.EQ.0) TMXX=1.1	014120
	IADD=70	014130
	Z=ZZ	014140
125	ZZ0=ZZ0+3*ZN	014150
	IF(ZZ0.EQ.0.) ZZ0=ZZ	014160
	Z0=ZZ0	014170
	GO TO 900	014180
130	24 NOPH=NOPH+1	014190
	LL4=NOPH/2	014200
	IF(LL4.GT.MAXP4) GO TO 855	014210
	IADD=90	014220
	Z=ZZ	014230
	ZZ0=ZZ0+3*ZN	014240
135	IF(ZZ0.EQ.0.) ZZ0=ZZ	014250
	Z0=ZZ0	014260
	GO TO 900	014270
	30 NOCAP=NOCAP+1	014280
140	IF(IJUNIT.EQ.2) ZZ=ZZ*BKVA/L	014290
	L=0.	014300
	Z=ZZ	014310
	IF(ZZ0.EQ.0) ZZ0=ZZ	014320
	Z0=ZZ0	014330
	GO TO 900	014340
145	35 NOREAC=NOREAC+1	014350
	IF(IJUNIT.EQ.2) ZZ=ZZ*BKVA/L	014360
	L=0.	014370
	Z=ZZ	014380
	IF(ZZ0.EQ.0) ZZ0=ZZ	014390
150	Z0=ZZ0	014400
	GO TO 900	014410
	800 WRITE(2,1001) N,S0,EB	014420
	GO TO 899	014430
155	853 WRITE(2,1003) N	014440
	IERR=IERR+1	014450
	GO TO 10	014460
	854 WRITE(2,1004) MAXLTC,N	014470
	GO TO 899	014480
160	855 WRITE(2,1005) NMAX,N	014490
	GO TO 899	014500
	856 WRITE(2,1006) MAXPH,N	014510
	GO TO 899	014520
	857 WRITE(2,1007) MAXTR,N	014530
	GO TO 899	014540
165	858 WRITE(2,1002) N,S0,EB	014550
	899 IERR=IERR+1	014560
	RETJRN	014570
	900 IF(IJTY.EQ.1.AND.IJUNIT.GE.1) Z=Z*VP*VP*1000/BKVA	014580
170	IF(IJTY.EQ.1.AND.IJUNIT.GE.1) Z0=Z0*VP*VP*1000/BKVA	014590
	IF(IJTY.EQ.1) GO TO 901	014600
	IF(IJUNIT.EQ.8) CALL PERUNIT(Z,Z0,PH,VP,L)	014610

	IF(OUT.EQ.10) GO TO 909	014620
	IF(OUT.EQ.2.OR.OUT.EQ.3.OR.OUT.EQ.5.OR.OUT.EQ.8) GO TO 909	014630
175	901 GO TO(902,902,902,902,903,904,905,906,907,908) ID	014640
	902 WRITE(1,1021) LL1,SB,EB,Z,Z0	014650
	GO TO 909	014660
	903 WRITE(1,1022) LL2,SB,EB,Z,Z0,TPI,C	014670
	GO TO 909	014680
180	904 WRITE(1,1023) LL3,SB,EB,Z,Z0,TPI,C	014690
	GO TO 909	014700
	905 WRITE(1,1024) NOLTC,SB,EB,Z,Z0,TPI,TMNN,T4XX,C	014710
	GO TO 909	014720
	906 WRITE(1,1025) LL4,SB,EB,Z,Z0,PHI	014730
185	907 WRITE(1,1026) NOCAP,SB,EB,Z,Z0	014740
	GO TO 909	014750
	908 WRITE(1,1027) MOREAC,SB,EB,Z,Z0	014760
	909 IF(OUT.EQ.1) CALL PERJNIT(Z,Z0,PH,VP,L)	014770
	IF(IPHASE(SB).NE.3.AND.SB.NE.0) IPHASE(SB)=PH	014780
	IF(IPHASE(EB).NE.3) IPHASE(EB)=4	014790
190	LINA(NL)=EB	014800
	LINB(NL)=SB	014810
	LINA(NL-1)=SB	014820
	LINB(NL-1)=EB	014830
	IF(PHI.EQ.0) GO TO 910	014840
195	LPH4(NOPH)=EB	014850
	LPH3(NOPH)=SB	014860
	LPH4(NOPH-1)=SB	014870
	LPH3(NOPH-1)=EB	014880
	PHANG(NOPH-1)=PHI*.01745329	014890
200	PHANG(NOPH)=-PHANG(NOPH-1)	014900
	910 IF(ID.LT.5.OR.ID.GT.8) GO TO 920	014910
	LTRA(NOTR)=SB	014920
	LTRB(NOTR)=EB	014930
	TAP(NOTR)=TPI	014940
205	TMN(NOTR)=TMNN	014950
	TMX(NOTR)=TMXX	014960
	IC(NOTR)=C+IADD	014970
	920 IF(Z.EQ.0.) GO TO 960	014980
	Z=1./Z	014990
210	Z(NL)=REAL(Y)	015000
	Z(NL)=AIMAG(Y)	015010
	ZOR(NL)=REAL(Z0)	015020
	ZOI(NL)=AIMAG(Z0)	015030
	G(NL-1)=G(NL)	015040
215	B(NL-1)=B(NL)	015050
	ZOR(NL-1)=ZOR(NL)	015060
	ZOI(NL-1)=ZOI(NL)	015070
	GO TO 10	015080
	930 IF(IERR.NE.0) RETURN	015090
220	CALL LSORT(2,LINA,LINB,NL,4,3,B,ZOR,ZOI,0)	015100
	CALL LSORT(2,LTRA,LTRB,NOTR,5,TAP,T4N,TMX,0,IC)	015110
	CALL LSORT(2,LPH4,LPH3,NOP4,1,PHANG,0,0,0,0)	015120
	DO 945 I=1,250	015130
	C IZ=NUMBER OF CONNECTIONS TO REF BUS	015140
225	945 CONEC(I)=0	015150
	IZ=0	015160
	J=1	015170
	DO 950 I=1,NL	015180

```

230          IF(LINA(I).EQ.0) G) TO 947                                015190
          940 IF(LINA(I).NE.J) G) TO 949                                015200
          30NEC(J)=CONEC(J)+1                                          015210
          50 T) 950                                                    015220
          949 J=J+1                                                    015230
          50 T) 948                                                    015240
235          947 IZ=IZ+1                                                015250
          950 CONTINUE                                                015260
          48JS=J                                                        015270
          RETURN                                                        015280
240          960 4RITE(2,1028) 58,EB                                    015290
          IERR=IERR+1                                                  015300
          RETURN                                                        015310
          EN)                                                            015320

1          OVERLAY(LODFLO,2,0)                                         015330
          PR)SRAM FOLFLOW                                              015340
          COMPLEX Y,VA,VB,S,Z                                          015350
          INTEGER CON,CHG,SCJP,OUT,CONEC,A,C                          015360
          DIMENSION IOUM(50)                                           015370
          COMMON /COMA/CON,CHG,LOOP,SCOP,INP,OUT,F,T,8KVA,RHO1,NMAX,MAXTR, 015380
          19AKLTC,MAXPH,ISYS,3KVA1
          COMMON /COMB/LINA(1450),LINA3(1450),G(1450),B(1450),P(250),Q(250), 015390
          1,PHA(50),PHANG(50),TRA(250),LTR3(250),TAP(250),TMM(250),V(250), 015400
          2THK(250),UBPP(250),ANG(250),IBJS(250),DPP(250),UBP(3000), 015410
          3BUSN44E(250),LPHA(50),LIST(250),IUP(250),2MIN(250),2MAX(250), 015420
          4JBP(250),UBPP(3000),JBP(3000),JBP(3000),ICC(250),DL(250), 015430
          5Z(1450),ZOR(1450),ROIA(250),CONEC(250),JLQ(250),IP4ASE(250) 015440
          COMMON/COMC/ NA(250),NR(250),JCO.(1000),DU(1000),ID(250)    015450
          COMMON/CONST/ N9JS,NL,ISS,I*V,LL1,LL2,LL3,LL4,NOTR,IZ,NOLTC 015460
          1,ITR1,ITR2,PTOL,2TOL,NLC                                     015470
          COMMON/SUB/ BH                                               015480
1000 FORMAT(/T16,"**** CONVERGENCE NOT OBTAINED INI"/20X,I4,      015490
          1" DELTA THETA AND"/20X,I4," DELTA V ITERATIONS."/1X,      015500
          2"BUS NO.",3X,"VOLT(MAG)",3X,"VOLT(ANG)",5X,"DELTA P",7X, 015510
          3"DELTA Q"/)                                                015520
1001 FORMAT(/1X,70(1H)/1X,1H,59X,14/1X,1H,T30,                    015530
          1"SYSTEM SUMMARY",T71,14/1X,14,58X,1H/1X,1H,T25,        015540
          2"CONVERGENCE OBTAINED INI",T71,14/1X,1H,T24,I4,1X,      015550
          3"DELTA THETA AND",T71,14/1X,14,T24,I4,1X,                015560
          4"DELTA V ITERATIONS.",T71,14/1X,14,T71,14/1X,70(1H)/)    015570
1002 FORMAT(/1X,70(1H)/1X,1H,59X,14/1X,14,T27,"CALCULATED LINE FLOW 015580
          1WS",T71,1H/1X,1H,T4,"(THE LINE FLOWS ARE DEFINED POSITIVE WHEN F015590
          2LOADING OUT FROM THE BUS)",T71,14/1X,1H,58X,14/1X,70(1H)/T5, 015600
          3"LINE",T22,"POWER",15X,"LINE",12X,"POWER"/2X,"FROM",2X,"TO", 015610
          45X,"REAL",7X,"REACTIVE",5X,"FROM",2X,"TO",5X,"REAL",7X, 015620
          5"REACTIVE"/)                                              015630
1003 FORMAT(1X,2(I3,2X),1X,F9.5,3X,F9.5,3X,I3,2X,I3,2(3X,F9.5)/) 015640
1004 FORMAT(T23,"***** SLACK BUS POWER *****"/T10,"BUS NO.",T24, 015650
          1"REAL",T37,"REACTIVE",T55,"MAG."/10X,I3,7X,F10.5,5X,F10.5,6X, 015660
          2F9.5//)                                                  015670
1005 FORMAT(/1X,70(1H)/1X,14,59X,1H/1X,1H,T14,"RESULTS OF FAST DE 015680
          1COUPLED LOAD FLOW ANALYSIS",T71,1H/1X,1H,                015690
          2T19,"ALL MAGNITUDE VALUES ARE PER-UNIT",T71,14/1X,1H, 015700
          3T19,"SYSTEM HAS ",I3," BUSES",I2," ARE TYPE 2.",T71,14/1X, 015710
          41H,T10,"NUMBER OF TIMES LOAD BUSES WILL BE CHANGED(NLC)", 015720
          5I3,"",T71,1H/1X,14,T26,"CONVERGENCE TOLERANCES",T71,1H/ 015730
          61X,1H,T31,"PTOL=",F7.5,T71,14/1X,1H,T31,"QTOL=",F7.5,T71,14/ 015740
          71X,14,58X,1H/1X,70(1H)/)                                  015750
1006 FORMAT(/1X,70(1H)/1X,14,59X,14/1X,14,T25,"OUTPUT TRANSFORMER 015760
          1DATA",T71,1H/1X,14,59X,14/1X,70(1H)/T16,"SB",T19,"EB",T27, 015770
          2"TAB",T36,"TAP(MIN)",T50,"TAP(MAX)"/(12X,I3,2X,I3,4X,F7.5,4X, 015780
          3F7.5,7X,F7.5))                                           015790
1007 FORMAT(/1X,70(1H)/1X,14,59X,14/1X,14,T23,"OUTPUT PHASE SHIFTE 015800
          12 DATA",T71,1H/1X,14,59X,14/1X,70(1H)/T17,"SB",T22,"EB",T29, 015810
          2"TAB",T37,"PHASE ANGLE( DEG)"/(13X,I3,2X,I3,3X,F7.5,6X,F9.4)) 015820
1008 FORMAT(T58,"POWER"/T3,"NO.",T7,"TYPE",T15,"NAME",T25,"V(MAG)", 015830
          1T35,"V(ANG-DEG)",T53,"REAL",T57,"REACTIVE"/(1X,I3,3X, 015840
          2I1,3X,A10,3X,F7.4,4X,F9.4,4X,F9.4,4X,F9.4,4X,F9.4)) 015850
1009 FORMAT(/1X,70(1H)/1X,14,59X,14/1X,14,T28,"OUTPUT BUS DATA", 015860
          1T71,1H/1X,14,59X,14/1X,70(1H)/)                          015870
1010 FORMAT(1X,I3,8X,"6.4,5X,F8.4,5X,"6.5,6X,F9.5)                015880

```

	1011 FORMAT(1X,"BUS",I3," SHOULD HAVE BEEN IN REORDERED BUS LIST,BUT	015900
	1WASN'T. CHECK INPUT BUS LIST"/)	015910
60	1016 FORMAT(1X,"ERROR: BUS VOLTAGE CALCULATED FOR BUS",I3," IS 0.	015920
	1ERROR IS IN LOAD FLOW ROUTINE WHERE DLP OR DL1 IS CALCULATED."/)	015930
	1017 FORMAT(R6,I3)	015940
	1018 FORMAT(I2,I3,R10,2F5.0,4F10.0)	015950
65	1019 FORMAT(1X,"ERROR: MAGNITUDE OF SLACK BUS POWER HAS EXCEEDED LIMIT	015960
	1 OF 0 OR 100000 P.J."/)	015970
	1020 FORMAT(I3,"IDB CAN'T BE GREATER THAN 1 IN LOAD CHANGE ROUTINE	015980
	1SEE BUS",I3," WITH BUSNAME",R10," ."/)	015990
	1021 FORMAT(/1X,70(1H*)/1X,70(1H*)/1X,14*,T27,"LOAD FLOW BUS CHANGE	016000
70	1",T71,1H*/1X,14*,T28,"CHANGE NUMBER",I3,T71,14*/1X,14*,T24	016010
	2,"NUMBER OF BUSES CHANGED",I3,T71,14*/1X,14*,68X,1H*/1X,70	016020
	3(14*)/)	016030
	1022 FORMAT(/1X,70(1H*)/1X,14*,59X,14*/1X,14*,T29,"BUS CHANGE DATA	016040
	1",T71,1H*/1X,14*,69X,14*/1X,70(14*)/)	016050
75	1023 FORMAT(I3,"PROGRAM CONTROL CARD NOT IN PROPER FORMAT OR LOCATION.	016060
	1CARD WITH KEYWORD ",R6," IS REJECTED."/)	016070
	1024 FORMAT(1X,I3,2X,I2,6X,R10,2X,F5.3,7X,F5.3,6X,F10.5,3X,F10.5)	016080
	ITRMAX1=ITR1	016090
	ITRMAX2=ITR2	016100
	ITER=ITERR=0	016110
80	WRITE(2,1005) NBUS,IPV,MLC,PYOL,ZTO.	016120
	C THE FOLLOWING SECTION TRIANGULATES AND STORES THE ELEMENTS OF	016130
	C THE B* AND B** MATRICES. THESE ELEMENTS REMAIN CONSTANT THROUGHOUT	016140
	C THE SOLUTION OF THE M4-THETA, AND MVAR-V EQUATIONS. THE FIRST	016150
85	C PORTION TRIANGULATES B*, AND THE SECOND PORTION DOES B**, WHICH	016160
	C DOES NOT HAVE PV BUSES REPRESENTED.	016170
	C THE FOLLOWING LOOP RESTORES THE QUANTITY NA VECTOR FOR THE B*	016180
	C TRIANGULATION.	016190
	DO 1 I=1,250	016200
90	1 NA(I)=0	016210
	4=J=1	016220
	DO 5 I=1,ML	016230
	IF(LINA(I).EQ.0.OR.LINB(I).EQ.0) GO TO 5	016240
	JCOL(I)=LINB(I)	016250
	Y=CMPLX(G(I),B(I))	016260
95	IF(Y.EQ.0) GO TO 909	016270
	IF(Y1.EQ.0.) GO TO 909	016280
	Y1=AINAG(1./Y)	016290
	IF(Y1.EQ.0) GO TO 909	016300
	IF(Y1.EQ.0.) GO TO 909	016310
100	JU(N)=-1./AINAG(1./Y)	016320
	4=4+1	016330
	2 IF(LINA(I).NE.J) GO TO 3	016340
	NA(J)=NA(J)+1	016350
	GO TO 5	016360
105	3 J=J+1	016370
	GO TO 2	016380
	5 CONTINUE	016390
	ISS=NL-2*IZ	016400
110	DO 7 I=1,NBUS	016410
	BSUM(I)=BSUM+0.	016420
	IAB=JADD(I)	016430
	IAC=IAB+NA(I)-1	016440
	DO 6 J=IAB,IAC	016450
	BSUM=BSUM+DJ(J)	016460

115	6	CONTINUE	016470
		301A(I)=-BSUM	016480
	7	CONTINUE	016490
	C	THE FOLLOWING DO 100 LOOP TRIANGULATES THE B* MATRIX. THE ORDER OF	016500
	C	TRIANGULATION FOLLOWS THE REORDERED BUS LIST (NB(X)) RETURNED BY	016510
120	C	SUBROUTINE ORDER. THE SLACK BUS IS NOT REPRESENTED IN B*, AND IS	016520
	C	PASSED OVER IN THE TRIANG. PROCESS. THE FIRST INNER LOOP IDENTIFIES	016530
	C	ALL ROWS BELOW THE I*TH ROW THAT WILL BE AFFECTED BY THE I*TH ROW	016540
	C	ELIMINATION. THE FOLLOWING STATEMENTS AND SECOND INNER LOOP STORE	016550
	C	THE DIAGONAL AND UPPER-TRIANGULAR TERMS FOR THE I*TH ROW. THE D0P(K)	016560
125	C	VECTOR CONTAINS A LIST OF THE DIAS. TERMS, AND THE IUSP(K) VECTOR	016570
	C	CONTAINS A LIST OF POINTERS TO THE FIRST ELEMENT OF THE UPPER-	016580
	C	TRIANGULAR LIST USP(KK) ASSOCIATED WITH THE I*TH ROW. THE JSP(KK)	016590
	C	VECTOR CONTAINS A LIST OF COLUMN IDENTIFIERS CORRESPONDING TO EACH	016600
	C	TERM IN THE USP(KK) VECTOR. THE INNER DO 50 LOOP COMPLETES THE	016610
130	C	ELIMINATION PROCESS ON THE ROWS BELOW THE I*TH ROW. THE I*TH COLUMN	016620
	C	ELEMENT IS DELETED FROM THE N*TH ROW (DO 30 LOOP), AND THEN AN	016630
	C	ELEMENT-BY-ELEMENT COMPARISON IS MADE BETWEEN THE I*TH AND N*TH ROWS.	016640
	C	N*TH ROW TERMS ARE MODIFIED, OR NEW TERMS ADDED AS APPROPRIATE	016650
	C	(DO 32 LOOP).	016660
135		4=KK=1	016670
		DO 100 I=1,NBUS	016680
		IRN=NB(I)	016690
		IF(I0B(IRN).EQ.3) GO TO 100	016700
		D0P(K)=1./DIA(IRN)	016710
140		IUSP(K)=KK	016720
		IF(I.EQ.NBUS) GO TO 100	016730
		KK=0	016740
		DO 10 N=1,ISS	016750
		IK=IRN(N)	016760
145		IF(I0B(IK).EQ.3.OR.JCOL(N).NE.IRN) GO TO 10	016770
		DO 9 LH=1,I	016780
		IF(IK.EQ.45(LH)) GO TO 10	016790
	9	CONTINUE	016800
		KK=KK+1	016810
150		IDUM(KK)=IK	016820
	10	CONTINUE	016830
		(I=JADD(IRN)	016840
		(P=KI+NA(IRN)-1	016850
		IF(KP.LT.KI) GO TO 16	016860
155		ITEST=KK	016870
		DO 15 J=KI,KP	016880
		IF(I09(JCOL(J)).EQ.3) GO TO 15	016890
		DU(J)=DU(J)*D0P(K)	016900
		USP(KK)=-DU(J)	016910
160		JSP(KK)=JCOL(J)	016920
		KK=KK+1	016930
	15	CONTINUE	016940
	16	(=K+1	016950
		IF(KK.NE.ITEST) GO TO 17	016960
165		JSP(KK)=0.	016970
		J0P(KK)=0	016980
		KK=KK+1	016990
	17	IF(KKK.EQ.0) GO TO 100	017000
		DO 50 N=1,KKK	017010
170		IJJ=0	017020
	20	KI=JADD(IRN)	017030

		KP=KI+NA(IRN)-1	017040
		DO 40 J=CI,KP	017050
175		IF(ID9(JCOL(J)),E2,3) GO TO 40	017060
		K1=JADD(IDUM(N))	017070
		K2=K1+NA(IDUM(N))-1	017080
		IF(K2.LT.<1) GO TO 34	017090
		IF(IJJ.EQ.1) GO TO 31	017100
180		DO 30 L=K1,<2	017110
		IF(JCOL(L).E2,IRN) GO TO 35	017120
	30	CONTINUE	017130
	31	IF(IRN(K1).E2,JCOL(J)) GO TO 36	017140
		DO 32 L=K1,<2	017150
185		IF(JCOL(L).E2,JCOL(J)) GO TO 39	017160
	32	CONTINUE	017170
		CALL ADR3L(L-1,1,J)	017180
		NA(IDU4(N))=NA(IDJM(N))+1	017190
		GO TO 40	017200
190	35	BM=DU(L)	017210
		CALL ADR3L(L,-1,0)	017220
		IJJ=1	017230
		NA(IDU4(N))=NA(IDJM(N))-1	017240
		GO TO 20	017250
195	36	BDIA(IRN(K1))=BDIA(IRN(K1))-BM*DU(J)	017260
		GO TO 40	017270
	34	BDIA(IRN(K1)-1)=BDIA(IRN(K1)-1)-BM*DU(J)	017280
		GO TO 40	017290
	39	DU(L)=DU(L)-BM*DU(J)	017300
	40	CONTINUE	017310
200	50	CONTINUE	017320
	100	CONTINUE	017330
	C	THE FOLLOWING DO 200 JDP TRIANGULATES B'' IN THE SAME MANNER AS B'	017340
	C	ABOVE, EXCEPT THAT BV BUSSES ARE NOT REPRESENTED IN B''.	017350
		DO 101 I=1,250	017360
205	101	VA(I)=0	017370
		V=J=1	017380
		DO 105 I=1,ML	017390
		IF(LINA(I).EQ.0) GO TO 105	017400
210		JCOL(N)=LINA(I)	017410
		DU(N)=B(I)	017420
		N=N+1	017430
	102	IF(LINA(I).NE.J) GO TO 103	017440
		VA(J)=VA(J)+1	017450
		GO TO 105	017460
215	103	J=J+1	017470
		GO TO 102	017480
	105	CONTINUE	017490
		ISS=NL-IZ	017500
220		DO 107 I=1,NBUS	017510
		BDIA(I)=BSUM=0.	017520
		IA3=JADD(I)	017530
		IAC=IA3+NA(I)-1	017540
		DO 106 J=IA3,IAC	017550
225		BSUM=BSUM+DJ(J)	017560
	106	CONTINUE	017570
		BDIA(I)=BSUM	017580
	107	CONTINUE	017590
		K=KK+1	017600

		30 200 I=1, NBUS	017610
230		IRN=NB(I)	017620
		IF(IDB(IRN), NE.1) GO TO 200	017630
		DBPP(K)=1./BOIA(IRN)	017640
		IUBPP(K)=KK	017650
		IF(I.EQ.NBUS) GO TO 200	017660
235		KKK=0	017670
		DO 110 M=1, ISS	017680
		IK=IRN(M)	017690
		IF(IDB(IK), NE.1. OR. JCOL(4), NE. IRN) GO TO 110	017700
		DO 109 L=1, I	017710
240		IF(IK.EQ.NB(LM)) GO TO 110	017720
	109	CONTINUE	017730
		KKK=KKK+1	017740
		IDUM(KKK)=IK	017750
	110	CONTINUE	017760
245		ITEST=KK	017770
		KI=JADD(IRN)	017780
		KF=KI+NA(IRN)-1	017790
		IF(KF.LT.KI) GO TO 116	017800
		DO 115 J=KI, KF	017810
250		IF(IDB(JCOL(J)), NE.1) GO TO 115	017820
		DU(J)=DU(J)*DBPP(K)	017830
		UBPP(KK)=J(J)	017840
		JBPP(KK)=JCOL(J)	017850
		KK=KK+1	017860
255	115	CONTINUE	017870
	116	KK=K+1	017880
		IF(KK.NE.ITEST) GO TO 117	017890
		JBPP(KK)=0.	017900
		JBPP(KK)=0	017910
260		KK=KK+1	017920
	117	IF(KKK.EQ.0) GO TO 200	017930
		DO 120 N=1, KKK	017940
		IJJ=0	017950
	120	KI=JADD(IRN)	017960
265		KF=KI+NA(IRN)-1	017970
		DO 140 J=KI, KF	017980
		IF(IDB(JCOL(J)), NE.1) GO TO 140	017990
		K1=JADD(IDUM(N))	018000
		K2=K1+NA(IDUM(N))-1	018010
270		IF(K2.LT.K1) GO TO 136	018020
		IF(IJJ.EQ.1) GO TO 171	018030
		DO 130 L=K1, K2	018040
		IF(JCOL(L).EQ.IRN) GO TO 135	018050
	130	CONTINUE	018060
275	131	IF(IRN(K1).EQ.JCOL(J)) GO TO 135	018070
		DO 132 L=K1, K2	018080
		IF(JCOL(L).EQ.JCOL(J)) GO TO 135	018090
	132	CONTINUE	018100
		CALL ADROL(L-1, 1, J)	018110
280		NA(IDUM(N))=NA(IDUM(N))+1	018120
		GO TO 140	018130
	135	BN=DJ(,)	018140
		CALL ADROL(L, -1, 0)	018150
		NA(IDUM(N))=NA(IDUM(N))-1	018160
285		IJJ=1	018170

		GO TO 120	010100
136		B0IA(IRN(K1))=B0IA(IRN(K1))-B4*JU(J)	010190
		GO TO 140	010200
134		B0IA(IRN(K1)-1)=B0IA(IRN(K1)-1)-B4*DU(J)	010210
290		GO TO 140	010220
139		DU(L)=JU(L)-B4*DU(J)	010230
140		CONTINUE	010240
190		CONTINUE	010250
200		CONTINUE	010260
295		DO 209 I=1,NBUS	010270
		LIST(I)=0	010280
		209 4(I)=CONEC(I)	010290
		C THE FOLLOWING SECTIONS ARE THE DIRECT SOLUTION ALGORITHMS	010300
		C STATEMENT 211 IS THE ENTRY POINT FOR THE DELTA-P/DELTA THETA	010310
300		C SOLUTION. THE DLP ARRAY IS FORMED AS: P(SPEC)/V - P(CALC)/V,	010320
		C AND THEN SOLVED FOR DELTA THETA. THE DELTA THETA SOLUTION FOR	010330
		C EACH BUS IS THE DLP ARRAY AFTER THE FORWARD AND BACKWARD SUBSTI-	010340
		C TUTIONS. THE DELTA THETA IS ADDED TO THE EXISTING BUS ANGLE TO	010350
		C UPDATE THE BUS ANGLE AFTER EACH SOLUTION.	010360
305		C STATEMENT 291 IS THE ENTRY POINT FOR THE DELTA-Q/DELTA V SOLUTION.	010370
		C THE D-Q ARRAY PERFORMS THE SAME FUNCTION AS DLP ABOVE, EXCEPT	010380
		C THAT THE SOLUTION DELTA V IS THE RESULTING ARRAY. THE DELTA V	010390
		C FOR EACH BUS IS ADDED TO THE EXISTING BUS VOLTAGE MAGNITUDE TO	010400
		C FORM THE UPDATED BUS V MAG. FOLLOWING EACH SOLUTION ITERATION.	010410
310		210 KP=K1-1	010420
		211 CP=0	010430
		CP=0	010440
		DO 250 I=1,NBUS	010450
		IRN=NB(I)	010460
315		IF(I00(IRN).EQ.3) GO TO 250	010470
		K=K+1	010480
		II=JADD(IRN)+IZ	010490
		IF=II+CONEC(IRN)-1	010500
		SS=BSUM=0.	010510
320		DO 225 J=II,IF	010520
		T=YAPR(IRN,LIN9(J))	010530
		GS=SS+T*G(J)	010540
		IF(LIN9(J).EQ.0) GO TO 225	010550
		THETA=ANG(IRN)-ANG(LIN9(J))	010560
325		IF(THETA.GT.94.2) THETA=94.2	010570
		IF(THETA.LT.-94.2) THETA=-94.2	010580
		BSUM=BSUM+V(LIN9(J))*(-G(J)*COS(THETA)-B(J)*SIN(THETA))	010590
		225 CONTINUE	010600
		PC=V(IRN)*(BSUM+V(IRN)*GS)	010610
330		IF(V(IRN).EQ.0) GO TO 906	010620
		IF(V(IRN).EQ.0.) GO TO 906	010630
		DLP(K)=P(IRN)/V(IRN)-PC/V(IRN)	010640
		DELTA=ABS(DLP(K))	010650
		IF(DELTA.GT.PTOL) KP=1	010660
335		250 CONTINUE	010670
		IF(KP.EQ.0) GO TO 400	010680
		IF(ITER.GT.ITRMAX1) GO TO 500	010690
		IS=NBUS-2	010700
		DO 270 I=1,IS	010710
340		K1=IUBP(I)	010720
		K2=IUBP(I+1)-1	010730
		DO 265 J=K1,K2	010740

	IF(JBP(J).EQ.0) GO TO 265	010750
	JJ=IPOS(JBP(J))	010760
345	DLP(JJ)=DLP(JJ)+USP(J)*DLP(I)	010770
	265 CONTINUE	010780
	270 CONTINUE	010790
	DO 275 I=1,K	010800
340	275 DLP(I)=DLP(I)*ORP(I)	010810
	DO 285 I=1,IS	010820
	TEMP=0.	010830
	K1=IJB(K-I)	010840
	K2=IJB(K-I+1)-1	010850
	DO 280 J=K1,K2	010860
395	IF(JBP(J).EQ.0) GO TO 290	010870
	JJ=IPOS(JBP(J))	010880
	TEMP=TEMP+USP(J)*DLP(JJ)	010890
	280 CONTINUE	010900
	DLP(K-I)=DLP(K-I)+TEMP	010910
360	285 CONTINUE	010920
	DO 290 I=1,NBUS	010930
	IF(IIB(I).EQ.3) GO TO 290	010940
	N=0	010950
	DO 288 J=1,NBUS	010960
365	IF(IIB(NB(J)).EQ.3) GO TO 290	010970
	N=N+1	010980
	IF(NB(J).EQ.2) GO TO 288	010990
	288 CONTINUE	010000
	289 ANG(I)=ANG(I)+DLP(I)	010010
370	IF(ANG(I).GT.94.26) ANG(I)=94.26	010020
	IF(ANG(I).LT.-94.26) ANG(I)=-94.26	010030
	290 CONTINUE	010040
	ITER=ITER+1	010050
	C STATEMENT 291 IS THE ENTRY POINT FOR THE DELTA Q-DELTA V SOLUTION	010060
375	C ROUTINE. ITER IS THE ITERATION COUNTER FOR THIS ROUTINE. THE	010070
	C DO 390 LOOP FORMS THE DELTA Q/V VECTOR FOR THE CURRENT ITERATION.	010080
	C IF Q=0 AT THE END OF THIS LOOP, Q1-QV CONVERGENCE IS OBTAINED.	010090
	C OTHERWISE, THE DO 378 THROUGH 393 LOOPS SOLVE FOR THE NEW DELTA V.	010100
	C THE DO 390 LOOP UPDATES THE BUS VOLTAGES.	010110
380	291 Q=0	010120
	Q=0	010130
	DO 390 I=1,NBUS	010140
	IR=NR(I)	010150
385	IF(IIB(IR).NE.1) GO TO 390	010160
	K=K+1	010170
	II=JADD(IRW)+IZ	010180
	IF=II+CONEC(IRW)-1	010190
	SS=BSUM=0.	010200
	DO 325 J=II,IF	010210
390	T=TAPR(IRW,LINB(J))	010220
	BS=BS+T*B(J)	010230
	IF(LINB(J).EQ.0) GO TO 325	010240
	THETA=ANG(IRW)-ANG(LINB(J))	010250
	IF(THETA.GT.6.3) THETA=6.3	010260
395	IF(THETA.LT.-6.3) THETA=-6.3	010270
	BSUM=BSUM+V(LINB(J))*(-G(J))*SIN(THETA)+B(J)*COS(THETA)	010280
	325 CONTINUE	010290
	2C=V(IRW)*(BSUM-V(IRW)*(SS))	010300
	IF(V(IRW).EQ.0.) GO TO 385	010310

400	IF(V(IRN).EQ.0) GO TO 906	019320
	DLQ(K)=Q(IRN)/V(IRN)-QC/V(IRN)	019330
	DELTA=ABS(DLQ(K))	019340
	IF(DELTA.GT.QTOL) KQ=1	019350
350	CONTINUE	019360
405	IF(K7.EQ.0) GO TO 401	019370
	IF(ITERR.GT.ITRMAX2) GO TO 500	019380
	IS=K-1	019390
	DO 370 I=1, IS	019400
	<1=IUBPP(I)	019410
410	K2=IUBPP(I+1)-1	019420
	DO 365 J=K1,K2	019430
	IF(JBPP(J).EQ.0) GO TO 365	019440
	JJ=IPOST(JBPP(J))	019450
	DLQ(JJ)=DLQ(JJ)+UBPP(J)*DLQ(I)	019460
415	CONTINUE	019470
370	CONTINUE	019480
	DO 375 I=1, K	019490
375	DL2(I)=DLQ(I)*DBPP(I)	019500
	DO 395 I=1, IS	019510
420	TEMP=0.	019520
	<1=IUBPP(K-I)	019530
	K2=IUBPP(K-I+1)-1	019540
	DO 380 J=K1,K2	019550
	IF(JBPP(J).EQ.0) GO TO 390	019560
425	JJ=IPOST(JBPP(J))	019570
	TEMP=TEMP+UBPP(J)*DLQ(JJ)	019580
	CONTINUE	019590
380	DLQ(K-I)=DLQ(K-I)+TEMP	019600
385	CONTINUE	019610
430	DO 390 I=1, NBUS	019620
	IF(I00(I).NE.1) GO TO 390	019630
	N=0	019640
	DO 388 J=1, NBUS	019650
435	IF(I00(NB(J)).NE.1) GO TO 390	019660
	N=N+1	019670
	IF(NB(J).EQ.I) GO TO 390	019680
	CONTINUE	019690
388	CONTINUE	019700
440	V(I)=V(I)+DLQ(N)	019710
	IF(V(I).LT.-100.) V(I)=-100.	019720
	IF(V(I).GT.100.) V(I)=100.	019730
390	CONTINUE	019740
	ITERR=ITERR+1	019750
445	C AFTER THE DELTA-V SOLUTION, SUBROUTINE LIMIT IS CALLED TO DETER-	019760
	C MINE IF ANY PV(TYPE 2) BUSES HAVE EXCEEDED THEIR Q LIMITS.	019770
	CALL LIMIT(0)	019780
	GO TO 211	019790
450	400 IF(K0.EQ.0) 450,291	019800
	401 IF(KP.EQ.0) 450,211	019810
450	C CONVERGENCE HAS BEEN OBTAINED. THE FOLLOWING SECTION CALCULATES	019820
	C THE SLACK BUS POWER, AND WRITES THE RESULTS ON THE OUTPUT FILE.	019830
450	450 4R1V(12,1001) ITR,ITERR	019840
	DO 452 I=1, NBUS	019850
	IF(I00(I).NE.3) GO TO 452	019860
450	<I=J000(I)+I2	019870
	KP=KI+CONEC(I)-1	019880

```

2S=BS-SUM=BSUM=0.                                019890
DO 451 J=KI,KF                                     019900
T=TAPR(I,LINB(J))                                  019910
460 GS=GS+T*G(J)                                    019920
BS=BS+T*B(J)                                       019930
IF(LINB(J).EQ.0) GO TO 451                          019940
THETA=ANG(I)-ANG(LINB(J))                          019950
465 SUM=SUM+V(LINB(J))*(-S(J)*COS(THETA)-B(J)*SIN(THETA)) 019960
BSUM=BSUM+V(LINB(J))*(-S(J)*SIN(THETA)+B(J)*COS(THETA)) 019970
451 CONTINUE                                        019980
P(I)=V(I)*(SUM+V(I)*GS)                            019990
Q(I)=V(I)*(BSUM-V(I)*BS)                            020000
SMI=P(I)*P(I)+Q(I)*Q(I)                            020010
470 IF(SMI.LT.0.) GO TO 910                          020020
IF(SMI.GT.10E9) GO TO 910                           020030
SM=SQRT(SMI)                                         020040
475 QUIT(2,1004) I,P(I),Q(I),SM                    020050
GO TO 453                                           020060
452 CONTINUE                                        020070
453 IF(DJT.EQ.6.OR.OUT.EQ.10) GO TO 456             020080
C CALCULATE LINE FLOWS***                           020090
WRIT(2,1002)                                       020100
NN=0                                                020110
480 DO 455 I=1,NL                                   020120
IF(LINA(I).EQ.0.OR.LINR(I).EQ.0) GO TO 455         020130
IF(LINA(I).GT.LINB(I)) GO TO 455                  020140
NN=NN+1                                             020150
485 T=CMPLX(G(I),R(I))                              020160
EA=V(LINA(I))*COS(ANG(LINA(I)))                   020170
EB=V(LINA(I))*SIN(ANG(LINA(I)))                   020180
VA=CMPLX(EA,EB)                                    020190
490 IA=V(LINR(I))*COS(ANG(LINR(I)))               020200
IB=V(LINR(I))*SIN(ANG(LINR(I)))                   020210
VB=CMPLX(IA,IB)                                    020220
DO 496 N=1,NTR                                     020230
495 IF(LINA(I).NE.LTRA(N)) GO TO 457               020240
IF(LINB(I).NE.LTRB(N)) GO TO 456                 020250
T=TAP(N)                                           020260
IF(T.EQ.0) T=1                                     020270
IF(T.EQ.0.) T=1                                    020280
S=T*VA*CONJG((T*VA-VB)*V/T)                       020290
R=VB*CONJG((VB-T*VA)*V/T)                         020300
GO TO 458                                          020310
500 457 IF(LINA(I).NE.LTRA(N)) GO TO 454           020320
IF(LINB(I).NE.LTRB(N)) GO TO 454                 020330
T=TAP(N)                                           020340
IF(T.EQ.0) T=1                                     020350
IF(T.EQ.0.) T=1                                    020360
505 S=VA*CONJG((VA-T*VB)*V/T)                     020370
R=T*VB*CONJG((T*VB-VA)*V/T)                      020380
GO TO 458                                          020390
454 CONTINUE                                        020400
C FALLING OUT THE BOTTOM OF THIS LOOP MEANS THAT THE BRANCH DEFINED
C BY LINA(I)-LINB(I) IS NOT A TRANSFORMER, AND A TAP CALCULATION IS
C NOT NECESSARY. CALCULATE THE LINE POWER FLOWS BELOW
S=VA*CONJG((VA-VB)*V)                              020410
R=VB*CONJG((VB-VA)*V)                              020420
020430
020440
020450

```

515	450 WRITE(2,1003) LINA(I),LINB(I),S,_INS(I),LINA(I),R	020460
	455 CONTINUE	020470
	456 DO 459 I=1,NBUS	020480
	459 ANG(I)=57.29578*ANG(I)	020490
	C WRITE SUMMARIZED OUTPUT DATA ON OUTPUT FILE	020500
520	WRITE(2,1009)	020510
	WRITE(2,1008) (IBUS(I),IOB(I),BUSNAME(I),V(I),ANG(I),P(I),Q(I),	020520
	I1=1,NBUS)	020530
	IF(OUT.EQ.10) GO TO 570	020540
	IF(LLL.EQ.0.OR.OJT.EQ.9) GO TO 451	020550
525	461 WRITE(2,1007) (LPHA(I),LPHB(I),P+ANG(I),I=1,LLL)	020560
	IF(NOTR.EQ.0.OR.JUT.EQ.9) GO TO 570	020570
	WRITE(2,1006) (LTRA(I),LTRB(I),TAP(I),TMN(I),TMX(I),I=1,NOTR)	020580
	GO TO 570	020590
	500 WRITE(2,1008) ITER,ITERR	020600
	_ODDP=-1	020610
530	DO 501 I=1,NBUS	020620
	501 ANG(I)=57.29578*ANG(I)	020630
	C THIS SECTION TO 502 IS TABLE DUMP ROUTINE WHEN PROBLEM DOES NOT	020640
	C CONVERGE WITHIN SPECIFIED ITERATIONS.	020650
535	DO 502 IB=1,NBUS	020660
	IF(IOB(IB).NE.1) GO TO 505	020670
	V1=V1+0	020680
	DO 502 JA=1,NBUS	020690
	IF(IOB(NB(JA)).NE.1) GO TO 503	020700
540	V1=V1+1	020710
	V1=V1+1	020720
	IF(NB(JA).EQ.IB) GO TO 519	020730
	503 CONTINUE	020740
	GO TO 900	020750
545	505 IF(IOB(IB).EQ.3) GO TO 510	020760
	V2=0	020770
	DO 506 JJJ=1,NBUS	020780
	IF(IOB(NB(JJJ)).EQ.3) GO TO 506	020790
	V2=V2+1	020800
550	IF(NB(JJJ).EQ.IB) GO TO 510	020810
	506 CONTINUE	020820
	GO TO 900	020830
	510 JQ=JQ+0	020840
	GO TO 520	020850
555	510 JP=DLP(M2)	020860
	JQ=0	020870
	GO TO 520	020880
	510 JP=DLP(M1)	020890
	DQ=DLT(M1)	020900
	GO TO 520	020910
560	520 WRITE(2,1010) I9JS(IB),V(I9),ANG(I9),OP,D1	020920
	502 CONTINUE	020930
	RETURN	020940
	570 CONTINUE	020950
565	C LOAD FLOW AUTO CHANGE ROUTINE. THE NUMBER OF CHANGES IS READ(NC)	020960
	C THEN NEW BUS DATA IS READ. BUS LIST UPDATED BY LOOP 600. LOAD	020970
	C FLOW ROUTINE IS AGAIN ENTERED AT 210 AND EXECUTED USING NEW DATA.	020980
	C NC IS NUMBER OF CHANGE. NC IS NUMBER OF BUSES EACH CHANGE.	020990
	C MCG IS CHANGE COUNTER. BUS CARDS USE SAME FORMAT AS IN BUSIN RT.	021000
570	IF(NC.LE.0) GO TO 999	021010
	DO 500 I=1,NBUS	021020

	ANG(T) = .01745329*ANG(I)	021030
500	CONTINUE	021040
	NCC=4CC+1	021050
575	READ(5,1017) A,NC	021060
	Z=6RUSCHG	021070
	IF(A.NE.C) GO TO 908	021080
	WRITE(2,1021) NCC,NC	021090
	WRITE(2,1022)	021100
	WRITE(2,1008)	021110
580	DO 602 IC=1,NC	021120
	READ(5,1018) IOBN,IBUSN,BJSN,VAN,AGN,PAN,ZAN,QMINN,QMAXN	021130
	IF(IOBN.GT.1) GO TO 907	021140
	ZMINN=QMINN/BKVA	021150
	ZMAXN=QMAXN/BKVA	021160
585	PN=PAN/BKVA	021170
	ZN=ZAN/BKVA	021180
	WRITE(2,1024) IBUSN,IOBN,BJSN,VAN,AGN,PN,ZN	021190
	ANGN = .01745329*ASN	021200
	IF(VAN.EQ.0) VAN=1.0	021210
590	VN=VAN	021220
	DO 600 JC=1,NBUS	021230
	IF(IBUS(JC).NE.IBUSN) GO TO 600	021240
	P(JC)=PN	021250
	Q(JC)=QN	021260
595	ZMIN(JC)=QMINN	021270
	ZMAX(JC)=QMAXN	021280
	ANG(JC)=ANGN	021290
	V(JC)=VN	021300
	BUSNAME(JC)=BUSN	021310
600	IOB(JC)=IOBN	021320
	CONTINUE	021330
	602 CONTINUE	021340
	NLC=NLC-1	021350
	DO T) 210	021360
605	908 WRITE(2,1011) IS	021370
	DO T) 908	021380
	906 WRITE(2,1016) IRW	021390
	DO T) 908	021400
	907 WRITE(2,1020) IBUSN,BUSN	021410
610	DO T) 908	021420
	908 WRITE(2,1023) C	021430
	DO T) 908	021440
	909 WRITE(2,1011) LINA(I)	021450
	DO T) 908	021460
615	910 WRITE(2,1019)	021470
	900 IERR=IERR+1	021480
	999 RETURN	021490
	END	021500
1	SUBROUTINE ADROL(IJ,IC,J)	021510
C	THIS SUBROUTINE ADDS OR DELETES ENTRIES FROM THE OU AND JCCL	021520
C	TABLES. OTHER TABLE ENTRIES ARE MOVED TO ALLOW FOR THE ADDED	021530
C	OR DELETED ENTRIES. THE COUNTER FOR THE NUMBER OF ENTRIES IS	021540
5	C ALSO ADJUSTED APPROPRIATELY.	021550
	COMMON/COMC/ NA(250),NB(250),JCL(1000),JJ(1000),IOB(250)	021560
	COMMON/CONST/ NBUS,NL,ISS,IPV,LL1,LL2,LL3,LL4,NCTR,IZ,NOLTC	021570
	1,ITR1,ITR2,PTOL,QTOL,NLC	021580
	COMMON/SUB/ BM	021590
10	IF(IZ.EQ.1) GO TO 20	021600
	LS=ISS-1	021610
	DO 10 I=IJ,LS	021620
	JCOL(I)=JCOL(I+1)	021630
	OU(I)=OU(I+1)	021640
15	10 CONTINUE	021650
	JCOL(ISS)=0	021660
	OU(ISS)=0.	021670
	ISS=ISS-1	021680
	RETURN	021690
20	20 LS=ISS-IJ-2	021700
	BM=OU(I)	021710
	JC=JCOL(I)	021720
	JCOL(ISS+1)=JCOL(ISS)	021730
	OU(ISS+1)=OU(ISS)	021740
25	JCOL(ISS)=JCOL(ISS-1)	021750
	OU(ISS)=OU(ISS-1)	021760
	DO 30 I=1,LS	021770
	JCOL(ISS-I)=JCOL(ISS-I-1)	021780
	OU(ISS-I)=OU(ISS-I-1)	021790
30	30 CONTINUE	021800
	JCOL(IJ+1)=JC	021810
	OU(IJ+1)=BM*BM	021820
	ISS=ISS+1	021830
	RETURN	021840
35	END	021850

```

1      SUBROUTINE LIMIT(IJK)                                021060
      INTEGER CONEC                                         021070
      DIMENSION SK(250)                                     021080
      COMMON /COMA/CON,CHG,LOOOP,SCOP,INP,OUT,F,T,BKVA,RHO1,NMAX,MAXTR, 021090
5      1VAKLTC,MAXPH,ISYS,BKVA1
      COMMON /COMB/LIN4(1450),LIN3(1450),S(1450),B(1450),P(250),Q(250), 021910
      1LP4B(50),PHANG(50),TRA(250),LTR3(250),TAP(250),TMN(250),V(250), 021920
      2THK(250),IUBPP(250),ANG(250),IRUS(250),D9P(250),UBP(3000), 021930
      3BUSNAME(250),LPHA(50),LIST(250),IUSP(250),QMIN(250),QMAX(250), 021940
10     4UBPP(250),UBPP(3000),J9P(3000),J3PP(3000),ICC(250),DLP(250), 021950
      5ZBI(1450),ZOR(1450),ROTA(250),CONEC(250),DLQ(250),IPHASE(250) 021960
      COMMON/CONC/ NA(250),NB(250),JC3L(1000),DJ(1000),IDB(250) 021970
      COMMON /CONST/ NBUS,NL,ISS,IPV,L-1,-L2,LL3,LL4,NOTR,IZ,NOLTC 021980
      1,ITR1,ITR2,PTOL,PTO,NLC                               021990
15     1000 FORMAT(T16,'**** BUS NO. ',I3,' EXCEEDED ITS MAXIMUM Q LIMIT:/'
      1T21,'Q SPECIFIED','F8.4,' PER-UNIT. '/T21,'Q CALCULATED',
      2F8.4,' PER-UNIT. '/T21,'AMOUNT EXCEEDED','F5.2,' PERCENT. '/')
      1001 FORMAT(T16,'**** BUS NO. ',I3,' EXCEEDED ITS MINIMUM Q LIMIT:/'
18     1T21,'Q SPECIFIED','F8.4,' PER-UNIT. '/T21,'Q CALCULATED',
      2F8.4,' PER-UNIT. '/T21,'AMOUNT EXCEEDED','F5.2,' PERCENT. '/')
20     C THIS SUBROUTINE DETERMINES IF ANY PV TYPE BUSES HAVE EXCEEDED
      C THEIR Q LIMITS. THE BUS LIST IS SCANNED FOR ALL TYPE 2 BUSES,
      C AND THE Q CALCULATED FOR EACH. THE CALCULATED Q IS COMPARED TO
      C THE SPECIFIED Q. IF THE CALC. Q EXCEEDS THE LIMIT, A CHECK IS
25     C THEN MADE TO SEE IF ANY TAP BRANCHES ARE CONNECTED TO THE BUS.
      C IF YES, THE TAP IS ADJUSTED (WITHIN TAP LIMITS) TO TRY TO HOLD
      C THE SPECIFIED BUS VOLTAGE. IF NO TAP'S ARE AVAILABLE, OR THE
      C TAP LIMITS HAVE BEEN REACHED, THE BUS VOLTAGE IS THEN ADJUSTED.
30     DO 100 I=1,NBUS
      IF(INB(I).NE.2) GO TO 100
      II=JADD(I)+IZ
      IF=II+CONEC(II)-1
      QS=SSUM=0.
      DO 10 J=II,IF
35         Y=TAPR(I,LINB(J))
         BS=BS+T*(R(J))
         IF(LINB(J).EQ.0) GO TO 10
         THETA=ANG(II)-ANG(LINB(J))
         BSUM=BSUM+V(LINB(J))*(-S(J)*SIN(THETA)+B(J)*COS(THETA))
40     CONTINUE
      Q(I)=V(I)*(BSUM-V(I)*BS)
      IF(QMAX(I).EQ.0.AND.QMIN(I).EQ.0) GO TO 100
      IF(QMAX(I)-Q(I)) 20,100,11
45     11 IF(QMIN(I)-Q(I)) 100,100,25
      20 DEL=QMAX(I)-Q(I)
      DO 21 L=1,LOOOP
      IF(LIST(L).EQ.I) GO TO 30
21     CONTINUE
      LOOP=LOOOP+1
50     EX=(-DEL/QMAX(I))*100.
      WRITE(2,1000) I,QMAX(I),Q(I),EX
      LIST(LOOP)=I
      GO TO 30
      25 DEL=QMIN(I)-Q(I)
95     DO 26 L=1,LOOOP
      IF(LIST(L).EQ.I) GO TO 50
26     CONTINUE

```

	LODOP=LODOP+1	022430
	LIST(LODOP)=I	022440
60	IF(QMIN(I).EQ.0.) GO TO 27	022450
	EX=ASS(100.*DEL/2MIN(I))	022460
	GO TO 26	022470
	27 EX=100.*DEL	022480
	28 4RIT(2,1001) I,2MIN(I),Q(I),EX	022490
65	50 IF(A9S(OEL).LE.01) GO TO 100	022500
	IF(SK(I).NE.0.) GO TO 90	022510
	K1=JADD(I)+IZ	022520
	K2=K1+CONEC(I)-1	022530
	C=0	022540
70	DO 60 L=1,NBUS	022550
	IRW=NB(L)	022560
	IF(I)9(IRW).NE.1) GO TO 60	022570
	C=C+1	022580
	DO 51 J=K1,K2	022590
75	IF(IRW.EQ.LINB(J)) GO TO 52	022600
	CONTINUE	022610
	JU(K)=0.	022620
	GO TO 60	022630
	52 JU(K)=0(J)	022640
80	60 CONTINUE	022650
	9S=0.	022660
	DO 51 L=K1,K2	022670
	T=TAPR(I,LINB(L))	022680
	61 3S=BS-T*B(L)	022690
85	IS=K-1	022700
	DO 70 N=1,IS	022710
	K1=IU9PP(N)	022720
	K2=IU9PP(N+1)-1	022730
	IF(K2.LT.K1) GO TO 70	022740
90	DO 65 J=K1,K2	022750
	IF(J9PP(J).EQ.0) GO TO 65	022760
	JJ=IPOS(J9PP(J))	022770
	DU(JJ)=DU(JJ)+U9PP(J)*DU(N)	022780
	CONTINUE	022790
95	70 CONTINUE	022800
	DO 75 N=1,K	022810
	JU(N)=DBPP(N)*DU(N)	022820
	75 CONTINUE	022830
	TEMP=0.	022840
100	DO 80 N=1,K	022850
	80 TEMP=TEMP+DU(N)*JU(N)/DBPP(N)	022860
	SK(I)=1./(BS-TEMP)	022870
	90 DV=SK(I)*DEL/V(I)	022880
	DO 91 L=1,NOTR	022890
105	IF(LTR(L).NE.I) GO TO 91	022900
	IF(ICC(L).GT.69.AND.ICC(L).LT.90) GO TO 92	022910
	91 CONTINUE	022920
	C FALLING OUT THE BOTTOM OF THE LOOP MEANS THAT BUS I DOES NOT HAVE	022930
	C AN _T_ BRANCH CONNECTED WITH THE TAPPING END AT BUS I. THEREFORE	022940
110	C CALCULATE ADJUSTED VOLTAGE FOR BUS I.	022950
	V(I)=V(I)+DV	022960
	GO TO 100	022970
	92 TNEW=TAP(L)+DV	022980
	IF(TMX(L)-TNEW) 94,96,93	022990
115	93 IF(TMN(L)-TNEW) 96,96,95	023000
	94 TAP(L)=TMX(L)	023010
	V(I)=V(I)+DV	023020
	GO TO 100	023030
	95 TAP(L)=TMN(L)	023040
120	V(I)=V(I)+DV	023050
	GO TO 100	023060
	C FUNCTION DISCRET CALC. THE NEAREST PHYSICAL (DISCRETE) TAP SETTING	023070
	C TO TNEW; THIS VALUE IS STORED AS THE NEW TAP SETTING FOR THE TRANS.	023080
	96 TAP(L)=DISCRET(TNEW)	023090
125	100 CONTINUE	023100
	RETJRW	023110
	END	023120

```

1      FUNCTION IPOS(IT)                                023130
COMMON/COMC/ NA(250),NB(250),JCD.(1000),DJ(1000),IDB(250) 023140
COMMON/CONST/ NBJS,NL,ISS,IPV,LL1,LL2,LL3,LL4,NOTR,IZ,NOLTC 023150
1,ITR1,ITR2,PTOL,ATOL,NLC                                023160
5      <=0
DO 10 I=1,NBUS                                         023180
IF(IDB(NB(I)).EQ.3) GO TO 10                            023190
K=K+1                                                    023200
IF(NB(I).EQ.IT) GO TO 11                               023210
10 CONTINUE                                           023220
11 IPOS=K                                               023230
RETURN                                                 023240
END                                                    023250

1      FUNCTION IPOS1(IT)                              023260
COMMON/COMC/ NA(250),NB(250),JCD.(1000),DJ(1000),IDB(250) 023270
COMMON/CONST/ NBJS,NL,ISS,IPV,LL1,LL2,LL3,LL4,NOTR,IZ,NOLTC 023280
1,ITR1,ITR2,PTOL,ATOL,NLC                                023290
5      <=0
DO 10 I=1,NBUS                                         023310
IF(IDB(NB(I)).NE.1) GO TO 10                           023320
K=K+1                                                    023330
IF(NB(I).EQ.IT) GO TO 11                               023340
10 CONTINUE                                           023350
11 IPOS1=K                                              023360
RETURN                                                 023370
END                                                    023380

1      FUNCTION TAPR(ITA,ITB)                          023390
C THIS FUNCTION DETERMINES IF THE BRANCH DEFINED BY ITA AND ITB IS 023400
C A TRANSFORMER BRANCH. IF SO, THE PROPER TAP RATIO IS ASSIGNED TO 023410
C TAPR AND RETURNED TO THE PROGRAM.                    023420
5      COMMON /COMB/LINA(1450),LINB(1450),G(1450),B(1450),P(250),Q(250), 023430
1,P43(50),P4ANG(50),LTRA(250),LTRB(250),TAP(250),TMN(250),V(250), 023440
2FMX(250),IUBPP(250),ANG(250),IPUS(250),OBP(250),UBP(3000), 023450
3BUSNAME(250),LPHA(50),LIST(250),IUBP(250),QMIN(250),QMAX(250), 023460
4JPP(250),UBPP(3000),JBP(3000),JPP(3000),ICC(250),OLP(250), 023470
5ZBI(1450),ZBR(1450),ADIA(250),CONEC(250),JLQ(250),IPHASE(250) 023480
10 COMMON /CONST/ NBUS,NL,ISS,IPV,LL1,LL2,LL3,LL4,NOTR,IZ,NOLTC 023490
1,ITR1,ITR2,PTOL,ATOL,NLC                                023500
TAPR=0.                                                  023510
IF(ITB.EQ.0) GO TO 25                                   023520
15 DO 20 I=1,NOTR                                       023530
IF(LTRA(I).NE.ITA) GO TO 10                            023540
IF(LTRB(I).EQ.ITB) TAPR=TAP(I)                        023550
GO TO 20                                               023560
20 IF(LTRB(I).NE.ITB) GO TO 20                         023570
IF(LTRA(I).EQ.ITA) TAPR=1./TAP(I)                    023580
20 CONTINUE                                           023590
25 IF(TAPR.EQ.0.) TAPR=1.0                             023600
RETURN                                                 023610
END                                                    023620

1      FUNCTION DISCRET(TT)                            023630
I1=1                                                    023640
K=TT/.00625-160.                                       023650
J=K                                                     023660
5 IF(ABS(X-J).GE.5) GO TO 5                            023670
DISCRET=1.0+.00625*J                                  023680
RETURN                                                 023690
5 IF(J.LT.0) I1=-1                                     023700
J=J+I1                                                 023710
10 DISCRET=1.0+.00625*J                                023720
RETURN                                                 023730
END                                                    023740

```

```

1      OVERLAY (SHRTCKT,3,0)                                023750
      PROGRAM FAULT                                         023760
C PROGRAM FAULT IS THE SHORT-CIRCUIT ANALYSIS SEGMENT OF THE OVERALL 023770
C SYSTEM ANALYSIS PROGRAM. THIS PROGRAM UTILIZES LARGE MATRIX TECH- 023780
C NIQUE AND ZBUS MATRIX SYMMETRY TO REDUCE PROGRAM STORAGE REQUIRE- 023790
C MENTS. ANY NETWORK UP TO 250 BUSES CAN BE USED AND THE PROGRAM 023800
C WILL COMPLETE THE FAULT STUDY BY TAKING 50-BUS "SUBSYSTEMS" OF THE 023810
C NETWORK AND REDUCING THE REMAINING SYSTEM TO A MESH EQUIVALENT. 023820
C BY TAKING CONSECUTIVE 50-BUS SEGMENTS, THE ENTIRE NETWORK CAN BE 023830
10 C STUDIED IN FIVE PASSES.                                023840
C THE PROGRAM USES THE METHOD OF SYMMETRICAL COMPONENTS, AND THE 023850
C POSITIVE AND ZERO SEQUENCE IMPEDANCE MATRICES ARE STORED AS ONE- 023860
C DIMENSIONAL ARRAYS, I.E. THE DIAGONAL ELEMENTS OF EACH ARE STORED 023870
C IN ZDIA AND ZODIA, AND THE UPPER TRIANGLES FOR EACH ARE STORED IN 023880
15 C ZBUS AND ZBUSB.                                       023890
      INTEGER COM,CHG,SCOP,OUT,CONEC,P4,A,C,AA,CC,A1,C1,A2,C2,A3,C3,D,DI023900
      COMPLEX ZZ,OPFDIAG,DIAG,CPLXV,V9,FA3,FA1,ZF,ZDIA,ZODIA,ZBUS,ZBUSB,023910
      IZH,EBUS,AMPA,EC,ZZE,ZC,ZCOJP,YCOJP,ZZO,ZG,ZZG(250),FI,FIG,PNS,ZS, 023920
      ZFIA                                                023930
20      DIMENSION IFB(25),L1(51),L2(51),JU(51),CUR(51)    023940
      COMMON /COMA/COM,CHG,LDOOP,SCOP,INP,OUT,F,T,BKVA,RHO1,NMAX,MAXTR, 023950
      IVALTC,MAXPH,ISYS,ICVA1                            023960
      COMMON /COMB/LINA(1450),LINB(1450),I(1450),B(1450),P(250),Q(250), 023970
      LPH(50),PHANG(50),LTRA(250),LTRB(250),TAP(250),THN(250),V(250), 023980
      THK(250),IUBPP(250),ANG(250),IRUS(250),O9P(250),UBP(3000), 023990
      BUSNAME(250),LPHB(50),LIST(250),IUP(250),MHN(250),MAY(250), 024000
      BUSPP(250),UBPP(3000),J3P(3000),J3PP(3000),ICC(250),JLP(250), 024010
      SZB(1450),ZOR(1450),BOIA(250),CONEC(250),JLQ(250),IPHASE(250) 024020
      COMMON/CONC/ MA(250),N9(250),JCL(1000),DJ(1000),IOB(250) 024030
30      COMMON /CONST/ NBUS,NL,ISS,IPV,LL1,LL2,LL3,LL4,NOTR,IZ,MOLTC 024040
      I,ITR1,ITR2,PTOL,ITOL,NLC                         024050
      COMMON /SAVE/ IERR                                  024060
      COMMON /ZERO/LA(25),LB(25),LC(25),LD(25),ZE(25),YCOUP(0,0), 024070
      ICCOUP(25),IJK(25),KJI(25),IT7(25),ISAVE(0),ZDIA(75),ZODIA(75), 024080
      ZBUS(2775),ZBUSB(2775),ENJS(250),Z(75)            024090
35      COMMON /ZCONST/ NOMU,NBS,IR3M,I4JT,IOUMH        024100
      1000 FORMAT(1X,"NUMBER OF LINES ADDED BY 'FAULT' TRANSFORMER ROU- 024110
      1LINE HAS"/1X,"EXCEEDED MAXIMUM LINE TABLE STORAGE CAPACITY"/ 024120
      21X,"LINE NO.",I4," OF TRANSFORMER TABLE."/)    024130
40      1001 FORMAT(1X,"NUMBER OF LINES ADDED BY 'FAULT' SOURCE IMPEDANCE ROU- 024140
      1LINE HAS"/1X,"EXCEEDED MAXIMUM STORAGE CAPACITY OF LINE TABLES"/ 024150
      21X,"CARD NO.",I4," OF SOURCE IMPEDANCE DATA."/) 024160
      1002 FORMAT(1X,"ERROR IN FEASIBLE BUS LIST ROUTINE"/1X,"NO. OF BUSES 024170
      1IN SYSTEM=",I4/1X,"FEASIBLE BUS LIST"/(14I5)) 024180
45      1003 FORMAT(1X,"NO. OF MUTUAL LINES DECLARED=",I4,"1"/1X, 024190
      1"MAXIMUM ALLOWED=251"/)                          024200
      1004 FORMAT(1X,"NO. OF BUSES IN SYSTEM AREA=",I4,"1"/1X, 024210
      1"MAXIMUM ALLOWED=501"/)                          024220
      1005 FORMAT(1X,"ERROR IN TRANSFORMER CONNECTION CODE! TXFR NO.",I4/ 024230
      11X,"SEE INPUT LINE LIST. WRONG CODE=",I4," CODE IS C+IADD."/) 024240
50      1006 FORMAT(1X,"ERROR BETWEEN TRANSFORMER AND LINE TABLES"/1X, 024250
      1"COULDN'T FIND TRANS. BRANCH (SB=",I4," AND EB=",I4,") CONNECTED 024260
      2TO BUS",I4," IN LINE TABLES."/)                024270
      1007 FORMAT(1X,"'FAULT' SOURCE IMPEDANCE ROUTINE HAS DETECTED A ZERO 024280
      1IMPJT FOR"/1X,"THREE-PH. FAULT CURRENT. THREE-PH. FAULT CURRENT 024290
      2MUST BE NON-ZERO"/1X,"SOURCE IMPEDANCE CARD NO.",I4/) 024300
95      1008 FORMAT(1X,"ERROR IN LINE TABLE! FIRST ENTRY IN LINA SHOULD BE 024310

```

```

1THE REF BUS(=0)."/1X,"ACTUAL TABLE ENTRY=",I4/) 024320
1009 FORMAT(26I3) 024330
60 1010 FORMAT(1M1/T23,29(14°)/T23,1M°,T25,"FAULT SUMMARY FOR BUS",I4,1X 024340
1,14°/T23,1M°,T27,"ZF= ("F5.2,"F5.2,") P.U.",T51,14°/T23,1M° 024350
2,T27,"ZG= ("F5.2,"F5.2,") P.J.",T51,14°/T23,29(14°)/ 024360
1011 FORMAT(T4,"THREE-PHASE",T23,"PHASE-GROUND",T42,"PHASE-PHASE",T50, 024370
1"PH-GND-GROUND"/T2,16(1M°),3X,16(1M°),3X,16(1M°),2X,16(1M°)/T2, 024380
2"IF(MAG)=",F8.4,T21,"IF(MAG)=",F8.4,T40"IF(MAG)=",F8.4,T58, 024390
3"IF(MAG)=",F8.4/T2,"X/R=",F9.7,T21,"X/R=",F8.3,T48,"X/R=",F8.3, 024400
4T51,"X/R=",F8.3/T2,16(1M°),3X,15(1M°),T40,"EF(A)=",F7.4,T58, 024410
5"EF(A)=",F7.4/T40,"EF(B)=",F7.4,T58,"EF(B)=",F7.4/T40,"EF(C)=", 024420
6F7.4,T59,"EF(C)=",F7.4/T40,16(14°),T58,"IF(B)=",F7.4/T59, 024430
7"X/R(B)=",F8.3/T58,"IF(C)=",F8.4/T59,"X/R(C)=",F8.3/T58,16(14°)) 024440
70 1012 FORMAT(T4,"BUS VOLTAGES",T22,"PHASE VOLTAGES"/T2,"BUS",T9, 024450
1"V(MAG)",T24,"A",T29,"B",T34,"C"/(1X,I3,3X,F9.4,5X,F4.2,1X, 024460
2"4.2,1X,F4.2)) 024470
1013 FORMAT(T22,"THREE-PHASE",T41,"PHASE-GROUND"/T20,16(1M°),3X,15(14°) 024480
1/T20,"IF(MAG)=",F8.4,T39,"IF(MAG)=",F8.4/T20,"X/R=",F8.3,T39, 024490
2"X/R=",F8.3/T20,16(1M°),3X,15(1M°)) 024500
1014 FORMAT(/T22,"BUS VOLTAGES",T40,"PHASE VOLTAGES"/T20,"BUS",T27, 024510
1"V(MAG)",T42,"A",T47,"B",T52,"C"/ 024520
2(19X,I3,3X,F9.4,5X,F4.2,1X,F4.2,1X,F4.2)) 024530
80 1015 FORMAT(I3,2X,6F9.2,4F5.1) 024540
1016 FORMAT(4I3,3X,2F10.5) 024550
1017 FORMAT(1X,"ERROR IN FEASIBLE BUS LIST ROUTINE!"/1X,"LINE LIST DOES 024560
1 NOT INCLUDE ALL LINES IN SYSTEM."/1X,"NO. OF LINES IN SYSTEM=", 024570
2I5/1X,"NO. OF LINES IN LIST=",I5/1X,"FEASIBLE LINE LIST"/(4I5)) 024580
95 1018 FORMAT(/1X,70(1M°)/1X,1M°,55X,14°/1X,1M°,T20, 024590
1"RESULTS OF SHORT CIRCUIT ANALYSIS",T71,14°/1X,1M°,T25, 024600
2"ALL VALUES ARE PER-UNIT",T71,14°/1X,1M°,T14, 024610
3"SYSTEM HAS",I4," BUSES. FAULT CODE(SCOP) IS",I3,".",T71, 024620
414°/1X,1M°,T18,"WHERE ARE",I4," SUBSYSTEM STUDIES(ISYS).", 024630
5T71,1M°/1X,1M°,58X,1M°/1X,70(1M°)//) 024640
90 1019 FORMAT(1X,70(1M°)/1X,1M°,68X,14°/1X,1M°,T25,"SUBSYSTEM STUDY NO.", 024650
1I4,".",T71,1M°/1X,14°,T19,"NUMBER OF BUSES IN THIS SYSTEM IS", 024660
2I3,".",T71,1M°/1X,14°,68X,14°/1X,70(1M°)//) 024670
95 1020 FORMAT(1X,70(1M°)/1X,1M°,65X,14°/1X,1M°,T22,"AUTOMATIC SHORT-CIRCUIT 024680
1IT STUDY",T71,14°/1X,1M°,T18,"ENTIRE NETWORK STUDY WILL BE COMPLE 024690
2TE)",T71,1M°/1X,1M°,T19,"IN",I2," PASSES OF SHORT-CIRCUIT PROGRAM. 024700
3",T71,1M°/1X,1M°,68X,1M°/1X,70(14°)//) 024710
1021 FORMAT(T31,"PHASE-GROUND"/T29,15(1M°)/T29,"IF(MAG)=",F7.4/T29, 024720
1"X/R=",F8.3/T29,16(1M°)) 024730
100 1022 FORMAT(/T27,"BUS",T33,"PHASE VOLTAGES"/T35,"A",T40,"B",T45,"C"/ 024740
1(25X,I3,3X,F4.2,1X,F4.2,1X,F4.2)) 024750
1030 FORMAT(T15,I3,1X,I3,1X,F8.4,5X,F9.4) 024760
1031 FORMAT(/T4,"LINE CURRENTS",T22,"LINE CURRENTS") 024770
1032 FORMAT(T2,"LINE",5X,"FAULT(I)",T22,"FAULT(I) PH-A") 024780
105 1033 FORMAT(1X,I3,1X,I3,1X,F8.4,5X,F9.4) 024790
1036 FORMAT(/1X,70(1M°)/1X,1M°,55X,14°/1X,1M°,T25,"SHORT CIRCUIT IMPDT 024800
1DATA",T71,1M°/1X,1M°,65X,14°/1X,70(1M°)//) 024810
1037 FORMAT(1X,"SOURCE IMPEDANCE BUS NO.",I3,5X,"VOLTS L-N(KV)",2F9.2/ 024820
11X,"3-PH FAULT CURRENT(AMPS)",2F9.2/1X,"PH-GND FAULT CURRENT(AMPS 024830
2)",2F9.2/1X,"FAULT Z(OHMS)",2F6.1,"1 NEUT Z(OHMS)",2F5.1//) 024840
110 1038 FORMAT(1X,"Z0R AND Z0I OF LINE",I4," TO",I4," SHOULD'N'T BE ZERO.") 024850
1039 FORMAT(1X,"Z0IA OF BUS",I4," SHOULD NOT BE ZERO.") 024860
1040 FORMAT(/T31,"LINE CURRENTS"/T29,"LINE",T38,"FAULT(I)") 024870
1041 FORMAT(T27,I3,1X,I3,4X,F7.6) 024880

```

115	1042 FORMAT(1X,"MUTUAL COUPLING DATA"/1X,"LA=",I3,"LB=",	024890
	I3,"LR=",I3,"LS=",I3/1X,"MUTUAL IMPEDANCE (OHMS)="",2F10.5/)	024900
	1043 FORMAT(1X,"FAULT IMPEDANCE(ZF)(OHMS)="",2(F6.3,1X)/1X	024910
	1,"PHASE=",I2,1X,"BUS VOLTAGE(KV)="",F6.2)	024920
	1044 FORMAT(R6,I3,2F6.0,I3,F6.0)	024930
120	1045 FORMAT(1X,"BUS",I3)	024940
	1046 FORMAT(/T22,"LINE CURRENTS",T41,"LINE CURRENTS"/T20,"LINE",	024950
	25X,"FAULT(I)",T42,"FAULT(I) "4-A")	024960
	1047 FORMAT(R6,2I3)	024970
	1048 FORMAT(I3,"PROGRAM CONTROL CARD NOT IN PROPER FORMAT OR LOCATION.	024980
125	1CARD WITH KEYWORD ",R6," IS REJECTED.")	024990
	1049 FORMAT(I3,"BUS B CAN NOT BE IN SJASYSYSTEM LIST. DOES NBS NUMBER	025000
	1EQUA. NUMBER OF BUSES LISTED ON SUBSYSTEM CARD?")	025010
	1050 FORMAT("PH PLEASE USE LU06...")	025020
	1051 FORMAT(I2,I3,A10,2F5.0,4F10.0)	025030
130	1052 FORMAT(R6,I3)	025040
	READ(5,1047) A,ISYS,SCOP	025050
	IF(OUT.EQ.11.OR.OUT.EQ.14) WRITE(2,1050)	025060
	3=SRMTCKT	025070
	IF(A.EQ.C) GO TO 5	025080
135	3=SRDLFLOW	025090
	IF(A.NE.D) GO TO 912	025100
	30 2 I=1,NBUS	025110
	READ(5,1051) IDB(I),IBUS(I),BUSNAME(I),V(I),ANG(I),P(I),Q(I),	025120
	IQMIN(I),QMAX(I)	025130
140	2 CONTINUE	025140
	READ(5,1047) A,ISYS,SCOP	025150
	31=SRBUSCHG	025160
	4 IF(A.EQ.C) GO TO 5	025170
	IF(A.NE.D) GO TO 912	025180
145	READ(5,1051) IDB(I),IBUS(I),BUSNAME(I),V(I),ANG(I),P(I),Q(I),	025190
	IQMIN(I),QMAX(I),I=1,ISYS)	025200
	READ(5,1047) A,ISYS,SCOP	025210
	30 TO 4	025220
	C THE FOLLOWING SECTION PROCESSES TRANSFORMER DATA TO ADD LINES	025230
150	C REFERENCE IF REQUIRED IN THE ZERO SEQUENCE.	025240
	5 IF(OUT.EQ.2.OR.OUT.EQ.10) GO TO 5	025250
	WRITE(1,1036)	025260
	6 MAXLIN=1450	025270
	IZADD=0	025280
155	30 7 I=1,NBUS	025290
	4A(I)=CONEC(I)	025300
	3LP(I)=DLQ(I)=0.	025310
	7 CONTINUE	025320
	IF(NOTR.EQ.0) GO TO 51	025330
160	30 50 I=1,NOTR	025340
	IF(ICC(I).LT.41) GO TO 905	025350
	IF(ICC(I).LT.61) GO TO 9	025360
	IF(ICC(I).LT.71) GO TO 10	025370
	IF(ICC(I).LT.91) GO TO 15	025380
165	IF(ICC(I).GT.100) GO TO 905	025390
	C FALL THROUGH: TRANSFORMER IS PHASE SHIFTER TYPE:	025400
	IF(ICC(I).EQ.91.OR.ICC(I).EQ.94.OR.ICC(I).GT.96) GO TO 20	025410
	C TRANSFORMER IS TYPE FIXED:	025420
170	9 IF(ICC(I).LT.44.OR.ICC(I).EQ.46.OR.ICC(I).GT.48) GO TO 34	025430
	IF(ICC(I).EQ.44) GO TO 30	025440
	IF(ICC(I).EQ.45.OR.ICC(I).EQ.48) GO TO 25	025450

	30 TO 985	025460
	C TRANSFORMER IS TYPE AUTO:	025470
175	10 IF(ICC(I).LT.63.JR.ICC(I).EQ.64.JR.ICC(I).EQ.67) GO TO 40	025480
	IF(ICC(I).EQ.63.JR.ICC(I).EQ.65.JR.ICC(I).EQ.66) GO TO 20	025490
	IF(ICC(I).EQ.68) GO TO 25	025500
	30 TO 985	025510
	C TRANSFORMER IS TYPE T3UL:	025520
180	15 IF(ICC(I).LT.74.JR.ICC(I).EQ.75.JR.ICC(I).GT.78) GO TO 50	025530
	IF(ICC(I).EQ.77) GO TO 20	025540
	IF(ICC(I).EQ.75.JR.ICC(I).EQ.78) GO TO 25	025550
	IF(ICC(I).EQ.74) GO TO 30	025560
	30 TO 985	025570
185	C ADD LINE FROM P TO REF: ADD LINE FROM 2 TO REF:	025580
	20 K1=JADD(LTRA(I))+IZ	025590
	K2=K1+NA(LTRA(I))-1	025600
	DO 21 J=K1,K2	025610
	IF(LINB(J).EQ.LTRA(I)) GO TO 22	025620
190	21 CONTINUE	025630
	30 TO 986	025640
	22 NL=NL+2	025650
	IF(NL.GT.MAXLIN) GO TO 900	025660
	LINA(NL)=0	025670
195	LINB(NL)=LTRA(I)	025680
	LINA(NL-1)=LTRA(I)	025690
	LINB(NL-1)=0	025700
	ZOR(NL)=.0*ZOR(J)	025710
	ZOI(NL)=.0*ZOI(J)+.1	025720
200	ZOR(NL-1)=ZOR(NL)	025730
	ZOI(NL-1)=ZOI(NL)	025740
	CONEC(LINA(J))=CONEC(LINA(J))+1	025750
	IZADD=IZADD+1	025760
	NL=NL+2	025770
205	IF(NL.GT.MAXLIN) GO TO 900	025780
	LINA(NL)=0	025790
	LINB(NL)=LTRA(I)	025800
	LINA(NL-1)=LTRA(I)	025810
	LINB(NL-1)=0	025820
210	ZOR(NL)=2.*ZOR(J)	025830
	ZOI(NL)=2.*ZOI(J)	025840
	ZOR(NL-1)=ZOR(NL)	025850
	ZOI(NL-1)=ZOI(NL)	025860
	CONEC(LINB(J))=CONEC(LINB(J))+1	025870
215	IZADD=IZADD+1	025880
	30 TO 90	025890
	C ADD LINE FROM Q TO REF: Q=EB	025900
	25 K1=JADD(LTRA(I))+IZ	025910
	K2=K1+NA(LTRA(I))-1	025920
220	DO 26 J=K1,K2	025930
	IF(LINB(J).EQ.LTRA(I)) GO TO 27	025940
	26 CONTINUE	025950
	30 TO 986	025960
225	27 NL=NL+2	025970
	IF(NL.GT.MAXLIN) GO TO 900	025980
	LINA(NL)=0	025990
	LINB(NL)=LTRA(I)	026000
	LINA(NL-1)=LTRA(I)	026010
	LINB(NL-1)=0	026020

230	ZOR(NL)=ZOR(J)	026070
	ZOI(NL)=ZOI(J)	026080
	ZOR(NL-1)=ZOR(NL)	026090
	ZOI(NL-1)=ZOI(NL)	026100
	CONEC(LIN9(J))=CONEC(LIN9(J))+1	026110
	IZADD=IZADD+1	026120
235	DO 73 40	026130
	C ADD LINE FROM P TO REF1	026140
	30 K1=JADD(LTRA(I))+IZ	026150
	K2=K1+NA(LTRA(I))-1	026160
240	DO 31 J=K1,K2	026170
	IF(LIN9(J).EQ.LTR9(I)) GO 73 32	026180
	31 CONTINUE	026190
	30 73 906	026200
	32 NL=NL+2	026210
	IF(NL.GT.MAXLIN) GO TO 900	026220
245	LIN9(NL)=0	026230
	LIN9(NL-1)=LTRA(I)	026240
	LIN9(NL-1)=LTRA(I)	026250
	LIN9(NL-1)=0	026260
250	ZOR(NL)=ZOR(J)	026270
	ZOI(NL)=ZOI(J)	026280
	ZOR(NL-1)=ZOR(NL)	026290
	ZOI(NL-1)=ZOI(NL)	026300
	CONEC(LINA(J))=CONEC(LINA(J))+1	026310
	IZADD=IZADD+1	026320
255	DO 73 40	026330
	34 IF(ICC(I).LT.43) 35,60	026340
	35 K1=JADD(LTRA(I))+IZ	026350
	K2=K1+NA(LTRA(I))-1	026360
260	DO 36 J=K1,K2	026370
	IF(LIN9(J).EQ.LTR9(I)) GO TO 37	026380
	36 CONTINUE	026390
	30 73 906	026400
	37 ZOR(J)=1.E10	026410
	ZOI(J)=0.	026420
265	K1=JADD(LTR9(I))+IZ	026430
	K2=K1+NA(LTR9(I))-1	026440
	DO 38 J=K1,K2	026450
	IF(LIN9(J).EQ.LTRA(I)) GO TO 39	026460
	38 CONTINUE	026470
270	30 73 906	026480
	39 ZOR(J)=1.E10	026490
	ZOI(J)=0.	026500
	DO 73 50	026510
275	40 IF(ICC(I).EQ.44.OR.ICC(I).EQ.45.OR.ICC(I).GT.47	026520
	1.AND.ICC(I).LT.51).OR.ICC(I).EQ.51.OR.ICC(I).EQ.60.OR.ICC(I).EQ.71	026530
	2.OR.ICC(I).EQ.72.OR.ICC(I).EQ.74.OR.ICC(I).EQ.75.OR.ICC(I).GT.	026540
	377.AND.ICC(I).LT.81) 41,50	026550
	C LTRA=SB,LTR9=ER	026560
	41 L9US=LTRA(I)	026570
280	IF(LTR9(I).GT.LTRA(I)) L9US=LTR9(I)	026580
	L9US=LTR9(I)	026590
	IF(LTR9(I).GT.LTRA(I)) L9JS=LTRA(I)	026600
	K1=JADD(L9US)+IZ	026610
	K2=K1+NA(L9US)-1	026620
285	DO 42 J=K1,K2	026630

	IF(LINE(J).EQ.LLOUS) GO TO 43	026600
	42 CONTINUE	026610
	GO TO 906	026620
290	43 ZOR(J)=1.E10	026630
	ZOI(J)=0.	026640
	50 CONTINUE	026650
	C THE FOLLOWING SECTION PROCESSES GENERATOR BUSES TO ADD SOURCE	026660
	C IMPEDANCES. THIS INCLUDES THE SLACK BUS. THE SOURCE IMPEDANCES	026670
	C ARE CALCULATED FROM THE THREE-PHASE AND SINGLE-PHASE TO GROUND	026680
295	C FAULTS FOR EACH BUS, WHICH ARE READ IN BY THE FOLLOWING ROUTINE.	026690
	C THIS ROUTINE MAY BE BYPASSED IF THE SOURCE IMPEDANCES ARE TO BE	026700
	C IGNORED, AND THERE IS AT LEAST ONE LINE TO REFERENCE IN THE LINE	026710
	C TABLE. THE ROUTINE IS BYPASSED IF IREF=0.	026720
	51 READ(9,1052) AA,IREF	026730
300	3C=6RCURSOR	026740
	IF(AA.NE.CC) GO TO 314	026750
	IF(IREF.EQ.0) GO TO 65	026760
	30 50 I=1,IREF	026770
	READ(9,1015) IBF,V3,FA3,FA1,ZF,Z3	026780
305	IF(3JT.EQ.2.OR.3JT.EQ.10) GO TO 52	026790
	WRITE(1,1037) IBF,V3,FA3,FA1,ZF,Z3	026800
	52 VP=CABS(V3)	026810
	Z3=3000.*VP*VP/3KVA	026820
	Z1=CABS(FA3)	026830
310	IF(Z1.EQ.0.) GO TO 907	026840
	IF(Z3.EQ.0) GO TO 907	026850
	ZZ=V3*1000./FA3-ZF	026860
	Z1=CABS(FA1)	026870
	IF(Z1.EQ.0.) GO TO 55	026880
315	ZZ0=(3000.*V3/FA1-2.*ZZ-3.*ZF)/Z3	026890
	ZZ=ZZ/Z3	026900
	GO TO 56	026910
	55 ZZ=CMPLX(1.E10,0.)	026920
	ZZ=ZF/Z3	026930
320	56 NL=NL+2	026940
	IF(NL.GT.MAXLIN) GO TO 901	026950
	LINA(NL)=0	026960
	LINA(NL)=IBF	026970
325	LINA(NL-1)=IBF	026980
	LINA(NL-1)=0	026990
	G(NL)=REAL(1./ZZ)	027000
	B(NL)=AIMAG(1./ZZ)	027010
	G(NL-1)=G(NL)	027020
	B(NL-1)=B(NL)	027030
330	ZOR(NL)=REAL(ZZ0)	027040
	ZOI(NL)=AIMAG(ZZ0)	027050
	ZOR(NL-1)=ZOR(NL)	027060
	ZOI(NL-1)=ZOI(NL)	027070
	ZZG(IBF)=ZF/Z3	027080
335	IZADD=IZADD+1	027090
	CONEC(IBF)=CONEC(IBF)+1	027100
	60 CONTINUE	027110
	IF(IERR.NE.0) RETURN	027120
340	C ADDED LINES ARE NOW COMPLETE; RESORT LINE TABLES INTO ASCENDING	027130
	C BUS ORDER	027140
	65 CALL LSORT(2,LINA,LINA,NL,4,3,0,ZOR,ZOI,0)	027150
	IZ=IZ+IZADD	027160

	30 65 I=1,NBUS	027170
	VA(I)=CONEC(I)	027180
345	66 CONTINUE	027190
	C THE FOLLOWING SECTION FORMS A FEASIBLE ORDERING OF THE BUS LISTS	027200
	C FOR THE BUS IMPEDANCE BUILDING ALGORITHM.	027210
	IK=1	027220
	IF(LINA(I).NE.0) GO TO 908	027230
350	.IST(1)=LINA(I)	027240
	K=2	027250
	30 75 I=1,NBUS	027260
	IR4=LIST(I)	027270
	Q1=JADD(IR4)+IZ	027280
355	Q2=K1+CONEC(IR4)-1	027290
	DO 75 J=K1,K2	027300
	IF(LINA(J).EQ.0) GO TO 74	027310
	KK=K-1	027320
	DO 71 N=1,KK	027330
360	IF(LIST(N).EQ.LINA(J)) GO TO 73	027340
	CONTINUE	027350
	LIST(K)=LINA(J)	027360
	K=K+1	027370
	73 IF(LINA(J).ST.LINA(J)) GO TO 75	027380
365	74 JBP(K)=J	027390
	IK=IK+1	027400
	75 CONTINUE	027410
	76 CONTINUE	027420
	IF(K.NE.(NBUS+1)) GO TO 902	027430
370	ISS=NL/2	027440
	IF(IK.NE.(ISS+1)) GO TO 903	027450
	C AFTER THE LINES ARE REORDERED INTO A FEASIBLE LIST, THE MUTUAL	027460
	C COUPLING DATA IS READ IN. THIS INCLUDES THE MUTUALS FOR ALL LINES	027470
	C IN THE SYSTEM, REGARDLESS OF HOW THE SHORT-CIRCUIT STUDY IS TO PRO-	027480
375	C CEE) (I.E. BY SUBSYSTEM AREAS OR COMPLETE SYSTEM). A MAXIMUM OF	027490
	C 25 MUTUALLY COUPLED LINES ARE ALLOWED FOR IN THE PROGRAM.	027500
	READ(5,1052) A1,NOMU	027510
	Z1=BRNOMUTL	027520
	IF(A1.NE.C1) GO TO 915	027530
380	IF(NOMU.EQ.0) GO TO 77	027540
	IF(NOMU.GT.25) GO TO 903	027550
	READ(5,1016) (LA(I),LB(I),LR(I),.S(I),ZN(I),I=1,NOMU)	027560
	IF(DJT.EQ.2.OR.DJT.EQ.10) GO TO 77	027570
385	WRITE(1,1042) (LA(I),L9(I),.R(I),LS(I),Z4(I),I=1,NOMU)	027580
	C THE FOLLOWING SECTION PROCESSES FAULT IMPEDANCE DATA TO BE USED IN:	027590
	C THE FAULT CALCULATIONS. THE FIRST CARD READ IN INDICATES HOW	027600
	C MANY FAULT IMPEDANCES WILL BE ENTERED(NOFALT). THE FAULT IMPEDANCE	027610
	C (OHMS), NO. OF PHASES, AND PHASE-GND VOLTAGE(KV). SUCCEEDING CARDS	027620
	C READ IN THE LIST OF BUS NUMBERS TO WHICH THIS FAULT DATA APPLIES.	027630
390	C A BANK CARD SIGNALS THE END OF THE FAULT IMPEDANCE DATA. NOFALT=NFT.	027640
	77 READ(5,1044) A2,NOFALT,ZF,PH,VP	027650
	Z2=BRNOFALT	027660
	IF(A2.NE.C2) GO TO 916	027670
	IF(NOFALT.EQ.0) GO TO 84	027680
395	IF(DUT.EQ.2.OR.DJT.EQ.10) GO TO 90	027690
	WRITE(1,1043) ZF,PH,VP	027700
	80 IF(PH.EQ.1) BKVA=BKVA/3.	027710
	ZB=(1000.*VP*VP)/BKVA	027720
	IF(PH.EQ.1) BKVA=BKVA1	027730

400	KVZ=NOFALT/26.+.99	027740
	INT=KVZ	027750
	DO 83 I=1,INT	027760
	READ(5,1009) IF8	027770
	DO 82 J=1,25	027780
405	IF(IF8(J).EQ.0) GO TO 83	027790
	IF(OUT.E2.2.OR.OJT.E3.10) GO TO 81	027800
	WRITE(1,1045) IF8(J)	027810
81	DLP(IF8(J))=REAL(ZF/Z9)	027820
	DLQ(IF8(J))=AIMAG(ZF/Z9)	027830
410	82	027840
	CONTINJE	027850
	83	027860
	CONTINUE	027860
	GO TO 77	027870
	C THE FOLLOWING SECTION BEGINS THE BUS BUILDING ALGORITHM FOR THE	027880
	C POSITIVE AND ZERO SEQUENCE IMPEDANCE MATRICES. IF SHORT-CIRCUIT	027890
415	C IS TO BE PERFORMED ON COMPLETE NETWORK (ISYS=0) SKIP TO 100.	027900
	C OTHERWISE, READ IN AREA BUS LIST(S). A MAXIMUM OF 50 BUSES PER	027910
	C AREA IS ALLOWED. VALUES OF ISYS FROM 1 TO 99 INDICATE HOW MANY	027920
	C AREAS ARE TO BE STUDIED. FIRST, THE BUS VOLTAGES ARE INITIALIZED:	027930
	84 IF(LODOP.EQ.-1.AND.SCOP.EQ.1) SCJP=0	027940
420	IF(LODOP.EQ.-1.AND.SCOP.EQ.3) SCJP=2	027950
	IF(SCOP.EQ.1.OR.SCJP.EQ.3) GO TO 86	027960
	DO 85 I=1,NBUS	027970
	85 VBJS(I)=CMPLX(1.,0.)	027980
	GO TO 90	027990
425	86 DO 87 I=1,NBUS	028000
	87 VBJS(I)=CMPLX(V(I),ANG(I))	028010
	88 WRITE(2,1010) NBUS,SCOP,ISYS	028020
	IF(ISYS.EQ.0) GO TO 100	028030
	ICOUNT=ISYS	028040
430	95 IF(ICOUNT.EQ.0) RETURN	028050
	IOJT=ISYS-ICOUNT+1	028060
	READ(5,1052) AS,NBS	028070
	N3=6*NBUS	028080
	IF(C3.NE.A3) GO TO 917	028090
435	IF(NBS.GT.50) GO TO 904	028100
	WRITE(2,1019) IOJT,NBS	028110
	READ(5,1009) (NB(I),I=1,NBS)	028120
	DO 95 I=1,NBS	028130
	IF(NB(I).EQ.0) GO TO 913	028140
440	96 CONTINUE	028150
	GO TO 105	028160
	100 DO 101 I=1,NBUS	028170
	VB(I)=LIST(I)	028180
445	101 CONTINUE	028190
	KVZ=NBUS/50.+.99	028200
	IOJT=KVZ	028210
	WRITE(2,1020) IOJT	028220
	NBS=50	028230
	IF(NBUS.LT.50) NBS=NBUS	028240
450	105 4=0	028250
	C THE POSITIVE SEQ. BUILDING ALGORITHM IS SUBROUTINE BUS. THE POSI-	028260
	C TIVE SEQ. MATRIX IS STORED AS A VECTOR OF DIAGONAL TERMS (ZDIA),	028270
	C AND AS A VECTOR OF THE UPPER-TRIANGLE TERMS (ZBUS).	028280
	106 CALL BUS(N)	028290
455	IF(IERR.NE.0) RETURN	028300
	C THE ZERO SEQ. BUILDING ALGORITHM IS SUBROUTINE BUS0. THE ZERO	

```

C SEQ. MATRIX IS ALSO STORED AS A VECTOR OF DIAGONAL TERMS (ZODIA), 020310
C AND A VECTOR OF UPPER TRIANGLE TERMS (ZBUS). MUTUAL COUPLING 020320
C IS HANDLED BY SUBROUTINE MUTEST, WHICH IS CALLED BY BUSO. 020330
460 CALL BUSO(N) 020340
    IF(IERR.NE.0) RETURN 020350
C THE FOLLOWING SECTION CALCULATES THE FAULT CURRENTS. THE OPTION 020360
C IS PROVIDED TO CALCULATE ALL FAULT TYPES (3-PH, L-2, L-L, AND 020370
C L-L-G), OR JUST THE FIRST TWO. 020380
465 DO 200 I=1,NBS 020390
    IR4=IUBP(I) 020400
    ZF=ZPLX(DLP(IR4),JL2(IR4)) 020410
    ZG=ZTG(IR4) 020420
    IF(IPHASE(IR4).EQ.1) GO TO 132 020430
470 C CALCULATE THE THREE-PHASE FAULT FOR THE ITH BUS: 020440
    ZZ=ZF+ZDIA(I) 020450
    IF(ZZ.EQ.0) GO TO 310 020460
    AMPA=EBUS(IR4)/ZZ 020470
    (R=ATNAG(ZZ)/REA.(ZZ) 020480
    FAJLTI=CABS(AMPA) 020490
475 C CHECK TO SEE IF BUS VOLTAGE SUMMARY IS TO BE COMPUTED: 020500
    IF(OUT.EQ.12.OR.OUT.EQ.10) GO TO 132 020510
C COMPUTE VOLTAGE SUMMARIES: 020520
480 DO 124 J=1,NBS 020530
    IRN1=IUBP(J) 020540
    IF(J.EQ.I) GO TO 123 020550
    N1=IADD(J,I) 020560
    EC=EBUS(IRN1)-ZBUS(N1)*E9JS(IR4)/ZZ 020570
    DU(J)=CABS(EC) 020580
    GO TO 124 020590
485 123 EC=ZF*EBUS(IR4)/ZZ 020600
    DU(I)=CABS(EC) 020610
    124 CONTINUE 020620
C COMPUTE PHASE-GROUND FAULT FOR ITH BUS: 020630
490 132 ZZ=ZODIA(I)+2.*ZDIA(I)+3.*ZF 020640
    AMPA=3.*EBUS(IR4)/ZZ 020650
    ZZE=ZZ/3. 020660
    (RLG=ATNAG(ZZE)/REA.(ZZE) 020670
    FAJLTLG=CABS(AMPA) 020680
495 C CHECK TO SEE IF BUS PHASE VOLTAGE SUMMARIES ARE TO BE COMPUTED: 020690
    133 IF(OUT.EQ.12.OR.OUT.EQ.10) GO TO 143 020700
C CALCULATE PHASE VOLTAGE SUMMARIES: 020710
500 DO 135 J=1,NBS 020720
    IRN1=IUBP(J) 020730
    IF(J.EQ.I) GO TO 134 020740
    N1=IADD(J,I) 020750
    EC=EBUS(IRN1)-EBUS(IR4)*(ZBUS(N1)+2.*ZBUS(N1))/ZZ 020760
    UMP(J)=CABS(EC) 020770
    IF(IPHASE(IRN1).EQ.1) GO TO 135 020780
505 EC=EBUS(IRN1)*(-.5,+.966)-ZBUS(IR4)*(ZBUS(N1)-ZBUS(N1))/ZZ 020790
    UMP(NBS+J)=CABS(EC) 020800
    EC=EBUS(IR4)*(-.5,+.966)-E9JS(IR4)*(ZBUS(N1)-ZBUS(N1))/ZZ 020810
    UMP(2*NBS+J)=CABS(EC) 020820
    GO TO 135 020830
510 134 EC=EBUS(IR4)*3.*ZF/ZZ 020840
    UMP(J)=CABS(EC) 020850
    IF(IPHASE(IR4).EQ.1) GO TO 135 020860
    EC=EBUS(IR4)*(-.5,+.966)-E9JS(IR4)*(ZODIA(I)-ZDIA(I))/ZZ 020870

```

```

          UBP(NBS+J)=CABS(EC)
915      EC=EBUS(IRW)*(-.9,.855)-EJUS(IRW)*(ZODIA(I)-ZDIA(I))/ZZ      020000
          UBP(2*NBS+J)=CABS(EC)      020090
135      CONTINUE      020900
C COMPUTE LINE FAULT CURRENTS FOR 3-PHASE, AND LINE-GND.      020920
C USE Z MATRIX(ZBUS AND ZDIA). SCAN LINE TABLE FOR LINES CONNECTING      020930
920 C BUSES IN ISYS, THEN CAL. FAULT CURRENT AS CJ(LL) AND CUR(LL).      020940
          NBS=NBS
          LL=1      020950
          JO 140 NN=1,51      020960
          LI(NN)=L2(NN)=0      020970
925      140 CONTINUE      020980
          JO 140 II=1,NBS      020990
          DO 147 L=1,NL      029000
          IF(LINA(L).NE.IJSP(II)) GO TO 147      029010
          IF(LINA(L).LT.LI48(L)) GO TO 147      029020
930      DO 146 JJ=1,N3A      029030
          IF(LINB(L).NE.IJSP(JJ)) GO TO 146      029040
          IF(CMPLX(ZOR(L),ZOI(L)).EQ.0) GO TO 911      029050
          IF(JJ.EQ.I) GO TO 142      029060
          IF(II.EQ.I) GO TO 143      029070
935      N1=IADD(II,I)      029080
          N2=IADD(JJ,I)      029090
          FI=ZBUS(N2)-ZBUS(N1)      029100
          PMS=2*(ZBUS(N1)-ZBUS(N2))*CMPLX(G(L),B(L))      029110
          ZS=(ZBUS(N1)-ZBUS(N2))/CMPLX(ZOR(L),ZOI(L))      029120
940      GO TO 145      029130
          142 N1=IADD(II,I)      029140
          FI=ZDIA(I)-ZBUS(N1)      029150
          GO TO 144      029160
          143 N1=IADD(JJ,I)      029170
          FI=ZBUS(N1)-ZDIA(I)      029180
          GO TO 144      029190
          144 PMS=2*(ZDIA(I)-ZBUS(N1))*CMPLX(G(L),B(L))      029200
          ZS=(ZDIA(I)-ZBUS(N1))/CMPLX(ZOR(L),ZOI(L))      029210
945      145 FIG=(PMS+ZS)/ZZ      029220
          FIA=FI*CMPLX(G(L),B(L))/(ZDIA(I)+ZF)      029230
          CU(LL)=CABS(FIA)      029240
          IF(IPHASE(IRW).EQ.1) CU(LL)=0      029250
          CUR(LL)=CABS(FIG)      029260
          LI(LL)=LINA(L)      029270
          LZ(LL)=LINB(L)      029280
950      LL=LL+1      029290
          146 CONTINUE      029300
          147 CONTINUE      029310
          148 CONTINUE      029320
955 C CHECK TO SEE IF L-L AND L-L-G FAULTS ARE TO BE CALCULATED!      029330
          149 IF(SCOP.EQ.2.OR.SCOP.EQ.3) GO TO 150      029340
          IF(IPHASE(IRW).EQ.1) GO TO 170      029350
          C CALCULATE L-L FAULT FOR ITH BUS(A AND 3 PHASES FAULTED)!      029360
          ZZ=ZF+2.*ZDIA(I)      029370
          IF(ZZ.EQ.0) GO TO 910      029380
          ANPA=1.732*EBUS(IRW)/ZZ      029390
          ZZE=ZZ/1.732      029400
          RLL=ATHAG(ZZE)/REAL(ZZE)      029410
          FAULTLL=CABS(ANPA)      029420
960      EC=EBUS(IRW)*(ZF+2.*ZDIA(I))/ZZ      029430
          029440

```

```

575      JBP(1001)=CABS(E)
          EG=EGUS(IRN)*((-5,-.866)*ZF-ZDIA(I))/ZZ
          JBP(1002)=CABS(E)
          EG=EGUS(IRN)*((-5,.866)*ZF-ZDIA(I))/ZZ
          JBP(1003)=CABS(E)
C COMPUTE L-L-G FAULT FOR ITH BUS:
      ZZ=ZDIA(I)+2.*(ZDIA(I)+3.*(ZF+ZG))
      AMPA=3.*EGUS(IRN)/ZZ
      FALTLG=CABS(AMPA)
580      XRLG=AIMAG(ZZ)/REAL(ZZ)
          ZZ=ZDIA(I)*ZDIA(I)+2.*ZDIA(I)*(ZDIA(I)+3.*(ZF+ZG))
          AMPA=1.732*EGUS(IRN)*(ZDIA(I)-(-5,.866)*ZDIA(I)+3.*(ZF+ZG))/ZZ
          FAJLTLG=CABS(AMPA)
          ZZ=ZZ/(1.732*(ZDIA(I)-(-5,.866)*ZDIA(I)+3.*(ZF+ZG)))
585      XRLG=AIMAG(ZZ)/REAL(ZZ)
          IF(XRLG.LT.0) XRLG=XRLG*(-1.)
          AMPA=1.732*EGUS(IRN)*(ZDIA(I)-(-5,-.866)*ZDIA(I)+3.*(ZF+ZG))/ZZ
          FAJLTLG=CABS(AMPA)
          ZZ=ZZ/(1.732*(ZDIA(I)-(-5,-.866)*ZDIA(I)+3.*(ZF+ZG)))
590      XRLG=AIMAG(ZZ)/REAL(ZZ)
          EG=3.*EGUS(IRN)*ZDIA(I)*(ZDIA(I)+2.*(ZF+ZG))/ZZ
          UBP(1004)=CABS(E)
          EG=-3.*EGUS(IRN)*ZDIA(I)*(ZF+ZG)/ZZ
          JBP(1005)=CABS(E)
595      C LIST ITH BUS FAULT SUMMARIES ON OUTPUT FILE:
          150 WRITE(2,1010) IRN,ZF,ZG
              WRITE(2,1011) FAJLTLG,FAULTLG,XR,XRLG,XRLG,XRLG,
              JBP(1001),UBP(1004),UBP(1002),UBP(1005),UBP(1003),UBP(1005),
              ZFAJLTLG,XRLG,FAULTLG,XRLG
          IF(OUT.EQ.12.OR.OUT.EQ.10) GO TO 200
          IF(OUT.EQ.13.OR.OUT.EQ.14) GO TO 151
          WRITE(2,1012) (IUBP(J),OU(J),UBP(J),UBP(NBS+J),UBP(2*NBS+J),
              J=1,NBS)
          151 WRITE(2,1031)
              WRITE(2,1032)
              MM=LL-1
              WRITE(2,1033) (L1(N),L2(N),CU(N),CUR(N),N=1,MM)
              GO TO 200
          160 WRITE(2,1010) IRN,ZF,ZG
              WRITE(2,1013) FAJLTLG,FAULTLG,XR,XRLG
          IF(OUT.EQ.12.OR.OUT.EQ.10) GO TO 200
          IF(OUT.EQ.13.OR.OUT.EQ.14) GO TO 151
          WRITE(2,1014) (IUBP(J),OU(J),UBP(J),UBP(NBS+J),UBP(2*NBS+J),
              J=1,NBS)
          161 MM=LL-1
              WRITE(2,1046)
              WRITE(2,1030) (L1(N),L2(N),CU(N),CUR(N),N=1,MM)
              GO TO 200
          170 WRITE(2,1010) IRN,ZF,ZG
              WRITE(2,1021) FAJLTLG,XRLG
          IF(OUT.EQ.12.OR.OUT.EQ.10) GO TO 200
          IF(OUT.EQ.13.OR.OUT.EQ.14) GO TO 171
          WRITE(2,1022) (IUBP(J),UBP(J),UBP(NBS+J),JBP(2*NBS+J),J=1,NBS)
          171 WRITE(2,1040)
              MM=LL-1
              WRITE(2,1041) (L1(N),L2(N),CU(N),N=1,MM)
          200 CONTINUE

```

	ICOUNT=ICOUNT-1	030020
	M=4*NBS	030030
630	IF(M.LT.NBUS) NBS=NBS-M	030040
	IF(NBS.GT.50) NBS=50	030050
	IF(M.EQ.NBUS) M=0	030060
	IF(ISYS.NE.0) GO TO 95	030070
	IF(M.EQ.0) RETURN	030080
635	GO TO 106	030090
	900 WRITE(2,1000) I	030100
	GO TO 950	030110
	901 WRITE(2,1001) I	030120
	GO TO 950	030130
640	902 WRITE(2,1002) N9JS,(LIST(I),I=1,(030140
	GO TO 950	030150
	903 WRITE(2,1003) N04U	030160
	GO TO 950	030170
	904 WRITE(2,1004) NBS	030180
645	GO TO 950	030190
	905 WRITE(2,1005) I,IC0(I)	030200
	GO TO 950	030210
	906 WRITE(2,1006) LTRA(I),LTRB(I),LTRA(I)	030220
	GO TO 950	030230
650	907 WRITE(2,1007) I	030240
	GO TO 950	030250
	908 WRITE(2,1008)	030260
	GO TO 950	030270
	909 WRITE(2,1017) ISS,IK,(JDP(I),I=1,IK)	030280
655	GO TO 950	030290
	910 WRITE(2,1039) IU3P(I)	030300
	GO TO 950	030310
	911 WRITE(2,1030) LINA(L),LINA(L)	030320
	GO TO 950	030330
660	912 WRITE(2,1040) C	030340
	GO TO 950	030350
	913 WRITE(2,1049)	030360
	GO TO 950	030370
	914 WRITE(2,1040) CC	030380
665	GO TO 950	030390
	915 WRITE(2,1040) C1	030400
	GO TO 950	030410
	916 WRITE(2,1040) C2	030420
	GO TO 950	030430
670	917 WRITE(2,1040) C3	030440
	990 IERR=IERR+1	030450
	RETURN	030460
	END	030470

```

1      SUBROUTINE BUS(M)                                030480
      INTEGER CONEC                                    030490
      COMPLEX ZBUS,ZOBUS,ZOIA,ZOIA,ZM,YCOUP,ZCOUP,EBUS,ZC,ZLL 030500
      COMMON /COMB/LINA(1450),LINA(1450),S(1450),R(1450),P(250),Q(250), 030510
5      LPH9(50),PHANG(50),LTRA(250),LTRA(250),TAP(250),T4N(250),V(250), 030520
      ZTK(250),IUBP(250),ANG(250),IBUS(250),JBP(250),UBP(3000), 030530
      BUSNAME(250),LPHA(50),LIST(250),IUP(250),ZMIN(250),ZMAX(250), 030540
      JBP(250),URPP(3000),JBP(3000),JBP(3000),ICC(250),OLP(250), 030550
      SZOT(1450),ZOR(1450),BOIA(250),CONEC(250),DLO(250),IPHASE(250) 030560
10     COMMON/CONC/ NA(250),N9(250),JCOL(1000),JJ(1000),IOB(250) 030570
      COMMON/CONST/ N9BUS,NL,ISS,IPV,LL1,LL2,LL3,LL4,NOTR,IZ,NOLTC 030580
      I,ITR1,ITR2,PTOL,ITOL,NLC                        030590
      COMMON /SAVE/ IERR                                030600
      COMMON /ZERO/LA(25),LB(25),LC(25),LD(25),ZE(25),YCOUP(8,8), 030610
15     ZCOUP(25),IJK(25),KJI(25),IT7(25),ISAVE(8),ZOIA(75),ZOOIA(75), 030620
      ZBUS(2775),ZOBUS(2775),EBUS(250),ZC(75)         030630
      COMMON /CONST/ NOMU,N9S,IRON,I4JT,IOU4M         030640
1000  FORMAT(1X,"ERROR IN POS. SEQ BUILDING ALGORITHM. BUS",I4, 030650
      " OR BUS",I4," SHOULD HAVE BEEN IN SUBSYSTEM BUS LIST BUT WASN'T") 030660
20     1001 FORMAT(1X,"ERROR IN POS. SEQ BUILDING ALGORITHM. LINE",I4," TO", 030670
      " I4," SHOULD HAVE G(L) OR B(L) GREATER THAN ZERO.") 030680
      1002 FORMAT(1X,"ERROR IN BUS(M) ROUTINE. NUMBER OF LINE ELEMENTS HAS EXCEEDED 030690
      1003 FORMAT(1X,"EXCEEDED ARRAY SIZE OF 75. REDUCE SIZE OF SUBSYSTEM.") 030700
25     C CLEAR ALL ARRAYS AND RESTORE NA ARRAY. IUBP IS BUILT TO BE SUB- 030710
      C SYSTEM BUS LIST. L IS LINE ENTRY NUMBER FROM G(L) B(L) LIST. 030720
      C LOOP STARTING AT 100 TAKES EACH LINE(L) AND BUILDS SYSTEM. 030730
      C IF BUS IS NOT IN SUBSYSTEM LIST, IT IS DROPPED AT 90 TO 100. 030740
      C THE L NUMBERS COME FROM JBP(I) IN LOOP 90 OF FAULT. 030750
30     DO 9 I=1,75
      ZOIA(I)=0
      IUBP(I)=0
      ZC(I)=0
      9 CONTINUE
35     DO 12 I=1,2775
      ZBUS(I)=0
      12 CONTINUE
      DO 11 I=1,N9BUS
      NA(I)=CONEC(I)
      11 CONTINUE
40     C=0
      DO 100 I=1,ISS
      ITEST=1
      L=JBP(I)
      JA=0
      JC=0
      JD=0
45     IF(G(L).EQ.0.AND.(L).EQ.0.) GO TO 88
      IF(LINA(L).EQ.0) GO TO 25
      DO 1 J=1,K
      IF(LINA(L).EQ.IUBP(J)) GO TO 2
50     1 CONTINUE
      ITEST=0
      2 JA=J
      DO 3 J=1,K
      IF(LINA(L).EQ.IUBP(J)) GO TO 4
55     3 CONTINUE
      IF(ITEST.EQ.0) GO TO 89
      030760
      030770
      030780
      030790
      030800
      030810
      030820
      030830
      030840
      030850
      030860
      030870
      030880
      030890
      030900
      030910
      030920
      030930
      030940
      030950
      030960
      030970
      030980
      030990
      031000
      031010
      031020
      031030
      031040

```

	C LINE IS A BRANCH FROM "P" TO "2"; THEREFORE SWAP P AND Q!	031090
	L1=LINA(L)	031090
60	L2=LINA(L)	031070
	JC=J	031080
	JD=JA	031090
	SO TO 5	031100
	4 L1=LINA(L)	031110
65	L2=LINA(L)	031120
	JC=JA	031130
	JD=J	031140
	IF(IYEST.EQ.1) GO TO 15	031150
	C LINE IS A BRANCH FROM Q TO P; ADD NEW BUS TO SYSTEM:	031160
70	5 JJ=K	031170
	K=K+1	031180
	C K IS CHECKED TO PREVENT OVER-RUN OF ARRAYS ZBUS,ZDIA,ZC.	031190
	C SIZE OF ZBUS IS DETERMINED BY K AND IADD ROUTINE.	031200
	IF(K.GT.75.) GO TO 302	031210
75	IUSP(K)=L1	031220
	DO 6 J=1, JJ	031230
	IF(IUSP(J).EQ.L2) GO TO 7	031240
	6 CONTINUE	031250
	SO TO 900	031260
80	7 GO TO N=1, JJ	031270
	N1=IADD(N, K)	031280
	IF(N.EQ.0) SO TO 5	031290
	N2=IADD(N, J)	031300
	ZBUS(N1)=ZBUS(N2)	031310
85	GO TO 19	031320
	ZBUS(N1)=ZDIA(J)	031330
	19 CONTINUE	031340
	10 CONTINUE	031350
	ZDIA(K)=ZDIA(J+1./CMPLX(G(L), B(L)))	031360
90	SO TO 90	031370
	C LINE IS A LOOP:	031380
	15 K1=K2=0	031390
	DO 16 J=1, K	031400
	IF(IUSP(J).EQ.L1) K1=J	031410
95	IF(IUSP(J).EQ.L2) K2=J	031420
	16 CONTINUE	031430
	DO 20 J=1, K	031440
	IF(J.EQ.K1) GO TO 17	031450
	IF(J.EQ.K2) GO TO 18	031460
100	N1=IADD(J, K1)	031470
	N2=IADD(J, K2)	031480
	ZC(J)=ZBUS(N1)-ZBUS(N2)	031490
	GO TO 20	031500
	17 N2=IADD(J, K2)	031510
105	ZC(J)=ZDIA(K1)-ZBUS(N2)	031520
	GO TO 20	031530
	18 N1=IADD(J, K1)	031540
	ZC(J)=ZBUS(N1)-ZDIA(K2)	031550
	20 CONTINUE	031560
110	N1=IADD(K1, K2)	031570
	IF(K2.LT.K1) N1=IADD(K2, K1)	031580
	IF(CMPLX(G(L), B(L)).EQ.0) SO TO 301	031590
	ZLL=ZDIA(K1)+ZDIA(K2)-2.*ZBUS(N1)+1./CMPLX(G(L), B(L))	031600
	SO TO 80	031610

115	C LINE 145 P=NOOE THE REF:	031620
	25 DO 26 J=1,K	031630
	IF(LINA(L).EQ.IUBP(J)) GO TO 30	031640
	26 CONTINUE	031650
	C LINE IS A BRANCH FROM REF. TO "Q": ADD NEW BUS TO SYSTEM:	031660
120	JJ=K	031670
	K=K+1	031680
	IUBP(K)=L2=LINA(L)	031690
	IF(CMPLX(G(L),B(L)).EQ.0) GO TO 301	031700
	ZDIA(K)=1./CMPLX(G(L),B(L))	031710
125	.1=0	031720
	GO TO 90	031730
	C LINE IS A LOOP CLOSING ELEMENT (REF. TO "Q"): ADD LOOP ELEMENT:	031740
	30 .1=0	031750
	L2=LINA(L)	031760
130	JD=J	031770
	DO 40 N=1,K	031780
	IF(J.EQ.N) GO TO 35	031790
	N1=IADD(N,J)	031800
	ZC(N)=ZBUS(N1)	031810
135	GO TO 40	031820
	35 ZC(N)=ZDIA(J)	031830
	40 CONTINUE	031840
	IF(CMPLX(G(L),B(L)).EQ.0) GO TO 301	031850
	ZLL=ZDIA(J)+1./CMPLX(G(L),B(L))	031860
140	C ELIMINATE LOOP AXIS BY KRON REDUCTION:	031870
	80 DO 85 II=1,K	031880
	DO 84 JJ=1,K	031890
	IF(II.EQ.JJ) GO TO 82	031900
	IF(II.GT.JJ) GO TO 83	031910
145	N1=IADD(II,JJ)	031920
	ZBUS(N1)=ZBUS(N1)-ZC(II)*ZC(JJ)/ZLL	031930
	GO TO 83	031940
	82 ZDIA(II)=ZDIA(II)-ZC(II)*ZC(II)/ZLL	031950
	83 CONTINUE	031960
150	84 CONTINUE	031970
	85 CONTINUE	031980
	GO TO 90	031990
	C REDUCE BUS CONNECTION COUNT FOR BUSES: ELIMINATE PTH OR QTH AXIS	032000
	C IF ALL CONNECTIONS TO THAT BUS ARE COMPLETE AND THE BUS IS NOT IN	032010
155	C THE AREA OF STUDY:	032020
	88 L1=LINA(L)	032030
	.2=LINA(L)	032040
	90 CONTINUE	032050
	IF(L1.EQ.0) GO TO 95	032060
160	NA(L1)=NA(L1)-1	032070
	IF(NA(L1).NE.0) GO TO 95	032080
	DO 92 N=1,NBS	032090
	IF(NB(N+NB).EQ.L1) GO TO 95	032100
	92 CONTINUE	032110
165	C ELIMINATE PTH AXIS:	032120
	CALL SWAPZ(JC,K)	032130
	K=K-1	032140
	95 NA(L2)=NA(L2)-1	032150
	IF(NA(L2).NE.0) GO TO 100	032160
170	DO 96 N=1,NBS	032170
	IF(NB(N+NB).EQ.L2) GO TO 100	032180
	96 CONTINUE	032190
	C ELIMINATE QTH AXIS AND ADJUST INDEX	032200
	IF(K.L7.JD) 50,60	032210
175	50 JD=JC	032220
	60 CALL SWAPZ(JD,K)	032230
	K=K-1	032240
	100 CONTINUE	032250
	RETURN	032260
180	899 WRITE(2,*) "BUS LIST FOLLOWS"	032270
	WRITE(2,*) (IUBP(J),J=1,K)	032280
	900 WRITE(2,1000) LINA(L),LINA(L)	032290
	GO TO 910	032300
185	901 WRITE(2,1001) LINA(L),LINA(L)	032310
	GO TO 910	032320
	902 WRITE(2,1002)	032330
	910 IERR=IERR+1	032340
	RETURN	032350
	END	032360

```

1      SUBROUTINE BUSO(M)                                032370
      INTEGER CONEC                                     032380
      COMPLEX ZZ, OFFDIAG, DIAG, ZLL, Z0IA, Z00IA, Z0JS, Z0BUS, YCOUP, ZC, 032390
      ZCOUP, ZM, EBUS                                    032400
5      COMMON /COMB/LINA(1450), LINA(1450), S(1450), B(1450), P(250), Q(250), 032410
      LPA(50), PHANG(50), LTRA(250), LTR3(250), TAP(250), TMN(250), V(250), 032420
      ZTMN(250), IUBPP(250), ANG(250), IYUS(250), OBP(250), UBP(3000), 032430
      BUSNAME(250), LPHA(50), LIST(250), IUP(250), QMIN(250), QMAX(250), 032440
      JBP(250), UBPP(3000), JBP(3000), JBP(3000), ICC(250), OLP(250), 032450
      SBI(1450), ZOR(1450), B0IA(250), CONEC(250), JLQ(250), IPHASE(250) 032460
10     COMMON/COMC/ NA(250), NR(250), JCO.(1000), OJ(1000), IOB(250) 032470
      COMMON /CONST/ NBUS, NL, ISS, IP, L1, L2, L3, LL4, NOTR, IZ, MOLTG 032480
      ITR1, ITR2, PTOL, QTOL, NLC                      032490
      COMMON /SAVE/ IERR                                032500
15     COMMON /ZERO/LA(25), LB(25), LP(25), LS(25), ZM(25), YCOUP(8,8), 032510
      ZCJP(25), IJK(25), KJI(25), IT7(25), ISAVE(8), Z0IA(75), Z00IA(75), 032520
      ZBUS(2775), Z0BUS(2775), EBUS(250), ZC(75)       032530
      COMMON /ZCONST/ NOMU, NBS, IR3H, I4JT, IDUMH     032540
20     1000 FORMAT(1X,"ERROR IN ZERO SEQ. BUILDING ALGORITHM. BUS",I4, 032550
      1"OR BUS",I4,"SHOUL HAVE BEEN IN SYSTEM LIST, BUT WASN'T!!") 032560
      1001 FORMAT(1X,"ERROR IN ZERO SEQ. BUILDING ALGORITHM. LINE FROM",I4, 032570
      1"TO",I4," MUST HAVE ZERO SEQ. IMPEDANCE NOT EQ. TO 0.") 032580
      1002 FORMAT(1X,"ERROR IN BUSO ROUTINE. NUMBER OF LINE ELEMENTS MASEXCEE 032590
      1JED 75 ARRAY SIZE. REDUCE SIZE OF SUBSYSTEM.") 032600
25     C SEE COMMENTS IN BUS(M) ROUTINE, SAME LOCATION AS HERE. DIFFERENCE 032610
      C IS THAT BUSO(M) HAS ROUTINES FOR MUTUAL IMPEDANCES. 032620
      DO 1 I=1,75                                       032630
      IUBP(I)=0                                         032640
      Z0IA(I)=0                                         032650
      ZC(I)=0                                           032660
30     1 CONTINUE                                       032670
      DO 11 II=1, NBUS                                   032680
      VA(II)=CONEC(II)                                  032690
35     11 CONTINUE                                       032700
      DO 3 I=1, 2775                                    032710
      Z0BUS(I)=0                                        032720
      9 CONTINUE                                       032730
      C=0                                               032740
      DO 100 I=1, ISS                                   032750
      ITEST=1                                           032760
      I=JBP(I)                                          032770
      JA=0                                               032780
      JC=0                                               032790
      JD=0                                               032800
45     IF(LINB(L).EQ.0) GO TO 50                        032810
      DO 2 J=1,K                                        032820
      IF(LINA(L).EQ.IUBP(J)) GO TO 3                   032830
      2 CONTINUE                                       032840
      ITEST=0                                           032850
50     3 JA=J                                           032860
      DO 4 J=1,K                                        032870
      IF(LINB(L).EQ.IUBP(J)) GO TO 5                 032880
      4 CONTINUE                                       032890
      IF(ITEST.EQ.0) GO TO 898                          032900
95     C LINE IS A BRANCH FROM "P" TO "J": S4AP P AND QI 032910
      L1=LINB(L)                                       032920
      L2=LINA(L)                                       032930

```

	JC=J	032940
	JD=JA	032950
60	DO TO 6	032960
	S L1=LINA(L)	032970
	L2=LINB(L)	032980
	JC=JA	032990
	JD=J	033000
65	IF(ITEST.EQ.1) GO TO 25	033010
	C LINE IS A BRANCH FROM "O" IN SYSTEM TO NEW BUS "P"; ADD NEW BUS;	033020
	C THEN CHK FOR MUTUALS:	033030
	6 JJ=K	033040
	K=K+1	033050
70	C K IS CHECKED TO PREVENT OVER-RUN OF ARRAYS ZBUS,ZODIA,ZC.	033060
	C SIZE OF ZBUS ARRAY IS DETERMINED BY K AND IADD ROUTINE.	033070
	IF(K.GT.75.) GO TO 902	033080
	IUBP(K)=L1	033090
75	ZZ=CMPLX(ZOR(L),ZOI(L))	033100
	IF(NOMU.EQ.0) GO TO 20	033110
	CALL MUTEST(L1,L2,ZZ)	033120
	IF(IERR.NE.0) RETURN	033130
	IF(IROW.EQ.0) GO TO 20	033140
80	C BRANCH HAS MUTUALS AND IS NOT THE FIRST LINE OF MUTUALLY COUPLED	033150
	C SET; ADD BRANCH N/MUTUALS:	033160
	DO 7 J=1,JJ	033170
	IF(IUBP(J.EQ.L2) GO TO 8	033180
	CONTINUE	033190
	7 GO TO 900	033200
85	8 IQ=J	033210
	DO 10 J=1,JJ	033220
	ZLL=OFFDIAG(JJ,J)	033230
	IF(IERR.NE.0) RETURN	033240
90	N1=IADD(J,IQ)	033250
	N2=IADD(J,K)	033260
	ZBUS(N2)=ZBUS(N1)+ZLL/YCOUP(1,1)	033270
	CONTINUE	033280
	10 ZLL=DIAG(JJ,K)	033290
	41=IADD(IQ,K)	033300
95	ZODIA(K)=ZBUS(N1)+((1.0,0.0)+ZLL)/YCOUP(1,1)	033310
	GO TO 90	033320
	C BRANCH DOES NOT HAVE MUTUALS, OR IS THE FIRST LINE OF A MUTUALLY	033330
	C COUPLED SET; ADD BRANCH NO/MUTUALS:	033340
100	20 DO 21 J=1,JJ	033350
	IF(IUBP(J.EQ.L2) GO TO 22	033360
	CONTINUE	033370
	21 GO TO 900	033380
	22 DO 24 N=1,JJ	033390
105	N1=IADD(N,K)	033400
	IF(N.EQ.J) GO TO 23	033410
	N2=IADD(N,J)	033420
	ZBUS(N1)=ZBUS(N2)	033430
	GO TO 24	033440
110	23 ZBUS(N1)=ZODIA(J)	033450
	CONTINUE	033460
	ZODIA(K)=ZODIA(J)+ZZ	033470
	GO TO 90	033480
	C LINE IS LOOP CLOSING ELEMENT; CHK FOR MUTUALS:	033490
	25 ZZ=CMPLX(ZOR(L),ZOI(L))	033500

115	IF(NOMU.EQ.0) GO TO 41	033510
	CALL MUTEST(L1,L2,ZZ)	033520
	IF(IERR.NE.0) RETURN	033530
	IF(IROW.EQ.0) GO TO 41	033540
	C LOOP 445 MUTUALS AND IS NOT THE FIRST LINE OF A MUTUALLY COUPLED	033550
120	C SET: ADD LOOP W/MUTJA.SI	033560
	K1=K2=0	033570
	DO 26 J=1,K	033580
	IF(IUSP(J).EQ.L1) K1=J	033590
	IF(IUSP(J).EQ.L2) K2=J	033600
125	CONTINUE	033610
	26 IF(K1.EQ.0.OR.K2.EQ.0) GO TO 900	033620
	DO 30 J=1,K	033630
	ZLL=OFFDIAG(K,J)	033640
	IF(IERR.NE.0) RETURN	033650
130	N1=IADD(J,K1)	033660
	N2=IADD(J,K2)	033670
	N3=IADD(J,K+1)	033680
	ZOBUS(N3)=ZOBUS(N1)-ZOBUS(N2)+ZLL/YCOUP(1,1)	033690
	30 CONTINUE	033700
135	ZL=ZDIAG(K,K)	033710
	N1=IADD(K1,K+1)	033720
	N2=IADD(K2,K+1)	033730
	ZDIA(K+1)=ZOBUS(N1)-ZOBUS(N2)+((1.0,0.0)+ZLL)/YCOUP(1,1)	033740
	20 TO 80	033750
140	C LOOP DOES NOT HAVE MUTUALS, OR IS THE FIRST LINE OF MUTUALLY	033760
	C COUPLED SET: ADD LOOP NO/MUTUAL.SI	033770
	41 K1=K2=0	033780
	DO 42 J=1,K	033790
	IF(IUSP(J).EQ.L1) K1=J	033800
145	IF(IUSP(J).EQ.L2) K2=J	033810
	CONTINUE	033820
	42 IF(K1.EQ.0.OR.K2.EQ.0) GO TO 900	033830
	DO 45 J=1,K	033840
	IF(J.EQ.K1) GO TO 43	033850
150	IF(J.EQ.K2) GO TO 44	033860
	N1=IADD(J,K1)	033870
	N2=IADD(J,K2)	033880
	ZC(J)=ZOBUS(N1)-ZOBUS(N2)	033890
	GO TO 45	033900
155	43 N2=IADD(J,K2)	033910
	ZC(J)=ZDIA(K1)-ZOBUS(N2)	033920
	GO TO 45	033930
	44 N1=IADD(J,K1)	033940
	ZC(J)=ZOBUS(N1)-ZDIA(K2)	033950
160	45 CONTINUE	033960
	N1=IADD(K1,K2)	033970
	ZLL=ZDIA(K1)+ZDIA(K2)-2.*ZOBUS(N1)+ZZ	033980
	20 TO 80	033990
165	C LINE HAS ONE NODE THE REPT CHK IF OTHER NODE IS IN SYSTEMI	034000
	50 ZZ=CMPLX(ZR(L),ZDI(L))	034010
	DO 51 J=1,K	034020
	IF(LINA(L).EQ.IUSP(J)) GO TO 60	034030
	51 CONTINUE	034040
170	C LINE IS BRANCH FROM REP. TO "2": ADD NEW BJS TO SYSTEMI	034050
	-1=0	034060
	JJ=K	034070

	K=K+1	034000
	IUBP(K)=L2=LINA(L)	034090
	Z00IA(K)=ZZ	034100
175	Z00IA(K)=ZZ	034110
	Z0 T> 99	034120
	C LINE IS LOOP CLOSING ELEMENT (REF. TO "Q"); ADD LOOP CLOSING	034130
	C ELEMENT:	034140
	00 -1=0	034150
180	L2=LINA(L)	034160
	DO 65 N=1,K	034170
	IF(IJ.EQ.M) GO TO 51	034180
	N1=IADD(N,J)	034190
	ZC(N)=Z0BUS(N1)	034200
185	GO TO 65	034210
	ZC(N)=Z00IA(J)	034220
	65 CONTINUE	034230
	ZL=Z00IA(J)+ZZ	034240
	C ELIMINATE LOOP AXIS BY KRON REDUCTION:	034250
190	00 IF(ZLL.EQ.0) GO TO 999	034260
	DO 65 II=1,K	034270
	DO 64 JJ=1,K	034280
	IF(II.EQ.JJ) GO TO 83	034290
	IF(II.GT.JJ) GO TO 84	034300
195	N1=IADD(II,JJ)	034310
	Z0BUS(N1)=Z0BUS(N1)-ZC(II)*ZC(JJ)/ZLL	034320
	GO TO 84	034330
	83 Z00IA(II)=Z00IA(II)-ZC(II)*ZC(II)/ZLL	034340
	84 CONTINUE	034350
	85 CONTINUE	034360
200	C REDUCE BUS CONNECTION COUNT FOR BUSES; ELIMINATE PTH OR QTH AXIS	034370
	C IF ALL CONNECTIONS TO THAT BUS ARE COMPLETE AND THE BUS IS NOT IN	034380
	C THE AREA OF STUDY:	034390
	90 IF(L1.EQ.0) GO TO 95	034400
205	NA(L1)=NA(L1)-1	034410
	IF(NA(L1).NE.0) GO TO 95	034420
	DO 92 N=1,N95	034430
	IF(ND(N+ND).EQ.L1) GO TO 95	034440
	CONTINUE	034450
210	CALL SWAPZ0(JC,K)	034460
	K=K-1	034470
	95 NA(L2)=NA(L2)-1	034480
	IF(NA(L2).NE.0) GO TO 100	034490
	DO 96 N=1,N95	034500
215	IF(ND(N+ND).EQ.L2) GO TO 100	034510
	CONTINUE	034520
	96 CONTINUE	034530
	C ELIMINATE QTH AXIS:	034540
	IF(K.LT.JD) 97,99	034550
	97 JD=JC	034560
220	98 CALL SWAPZ0(JD,K)	034570
	K=K-1	034580
	100 CONTINUE	034590
	RETURN	034600
225	096 WRITE(2,*) "BUS LIST FOLLOWS"	034610
	WRITE(2,*) (IUBP(J), J=1,K)	034620
	Z0 T> 999	034630
	099 WRITE(2,1001) L1,L2	034640
	Z0 T> 910	
230	900 WRITE(2,1000) LINA(L),LINA(L)	034650
	Z0 T> 910	034660
	902 WRITE(2,1002)	034670
	910 IERR=IERR+1	034680
	RETURN	034690
	END	034700

1	SUBROUTINE MUTEST(I1,I2,ZZ)	034710
	INTEGER S1,S2	034720
	COMPLEX YCOUP,ZM,ZZ,ZCOUP,ZBUS,ZBUS,ZDIA,ZDIA,EBUS,ZC	034730
	COMMON /ZERO/LA(25),L9(25),LR(25),L3(25),Z4(25),YCOUP(8,8),	034740
5	ZCOUP(25),IJK(25),KJI(25),ITZ(25),ISAVE(8),ZDIA(75),ZDIA(75),	034750
	ZBUS(2775),ZBUS(2775),EBUS(250),ZC(75)	034760
	COMMON /ZCONST/ NOFU,NBS,IR3H,I4JT,IOUHM	034770
	COMMON /SAVE/ IERR	034780
	IR3H=0	034790
10	DO 10 I=1,NOFU	034800
	IF(LA(I).NE.I1) GO TO 1	034810
	IF(L9(I).EQ.I2) GO TO 11	034820
	GO TO 2	034830
	1 IF(L9(I).NE.I1) GO TO 2	034840
15	IF(LA(I).EQ.I2) GO TO 11	034850
	2 IF(LR(I).NE.I1) GO TO 3	034860
	IF(L3(I).EQ.I2) GO TO 12	034870
	GO TO 10	034880
	3 IF(L3(I).NE.I1) GO TO 10	034890
20	IF(L3(I).EQ.I2) GO TO 12	034900
	10 CONTINUE	034910
	C LINE DOES NOT HAVE MUTUALS; RETURN TO BUS0 TO CONTINUE PROCESSING	034920
	C LINE0	034930
	RETURN	034940
25	C LINE DOES HAVE MUTUALS; THE OTHER LINE IS DEFINED BY LR(I) AND L3(I);	034950
	11 L1=LR(I)	034960
	L2=L3(I)	034970
	GO TO 15	034980
	C LINE DOES HAVE MUTUALS; THE OTHER LINE IS DEFINED BY LA(I) AND L9(I);	034990
30	12 L1=LA(I)	035000
	L2=L9(I)	035010
	C CHK TO SEE IF OTHER LINE HAS ALREADY BEEN ADDED TO SYSTEM;	035020
	15 IF(IMUT.EQ.0) GO TO 20	035030
	DO 17 I=1,IMUT	035040
35	IF(IJK(I).NE.L1) GO TO 16	035050
	IF(KJI(I).EQ.L2) GO TO 21	035060
	GO TO 19	035070
	16 IF(KJI(I).NE.L1) GO TO 19	035080
	IF(IJK(I).EQ.L2) GO TO 21	035090
40	19 CONTINUE	035100
	C THIS IS THE FIRST LINE OF THE MUTUALLY COUPLED SET; ADD LINE TO	035110
	C MUTJAL. BUILDING TAB.E; RETURN;	035120
	20 INJT=IMUT+1	035130
	IOUHM=IOUHM+1	035140
45	ITZ(IMUT)=IOUHM	035150
	IJK(IMUT)=I1	035160
	KJI(IMUT)=I2	035170
	ZCJJP(IMUT)=ZZ	035180
	RETURN	035190
50	C NEW LINE IS COUPLED TO LINE ALREADY IN SYSTEM; ADD NEW LINE TO	035200
	C MUTJAL. BUILDING TAB.E; FORM MUTJAL COUPLING MATRIX FOR THIS	035210
	C COUPLED SET;	035220
	21 INJT=IMUT+1	035230
	IST=IMUT+1	035240
55	ITZ(IMUT)=ITZ(I)	035250
	IJK(IMUT)=I1	035260
	KJI(IMUT)=I2	035270

	ZCOUP(INUT)=Z2	035280
	C THE FOLLOWING SECTION CONSTRUCTS THE ACTUAL COUPLING MATRIX TO BE	035290
60	C USED BY THE BUS BUILDING ALGORITHM FOR LINES WITH MUTUALS:	035300
	II=K+1	035310
	DO 80 I=1,IST	035320
	IF(I.EQ.IST) GO TO 80	035330
	JJ=IST-I	035340
65	IF(ITZ(JJ).NE.ITZ(INUT)) GO TO 90	035350
	IRJN=IRON+1	035360
	ISAVE(IRJN)=JJ	035370
	YCOUP(IRJN,IRON)=ZCOUP(JJ)	035380
	IF(JJ.EQ.INUT) GO TO 80	035390
70	S1=II	035400
	S2=I?	035410
	JK=1	035420
	22 DO 50 J=1,NMJJ	035430
	IF(LA(J).NE.S1) GO TO 25	035440
75	IF(LB(J).NE.S2) GO TO 30	035450
	IF(LR(J).NE.IJK(JJ)) GO TO 40	035460
	IF(LS(J).EQ.KJI(JJ)) GO TO 55	035470
	GO TO 50	035480
	25 IF(LB(J).NE.S1) GO TO 30	035490
80	IF(LA(J).NE.S2) GO TO 30	035500
	IF(LR(J).NE.IJK(JJ)) GO TO 40	035510
	IF(LS(J).EQ.KJI(JJ)) GO TO 55	035520
	GO TO 50	035530
	30 IF(LR(J).NE.S1) GO TO 35	035540
85	IF(LS(J).NE.S2) GO TO 30	035550
	IF(LA(J).NE.IJK(JJ)) GO TO 45	035560
	IF(LB(J).EQ.KJI(JJ)) GO TO 55	035570
	GO TO 50	035580
	35 IF(LS(J).NE.S1) GO TO 30	035590
90	IF(LR(J).NE.S2) GO TO 30	035600
	IF(LA(J).NE.IJK(JJ)) GO TO 45	035610
	IF(LB(J).EQ.KJI(JJ)) GO TO 55	035620
	GO TO 50	035630
	40 IF(LS(J).NE.IJK(JJ)) GO TO 30	035640
95	IF(LR(J).EQ.KJI(JJ)) GO TO 55	035650
	GO TO 50	035660
	45 IF(LB(J).NE.IJK(JJ)) GO TO 30	035670
	IF(LA(J).EQ.KJI(JJ)) GO TO 55	035680
	50 CONTINUE	035690
100	YCOUP(JK,IRON)=CHPLX(0.,0.)	035700
	YCOUP(IRON,JK)=CHPLX(0.,0.)	035710
	GO TO 56	035720
	55 YCOUP(JK,IRON)=YCOUP(IRON,JK)=Z4(IJ)	035730
	56 JK=JK+1	035740
105	IF(JK.EQ.IRON) GO TO 80	035750
	S1=IJK(ISAVE(JK))	035760
	S2=KJI(ISAVE(JK))	035770
	GO TO 22	035780
	80 CONTINUE	035790
110	C MUTUAL IMPEDANCE COUPLING MATRIX IS COMPLETE! CALL SUBROUTINE	035800
	C TO INVERT IMPEDANCE COUPLING MATRIX TO FORM ADMITTANCE COUPLING	035810
	C MATRIX:	035820
	NN=2*IRON	035830
	CALL CPLXINV(YCOUP,IRON,NN)	035840
115	RETURN	035850
	END	035860

```

1      SUBROUTINE CPLXIV(A,N,MN)
      COMPLEX A(N,MN),AL
      EPSIL=1.E-8
      DO 3 I=1,N
      DO 1 J=1,MN
      IF(I.EQ.J) GO TO 2
      A(I,J)=CMPLX(0.,0.)
      GO TO 3
      2 A(I,J)=CMPLX(1.,0.)
      3 CONTINUE
      DO 50 IP=1,N
      IM=IP
      IST=IP+1
      IF(IP.EQ.MN) GO TO 11
      DO 10 I=IST,MN
      Z1=CABS(A(I,IP))
      Z2=CABS(A(I,IP))
      IF(Z1.GE.Z2) GO TO 10
      IM=I
      10 CONTINUE
      11 Z1=CABS(A(IM,IP))
      IF(Z1.GE.EPSIL) GO TO 13
      WRITE(1,1000) A(IM,IP)
      1000 FORMAT(1X,"PIVOT ELEMENT=",Z20.10/)
      Z1=CABS(A(IM,IP))
      IF(Z1.EQ.0.) GO TO 70
      13 IF(IM.EQ.IP) GO TO 20
      DO 15 J=IP,MN
      AL=A(IP,J)
      A(IP,J)=A(IM,J)
      A(IM,J)=AL
      15 AL=A(IP,IP)
      20 AL=A(IP,IP)
      A(IP,IP)=CMPLX(1.,0.)
      DO 25 J=IST,MN
      A(IP,J)=A(IP,J)/AL
      DO 40 I=1,I
      IF(I.EQ.IP) GO TO 40
      AL=A(I,IP)
      DO 35 J=IP,MN
      A(I,J)=A(I,J)-AL*A(IP,J)
      35 CONTINUE
      40 CONTINUE
      DO 50 I=1,N
      DO 50 J=1,MN
      50 A(I,J)=A(I,J)+0
      70 RETURN
      END

```

```

1      SUBROUTINE SHAPZ(IR,K)
      INTEGER CONEC
      COMPLEX ZBUS,ZBUS,ZDIA,ZDIA,ZN,VC0UP,ZC0UP,EBJS,ZC
      COMMON /COMB/LINA(1450),LINA(1450),S(1450),B(1450),P(250),Q(250),
      5 I,P4B(50),PHANG(50),LTRA(250),LTRA(250),TAP(250),TNN(250),V(250),
      ZTK(250),IUBPP(250),ANG(250),IUS(250),DB(250),UBP(3000),
      10 BUSNAME(250),LPHA(50),LIST(250),IUSP(250),QMIN(250),QMAX(250),
      630PP(250),UBPP(3000),JAP(3000),JBP(3000),ICC(250),DLP(250),
      920I(1450),ZOR(1450),BDIA(250),CONEC(250),JLQ(250),IPHASE(250)
      20 COMMON /ZERO/LA(25),LQ(25),LR(25),LS(25),L4(25),VC0UP(0,0),
      ZC0UP(25),IJK(25),KJI(25),ITZ(25),ISAVE(0),ZDIA(75),ZDIA(75),
      ZBUS(2775),ZBUS(2775),EBJS(250),ZC(75)
      COMMON /ZCONST/ N04J,N05,IRJN,I4JT,IDUMH
      IF(IR.EQ.K) GO TO 25
      DO 20 I=1,K
      IF(I.EQ.IR) GO TO 15
      IF(I.EQ.K) GO TO 20
      15 Z1=IADD(I,IR)
      20 Z2=IADD(I,K)
      ZBUS(N1)=ZBUS(N2)
      ZBUS(N2)=CMPLX(0.,0.)
      GO TO 20
      15 ZDIA(IR)=ZDIA(K)
      ZDIA(K)=CMPLX(0.,0.)
      20 CONTINUE
      IUSP(IR)=IUSP(K)
      GETJRN
      25 DO 25 J=1,K
      IF(J.EQ.K) GO TO 30
      30 Z1=IADD(J,K)
      ZBUS(N1)=CMPLX(0.,0.)
      25 CONTINUE
      30 ZDIA(K)=CMPLX(0.,0.)
      GETJRN
      END

```

```

1      SUBROUTINE SNAPZ0(IR,K)                                036690
      INTEGER CONEC                                          036700
      COMPLEX ZBUS,ZBUS,Z0IA,Z0IA,ZH,YCOUP,ZCOUP,ZBUS,ZC    036710
      COMMON /COMB/LINA(1450),LINA(1450),G(1450),G(1450),P(250),Q(250),
5      1.P43(50),PHANG(50),LTRA(250),LTRB(250),TAP(250),TMM(250),V(250),    036720
      2TMM(250),IUBPP(250),ANG(250),IYUS(250),OBP(250),UBP(3000),        036730
      3BUSNAME(250),LPHA(50),LIST(250),IUP(250),IMIN(250),IMAX(250),    036740
      4JBP(250),UBPP(3000),JPP(3000),JPP(3000),ICC(250),OLP(250),      036750
      5Z0I(1450),Z0R(1450),0DIA(250),CONEC(250),JLQ(250),IPHASE(250)    036760
      COMMON /ZERO/LA(25),LB(25),LR(25),LS(25),Z4(25),YCOUP(0,0),      036770
      1ZCOUP(25),IJK(25),KJI(25),ITZ(25),ISAVE(0),Z0IA(75),Z00IA(75),    036780
      2ZBUS(2775),ZBUS(2775),EBUS(250),ZC(75)                    036790
      COMMON /ZCONST/ NOMU,NBS,IRCN,I4JT,IOUMH                    036800
      IF(IR.EQ.K) GO TO 25                                         036810
      DO 20 I=1,K                                                  036820
      IF(I.EQ.IR) GO TO 15                                         036830
      IF(I.EQ.K) GO TO 20                                         036840
      V1=IADD(I,IR)                                               036850
      V2=IADD(I,K)                                               036860
      ZBUS(N1)=ZBUS(V2)                                           036870
      ZBUS(N2)=CHPLX(0.,0.)                                       036880
      DO 15 J=1,K                                                  036890
      Z0IA(IR)=Z0IA(K)                                           036900
      Z0IA(K)=CHPLX(0.,0.)                                       036910
      20 CONTINUE                                               036920
      IUBP(IR)=IUBP(K)                                           036930
      RETJRN                                                    036940
      25 DO 25 J=1,K                                              036950
      IF(J.EQ.K) GO TO 30                                         036960
      V1=IADD(J,K)                                               036970
      ZBUS(N1)=CHPLX(0.,0.)                                       036980
      30 CONTINUE                                               036990
      Z0IA(K)=CHPLX(0.,0.)                                       037000
      RETURN                                                    037010
      END                                                        037020
      END                                                        037030

1      COMPLEX FUNCTION OFFDIAG(JJ,JC)                        037040
      INTEGER CONEC,S1,S2                                        037050
      COMPLEX ZBUS,ZBUS,Z0IA,Z0IA,ZH,YCOUP,ZCOUP,ZBUS,ZC    037060
      COMMON /COMB/LINA(1450),LINA(1450),G(1450),G(1450),P(250),Q(250),    037070
5      1.PH4(50),PHANG(50),LTRA(250),LTRB(250),TAP(250),TMM(250),V(250),    037080
      2TMM(250),IUBPP(250),ANG(250),IYUS(250),OBP(250),UBP(3000),        037090
      3BUSNAME(250),LPHA(50),LIST(250),IUP(250),IMIN(250),IMAX(250),    037100
      4JBP(250),UBPP(3000),JPP(3000),JPP(3000),ICC(250),OLP(250),      037110
      5Z0I(1450),Z0R(1450),0DIA(250),CONEC(250),JLQ(250),IPHASE(250)    037120
      COMMON /ZERO/LA(25),LB(25),LR(25),LS(25),Z4(25),YCOUP(0,0),      037130
      1ZCOUP(25),IJK(25),KJI(25),ITZ(25),ISAVE(0),Z0IA(75),Z00IA(75),    037140
      2ZBUS(2775),ZBUS(2775),EBUS(250),ZC(75)                    037150
      COMMON /SAVE/ IERR                                          037160
      COMMON /ZCONST/ NOMU,NBS,IRCN,I4JT,IOUMH                    037170
      OFFDIAG=CHPLX(0.,0.)                                       037180
      DO 10 I=2,IROW                                             037190
      S1=S2=0                                                    037200
      DO 1 J=1,JJ                                                037210
      IF(IUBP(J).EQ.IJK(ISAVE(I))) S1=J                          037220
      IF(IUBP(J).EQ.KJI(ISAVE(I))) S2=J                          037230
      1 CONTINUE                                               037240
      IF(S1.EQ.0.OR.S2.EQ.0) GO TO 20                             037250
      IF(S1.EQ.JC) GO TO 5                                        037260
      IF(S2.EQ.JC) GO TO 6                                        037270
      V1=IADD(JC,S1)                                             037280
      V2=IADD(JC,S2)                                             037290
      OFFDIAG=OFFDIAG+(ZBUS(N1)-ZBUS(N2))*YCOUP(1,I)          037300
      DO 10 I=1,10                                              037310
      V2=IADD(JC,S2)                                             037320
      OFFDIAG=OFFDIAG+(Z0IA(S1)-ZBUS(N2))*YCOUP(1,I)          037330
      DO 10 I=1,10                                              037340
      V1=IADD(JC,S1)                                             037350
      OFFDIAG=OFFDIAG+(ZBUS(N1)-Z0IA(S2))*YCOUP(1,I)          037360
      10 CONTINUE                                               037370
      RETURN                                                    037380
      20 IERR=1,100 I,IJK(ISAVE(I)),KJI(ISAVE(I))              037390
      IERR=IERR+1                                               037400
      100 FORMATT('SUBROUTINE SUBS: ERROR BETWEEN MUTUAL TABLE AND SYSTEM',
      1BUS LIST (IUBP, /TS, 'WORKING ON ROD', IS, ' YCOUP MATRIX. /
      2TS, 'R=', I4, ' AND S=', I4, /)
      RETURN                                                    037410
      END                                                        037420
      END                                                        037430
      END                                                        037440
      END                                                        037450

```

```

1      COMPLEX FUNCTION DIAG(JJ,JS)                                037460
      INTEGER CONEC,S1,S2                                        037470
      COMPLEX ZBUS,ZBUSJ,ZDIA,ZDIA,ZM,VCJUP,ZCJUP              037480
      COMMON /COMB/LIN4(1450),LI43(1450),J(1450),B(1450),P(250),Q(250), 037490
      1LP49(50),PHANG(50),TRA(250),LTR3(250),TAP(250),TMN(250),V(250), 037500
      2Y4(250),IUBPP(250),ANG(250),T4JS(250),O9(250),UBP(3000), 037510
      3BUSNAME(250),LPH4(50),LIST(250),IUBP(250),QMIN(250),QMAX(250), 037520
      4)9PP(250),URPP(3000),J9P(3000),J3PP(3000),ICC(250),DLP(250), 037530
      5Z0I(1450),Z0R(1450),RDIA(250),CONEC(250),DLQ(250),IPHASE(250) 037540
      COMMON /ZERO/LA(25),LB(25),R(25),LS(25),ZH(25),VCJUP(0,0), 037550
      1ZCJUP(25),IJK(25),KJI(25),ITZ(25),ISAVE(0),ZDIA(75),ZDIA(75), 037560
      2ZBUS(2775),ZBUSJ(2775),E9US(250),ZC(75)                037570
      COMMON /ZCONST/ NOMU,N9S,IR34,INUT,IOUM4                037580
      DIAG=CMPLX(0.,0.)                                         037590
      DO 10 I=2,IRON                                            037600
          DO 5 J=1,JJ                                           037610
              IF(IUBP(J).EQ.IJK(ISAVE(I))) S1=J                037620
              IF(IUBP(J).EQ.KJI(ISAVE(I))) S2=J                037630
          5      CONTINUE                                         037640
      V1=IAOD(S1,JJ+1)                                          037650
      V2=IAOD(S2,JJ+1)                                          037660
      DIAG=DIAG+(ZBUS(M1)-ZBUS(V2))*VCJUP(I,I)                 037670
10     CONTINUE                                                 037680
      RETURN                                                    037690
      END                                                        037700

1      COMPLEX FUNCTION CPLXV(EA,E9)                                037710
      ETEMP=EA                                                  037720
      ITI4=0.01745329*ETEMP                                     037730
      E1=EA*COS(ETEMP)                                         037740
      E2=EA*SIN(ETEMP)                                         037750
      CPLXV=CMPLX(E1,E2)                                       037760
      RETURN                                                    037770
      END                                                        037780

1      FUNCTION IAOD(I,J)                                        037790
      III=I                                                       037800
      JJJ=J                                                       037810
      IF(III.LT.JJJ) GO TO 10                                    037820
      TE4P=JJJ                                                  037830
      JJJ=III                                                    037840
      III=TE4P                                                  037850
10     IAOD=75*(III-1)-((III*III-III)/2)+JJJ-III              037860
      RETURN                                                    037870
      END                                                        037880

```

Vita

Jesse Allen Underwood, Jr. was born in Renovo, Pennsylvania on June 29, 1941. He graduated from Northwestern High School in Albion, Pennsylvania in 1959 and then entered Edinboro State College, Edinboro, Pennsylvania. Graduating in 1963 with a Bachelor of Science degree in Education, he entered the United States Air Force through the Officer Training School at Lackland AFB, Texas, receiving his commission in 1964. After attending the Electronics Systems Officer Course at Keesler, AFB, Mississippi in 1965, he served at radar stations in Benton, Pennsylvania and Antigo, Wisconsin as the Communications-Electronics officer. He entered the Air Force Institute of Technology Post Graduate Program in June 1974.

Permanent Address: RD #1, Ridge Road
West Springfield, PA 16443

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER GE/EE/76-43	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A USER-ORIENTED POWER DISTRIBUTION SYSTEM ANALYSIS PROGRAM		5. TYPE OF REPORT & PERIOD COVERED MS Thesis
7. AUTHOR(s) Jesse A. Underwood, Jr. Major, USAF		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS Air Force Institute of Technology (AFIT-EM) Wright-Patterson AFB, Ohio 45433		8. CONTRACT OR GRANT NUMBER(s)
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Institute of Technology (AFIT/EN) Wright-Patterson AFB, Ohio 45433		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE December 1976
		13. NUMBER OF PAGES 257
		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for Public Release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES Approved for public release; IAW AFR 190-17 Jerral F. Guess, Captain, USAF Director of Information		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Electric Power Transmission Electrical Wires Transmission Lines Electric Power Short Circuit Load Flow Analysis Electrical Impedance		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This paper is a revision of a digital computer program written to perform a load flow and/or short circuit analysis of a power distribution system. The program has been named Power Distribution System Analysis Program (PDSAP). The program capacity is 250 buses and 500 line elements, with 250 of the line elements being transformers. Input routines accept data as impedances (ohms or per-unit), or as descriptive information such as wire size, length, or transformer ratings. For descriptive data, the program uses pre-calculated approximations to derive the impedance values due to neutral conductors in		

