General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.

- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.

- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.

- This document is paginated as submitted by the original source.

- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Report 6

MTRAC—A COMPUTER PROGRAM FOR ANALYSIS OF CIRCUITS INCLUDING MAGNETIC CORES

Volume II

Input Data and Program Listing

By: D. NITZAN and J. R. HERNDON

Prepared for:

JET PROPULSION LABORATORY
4800 OAK GROVE DRIVE
PASADENA, CALIFORNIA 91103

CONTRACT 951840 UNDER NAS7-100

(Extension of work under Contracts 950095 under NASw-6 and 950943 and 951383 under NAS7-100)

This work was performed for the Jet Propulsion Laboratory, California Institute of Technology, sponsored by the National Aeronautics and Space Administration under Contract NAS7-100.

STANFORD RESEARCH INSTITUTE
Menlo Park, California 94025 • U.S.A.

STANFORD RESEARCH INSTITUTE
Menlo Park, California 94025 · U.S.A.

Report 6 *June 1969*

MTRAC—A COMPUTER PROGRAM FOR ANALYSIS OF CIRCUITS INCLUDING MAGNETIC CORES

Volume II

Input Data and Program Listing

By: D. NITZAN and J. R. HERNDON

Prepared for:

JET PROPULSION LABORATORY
4800 OAK GROVE DRIVE
PASADENA, CALIFORNIA 91103

CONTRACT 951840 UNDER NAS7-100

(Extension of work under Contracts 950095 under NASw-6 and 950943 and 951383 under NAS7-100)

SRI Project 6408

Approved:

F. J. KAMPHOEFNER
Director
Engineering Sciences Laboratory

TORBEN MEISLING
Executive Director
Information Science and Engineering

CONTENTS

I MTRAC INPUT DATA

Instructions for the order and contents of MTRAC input-data cards are given as follows. The standard input-data cards are listed in Sec. I-A. In Sec. I-B are listed the cards for parameter deviations in a worst-case analysis.

A. MTRAC INPUT-DATA CARDS

Card 00000

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	A6,6X	CKT	Name (up to 6 characters) of circuit or circuit mode.	CKT5IA
13-24	I12	LIMITT	Time limit (in seconds): 20 seconds before LIMITT is reached, results are plotted (if NPLTVT ≥ 1), continued-run cards are punched, and the run terminates.	120
25-36	I12	KONT	1 - continued run; 0 - otherwise.	1
37-48	F12.0	DMAX	Maximum deviation (in percent) of parameter values. If DMAX > 0, then deviation signs must follow cards of variable sources and each circuit-element type.	5
49-60				
61-72				
73-80				00000

Card 00001: MONITOR WRITE specifications for debugging purposes.

1 - WRITE statements to be executed in the corresponding subprogram(s) if NDTMIN < NDELT ≤ NDTMAX;
0 (or blank) - otherwise.
Commas may be replaced by blanks.

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	6(A1,1X)	EMVALU(I)I=1,6	(1)Main program and RWRUNC, (2)AUX(K), (3)ELEM, (4)SEQSOL, (5)IVS, (6)RES,	1,0,1,0,1,1,
13-24	6(A1,1X)	EMVALU(I)I=7,12	(7)CAP, (8)IND, (9)ZEN, (10)DIOD, (11)TRAN, (12)CORE.	1 1 1 1
25-36	I12	NDTMIN	Index number of Δt before beginning the MONITOR printout.	0
37-48	I12	NDTMAX	Index number of Δt before ending the MONITOR printout.	10
49-60				
61-72				
73-80				0001

Card 00002

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NSVVAR	1 - save the final values of the previous mode for initial values; 0 - otherwise. If NG = -1, NSVVAR = 1 is forbidden.	1
13-24	I12	NSVPHI	1 - save the final magnetic-core Ø values of the previous mode for initial values; 0 - otherwise.	1
25-36	I12	NSVIL	1 - save the inductor-current values of the previous mode for initial values; 0 - otherwise.	1
37-48				
49-60				
61-72				
73-80				00002

Card 00003

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	ITRTST	Number of iterations below which Δt is multiplied by 2 (if not exceeding the value specified in Cards 01200-01230) and above which Δt is divided by 2. If blank, then ITRTST = 10 + KNL, where KNL = number of nonlinear elements (diodes, transistor junctions, and core windings).	8
13-24	I12	MAXMIN	Initial (lowest) maximum number of iterations above which the variable values are reset and Δt is divided by 4. If blank, then MAXMIN = 2(10+KNL).	20
25-36	I12	KFMAX	Maximum number of trials before ITRMAX is changed (doubled for KFMAX successive convergence failures; halved for KFMAX successive convergence successes with less than ITRTST iterations). If blank, then KFMAX = 3.	3
37-48	E12.8	RELER	Relative voltage error in the solution of initial conditions. If blank, then RELER = 10^{-6}.	1.0 -04
49-60	E12.8	ABSER	Absolute voltage error in the solution of initial conditions. If blank, then ABSER = 10^{-6} volt.	1.0 -05
61-72	I12	N3B	Solution of initial conditions: -1 - no transient solution; 0 - if no convergence, terminate mode run; 1 - if no convergence, print errors and proceed with the transient solution.	1
73-80				00003

Card 01000

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NV	Number of unknown nodal voltages.	15
13-24	I12	NAUX	Number of auxiliary unknowns. (max. NV + NAUX = 70).	23
25-36	I12	NG	Initial conditions: - 1 - calculate; 0 - zero; 1 - read in.	1
37-48	I12	ND	Number of variable sources and grounded dc voltage sources (max.18).	2
49-60	I12	NB	Ratio of computed points to plotted points (NB ≥ 1).	1
61-72	I12	NBB	Ratio of plotted points to printed points (NBB ≥ 1).	1
73-80				01000

Card 01010

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NPLTVT	Number of variables to be plotted (max.10).	5
13-24	I12	N1B	1 - plot phase planes ($\dot{V}$ vs. V) of the variables that are plotted vs. time; 0 - otherwise.	1
25-36	I12	NPLTVV	Number of variable-vs.-variable plots (max. 10).	
37-48				
49-60				
61-72				
73-80				01010

Card 01011

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	KPLOT	1 - plot at the end of mode; 0 - otherwise. (To plot variables in succeeding modes, let KPLOT = 1 for the last mode only.)	1
13-24	I12	KPRPLT	1 - print-plot at the end of mode; 0 - otherwise. (To plot variables in succeeding modes, let KPRPLT = 1 for the last mode only.) If KPRPLT = 0 and NPLTVT > 0, then KPRPLT is set to 1 if the number of plot points exceeds IPX or if the run is about to be terminated due to either a run-time limit (t > LIMITT-20 seconds, Card 00000) or a failure to converge.	1
25-36	F12.0	SCALE	Plot scales: 1 - specified by the user; 0 - determined by the program on the basis of extreme values. (Scales for print-plots are determined by the program only.)	1
37-48	I12	IPX	Number of plotted points per frame (max. 350).	350
49-60	11X,A1	PRSYMB	A symbol to be used in print-plots (if blank, then *).	*
61-72	I12	KPFIT	Scales and frame boundaries of V-vs.-t print-plots (and plots if SCALE=0) if the number of plot points exceeds IPX: 0 - independent (among frames); 1 - X scale and boundaries independent; Y scale and boundaries fixed by first frame; 2 - X boundaries independent; X scale, Y scale, and Y boundaries fixed by first frame; 3 - fixed by first frame. Scales and boundaries for V-vs.-V and V-vs.-V plots are independent if KPFIT = 0, but fixed by first frame if KPFIT = 1, 2, or 3.	
73-80				01011

Cards 01020 and 01021 (0 to 2 cards): NPLTVT (max.10) variables to be plotted vs. time. Omit cards with no data (i.e., omit card 01021 if NPLTVT ≤ 6; omit both cards if NPLTVT ≤ 0, Card 01010).

Variables to be plotted vs. each other must be included in these cards. If any of these variables should not be plotted vs. time, add a minus sign to the left of its index number.

Card 01020

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	A6, I6	AAAA(1),NPP(1)	Name and index number of the 1st variable.	ALPHA 18
13-24	A6, I6	AAAA(2),NPP(2)	Name and index number of the 2nd variable.	AMF 1 -25
25-36	A6, I6	AAAA(3),NPP(3)	Name and index number of the 3rd variable.	PHI 1 26
37-48	A6, I6	AAAA(4),NPP(4)	Name and index number of the 4th variable.	PHIDOT1 27
49-60	A6, I6	AAAA(5),NPP(5)	Name and index number of the 5th variable.	PHIDOT2 30
61-72	A6, I6	AAAA(6),NPP(6)	Name and index number of the 6th variable.	
73-80				01020

Card 01021

1-12	A6, I6	AAAA(7),NPP(7)	Name and index number of the 7th variable.	
13-24	A6, I6	AAAA(8),NPP(8)	Name and index number of the 8th variable.	
25-36	A6, I6	AAAA(9),NPP(9)	Name and index number of the 9th variable.	
37-48	A6, I6	AAAA(10),NPP(10)	Name and index number of the 10th variable.	
49-60				
61-72				
73-80				01021

Cards 01030-01033 (0 to 4 cards): NPLTVV (max.10) plots of Variable V_y vs. Variable V_x. Both variables must be listed in Cards 01020-01021. Omit cards with no data. Omit cards if NPLTVT≤0 (Card 01010).

Card 01030

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NP1(1,1)	V_x index number of 1st plot.	25
13-24	I12	NP1(1,2)	V_y index number of 1st plot.	26
25-36	I12	NP1(2,1)	V_x index number of 2nd plot.	
37-48	I12	NP1(2,2)	V_y index number of 2nd plot.	
49-60	I12	NP1(3,1)	V_x index number of 3rd plot.	
61-72	I12	NP1(3,2)	V_y index number of 3rd plot.	
73-80				01030

Card 01031

1-12	I12	NP1(4,1)	V_x index number of 4th plot.	
13-24	I12	NP1(4,2)	V_y index number of 4th plot.	
25-36	I12	NP1(5,1)	V_x index number of 5th plot.	
37-48	I12	NP1(5,2)	V_y index number of 5th plot.	
49-60	I12	NP1(6,1)	V_x index number of 6th plot.	
61-72	I12	NP1(6,2)	V_y index number of 6th plot.	
73-80				01031

etc.

Cards 01040-01099 (0 to 60 cards): User's specifications for plot units, scales, and frames. Number of cards is twice the total number of plots, i.e., 2[NPLTVT(1+N1B) + NPLTVV]. Omit cards with no data. Omit all cards if SCALE = 0 (Card 01011) or if NPLTVT≤0 (Card 01010). Plot No. (Columns 61-72) is not read in and may be deleted.

Card 01040

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	A6,6X	Q1	Name of X variable.	TIME
13-24	2R6	XUNIT(J,1)J=1,2	Units of X variable.	MICROSECONDS
25-36	E12.5	XSCALE(1)	Scale (in mks units/inch) of X variable.	2.54-06
37-48	E12.5	XLEFT(1)	Left boundary of frame (in mks units).	-1.0-06
49-60	E12.5	XRIGHT(1)	Right boundary of frame (in mks units).	12.0-06
61-72			Plot No. 1.	PLOT NO. 1
73-80				01040

Card 01041

1-12	A6,6X	Q2	Name of Y variable.	ALPHA
13-24	2R6	YUNIT(J,1)J=1,2	Units of Y variable.	AMPERE
25-36	E12.6	YSCALE(1)	Scale (in mks units/inch) of Y variable.	1.27
37-48	E12.6	YBOTTM(1)	Bottom boundary of frame (in mks units).	-0.5
49-60	E12.6	YTOP(1)	Top boundary of frame (in mks units).	5.0
61-72			Plot No. 1.	PLOT NO. 1
73-80				01041

Card 01042

1-12	A6,6X	Q1	Name of X variable.	TIME
13-24	2R6	XUNIT(J,2)J=1,2	Units of X variable.	MICROSECONDS
25-36	E12.6	XSCALE(2)	Scale (in mks units/inch) of X variable	2.54-06
37-48	E12.6	XLEFT(2)	Left boundary of frame (in mks units).	-1.0-06
49-60	E12.6	XRIGHT(2)	Right boundary of frame (in mks units).	12.0-06
61-72			Plot No. 2.	PLOT NO. 2
73-80				01042

Card 01043

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	A6,6X	Q2	Name of Y variable.	MMF 1
13-24	2R6	YUNIT(J,2)J=1,2	Units of Y variable.	AMPERE TURN
25-36	E12.6	YSCALE(2)	Scale (in mks units/inch) of Y variable.	0.127
37-48	E12.6	YBOTTM(2)	Bottom boundary of frame (in mks units).	-0.05
49-60	E12.6	YTOP(2)	Top boundary of frame (in mks units).	0.5
61-72			Plot No. 2.	PLOT NO. 2
73-80				01043

Card 01044

1-12	A6,6X	Q1	Name of X variable.	TIME
13-24	2R6	XUNIT(J,3)J=1,2	Units of X variable.	MICROSECONDS
25-36	E12.5	XSCALE(3)	Scale (in mks units/inch) of X variable.	2.54 -06
37-48	E12.5	XLEFT(3)	Left boundary of frame (in mks units).	-1.0 -06
49-60	E12.5	XRIGHT(3)	Right boundary of frame (in mks units).	12.0 -06
61-72			Plot No. 3.	PLOT NO. 3
73-80				01044

Card 01045

1-12	A6,6X	Q2	Name of Y variable.	PHI 1
13-24	2R6	YUNIT(J,3)J=1,2	Units of Y variable.	MAXWELLS
25-36	E12.6	YSCALE(3)	Scale (in mks units/inch) of Y variable.	20.0 -08
37-48	E12.6	YBOTTM(3)	Bottom boundary of frame (in mks units).	-40.0 -08
49-60	E12.6	YTOP(3)	Top boundary of frame (in mks units).	40.0 -08
61-72			Plot No. 3.	PLOT NO. 3
73-80				01045

etc.

Card 01100

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	E12.5	ABTEMP	Absolute temperature (°K).	298.0
13-24	I12	NDTSPC	Number of Δt-specification periods (max. 10). If blank, NDTSPC = 1.	3
25-36				
37-48				
49-60				
61-72				
73-80				01100

Cards 01200-01203 (1 to 4 cards): Specification for maximum Δt in NDTSPC periods (max. 10). If a cycle of (NDTSPC-1) periods is to be repeated n times, let the last Δt be -n.0. Omit cards with no data.

Card 01200

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	E12.8	CT(1,1)	Maximum Δt during $0 < t \leq t_1$ (1st period).	1.0 -08
13-24	E12.8	CT(1,2)	t_1	2.5 -06
25-36	E12.8	CT(2,1)	Maximum Δt during $t_1 < t \leq t_2$ (2nd period).	1.0 -06
37-48	E12.8	CT(2,2)	t_2	23.0 -06
49-60	E12.8	CT(3,1)	Maximum Δt during $t_2 < t \leq t_3$ (3rd period).	1.0 -07
61-72	E12.8	CT(3,2)	t_3	24.0 -06
73-80				01200

Card 01201

1-12	E12.8	CT(4,1)	Maximum Δt during $t_3 < t \leq t_4$ (4th period).	
13-24	E12.8	CT(4,2)	t_4	
25-36	E12.8	CT(5,1)	Maximum Δt during $t_4 < t \leq t_5$ (5th period).	
37-48	E12.8	CT(5,2)	t_5	
49-60	E12.8	CT(6,1)	Maximum Δt during $t_5 < t \leq t_6$ (6th period).	
61-72	E12.8	CT(6,2)	t_6	
73-80				01201

etc.

Cards 02000, 02001, 02002,... (0 to 10 cards): Enter only if NG=1 (i.e., if initial unknown voltages and auxiliary unknowns are to be read in). Number of cards $=[(NV+NAUX)/6+\alpha]$, where $0 \leq \alpha \leq 5/6$.

Card 02000

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	E12.8	V1(1)	Initial value of unknown No. 1.	
13-24	E12.8	V1(2)	Initial value of unknown No. 2.	
25-36	E12.8	V1(3)	Initial value of unknown No. 3.	
37-48	E12.8	V1(4)	Initial value of unknown No. 4.	
49-60	E12.8	V1(5)	Initial value of unknown No. 5.	
61-72	E12.8	V1(6)	Initial value of unknown No. 6.	
73-80				02000

Card 02001

1-12	E12.8	V1(7)	Initial value of unknown No. 7.	
13-24	E12.8	V1(8)	Initial value of unknown No. 8.	
25-36	E12.8	V1(9)	Initial value of unknown No. 9.	
37-48	E12.8	V1(10)	Initial value of unknown No. 10.	
49-60	E12.8	V1(11)	Initial value of unknown No. 11.	
61-72	E12.8	V1(12)	Initial value of unknown No. 12.	
73-80				02001

Card 02002

1-12	E12.8	V1(13)	Initial value of unknown No. 13.	
13-24	E12.8	V1(14)	Initial value of unknown No. 14.	
25-36	E12.8	V1(15)	Initial value of unknown No. 15.	
37-48	E12.8	V1(16)	Initial value of unknown No. 16.	
49-60	E12.8	V1(17)	Initial value of unknown No. 17.	
61-72	E12.8	V1(18)	Initial value of unknown No. 18.	
73-80				02002

etc.

Cards 03000-03006 (1 to 7 cards): Data on Radiation photocurrent Ipp.
If there are no radiation runs, enter only the first card (blank).

Card 03000

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NI	Number of break points in Ipp waveform (2 to 16); NI=0 if no rad.run.	2
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				03000

Card 03001

1-12	E12.8	E(1,1,1)	Initial Ipp value (always zero).	0.0
13-24	E12.8	E(1,1,2)	Ipp scale factor.	0.0
25-36	E12.8	E(1,2,1)	Normalized Ipp value at 2nd point.	0.0
37-48	E12.8	E(1,2,2)	Time at 2nd point.	1.0
49-60	E12.8	E(1,3,1)	Normalized Ipp value at 3rd point.	
61-72	E12.8	E(1,3,2)	Time at 3rd point.	
73-80				03001

Card 03002

1-12	E12.8	E(1,4,1)	Normalized Ipp value at 4th point.	
13-24	E12.8	E(1,4,2)	Time at 4th point.	
25-36	E12.8	E(1,5,1)	Normalized Ipp value at 5th point.	
37-48	E12.8	E(1,5,2)	Time at 5th point.	
49-60	E12.8	E(1,6,1)	Normalized Ipp value at 6th point.	
61-72	E12.8	E(1,6,2)	Time at 6th point.	
73-80				03002

etc.

Cards 03010-03016 (2 to 7 cards): Data on 1st variable source (or grounded dc voltage source). If dc, enter first two cards only.

Card 03010

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NI	Number of break points (1 if dc; 2 to 16 if variable).	1
13-24	I12	NI2	-1 - dc; 0 - hold last value; 1 - cycle.	-1
25-36				
37-48				
49-60				
61-72				
73-80				03010

Card 03011: If dc - Columns 13-73 are blank.

1-12	E12.8	E(2,1,1)	Initial source value at 1st point; dc value if dc.	23.0
13-24	E12.8	E(2,1,2)	Initial time at 1st point (usually zero).	
25-36	E12.8	E(2,2,1)	Source value at 2nd point.	
37-48	E12.8	E(2,2,2)	Time at 2nd point.	
49-60	E12.8	E(2,3,1)	Source value at 3rd point.	
61-72	E12.8	E(2,3,2)	Time at 3rd point.	
73-80				03011

Card 03012

1-12	E12.8	E(2,4,1)	Source value at 4th point.	
13-24	E12.8	E(2,4,2)	Time at 4th point.	
25-36	E12.8	E(2,5,1)	Source value at 5th point.	
37-48	E12.8	E(2,5,2)	Time at 5th point.	
49-60	E12.8	E(2,6,1)	Source value at 6th point.	
61-72	E12.8	E(2,6,2)	Time at 6th point.	
73-80				03012

etc.

Cards 03020-03026 (2 to 7 cards): Data on 2nd variable source (or grounded dc voltage source). If uc, enter first two cards only.

Card 03020

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NI	Number of break points (1 if dc; 2 to 16 if variable).	4
13-24	I12	NI2	-1 - dc; 0 - hold last value; 1 - cycle.	
25-36				
37-48				
49-60				
61-72				
73-80				03020

Card 03021: If dc, Columns 13-72 are blank.

1-12	E12.8	E(3,1,1)	Initial source value at 1st point; dc value if dc.	
13-24	E12.8	E(3,1,2)	Initial time at 1st point (usually zero).	
25-36	E12.8	E(3,2,1)	Source value at 2nd point.	5.0
37-48	E12.8	E(3,2,2)	Time at 2nd point.	0.5 -06
49-60	E12.8	E(3,3,1)	Source value at 3rd point.	5.0
61-72	E12.8	E(3,3,2)	Time at 3rd point.	23.0 -06
73-80				03021

Card 03022

1-12	E12.8	E(3,4,1)	Source value at 4th point.	
13-24	E12.8	E(3,4,2)	Time at 4th point.	23.5 -06
25-36	E12.8	E(3,5,1)	Source value at 5th point.	
37-48	E12.8	E(3,5,2)	Time at 5th point.	
49-60	E12.8	E(3,6,1)	Source value at 6th point.	
61-72	E12.8	E(3,6,2)	Time at 6th point.	
73-80				03022

etc.

Cards 03500, 03510, 03511,... (1 or 3 to 31 cards): Normal-Termination conditions: 0 to 5 OR statements, each with 1 to 5 AND conditions. Omit cards with no data.

Card 03500

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	IOC	Number of OR statements.	4
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				03500

Card 03510

1-12	I12	IAC(1)	Number of AND conditions of 1st statement.	3
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				03510

Card 03511: 1st condition of 1st statement.

1-12	A1,8X,I3	A1,NOVAR(1,1)	ABS (absolute value) or blank (otherwise); Unknown V No.	ABS 21
13-24	A1,11X	A2	GT (i.e., >) or LE (i.e., ≤).	LE
25-36	E12.5	BVAL(1,1)	Boundary value.	3.35 -04
37-48				
49-60				
61-72				
73-80				03511

Card 03512: 2nd condition of 1st statement.

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	A1,8X,I3	A1,NOVAR(1,2)	ABS (absolute value) or blank (otherwise); Unknown V No.	20
13-24	A1,11X	A2	GT (i.e., >) or LE (i.e., ≤).	GT
25-36	E12.5	BVAL(1,2)	Boundary value.	0.0
37-48				
49-60				
61-72				
73-80				03512

etc.

Card 03520

1-12	I12	IAC(2)	Number of AND conditions of 2nd statement.	2
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				03520

Card 03521: 1st condition of 2nd statement.

1-12	A1,8X,I3	A1,NOVAR(2,1)	ABS (absolute value) or blank (otherwise); Unknown V No.	23
13-24	A1,11X	A2	GT (i.e., >) or LE (i.e., ≤).	LE
25-36	E12.5	BVAL(2,1)	Boundary value.	-6.55-08
37-48				
49-60				
61-72				
73-80				03521

etc.

Cards 03600, 03610, 03611, ... (1 or 3 to 31 cards): Failure-termination conditions: 0 to 5 OR Statements, each with 1 to 5 AND conditions. Omit cards with no data.

Card 03600

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	IOCF	Number of OR statements.	1
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				03600

Card 03610

1-12	I12	IACF(1)	Number of AND conditions of 1st statement.	3
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				03610

Card 03611: 1st condition of 1st statement.

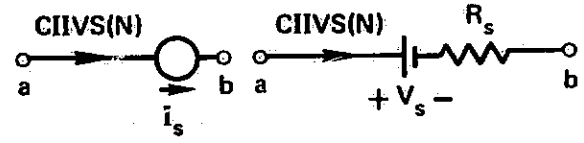
1-12	A1,8X,I3	A1,NOVARF(1,1)	ABS (absolute value) or blank (otherwise); Unknown V No.	7
13-24	A1,11X	A2	GT (i.e., >) or LE (i.e., ≤).	GT
25-36	E12.5	BVALF(1,1)	Boundary value.	140.0
37-48				
49-60				
61-72				
73-80				03611

etc.

Cards 10000, 10010,....(1 to 21 cards): Current sources and floating voltage sources (0 to 20).
If any source varies with time, it must also be entered as a variable source (in Cards 03010,...).
Terminal-type designation: blank - floating; S - known voltage sources; G - ground.
Enter the first card only, if there are no such sources.

Card 10000

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NIV	Number of current sources and floating voltage sources (max.20).	3
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				10000



Card 10010: 1st current or floating voltage source.

1-12	I12	K4(1)	1 - current source; 0 - floating voltage source.	1
13-24	I12	K3(1)	0 (if dc) or index no. of variable source (if variable).	2
25-36	A6,I6	A1,K1(1)	Type (blank, S, or G) and index number of Terminal a.	S 1
37-48	A6,I6	A2,K2(1)	Type (blank, S, or G) and index number of Terminal b.	1
49-60	E12.8	VI(1)	Magnitude of a dc source; blank if a variable source.	
61-72	E12.8	RVS (1)	R_S of voltage source (0.1 Ω if blank); blank if current source.	
73-80				10010

Card 10020: 2nd current or floating voltage source.

1-12	I12	K4(2)	1 - current source; 0 - floating voltage source.	
13-24	I12	K3(2)	0 (if dc) or index no. of variable source (if variable).	
25-36	A6,I6	A1,K1(2)	Type (blank, S, or G) and index number of Terminal a.	8
37-48	A6,I6	A2,K2(2)	Type (blank, S, or G) and index number of Terminal b.	10
49-60	E12.8	VI(2)	Magnitude of a dc source; blank if a variable source.	6.0
61-72	E12.8	RVS(2)	R_S of voltage source (0.1 Ω if blank); blank if current source.	0.05
73-80				10020

etc.

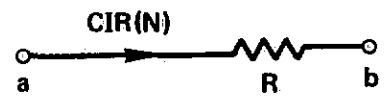
Cards 20000,20010,... (1 to 61 cards): Resistors (0 to 40).

Terminal-type designation: blank - floating; S - known voltage source; G - ground.

Up to 5 resistors may vary with time - see next page.

Enter the first card only, if there are no resistors.

Card 20000

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NR	Number of resistors (max.40).	4
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				20000

Card 20010: 1st resistor, R1.

1-12	A6,I6	F1,K1(1)	Type (blank, S, or G) and index number of Terminal a.	5
13-24	A6,I6	R2,K2(1)	Type (blank, S, or G) and index number of Terminal b.	6
25-36	E12.8	R(1)	Resistance R (if blank, R = 0.001Ω).	104.7
37-48	I12	NRP(1)	No. of break points if R varies with time.	
49-60				
61-72				
73-80				20010

Card 20020: 2nd resistor, R2.

1-12	A6,I6	F1,K1(2)	Type (blank, S, or G) and index number of Terminal a.	2
13-24	A6,I6	R2,K2(2)	Type (blank, S, or G) and index number of Terminal b.	G 0
25-36	E12.8	R(2)	Resistance R (if blank, R = 0.001 Ω).	3.9 06
37-48	I12	NRP(2)	No. of break points if R varies with time.	
49-60				
61-72				
73-80				20020

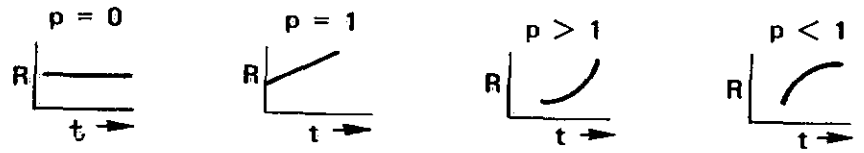
An Nth resistor may vary with time in up to 4 modes, in each of which $R(t) = R_0 + (R_f - R_0)\left(\frac{t-t_0}{t_f-t_0}\right)^p$.

In this case:

(1) NRP(N) ≥ 2, where NRP(N) = No. of break points = 1 + No. of R(t) modes;

(2) Initial R_0 = R (Columns 25-36) and initial t_0 = t;

(3) NRP(N)-1 cards must follow, each with the values of R_f, t_f, and p.



Card 20030: 3rd resistor, R3.

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	A6,I6	F1,K1(3)	Type (blank, S, or G) and index number of Terminal a.	14
13-24	A6,I6	R2,K2(3)	Type (blank, S, or G) and index number of Terminal b.	S 2
25-36	E12.8	R(3)	Resistance R (if blank, R = 0.001 Ω).	0.5
37-48	I12	NRP(3)	No. of break points if R varies with time.	3
49-60				
61-72				
73-80				20030

Card 20031: 1st mode of R3(t).

1-12	E12.5	RF(1,2)	R_f (at 2nd break point).	1.0 +06
13-24	E12.5	TFR(1,2)	t_f (at 2nd break point).	2.0 -06
25-36	E12.5	XRES(1,2)	p value.	0.5
37-48				
49-60				
61-72				
73-80				20031

Card 20032: 2nd mode of R3(t).

1-12	E12.5	RF(1,3)	R_f (at 3rd break point).	1.0 +06
13-24	E12.5	TFR(1,3)	t_f (at 3rd break point).	12.0 -06
25-36	E12.5	XRES(1,3)	p value.	
37-48				
49-60				
61-72				
73-80				20032

etc.

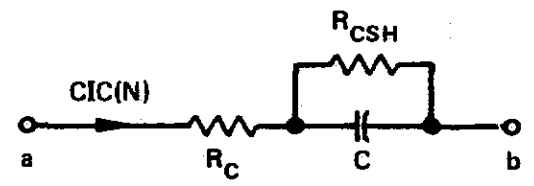
Cards 30000, 30010, ... (1 to 21 cards): Capacitors (0 to 20).

Terminal-type designation: blank - floating; S - known voltage source; G - ground.

Enter the first card only, if there are no capacitors.

Card 30000

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NC	Number of capacitors (max.20)	1
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				30000



Card 30010: 1st capacitor, C1.

1-12	A6, I6	F1, K1(1)	Type (blank, S, or G) and index number of Terminal a.	G 0
13-24	A6, I6	R2, K2(1)	Type (blank, S, or G) and index number of Terminal b.	6
25-36	E12.8	CAPC(1)	Capacitance C.	22.0-06
37-48	E12.8	RC(1)	Series resistance R_C (if blank, then R_C = 0.1 Ω).	0.35
49-60	E12.8	RCSH(1)	Shunt resistance R_{CSH} (if blank, then $R_{CSH} = 10^9$ Ω).	
61-72	E12.8	CIO	Initial current.	
73-80				30010

Card 30020: 2nd capacitor, C2.

1-12	A6, I6	F1, K1(2)	Type (blank, S, or G) and index number of Terminal a.	
13-24	A6, I6	R2, K2(2)	Type (blank, S, or G) and index number of Terminal b.	
25-36	E12.8	CAPC(2)	Capacitance C.	
37-48	E12.8	RC(2)	Series resistance R_C (if blank, then R_C = 0.1 Ω).	
49-60	E12.8	RCSH(2)	Shunt resistance R_{CSH} (if blank, then $R_{CSH} = 10^9$ Ω).	
61-72	E12.8	CIO	Initial current.	
73-80				30020

etc.

Cards 40000, 40010,... (1 to 41 cards): Linear and nonlinear inductors (0 to 20).

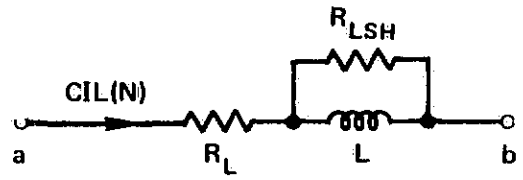
Terminal-type designation: blank - floating; S - known voltage source; G - ground.

For nonlinear inductors, $L = L_0 e^{-i/I_{con}}$.

Enter the first card only, if there are no inductors.

Card 40000

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NL	Number of inductors (max. 20).	2
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				40000



Cards 40010 and 40011: 1st inductor, L1.

Card 40010

1-12	A6, I6	F1,K1(1)	Type (blank, S, or G) and index number of Terminal a.	10
13-24	A6, I6	R2,K2(1)	Type (blank, S, or G) and index number of Terminal b.	1
25-36	E12.8	INDL(1)	Inductance L (linear inductor) or L_0 (nonlinear inductor).	365.0-06
37-48	E12.8	RL(1)	Series resistance R_L (if blank, R_L = 1.0 Ω).	0.5
49-60	E12.8	RLSH(1)	Shunt resistance R_{LSH} (if blank, $R_{LSH} = 10^6$ Ω).	
61-72	E12.8	VICON(1)	blank (linear inductor) or $1/I_{con}$ (nonlinear inductor).	7.24
73-80				40010

Card 40011

1-12	E12.5	CILO	Initial current.	0.152
13-24	I12	INCIL(1)	1 - Initial CIL(N) = CILO regardless of NSVVAR or NSVIL; 0 - otherwise	1
25-36				
37-48				
49-60				
61-72				
73-80				40011

Cards 40020 and 40021: 2nd inductor, L2.

Card 40020

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	A6, I6	F1, K1(2)	Type (blank, S, or G) and index number of Terminal a.	15
13-24	A6, I6	R2, K2(2)	Type (blank, S, or G) and index number of Terminal b.	G 0
25-36	E12.8	INDL(2)	Inductance L (linear inductor) or L_0 (nonlinear inductor).	11.7 -06
37-48	E12.8	RL(2)	Series resistance R_L (if blank, R_L = 1.0 Ω).	
49-60	E12.8	RLSH(2)	Shunt resistance R_{LSH} (if blank, $R_{LSH} = 10^6$ Ω).	2.5 7
61-72	E12.8	VICON(2)	blank (linear inductor) or $1/I_{con}$ (nonlinear inductor).	
73-80				40020

Card 40021

1-12	E12.5	CIL0	Initial current.	
13-24	I12	INCIL(2)	1 - Initial CIL(N) = CIL0 regardless of NSVVAR or NSVIL; 0 - otherwise.	
25-36				
37-48				
49-60				
61-72				
73-80				40021

etc.

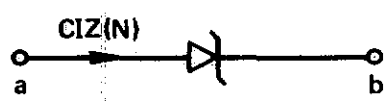
Cards 50000, 50010,... (1 to 21 cards): Zener diodes (0 to 20).

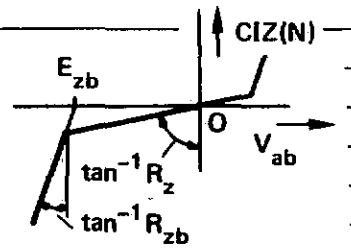
Terminal-type designation: blank - floating; S - known voltage source; G - ground.

Enter the first card only, if there are no zener diodes.

Card 50000

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NZEN	Number of zener diodes (max. 20).	2
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				50000





Card 50010: 1st zener diode, Z1.

1-12	A6, I6	F1, K1(1)	Type (blank, S, or G) and index number of Terminal a.	G 0
13-24	A6, I6	R2, K2(1)	Type (blank, S, or G) and index number of Terminal b.	9
25-36	E12.8	RZ(1)	High zener resistance R_z.	0.18 +06
37-48	E12.8	RZB(1)	Low back zener resistance R_{zb}.	0.5
49-60	E12.8	EZB(1)	Extrapolated back zener voltage E_{zb} (negative value).	-4.65
61-72				
73-80				50010

Card 50020: 2nd zener diode, Z2.

1-12	A6, I6	F1, K1(2)	Type (blank, S, or G) and index number of Terminal a.	G 0
13-24	A6, I6	R2, K2(2)	Type (blank, S, or G) and index number of Terminal b.	14
25-36	E12.8	RZ(2)	High zener resistance R_z.	0.35 +06
37-48	E12.8	RZB(2)	Low back zener resistance R_{zb}.	0.55
49-60	E12.8	EZB(2)	Extrapolated back zener voltage E_{zb} (negative value).	-4.68
61-72				
73-80				50020

etc.

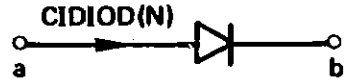
Cards 60000, 60010,... (1 to 41 cards): Diodes (0 to 20).

Terminal-type designation: blank - floating; S - known voltage source; G - ground.

Two cards per diode if the parameter values are specified; One card per diode if the paramater values are identical with those of another diode (of lower index number).

Enter the first card only, if there are no diodes.

Card 60000

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NDIODE	Number of diodes (max. 20).	4
13-24				
25-36				
37-48			CIDIOD(N) 	
49-60				
61-72				
73-80				60000

Cards 60010 and 60011: 1st diode, D1.

Card 60010

1-12	A6, I6	F1, K1(1)	Type (blank, S, or G) and index number of Terminal a.	G 0
13-24	A6, I6	R2, K2(1)	Type (blank, S, or G) and index number of Terminal b.	8
25-36	I12	K3	Parameter values: 0 - specify; N - same as those of Nth diode.	
37-48	E12.8	QCR(1,6)	Photocurrent I_{pp}.	
49-60				
61-72				
73-80				60010

Card 60011: Parameter values of D1.

1-12	E12.8	QCR(1,1)	I_{sd}	9.717 -09
13-24	E12.8	QCR(1,2)	m_d	1.96
25-36	E12.8	QCR(1,3)	$R_{\ell d}$	6.00 +09
37-48	E12.8	QCR(1,4)	C_{j0d}	1.0 -12
49-60	E12.8	QCR(1,5)	$V_{\varphi d}$	0.8
61-72	E12.8	QCR(1,7)	T_d	1.0 -09
73-80				60011

Cards 60020 and 60021: 2nd diode, D2.

Card 60020

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	A6, I6	F1, K1(2)	Type (blank, S, or G) and index number of Terminal a.	1
13-24	A6, I6	R2, K2(2)	Type (blank, S, or G) and index number of Terminal b.	2
25-36	I12	K3	Parameter values: 0 - specify; N - same as those of Nth diode.	1
37-48	E12.8	QCR(2,6)	Photocurrent I_{pp}.	
49-60				
61-72				
73-80				60020

Card 60021: Parameter values of D2. Omit this card if parameter values are identical with those of D1 (K3=1).

1-12	E12.8	QCR(2,1)	I_{sd}	
13-24	E12.8	QCR(2,2)	m_d	
25-36	E12.8	QCR(2,3)	$R_{\ell d}$	
37-48	E12.8	QCR(2,4)	C_{j0d}	
49-60	E12.8	QCR(2,5)	$V_{\varphi d}$	
61-72	E12.8	QCR(2,7)	T_d	
73-80				60021

etc.

Cards 70000, 70010, ... (1 to 81 cards): npn and pnp Transistors (0 to 20).

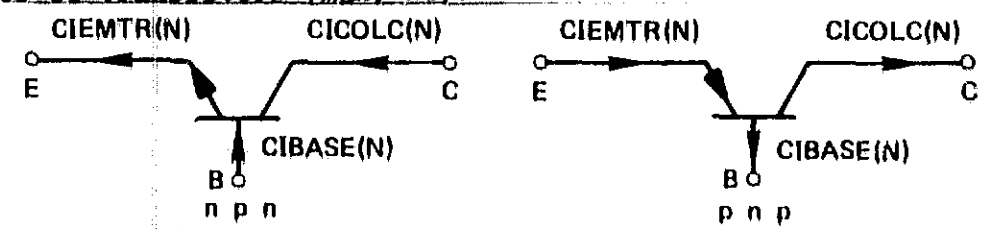
Terminal-type designation: blank - floating; S - known voltage source; G - ground

Four cards per transistor if the parameter values are specified; one card per transistor if the parameter values are identical with those of another transistor (of lower index number).

Enter the first card only, if there are no transistors.

Card 70000

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NTRANS	Number of transistors (max. 20).	
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				



Cards 70010 through 70013: 1st transistor, T1.

Card 70010

1-12	I12	K1(1)	0 - npn; 1 - pnp.	
13-24	A6, I6	BASE, K2(1)	Type (blank, S, or G) and index number of Terminal B (Base).	
25-36	A6, I6	COLC, K3(1)	Type (blank, S, or G) and index number of Terminal C (Collector).	
37-48	A6, I6	EMTR, K4(1)	Type (blank, S, or G) and index number of Terminal E (Emitter).	G
49-60	I12	K5	Parameter values: 0 - specify; N - same as those of Nth transistor.	
61-72				
73-80				

Cards 70011, 70012, and 70013: Parameter values of T1.

Card 70011

1-12	E12.8	QTAN(1,1)	β_n	50.0
13-24	E12.8	QTAN(1,2)	β_i	0.1
25-36	E12.8	QTAN(1,3)	T_e	1.0E-09
37-48	E12.8	QTAN(1,4)	T_c	6.9E-06
49-60	E12.8	QTAN(1,5)	I_{sc}	2.1E-09
61-72	E12.8	QTAN(1,6)	m_c	1.75
73-80				

Card 70012

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	E12.8	QTAN(1,7)	C_{j0c}	0.66 -12
13-24	E12.8	QTAN(1,8)	$V_{\varphi c}$	0.7
25-36	E12.8	QTAN(1,9)	$R_{\ell c}$	0.3 +12
37-48	E12.8	QTAN(1,10)	I_{se}	54.01 -09
49-60	E12.8	QTAN(1,11)	m_e	2.12
61-72	E12.8	QTAN(1,12)	C_{j0e}	57.0 -12
73-80				70012

Card 70013

1-12	E12.8	QTAN(1,13)	$V_{\varphi e}$	0.6
13-24	E12.8	QTAN(1,14)	$R_{\ell e}$	0.2 +12
25-36	E12.8	QTAN(1,15)	Collector photocurrent I_{ppc}.	
37-48	E12.8	QTAN(1,16)	Emitter photocurrent I_{ppe}.	
49-60				
61-72				
73-80				70013

Cards 70020 through 70023; 2nd transistor, T2.

Card 70020

1-12	I12	K1(2)	0 - npn; 1 - pnp.	
13-24	A6, I6	BASE, K2(2)	Type (blank, S, or G) and index number of Terminal B (Base).	13
25-36	A6, I6	COLC, K3(2)	Type (blank, S, or G) and index number of Terminal C (Collector).	12
37-48	A6, I6	EMTR, K4(2)	Type (blank, S, or G) and index number of Terminal E (Emitter).	14
49-60	I12	K5	Parameter values: 0 - specify; N - same as those of Nth transistor.	1
61-72				
73-80				70020

Cards 70021, 70022, and 70023: Parameter values of T2. Omit these three cards if the parameter values are identical with those of T1 (K5 = 1).

Card 70021

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	E12.8	QTAN(2,1)	β_n	
13-24	E12.8	QTAN(2,2)	β_i	
25-36	E12.8	QTAN(2,3)	T_e	
37-48	E12.8	QTAN(2,4)	T_c	
49-60	E12.8	QTAN(2,5)	I_{sc}	
61-72	E12.8	QTAN(2,6)	m_c	70021
73-80				

Card 70022

1-12	E12.8	QTAN(2,7)	C_{j0c}	
13-24	E12.8	QTAN(2,8)	$V_{\varphi c}$	
25-36	E12.8	QTAN(2,9)	$R_{\ell c}$	
37-48	E12.8	QTAN(2,10)	I_{se}	
49-60	E12.8	QTAN(2,11)	m_e	
61-72	E12.8	QTAN(2,12)	C_{j0e}	70022
73-80				

Card 70023

1-12	E12.8	QTAN(2,13)	$V_{\varphi e}$	
13-24	E12.8	QTAN(2,14)	$R_{\ell e}$	
25-36	E12.8	QTAN(2,15)	Collector photocurrent I_{ppc}.	
37-48	E12.8	QTAN(2,16)	Emitter photocurrent I_{ppe}.	
49-60				
61-72				70023
73-80				

etc.

(1) $\underline{\emptyset_j(F) \text{ and } \dot{\emptyset}_p(F) \text{ Input Core Parameters (PARIN = PHIF or blank).}}$

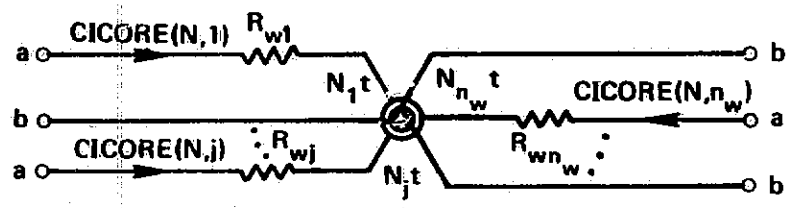
Cards 80000, 80001, ... (1 to 322 cards): Magnetic cores (0 to 20), each with n_w windings (1 to 10).

Terminal-type designation: blank - floating; S - known voltage source; G - ground.

Use $6 + n_w$ cards per core if the parameter values are specified; $1 + n_w$ cards per core if the parameter values are identical with those of another core (of lower index number). Since $1 \leq n_w \leq 10$, there may be 2 to 16 cards per core. Enter the first card only, if there are no cores.

Card 80000

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NCORE	Number of cores (max. 20).	4
13-24	A6	PARIN	Parameter input-data kind: PHIF or blank - ØF data; BH - BH data.	PHIF
25-36				
37-48				
49-60				
61-72				
73-80				80000



Card 80001

1-12	E12.5	RELERR	Relative error for convergence of winding currents (10^{-4} if blank).	0.00001
13-24	E12.5	ABSERR	Absolute error for convergence of winding currents (10^{-4}A if blank).	0.00001
25-36	E12.5	PSTEP	Weight of previous-iteration results on computed winding current, varying from PSTEP = 0 (no effect) to |PSTEP| = 1.0 (i = $i_{<-1>}$). If "1/2-factor" rather than Aitken formula is to be used, PSTEP is negative.	0.5
37-48	F12.0	STSWEX	1 - terminate mode run if flux switching of all cores is completed and t > NUDT · Δt; 0 - otherwise.	1
49-60	I12	NUDT	Number of Δt's above which STSWEX is effective.	10
61-72	I12	LASTIC	1 - elastic Ø is included; 0 - elastic Ø neglected.	1
73-80				80001

Cards 80100, 80101, ...: 1st magnetic core, Core 1.

Card 80100

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NW(1)	Number of windings.	4
13-24	I12	KC1	Parameter values: 0 - specify; N - same as those of Nth core.	
25-36	E12.5	U0(1)	Initial [illegible]r.	1.0
37-48				
49-60				
61-72				
73-80				80100

Card 80101: 1st winding of Core 1.

1-12	A6, I6	F1C,K1C(1,1)	Type (blank, S, or G) and index number of Terminal a (bottom).	4
13-24	A6, I6	R2C,K2C(1,1)	Type (blank, S, or G) and index number of Terminal b (top).	5
25-36	F12.0	XNT(1,1)	Number of turns.	4
37-48	E12.5	RW(1,1)	Winding resistance (0.1Ω if blank).	0.015
49-60	I6, I6	NP(1,1);NS(1,1)	No. of identically switching cores in parallel; in series (1 if blank).	6
61-72	E12.5	CI0	Initial current.	
73-80				80101

Card 80102: 2nd winding of Core 1.

1-12	A6, I6	F1C,K1C(1,2)	Type (blank, S, or G) and index number of Terminal a (bottom).	7
13-24	A6, I6	R2C,K2C(1,2)	Type (blank, S, or G) and index number of Terminal b (top).	8
25-36	F12.0	XNT(1,2)	Number of turns.	20
37-48	E12.5	RW(1,2)	Winding resistance (0.1Ω if blank).	0.3
49-60	I6, I6	NP(1,2);NS(1,2)	No. of identically switching cores in parallel; in series (1 if blank).	
61-72	E12.5	CI0	Initial current.	
73-80				80102

etc.

Cards 80120 through 80124: Parameters of Core 1.

Use mks units (Ø in weber, F in ampere-turn, H in ampere-turn/meter, etc.), except for core diameters.

Cards 80120, 80121, and 80122: Static $\phi_d(F)$ parameters.

Card 80120

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	E12.5	XOD(1)	Outside diameter (in inch).	0.147
13-24	E12.5	XID(1)	Inside diameter (in inch).	0.09
25-36	E12.5	PHIS(1)	ϕ_s - Saturation flux (if blank, then $\phi_s = 1.1\phi_r$).	39.6 -08
37-48	E12.5	PHIR(1)	ϕ_r - Maximum remanent flux.	36.0 -08
49-60	11X, A1	TYPE (1)	$\phi_d(F)$ - curve type: G, A, B, C, D, E, or F (if blank, then B).	B
61-72	A1	CORMAT(1)	Core material: F or blank - Ferrite; T - Tape-wound.	FERRITE
73-80				80120

Card 80121

1-12	E12.5	FD1(1)	F_{d1} - Threshold mmf; blank for Type D.	0.21
13-24	E12.5	FDZ(1)	F_{dz} - May be blank if $F_{dz} = F_{d1}$ or $F_{dz} = \frac{1}{2}(F_{d1} + F_{d2})$; blank for Types A, D, E	
25-36	E12.5	FD2(1)	F_{d2} - May be blank if $F_{d2} = F_{d1}$; blank for Types A, D, and E.	
37-48	E12.5	FD3(1)	F_{d3} - Blank for Types B, D, and F.	
49-60	E12.5	PHID2(1)	ϕ_{d2}: ϕ_d @ F = F_{d2}; blank for Types A, D, E, and F.	-13.0 -08
61-72				
73-80				80121

Card 80122

1-12	E12.5	HA(1)	H_a - Material saturation parameter.	147.1
13-24	E12.5	HQ(1)	H_q - Material nonsaturation parameter; blank for Types C, E, and F.	19.81
25-36	E12.5	HN(1)	H_n - Material nonsaturation parameter; blank for Types C, E, and F.	15.23
37-48				
49-60				
61-72				
73-80				80122

Cards 80123 and 80124: Dynamic $\phi_p(F)$ parameters. Values of parameters F_0'', F_B, F_0 and ρ_p must be entered. Close values of the remaining parameters will be computed by the program if not entered.

Card 80123

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	E12.5	XLMDAD(1)	λ_d (blank if not known).	2.0508
13-24	E12.5	XNUD(1)	ν_d (blank if not known).	1.8569
25-36	E12.5	FDB(1)	F_{dB} (blank if not known).	0.3290
37-48	E12.5	FOPP(1)	F_0''	0.2863
49-60	E12.5	XLAMDA(1)	λ (blank if not known).	1.8076
61-72	E12.5	XNU(1)	ν (blank if not known).	1.2423
73-80				80123

Card 80124

1-12	E12.5	FB(1)	F_B	1.7897
13-24	E12.5	FO(1)	F_0	0.5735
25-36	E12.5	ROP(1)	ρ_p	2.4768
37-48	E12.5	FB1(1)	F_{B1} (blank if no Region 5).	
49-60	E12.5	FO1(1)	F_{01} (blank if F_{B1} is blank).	
61-72	E12.5	ROP1(1)	ρ_{p1} (blank if F_{B1} is blank).	
73-80				80124

Cards 80200, 80201, ...: 2nd magnetic core, Core 2.

Card 80200

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NW(2)	Number of windings.	2
13-24	I12	KC1	Parameter values: 0 - specify; N - same as those of Nth core.	1
25-36	E12.5	U0(2)	Initial ϕ/ϕ_r.	-1.0
37-48				
49-60				
61-72				
73-80				80200

Card 80201: 1st winding of Core 2.

1-12	A6,I6	F1C,K1C(2,1)	Type (blank, S, or G) and index number of Terminal a (bottom).	9
13-24	A6,I6	R2C,K2C(2,1)	Type (blank, S, or G) and index number of Terminal b (top).	10
25-36	F12.0	XNT(2,1)	Number of turns.	20
37-48	E12.5	RW(2,1)	Winding resistance (0.1Ω if blank).	0.3
49-60	I6,I6	NP(2,1);NS(2,1)	No. of identically switching cores in parallel; in series (1 if blank).	
61-72	E12.5	CI0	Initial current.	
73-80				80201

Card 80202: 2nd winding of Core 2.

1-12	A6,I6	F1C,K1C(2,2)	Type (blank, S, or G) and index number of Terminal a (bottom).	G 0
13-24	A6,I6	R2C,K2C(2,2)	Type (blank, S, or G) and index number of Terminal b (top).	7
25-36	F12.0	XNT(2,2)	Number of turns.	4
37-48	E12.5	RW(2,2)	Winding resistance (0.1Ω if blank).	0.06
49-60	I6,I6	NP(2,2);NS(2,2)	No. of identically switching cores in parallel; in series (1 if blank).	
61-72	E12.5	CI0	Initial current.	
73-80				80202

etc.

Cards 80220 through 80224: Parameters of Core 2. Omit these five cards if the parameter values are identical with those of Core 1 (KC1 = 1).

Use mks units (Ø in weber, F in ampere-turn, H in ampere-turn/meter, etc.), except for core diameters.

Cards 80220, 80221, and 80222: Static $\phi_d(F)$ parameters.

Card 80220

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	E12.5	XOD(2)	Outside diameter (in inch).	
13-24	E12.5	XID(2)	Inside diameter (in inch).	
25-36	E12.5	PHIS(2)	ϕ_s - Saturation flux (if blank, then $\phi_s = 1.1\ \phi_r$).	
37-48	E12.5	PHIR(2)	ϕ_r - Maximum remanent flux.	
49-60	11X,A1	TYPE(2)	$\phi_d(F)$ - curve type: G, A, B, C, D, E, or F (if blank, then B).	
61-72	A1	CORMAT(2)	Core material: F or blank - Ferrite; T - tape-wound.	
73-80				80220

Card 80221

1-12	E12.5	FD1(2)	F_{d1} - Threshold mmf; blank for Type D.	
13-24	E12.5	FDZ(2)	F_{dz} - May be blank if $F_{dz} = F_{d1}$ or $F_{dz} = \frac{1}{2}(F_{d1} + F_{d2})$; blank for Types A, D, E.	
25-36	E12.5	FD2(2)	F_{d2} - May be blank if $F_{d2} = F_{d1}$; blank for Types A, D, and E.	
37-48	E12.5	FD3(2)	F_{d3} - Blank for Types B, D, and F.	
49-60	E12.5	PHID2(2)	ϕ_{d2}; ϕ_d @ $F = F_{d2}$; blank for Types A, D, E, and F.	
61-72				
73-80				80221

Card 80222

1-12	E12.5	HA(2)	H_a - Material saturation parameter.	
13-24	E12.5	HQ(2)	H_q - Material nonsaturation parameter; blank for Types C, E, and F.	
25-36	E12.5	HN(2)	H_n - Material nonsaturation parameter; blank for Types C, E, and F.	
37-48				
49-60				
61-72				
73-80				80222

Cards 80223 and 80224: Dynamic $\phi_p(F)$ parameters. Values of Parameters F_0'', F_B, F_0, and ρ_p must be entered.
Close values of the remaining paramaters will be computed by the program if not entered.

Card 80223

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	E12.5	XLMDAD(2)	λ_d (blank if not known).	
13-24	E12.5	XNUD(2)	ν_d (blank if not known).	
25-36	E12.5	FDB(2)	F_{dB} (blank if not known).	
37-48	E12.5	FOPP(2)	F_0''	
49-60	E12.5	XLAMDA(2)	λ (blank if not known).	
61-72	E12.5	XNU(2)	ν (blank if not known).	
73-80				80223

Card 80224

1-12	E12.5	FB(2)	F_B	
13-24	E12.5	FO(2)	F_0	
25-36	E12.5	ROP(2)	ρ_p	
37-48	E12.5	FB1(2)	F_{B1} (blank if no Region 5).	
49-60	E12.5	FO1(2)	F_{01} (blank if F_{B1} is blank).	
61-72	E12.5	ROP1(2)	ρ_{p1} (blank if F_{B1} is blank).	
73-80				80224

etc.

(2) Static and Dynamic B(H)-Loop Input Core Parameters (PARIN = BH).

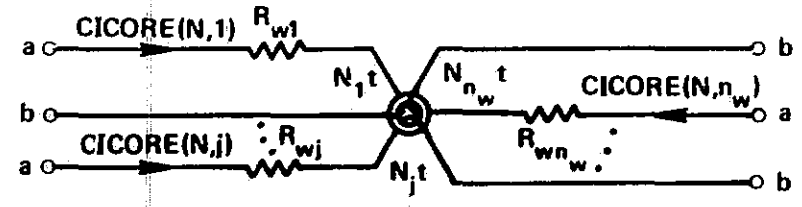
Cards 80000, 80001, ...: Magnetic cores (0 to 20), each with n_w windings (1 to 10).

Terminal-type designation: blank - floating; S - known voltage source; G - ground.

Enter the first card only, if there are no cores.

Card 80000

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NCORE	Number of cores (max. 20).	4
13-24	A6	PARIN	Parameter input-data kind: PHIF or blank - ØF data; BH - BH data.	PHIF
25-36				
37-48				
49-60				
61-72				
73-80				80000



Card 80001

1-12	E12.5	RELERR	Relative error for convergence of winding currents (10^{-4} if blank).	0.00001
13-24	E12.5	ABSERR	Absolute error for convergence of winding currents (10^{-4}A if blank).	0.00001
25-36	E12.5	PSTEP	Weight of previous-iteration results on computed winding current, varying from PSTEP = 0 (no effect) to |PSTEP| = 1.0 (i = $i_{<-1>}$). If "1/2-factor" rather than Aitken formula is to be used, PSTEP is negative.	0.5
37-48	F12.0	STSWEX	1 - terminate mode run if flux switching of all cores is completed and t > NUDT · Δt; 0 - otherwise.	1
49-60	I12	NUDT	Number of Δt's above which STSWEX is effective.	10
61-72	I12	LASTIC	1 - elastic Ø is included; 0 - elastic Ø neglected.	1
73-80				80001

Cards 80100, 80101, ...: 1st magnetic core, Core 1.

Card 80100

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NW(1)	Number of windings.	4
13-24	I12	KC1	Parameter values: O - specify; N - same as those of Nth core.	
25-36	E12.5	U0(1)	Initial B/B_r.	1.0
37-48				
49-60				
61-72				
73-80				80100

Card 80101: 1st winding of Core 1.

1-12	A6, I6	F1C,K1C(1,1)	Type (blank, S, or G) and index number of Terminal a (bottom).	4
13-24	A6, I6	R2C,K2C(1,1)	Type (blank, S, or G) and index number of Terminal b (top).	5
25-36	F12.0	XNT(1,1)	Number of turns.	4
37-48	E12.5	RW(1,1)	Winding resistance (0.1Ω if blank).	0.015
49-60	I6, I6	NP(1,1);NS(1,1)	No. of identically switching cores in parallel; in series (1 if blank).	6
61-72	E12.5	CI0	Initial current.	
73-80				80101

Card 80102: 2nd winding of Core 1.

1-12	A6, I6	F1C,K1C(1,2)	Type (blank, S, or G) and index number of Terminal a (bottom).	7
13-24	A6, I6	R2C,K2C(1,2)	Type (blank, S, or G) and index number of Terminal b (top).	8
25-36	F12.0	XNT(1,2)	Number of turns.	20
37-48	E12.5	RW(1,2)	Winding resistance (0.1Ω if blank).	0.3
49-60	I6, I6	NP(1,2);NS(1,2)	No. of identically switching cores in parallel; in series (1 if blank).	
61-72	E12.5	CI0	Initial current.	
73-80				80102

etc.

Cards 80120 through 80122, 80130, 80131, ... ($4 + n_{dynp}$ cards): Parameters of Core 1.

Use units that are common in core-manufacturer catalogs: B in gausses, H in oersted, diameter in inch, and cross-sectional area in cm^2.

Cards 80120, 80121, and 80122: Static $B_d(H)$ parameters.

Card 80120

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	E12.5	XOD(1)	Outside diameter (in inch).	0.147
13-24	E12.5	XID(1)	Inside diameter (in inch).	0.090
25-36	E12.5	AIN(1)	Effective cross-sectional area (in cm^2).	0.01452
37-48	E12.5	BRIN(1)	$B_{r,in}$- Maximum remanent flux density (in gausses).	6200.0
49-60	11X,A1	TYPE(1)	$B_d(H)$ - curve type: G, A, B, C, D, E, or F (if blank, then G).	G
61-72	A1	CORMAT(1)	Core material: F - Ferrite; T or blank - Tape-wound.	TAPE-WOUND
73-80				80120

Card 80121: Threshold and boundary H values (in oersted).

1-12	E12.5	HD1IN(1)	$H_{d1,in}$- H threshold; blank for Type D.	0.01
13-24	E12.5	HDZIN(1)	$H_{dz,in}$- May be blank if $H_{dz}=H_{d1}$ or $H_{dz}=\frac{1}{2}(H_{d1}+H_{d2})$; blank for Types A, D, and E	0.018
25-36	E12.5	HD2IN(1)	$H_{d2,in}$- May be blank if $H_{d2}=H_{d1}$; blank for Types A, D and E.	0.022
37-48	E12.5	HD3IN(1)	$H_{d3,in}$- Blank for Types B and F.	0.038
49-60	E12.5	HMIN(1)	$H_{m,in}$- Blank for Types C, E, and F.	0.16
61-72				
73-80				80121

Card 80122: Boundary B values (in gausses).

1-12	E12.5	BD2IN(1)	$B_{d2,in}$: B_d @ $H = H_{d2}$; blank for Types A, D, E, and F.	5000.0
13-24	E12.5	BD3IN(1)	$B_{d3,in}$: B_d @ $H = H_{d3}$; blank for Types B, C, E, and F.	4000.0
25-36	E12.5	BMIN(1)	$B_{m,in}$: B_d @ $H = H_m$ $[B_m \approx \frac{1}{2}(B_r+B_s)]$; blank for Types C, E, and F.	7000.0
37-48				
49-60				
61-72				
73-80				80122

Cards 80130, 80131, 80132, ...: Dynamic B(H)-curve parameters.

Card 80130

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NDYNP(1)	n_{dynp} - Number of dynamic B(H) curves vs. frequency.	4
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				80130

Card 80131

1-12	E12.5	FREQ(1,1)	Frequency (in Hz) of 1st point.	400.0
13-24	E12.5	HOBIN(1,1)	H @ B = 0 (in oersted) of 1st point.	0.036
25-36				
37-48				
49-60				
61-72				
73-80				80131

Card 80132

1-12	E12.5	FREQ(1,2)	Frequency (in Hz) of 2nd point.	1000.0
13-24	E12.5	HOBIN(1,2)	H @ B = 0 (in oersted) of 2nd point.	0.045
25-36				
37-48				
49-60				
61-72				
73-80				80132

etc.

Cards 80200, 80201, ... : 2nd magnetic core, Core 2.

Card 80200

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NW(2)	Number of windings.	2
13-24	I12	KC1	Parameter values: 0 - specify; N - same as those of Nth core.	1
25-36	E12.5	U0(2)	Initial B/B_r.	-1.0
37-48				
49-60				
61-72				
73-80				80200

Card 80201: 1st winding of Core 2.

1-12	A6, I6	F1C,K1C(2,1)	Type (blank, S, or G) and index number of Terminal a (bottom).	9
13-24	A6, I6	R2C,K2C(2,1)	Type (blank, S, or G) and index number of Terminal b (top).	10
25-36	F12.0	XNT(2,1)	Number of turns.	20
37-48	E12.5	RW(2,1)	Winding resistance (0.1Ω if blank).	0.3
49-60	I6, I6	NP(2,1);NS(2,1)	No. of identically switching cores in parallel; in series (1 if blank).	
61-72	E12.5	CI0	Initial current.	
73-80				80201

Card 80202: 2nd winding of Core 2.

1-12	A6,I6	F1C,K1C(2,2)	Type (blank, S, or G) and index number of Terminal a (bottom).	G 0
13-24	A6,I6	R2C,K2C(2,2)	Type (blank, S, or G) and index number of Terminal b (top).	7
25-36	F12.0	XNT(2,2)	Number of turns.	4
37-48	E12.5	RW(2,2)	Winding resistance (0.1Ω if blank).	0.06
49-60	I6,I6	NP(2,2);NS(2,2)	No. of identically switching cores in parallel; in series (1 if blank).	
61-72	E12.5	CI0	Initial current.	
73-80				80202

etc.

Cards 80220 through 80222, 80230, 80231, ... [0 or (4 + n_{dynp}) cards]: Parameters of Core 2.

Omit these (4+n_{dynp}) cards if the parameter values are identical with those of Core 1 (KCI=1).

Use units that are common in core-manufacturer catalogs: B in gausses, H in oersted, diameter in inch, and cross-sectional area in cm^2.

Cards 80220, 80221, and 80222: Static $B_d(H)$ parameters.

Card 80220

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	E12.5	XOD(2)	Outside diameter (in inch).	
13-24	E12.5	XID(2)	Inside diameter (in inch).	
25-36	E12.5	AIN(2)	Effective cross-sectional area (in cm^2).	
37-48	E12.5	BRIN(2)	$B_{r,in}$, Maximum remanent flux density (in gausses).	
49-60	11X,A1	TYPE(2)	$B_d(H)$ - curve type: G, A, B, C, D, E, or F (if blank, then G).	
61-72	A1	CORMAT(2)	Core material: F - Ferrite; T or Blank - Tape-wound.	
73-80				80220

Card 80221: Threshold and boundary H values (in oersted).

1-12	E12.5	HD1IN(2)	$H_{d1,in}$, H threshold; blank for Type D.	
13-24	E12.5	HDZIN(2)	$H_{dz,in}$ - May be blank if $H_{dz}=H_{d1}$ or $H_{dz}=\frac{1}{2}(H_{d1}+H_{d2})$; blank for Types A, D, and E.	
25-36	E12.5	HD2IN(2)	$H_{d2,in}$ - May be blank if $H_{d2}=H_{d1}$; blank for Types A, D, and E.	
37-48	E12.5	HD3IN(2)	$H_{d3,in}$ - Blank for Types B and F.	
49-60	E12.5	HMIN(2)	$H_{m,in}$ - Blank for Types C, E, and F.	
61-72				
73-80				80221

Card 80222: Boundary B values (in gausses).

1-12	E12.5	BD2IN(2)	$B_{d2,in}$: B_d @ $H = H_{d2}$; blank for Types A, D, E, and F.	
13-24	E12.5	BD3IN(2)	$B_{d3,in}$: B_d @ $H = H_{d3}$; blank for Types B, C, E, and F.	
25-36	E12.5	BMIN(2)	$B_{m,in}$: B_d @ $H = H_m$ [$B_m \approx \frac{1}{2}(B_r+B_s)$.]; blank for Types C, E, and F.	
37-48				
49-60				
61-72				
73-80				80222

Cards 80230, 80231, 80232, ...: Dynamic B(H)-curve parameters.

Card 80230

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	I12	NDYNP(2)	n_{dynp}- Number of dynamic B(H) curves vs. frequency.	80230
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				

Card 80231

1-12	E12.5	FREQ(2,1)	Frequency (in Hz) of 1st point.	80231
13-24	E12.5	HOBIN(2,1)	H @ B = 0 (in oersted) of 1st point.	
25-36				
37-48				
49-60				
61-72				
73-80				

Card 80232

1-12	E12.5	FREQ(2,2)	Frequency (in Hz) of 2nd point.	80232
13-24	E12.5	HOBIN(2,2)	H @ B = 0 (in oersted) of 2nd point.	
25-36				
37-48				
49-60				
61-72				
73-80				

etc.

B. PARAMETER DEVIATION CARDS

Cards S03001, S03011, ... S10010, S10020, ... etc. (1 to 300 cards): Signs of maximum deviation of parameter values.

Enter only if DMAX > 0.

Each group of cards follows the last parameter card of the corresponding circuit element. For variable sources and grounded dc voltage sources, the sign card follows the corresponding source data cards.

Use 1 for a + sign, blank for no change, and -1 for a -sign. Use different values for deviations other than DMAX percent.

Field location corresponds to the location of each parameter in the corresponding input-data card.

Specified parameter values will be multiplied by [1 + (S·DMAX/100)] in the corresponding subprogram.

Cards S03001, S03011, S03021, ... (1 to 19 cards): Variable sources and grounded dc sources.

Card S03001: Follows photocurrent cards.

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	F12.0	S1	Sign S of photocurrent.	-1
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				S03001

Card S03011: Follows cards of 1st variable source.

1-12	F12.0	S1	Sign S of 1st variable source.	1
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				S03011

Card S03021: Follows cards of 2nd variable source.

1-12	F12.0	S1	Sign S of 2nd variable source.	
13-24				
25-36				
37-48				
49-60				
61-72				
73-80				S03021

etc.

Cards S10010, S10020, S10030, ... (1 to 20 cards): Current sources and floating voltage sources.
These cards follow the last source card in the 10000 series.

Card S10010: 1st current or floating-voltage source.

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	12X			
13-24	12X			
25-36	12X			
37-48	12X			
49-60	F12.0	S1	Sign S of magnitude of a dc source; blank if a variable source.	1
61-72	F12.0	S2	Sign S of R_S of a voltage source; blank if current source.	-1
73-80				S10010

Card S10020: 2nd current or floating-voltage source.

1-12	12X			
13-24	12X			
25-36	12X			
37-48	12X			
49-60	F12.0	S1	Sign S of magnitude of a dc source; blank if a variable source.	-1
61-72	F12.0	S2	Sign S of R_S of a voltage source; blank if current source.	1
73-80				S10020

Card S10030: 3rd current or floating-voltage source.

1-12	12X			
13-24	12X			
25-36	12X			
37-48	12X			
49-60	F12.0	S1	Sign S of magnitude of a dc source; blank if a variable source.	
61-72	F12.0	S2	Sign S of R_S of a voltage source; blank if current source.	
73-80				S10030

etc.

Cards S20010, S20020, S20030, ... (1 to 40 cards): Resistors.

These cards follow the last resistor card in the 20000 series.

Card S20010: 1st resistor, R1.

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	12X			
13-24	12X			
25-36	F12.0	S1	Sign S of resistance R (and R_f, if R varies with time).	-1
37-48				
49-60				
61-72				
73-80				S20010

Card S20020: 2nd resistor, R2.

1-12	12X			
13-24	12X			
25-36	F12.0	S1	Sign S of resistance R (and R_f, if R varies with time).	
37-48				
49-60				
61-72				
73-80				S20020

Card S20030: 3rd resistor, R3.

1-12	12X			
13-24	12X			
25-36	F12.0	S1	Sign S of resistance R (and R_f, if R varies with time).	1
37-48				
49-60				
61-72				
73-80				S20030

etc.

Cards S30010, S30020, S30030, ... (1 to 20 cards): Capacitors.

These cards follow the last capacitor card in the 30000 series.

Card S30010: 1st capacitor, C1.

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	12X			
13-24	12X			
25-36	F12.0	S1	Sign S of capacitance C.	
37-48	F12.0	S2	Sign S of series resistance R_C.	-1
49-60	F12.0	S3	Sign S of shunt resistance R_{CSH}.	1
61-72				
73-80				S30010

Card S30020: 2nd capacitor, C2.

1-12	12X			
13-24	12X			
25-36	F12.0	S1	Sign S of capacitance C.	1
37-48	F12.0	S2	Sign S of series resistance R_C.	1
49-60	F12.0	S3	Sign S of shunt resistance R_{CSH}.	-1
61-72				
73-80				S30020

Card S30030: 3rd capacitor, C3.

1-12	12X			
13-24	12X			
25-36	F12.0	S1	Sign S of capacitance C.	-1
37-48	F12.0	S2	Sign S of series resistance R_C.	-1
49-60	F12.0	S3	Sign S of shunt resistance R_{CSH}.	
61-72				
73-80				S30030

etc.

Cards S40010, S40020, S40030, ... (1 to 20 cards): Inductors.

These cards follow the last inductor card in the 40000 series.

Card S40010: 1st inductor, L1.

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	12X			
13-24	12X			
25-36	F12.0	S1	Sign S of Inductance L (or L_0 if L is nonlinear).	1
37-48	F12.0	S2	Sign S of series resistance R_L.	1
49-60	F12.0	S3	Sign S of shunt resistance R_{LSH}.	-1
61-72	F12.0	S4	Sign S of $1/I_{con}$; blank if L is linear.	-1
73-80				S40010

Card S40020: 2nd inductor, L2.

1-12	12X			
13-24	12X			
25-36	F12.0	S1	Sign S of Inductance L (or L_0 if L is nonlinear).	-1
37-48	F12.0	S2	Sign S of series resistance R_L.	-1
49-60	F12.0	S3	Sign S of shunt R_{LSH}.	1
61-72	F12.0	S4	Sign S of $1/I_{con}$; blank if L is linear.	1
73-80				S40020

Card S40030: 3rd inductor, L3.

1-12	12X			
13-24	12X			
25-36	F12.0	S1	Sign S of Inductance L (or L_0 if L is nonlinear).	
37-48	F12.0	S2	Sign S of series resistance R_L.	
49-60	F12.0	S3	Sign S of shunt resistance R_{LSH}.	
61-72	F12.0	S4	Sign S of $1/I_{con}$; blank if L is linear.	
73-80				S40030

etc.

Cards S50010, S50020, S50030, ... (1 to 20 cards): Zener diodes.

These cards follow the last zener-diode card in the 50000 series.

Card S50010: 1st zener diode, Z1.

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	12X			
13-24	12X			
25-36	F12.0	S1	Sign S of high zener resistance R_z.	1
37-48	F12.0	S2	Sign S of low back zener resistance R_{zb}.	-1
49-60	F12.0	S3	Sign S of back zener voltage E_{zb}.	-1
61-72				
73-80				S50010

Card S50020: 2nd zener diode, Z2.

1-12	12X			
13-24	12X			
25-36	F12.0	S1	Sign S of high zener resistance R_z.	-1
37-48	F12.0	S2	Sign S of low back zener resistance R_{zb}.	1
49-60	F12.0	S3	Sign S of back zener voltage E_{zb}.	1
61-72				
73-80				S50020

Card S50030: 3rd zener diode, Z3.

1-12	12X			
13-24	12X			
25-36	F12.0	S1	Sign S of high zener resistance R_z.	
37-48	F12.0	S2	Sign S of low back zener resistance R_{zb}.	
49-60	F12.0	S3	Sign S of back zener voltage E_{zb}.	
61-72				
73-80				S50030

etc.

Cards S60011, S60021, S60031, ... (1 to 20 cards): Diodes.

These cards follow the last diode card in the 60000 series.

Card S60011: 1st diode, D1.

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	F12.0	S1	Sign S of I_{sd}.	
13-24	F12.0	S2	Sign S of m_d.	
25-36	F12.0	S3	Sign S of $R_{\ell d}$.	
37-48	F12.0	S4	Sign S of C_{j0d}.	
49-60	F12.0	S5	Sign S of $V_{\varphi d}$.	
61-72	F12.0	S7	Sign S of T_d.	
73-80				S60011

Card S60021: 2nd diode, D2.

1-12	F12.0	S1	Sign S of I_{sd}.	
13-24	F12.0	S2	Sign S of m_d.	
25-36	F12.0	S3	Sign S of $R_{\ell d}$.	
37-48	F12.0	S4	Sign S of C_{j0d}.	
49-60	F12.0	S5	Sign S of $V_{\varphi d}$.	
61-72	F12.0	S7	Sign S of T_d.	
73-80				S60021

Card S60031: 3rd diode, D3.

1-12	F12.0	S1	Sign S of I_{sd}.	
13-24	F12.0	S2	Sign S of m_d.	
25-36	F12.0	S3	Sign S of $R_{\ell d}$.	
37-48	F12.0	S4	Sign S of C_{j0d}.	
49-60	F12.0	S5	Sign S of $V_{\varphi d}$.	
61-72	F12.0	S7	Sign S of T_d.	
73-80				S60031

etc.

Cards S70011, S70012, S70013, S70021, ... (3 to 60 cards): Transistors.

These cards follow the last transistor card in the 70000 series.

Cards S70011, S70012, and S70013: 1st transistor, T1.

Card S70011

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	F12.0	ESSES(1)	Sign S of β_n.	1
13-24	F12.0	ESSES(2)	Sign S of β_i.	-1
25-36	F12.0	ESSES(3)	Sign S of T_e.	1
37-48	F12.0	ESSES(4)	Sign S of T_c.	-1
49-60	F12.0	ESSES(5)	Sign S of I_{sc}.	1
61-72	F12.0	ESSES(6)	Sign S of m_c.	-1
73-80				S70011

Card S70012

1-12	F12.0	ESSES(7)	Sign S of C_{j0c}.	-1
13-24	F12.0	ESSES(8)	Sign S of $V_{\varphi c}$.	-1
25-36	F12.0	ESSES(9)	Sign S of $R_{\ell c}$.	-1
37-48	F12.0	ESSES(10)	Sign S of I_{se}.	-1
49-60	F12.0	ESSES(11)	Sign S of m_e.	1
61-72	F12.0	ESSES(12)	Sign S of C_{j0e}.	-1
73-80				S70012

Card S70013

1-12	F12.0	ESSES(13)	Sign S of $V_{\varphi e}$.	1
13-24	F12.0	ESSES(14)	Sign S of $R_{\ell e}$.	-1
25-36				
37-48				
49-60				
61-72				
73-80				S70013

Cards S70021, S70022, and S70023: 2nd transistor, T2.

Card S70021

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	F12.0	ESSES(1)	Sign S of β_n.	-1
13-24	F12.0	ESSES(2)	Sign S of β_i.	1
25-36	F12.0	ESSES(3)	Sign S of T_e.	
37-48	F12.0	ESSES(4)	Sign S of T_c.	1
49-60	F12.0	ESSES(5)	Sign S of I_{sc}.	-1
61-72	F12.0	ESSES(6)	Sign S of m_c.	1
73-80				S70021

Card S70022

1-12				1
13-24	F12.0	ESSES(7)	Sign S of C_{j0c}.	1
25-36	F12.0	ESSES(8)	Sign S of $V_{\varphi c}$.	1
37-48	F12.0	ESSES(9)	Sign S of $R_{\ell c}$.	1
49-60	F12.0	ESSES(10)	Sign S of I_{se}.	
61-72	F12.0	ESSES(11)	Sign S of m_e.	
73-80		ESSES(12)	Sign S of C_{j0e}.	S70022

Card S70023

1-12	F12.0	ESSES(13)	Sign S of $V_{\varphi e}$.	-1
13-24	F12.0	ESSES(14)	Sign S of $R_{\ell e}$.	1
25-36				
37-48				
49-60				
61-72				
73-80				S70023

etc.

(1) ϕ_d(F) and ϕ_p(F) Input Core Parameters (PARIN = PHIF or blank).

Cards S80120 through S80124, S80220 through S80224, ... (5 to 100 cards): Magnetic cores.

These cards follow the last core card in the 80000 series.

Cards S80120 through S80124: 1st magnetic core, Core 1.

Card S80120

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	F12.0	ESSES(1)	Sign S of outside diameter.	1
13-24	F12.0	ESSES(2)	Sign S of inside diameter.	1
25-36	F12.0	ESSES(3)	Sign S of saturation flux ϕ_s.	-1
37-48	F12.0	ESSES(4)	Sign S of maximum remanent flux ϕ_r.	-1
49-60				
61-72				
73-80				S80120

Card S80121

1-12	F12.0	ESSES(5)	Sign S of threshold mmf F_{d1}; blank for Type D.	1
13-24	F12.0	ESSES(6)	Sign S of F_{dz}; blank for Types A, D, and E.	1
25-36	F12.0	ESSES(7)	Sign S of F_{d2}; blank for Types A, D, and E.	1
37-48	F12.0	ESSES(8)	Sign S of F_{d3}; blank for Types B, D, and F.	1
49-60	F12.0	ESSES(9)	Sign S of ϕ_{d2}; blank for Types A, D, E, and F.	-1
61-72				
73-80				S80121

Card S80122

1-12	F12.0	ESSES(10)	Sign S of H_a.	1
13-24	F12.0	ESSES(11)	Sign S of H_q; blank for Types C, E, and F.	1
25-36	F12.0	ESSES(12)	Sign S of H_n; blank for Types C, E, and F.	1
37-48				
49-60				
61-72				
73-80				S80122

Card S80123

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	F12.0	ESSES(13)	Sign S of λ_d.	-1
13-24	F12.0	ESSES(14)	Sign S of ν_d.	-1
25-36	F12.0	ESSES(15)	Sign S of F_{dB}.	1
37-48	F12.0	ESSES(16)	Sign S of F_0''.	1
49-60	F12.0	ESSES(17)	Sign S of λ.	-1
61-72	F12.0	ESSES(18)	Sign S of ν.	-1
73-80				S80123

Card S80124

1-12	F12.0	ESSES(19)	Sign S of F_B.	1
13-24	F12.0	ESSES(20)	Sign S of F_0.	1
25-36	F12.0	ESSES(21)	Sign S of ρ_p.	-1
37-48	F12.0	ESSES(22)	Sign S of F_{B1}; blank if no Region 5.	
49-60	F12.0	ESSES(23)	Sign S of F_{01}; blank if no Region 5.	
61-72	F12.0	ESSES(24)	Sign S of ρ_{p1}; blank if no Region 5.	
73-80				S80124

Cards S80220 through S80224: 2nd magnetic core, Core 2.

Card S80220

1-12	F12.0	ESSES(1)	Sign S of outside diameter.	1
13-24	F12.0	ESSES(2)	Sign S of inside diameter.	1
25-36	F12.0	ESSES(3)	Sign S of saturation flux ϕ_s.	-1
37-48	F12.0	ESSES(4)	Sign S of maximum remanent flux ϕ_r.	-1
49-60				
61-72				
73-80				S80220

etc.

(2) Static and Dynamic B(H)-Loop Input Core Parameters (PARIN = BH)

Cards S80120 through S80122, S80130, S80220 through S80222, S80230, ... : Magnetic cores.

These cards follow the last core card in the 80000 series.

Cards S80120 through S80122, S80130: 1st magnetic core, Core 1.

Card S80120

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	F12.0	ESSES(1)	Sign S of outside diameter.	1
13-24	F12.0	ESSES(2)	Sign S of inside diameter.	1
25-36	F12.0	ESSES(3)	Sign S of effective cross-sectional area.	1
37-48	F12.0	ESSES(4)	Sign S of maximum remanent flux density B_r.	-1
49-60				
61-72				
73-80				S80120

Card S80121

1-12	F12.0	ESSES(5)	Sign S of threshold H_{d1}; blank for Type D.	1
13-24	F12.0	ESSES(6)	Sign S of H_{dz}; blank for Types A, D, and E.	1
25-36	F12.0	ESSES(7)	Sign S of H_{d2}; blank for Types A, D, and E.	1
37-48	F12.0	ESSES(8)	Sign S of H_{d3}; blank for Types B and F.	1
49-60	F12.0	ESSES(9)	Sign S of H_m; blank for Types C, E, and F.	1
61-72				
73-80				S80121

Card S80122

1-12	F12.0	ESSES(10)	Sign S of B_{d2}; blank for Types A, D, E, and F.	-1
13-24	F12.0	ESSES(11)	Sign S of B_{d3}; blank for Types B, C, E, and F.	-1
25-36	F12.0	ESSES(12)	Sign S of B_m; blank for Types C, E, and F.	-1
37-48				
49-60				
61-72				
73-80				S80122

Card S80130

COLUMNS	FORMAT	PROGRAM NAME	DESCRIPTION	EXAMPLE
1-12	12X			
13-24	F12.0	ESSES(13)	Sign S of values of H @ B = 0 for all frequencies.	1
25-36				
37-48				
49-60				
61-72				
73-80				S80130

Cards S80220 through S80222, S80230: 2nd magnetic core, Core 2.

Card S80220

1-12	F12.0	ESSES(1)	Sign S of outside diameter.	1
13-24	F12.0	ESSES(2)	Sign S of inside diameter.	1
25-36	F12.0	ESSES(3)	Sign S of effective cross-sectional area.	1
37-48	F12.0	ESSES(4)	Sign S of maximum remanent flux density B_r.	-1
49-60				
61-72				
73-80				S80220

Card S80221

1-12	F12.0	ESSES(5)	Sign S of threshold H_{d1}; blank for Type D.	1
13-24	F12.0	ESSES(6)	Sign S of H_{dz}; blank for Types A, D, and E.	1
25-36	F12.0	ESSES(7)	Sign S of H_{d2}; blank for Types A, D, and E.	1
37-48	F12.0	ESSES(8)	Sign S of H_{d3}; blank for Types B and F.	1
49-60	F12.0	ESSES(9)	Sign S of H_m; blank for Types C, E, and F.	1
61-72				
73-80				S80221

etc.

II GLOSSARY OF VARIABLES AND PROGRAM LISTING

A. Glossary of Variables

The glossary of the variables in the MTRAC program is divided into two parts, one listing the COMMON variables and the other listing the local variables. Each list is presented alphabetically. Array sizes are indicated by numbers in parentheses.

1. MTRAC's COMMON Variables

AAAA(10)	Alphanumeric names of unknowns to be plotted and/or print-plotted.
ABC	Convergence indicator: 0 - convergence achieved; 1 - convergence not achieved yet; -3 - convergence not achieved in ITRMAX iterations.
ABSER	Largest absolute error allowed in the solution of initial conditions.
ABSERR	Largest absolute error allowed in the transient solution of core-winding currents.
ABTEMP	Absolute temperature (degrees Kelvin).
AF(20)	α_n of a transistor.
AR(20)	α_i of a transistor.
BCDT	Name for X-axis in plot.
BCDT1	Name for Y-axis in plot.
BCI(20)	Base-collector current, $i_{fc<-1>}$.
BCR(20)	Slope of i_{fc} *vs.* V_c at the previous iteration, $(i_{fc<-1>} + I_{sc})/\theta_{mc}$.
BCV(20)	Base-collector voltage, $V_{c<-1>}$.
BEI(20)	Base-emitter current, $i_{fe<-1>}$.
BER(20)	Slope of i_{fe} *vs.* V_e at the previous iteration, $(i_{fe<-1>} + I_{se})/\theta_{me}$.
BEV(20)	Base-emitter voltage, $V_{e<-1>}$.
BVAL(6,6)	Boundary value specified for normal termination.

BVALF(6,6)	Boundary value specified for circuit-failure termination.
CAPC(20)	Capacitance of a capacitor.
CC1(20)	$c_{1c} = 0.9\ V_{\varphi c}$ of a transistor.
CC2(20)	$c_{2c} = 1/\theta_{mc} = 11,590/(Tm_c)$ of a transistor.
CC3(20)	$c_{3c} = 1/R\ell_c$ of a transistor.
CC4(20)	$c_{4c} = 1/V_{\varphi c}$ of a transistor.
CDT1(20)	$d_{t1c} = T_c/\Delta t$ of a transistor.
CDT2(20)	$d_{t2c} = 1 + (T_c/\Delta t)$ of a transistor.
CDT3(20)	$d_{t3c} = C_{j0c}/\Delta t$ of a transistor.
CIBASE(20)	Base current.
CIC(20)	Capacitor current.
CICOLC(20)	Collector current of a transistor.
CICORE(20,10)	Core-winding current.
CICOR1(20,10)	Core-winding current at the previous Δt.
CIDIOD(20)	Diode current.
CIEMTR(20)	Emitter current of a transistor.
CIVS(20)	Current through a current source or a voltage source.
CIL(20)	Inductor current.
CIR(40)	Resistor current.
CIZ(20)	Zener-diode current.
CT(11,2)	CT(J,1)-maximum Δt for the Jth period; CT(J,2)-termination time of the Jth period.
C1(20)	$c_1 = 0.9\ V_{\varphi d}$ of a diode.
C2(20)	$c_2 = 1/\theta_{md} = 11,590/(Tm_d)$ of a diode.
C3(20)	$c_3 = 1/R\ell_d$ of a diode.
C4(20)	$c_4 = 1/V_{\varphi d}$ of a diode.
DELT	Time step, Δt.
D1(20)	Diode current, $i_{fd<-1>}$.
DMAX	Maximum deviation (in percent) of parameter values.
DR(20)	Slope of i_{fd} *vs.* V_d at the previous iteration, $(i_{fd<-1>} + I_{sd})/\theta_{md}$.
DTM1	$\Delta t_{(-1)}$.

DTM2	$\Delta t_{(-2)}$.
DT1(20)	$d_{t1} = T_d/\Delta t$ of a diode.
DT2(20)	$d_{t2} = 1 + (T_d/\Delta t)$ of a diode.
DT3(20)	$d_{t3} = C_{j0d}/\Delta t$ of a diode.
DV(20)	$V_{d<-1>}$ of a diode.
E(19,16,2)	Data of the Jth time-variable source at the Lth break point; E(J,L,1) = Source magnitude; E(J,L,2) = Break-point time. J = 1 for photo current; J = K + 1 for Kth source.
EC1(20)	$c_{1e} = 0.9\ V_{\varphi e}$ of a transistor.
EC2(20)	$c_{2e} = 1/\theta_{me} = 11{,}590/(Tm_e)$ of a transistor.
EC3(20)	$c_{3e} = 1/R\ell_e$ of a transistor.
EC4(20)	$c_{4e} = 1/V_{\varphi e}$ of a transistor.
EDT1(20)	$d_{t1e} = T_e/\Delta t$ of a transistor.
EDT2(20)	$d_{t2e} = 1 + (T_e/\Delta t)$ of a transistor.
EDT3(20)	$d_{t3e} = C_{j0e}/\Delta t$ of a transistor.
E1(20)	Present value of a time-variable source.
EP1(20)	Value of a time-variable source at the previous Δt.
EZB(20)	Extrapolated back zener voltage E_{zb}.
E2(19)	Present slope of a time-variable source.
F(20)	MMF of a core.
FB(20)	F_B of a core.
FB1(20)	F_{B1} of a core.
FDB(20)	F_{dB} of a core.
FDZ(20)	F_{dz} of a core.
FD1(20)	Threshold F_{d1} of a core.
FD2(20)	F_{d2} of a core.
FD3(20)	F_{d3} of a core.
FD4(20)	F_{d4} of a core.
FM1(20)	$F_{(-1)}$ of a core.
FM2(20)	$F_{(-2)}$ of a core.
FM3(20)	$F_{(-3)}$ of a core.

F0(20)	F_0 of a core.
F0PP(20)	F_0'' of a core.
F01(20)	F_{01} of a core.
H(60,60)	Conductance matrix of the circuit.
IAC(6)	Number of AND conditions of a normal-termination statement.
IACF(6)	Number of AND conditions of a circuit-failure-termination statement.
IBC	Control set in Subroutine ELEM: 0 - fill H and T arrays [Part (1)]; 1 - compute currents and test convergence [Part (2)].
ICS(6,6)	Indicator of normal-termination condition: 1 if V( ) $\leq$; 2 if V( ) >; -1 if |V( )| $\leq$; -2 if |V( )| >.
ICSF(6,6)	Indicator of circuit-failure-termination condition: 1 if V( ) $\leq$; 2 if V( ) >; -1 if |V( )| $\leq$; -2 if |V( )| >.
IGRAPH	Count of graphs plotted.
INCIL(20)	1 - initial inductor current CIL(N) = CIL0 regardless of NSVVAR or NSVIL; 0 - otherwise.
INDL(20)	Inductance of an inductor.
IOC	Number of OR statements describing normal-termination conditions.
IOCF	Number of OR statements describing circuit-failure-termination conditions.
IPX	Number of points to be plotted per frame.
ITRMAX	Maximum number of iterations above which variable values are reset and Δt is divided by 4.
ITRTST	Number of iterations below which Δt is multiplied by 2 (if not exceeding TI) and above which Δt is divided by 2.
JX(5)	Index number of a break point for a time-variable resistor.
JCAT	Index number of a period for which Δt_{max} is specified.
JG(19)	Index number of a variable-source mode.
JKKJ	Solution of initial conditions: 0 - converged; 1 - did not converge.
JKLJ	Number of initial-condition iterations.

JSJ	Control indicator: 0 – initially; 1 – prior to call on Subroutine INTIAL; -1 – throughout the transient solution.
JV	Present number of stored plot points.
KCAP(20)	Capacitor nodal voltages indicator: 1 – unknown; 2 – $V_b = V_{sb}$; 3 – $V_a = V_{sa}$.
KCORE(20,10)	Core-winding nodal voltages indicator: 1 – unknown; 2 – $V_b = V_{sb}$; 3 – $V_a = V_{sa}$.
KDIOD(20)	Diode nodal voltages indicator: 1 – unknown; 2 – $V_b = V_{sb}$; 3 – $V_a = V_{sa}$.
KFAIL	Number of variable resettings with no convergence before ITRMAX is changed.
KFMAX	Maximum number of trials before ITRMAX is changed (doubled if KFAIL = KFMAX; halved if KONVRG = KFMAX).
KIND(20)	Inductor nodal voltages indicator: 1 – unknown; 2 – $V_b = V_{sb}$; 3 – $V_a = V_{sa}$.
KIVS(20)	Indicator of nodal voltages of current and voltage sources: 1 – unknown; 2 – $V_b = V_{sb}$; 3 – $V_a = V_{sa}$.
KONT	1 – continued run; 0 – otherwise.
KONVRG	Count of successive time-step solutions each with less than ITRTST iterations.
KPFIT	Dependency of scales and boundaries of plot frames on those of first frame: 0 – independent; 1 – X independent; Y fixed by first frame; 2 – fixed by first frame except for independent X boundaries; 3 – fixed by first frame.
KPLOT	1 – plot at the end of present mode; 0 – otherwise.
KPRPLT	1 – print-plot at the end of present mode; 0 – otherwise (KPRPLT is set to 1 if JV > IPX or CPTIME > TLIMIT or NAW > 5).
KPUNCH	Set to 1 when the processor time of a mode run exceeds the time limit, in order to punch cards for a continued run; 0 – otherwise.
KRES(40)	Resistor nodal voltages indicator: 1 – unknown; 2 – $V_b = V_{sb}$; 3 – $V_a = V_{sa}$.
KTRAN(20)	Transistor nodal voltages indicator: 1 – unknown; 2 – $V_b = V_{sb}$; 3 – $V_c = V_{sc}$; 4 – $V_e = V_{se}$.
KXT	1 – if PLAPR(KEND) has been called with KEND = 0; 0 – otherwise.

KZEN(20)	Zener-diode nodal voltages indicator: 1 – unknown; 2 – $V_b = V_{sb}$; 3 – $V_a = V_{sa}$.
K1CAP(20)	Index number of Terminal *a* of a capacitor.
K1CORE(20,10)	Index number of Terminal *a* of a core winding.
K1DIOD(20)	Index number of Terminal *a* of a diode.
K1IND(20)	Index number of Terminal *a* of an inductor.
K1IVS(20)	Index number of Terminal *a* of a current or voltage source.
K1RES(40)	Index number of Terminal *a* of a resistor.
K1TRAN(20)	Transistor type: 0 – *npn*; 1 – *pnp*.
K1ZEN(20)	Index number of Terminal *a* of a zener diode.
K2CAP(20)	Index number of Terminal *b* of a capacitor.
K2CORE(20,10)	Index number of Terminal *b* of a core winding.
K2DIOD(20)	Index number of Terminal *b* of a diode.
K2IND(20)	Index number of Terminal *b* of an inductor.
K2IVS(20)	Index number of Terminal *b* of a current or voltage source.
K2RES(40)	Index number of Terminal *b* of a resistor.
K2TRAN(20)	Index number of Terminal *B* (Base) of a transistor.
K2ZEN(20)	Index number of Terminal *b* of a zener diode.
K3IVS(20)	Index number of a time-variable source (0 if *dc*).
K3TRAN(20)	Index number of Terminal *C* (Collector) of a transistor.
K4IVS(20)	Type of a variable source: 1 – current; 0 – voltage.
K4TRAN(20)	Index number of Terminal *E* (Emitter) of a transistor.
LASTIC	1 – elastic ϕ is included; 0 – elastic ϕ is neglected.
MAXMIN	Initial (lowest) maximum number of iterations above which the variable values are reset and Δt is divided by 4.
MONC	Logical control of MONITOR print-out in Subroutine CAP
MONCOR	Logical control of MONITOR print-out in Subroutines RWCORF, RWCORH, and CORE.
MOND	Logical control of MONITOR print-out in Subroutine DIODE.
MONELM	Logical control of MONITOR print-out in Subroutine ELEM.
MONIV	Logical control of MONITOR print-out in Subroutine IVS.

MONL	Logical control of MONITOR print-out in Subroutines RWIND and IND.
MONMA	Logical control of MONITOR print-out in the main program and Subroutines RWRUNC and INTIAL..
MONR	Logical control of MONITOR print-out in Subroutine RES.
MONS	Logical control of MONITOR print-out in Subroutine AUX(K).
MONT	Logical control of MONITOR print-out in Subroutines RWTRAN and TRAN.
MONTOR	Logical indicator of whether or not NDTMIN < NEDELT $\leq$ NDTMAX.
MONZEN	Logical control of MONITOR print-out in Subroutine ZEN.
MVALUE(14)	Logical control values specifying in which subprograms MONITOR print-out will be executed if NDTMIN < NDELT $\leq$ NDTMAX.
NAW	Number of times variables have been reset and Δt divided by 4 due to convergence failure. For each Δt, if NAW = 0 at the first iteration, then NAW = 1 in the following iterations until convergence is achieved.
NB	Ratio of computed points to plotted points.
NBB	Ratio of plotted points to printed points.
NC	Number of capacitors.
NCORE	Number of cores.
ND	Number of variable sources.
NDELT	Index number of the current time step, Δt.
NDIODE	Number of diodes.
NDTMAX	Value of NDELT beyond which MONITOR print-out stops.
NDTMIN	Value of NDELT before MONITOR print-out starts.
ND1	ND-1.
NG	Initial variable values: -1 - compute; 0 - zero; 1 - read in.
NITER	Iteration count during a given time step.
NITRSM	Total count of transient-solution iterations.
NIV	Number of current or voltage sources.
NL	Number of inductors.

NOEXIT	Control indicator: 1 – until inelastic flux switching begins, thus preventing stop-switching exit; 0 – otherwise.
NOVAR(6,6)	Index number of a variable whose value is tested in a condition for normal termination.
NOVARF(6,6)	Index number of a variable whose value is tested in a condition for circuit-failure termination.
NP(20,10)	Number of identically switching cores in parallel represented by a single core across a given winding.
NPLTVT	Number of variables to be plotted.
NPLTVV	Number of variable-*vs.*-variable plots.
NPP(10)	Index numbers of variables to be plotted (if negative, then no V-*vs.*-*t* plot).
NP1(10,2)	Index numbers of variables plotted *vs.* each other: NP1(J,1) for V_x; NP1(J,2) for V_y.
NR	Number of resistors.
NRP(40)	Number of break points if *R* varies with time.
NS(20,10)	Number of identically switching cores in series represented by a single core across a given winding.
NSVIL	Indicates whether the present mode will initially retain the inductor currents of the previous mode: 1 – yes; 0 – no.
NSVPHI	Indicates whether the present mode will initially retain all magnetic flux values of the previous mode: 1 – yes; 0 – no.
NSVVAR	Indicates whether the present mode will initially retain all the final time variables of the previous mode: 1 – yes; 0 – no.
NTRANS	Number of transistors.
NV	Number of unknown nodal voltages in the circuit.
NW(20)	Number of windings of a core.
NZ	Total number of unknown nodal voltages and auxiliary unknowns.
NZEN	Number of zener diodes.
N1B	1 – include phase-plane plots; 0 – otherwise.
N3B	-1 – solve initial conditions only; 0 – halt if error in initial condition solution; 1 – print if error in initial condition solution but proceed with transient solution.

PDTEM1(20)	$\dot{\phi}_{\epsilon(-1)}$ of a core.
PHDTE(20)	$\dot{\phi}_{\epsilon}$ of a core.
PHDTM1(20)	$\dot{\phi}_{(-1)}$ of a core.
PHDTM2(20)	$\dot{\phi}_{(-2)}$ of a core.
PHI(20)	ϕ of a core.
PHID(20)	ϕ_d of a core.
PHIDOT(20)	$\dot{\phi}$ of a core.
PHIM1(20)	$\phi_{(-1)}$ of a core.
PHIM2(20)	$\phi_{(-2)}$ of a core.
PHIM3(20)	$\phi_{(-3)}$ of a core.
PHIR(20)	ϕ_r of a core.
PHIS(20)	ϕ_s of a core.
PHI0(20)	Initial ϕ of a core.
PPIC(20)	Collector-base photocurrent.
PPIC1(20)	Collector-base photocurrent at the previous time step.
PPID(20)	Diode photocurrent.
PPID1(20)	Diode photocurrent at the previous time step.
PPIE(20)	Emitter-base photocurrent.
PPIE1(20)	Emitter-base photocurrent at the previous time step.
PRSYMB	A printer symbol used to print-plot specified output.
PSTEP	Weight of previous-iteration results on computed winding current of a core.
PWCR(20)	Dissipated power in a diode.
PWTR(20)	Dissipated power in a transistor.
P1(20)	Auxiliary core parameter p_1.
P2(20)	Auxiliary core parameter p_2.
P3(20)	Auxiliary core parameter p_3.
P4(20)	Auxiliary core parameter p_4.
P5(20)	Auxiliary core parameter p_5.
P6(20)	Auxiliary core parameter p_6.
P7(20)	Auxiliary core parameter p_7.

P8(20)	Auxiliary core parameter p_8.
P9(20)	Auxiliary core parameter p_9.
P10(20)	Auxiliary core parameter p_{10}.
P11(20)	Auxiliary core parameter p_{11}.
P12(20)	Auxiliary core parameter p_{12}.
P13(20)	Auxiliary core parameter p_{13}.
P14(20)	Auxiliary core parameter p_{14}.
P15(20)	Auxiliary core parameter p_{15}.
P16(20)	Auxiliary core parameter p_{16}.
P17(20)	Auxiliary core parameter p_{17}.
P18(20)	Auxiliary core parameter p_{18}.
P19(20)	Auxiliary core parameter p_{19}.
P20(20)	Auxiliary core parameter p_{20}.
QCR(20,7)	Diode parameters: I_{sd}, m_d, $R\ell_d$, C_{j0d}, $V_{\varphi d}$, I_{pp}, T_d.
QTAN(20,16)	Transistor parameters: β_n, β_i, T_e, T_c, I_{sc}, m_c, C_{j0c}, $V_{\varphi c}$, $R\ell_c$, I_{se}, m_e, C_{j0e}, $V_{\varphi e}$, $R\ell_e$, I_{ppc}, I_{ppe}.
R(40)	Resistance of a resistor.
RC(20)	Series resistance of a capacitor.
RCSH(20)	Shunt resistance of a capacitor.
RELER	Relative error allowed in a solution of initial conditions.
RELERR	Relative error allowed in a transient solution of core-winding currents.
RF(5,5)	R_f of a resistor whose resistance varies with time.
RL(20)	Series resistance of an inductor.
RLSH(20)	Shunt resistance of an inductor.
RMAG	I_{pp} scale factor = E(1,1,2).
RNUDT	Number of Δt's above which STSWEX is effective.
ROP(20)	ρ_p of a core.
ROP1(20)	ρ_{p1} of a core.
RVS(20)	Series resistance of a voltage source.
RW(20,10)	Resistance of a core winding.

RZ(20)	High resistance R_z of a zener diode.
RZB(20)	Low back resistance R_{zb} of a zener diode.
S(20)	Inductance value of an inductor.
SAVEXL(10,3)	Values of XL in the previous call on plot routines.
SAVEXR(10,3)	Values of XR in the previous call on plot routines.
SAVEYB(10,3)	Values of YB in the previous call on plot routines.
SAVEYT(10,3)	Values of YT in the previous call on plot routines.
ST(19,2)	Variable-source data. ST(J,1) = NI = number of break points (1 if *dc*); ST(J,2) indicates type: -1 - *dc*; 0 - hold last value; 1 - cycle.
STARTS	Central processor time at the beginning of each mode run.
STSWEX	1 - terminate mode run if flux switching of all cores is completed and $t >$ NUDT $\cdot \Delta t$; 0 - otherwise.
T(60)	Current matrix of a circuit.
TE	Time.
TF(354)	Storage of the X-axis values (usually time) for plotting.
TFR(5,5)	t_f of a resistor whose resistance varies with time.
TI	Maximum Δt, CT(JCAT,1).
TIMEIN	Time at the beginning of a mode (used to prevent stop-switching exit if TE < TIMEIN + NUDT · DELT).
TITLE(12)	Circuit name, saved for plot labeling.
TLIMIT	Maximum central processor time of a mode run before results are print-plotted, continued-run cards are punched, and the run terminates.
TOO(19)	Overall time of previously completed cycles.
TOOLD	Value of Δt at the last time Δt-dependent diode terms were set.
TOI	Previous Δt; controls tasks in Subroutine ELEM: > 0 - Part (1), to fill H and T arrays; < 0 - Part (2), to compute currents and test convergence.
TTOOLD	Value of Δt at the last time Δt-dependent transistor terms were set.
V(70)	Unknown nodal voltages and auxiliary unknowns.
VI(20)	Magnitude of a *dc* source in Subroutines RWIVS and IVS.

VICON(20)	$1/I_{con}$ of a nonlinear inductor (zero for a linear inductor).
VP(10,350)	Values of variables stored for plotting.
VZB(20)	Reverse boundary-value V_{zb} of a zener diode.
VZF(20)	Forward boundary-value V_{zf} of a zener diode.
V1(70)	$V_{(-1)}$.
V2(70)	$V_{(-2)}$.
WSQSOL	Logical control of MONITOR print-out in Subroutine SEQSOL.
XL	X value of the leftmost plotted point.
XLAMDA(20)	λ of a core.
XLEFT(20)	Left boundary of plot frame (in mks units).
XLMDAD(20)	λ_d of a core.
XNT(20,10)	Number of turns in a core winding.
XNU(20)	ν of a core.
XNUD(20)	ν_d of a core.
XP(713)	Y values for points to be plotted; set up in Subroutine PLAPR from Array VP(10,350).
XR	X value of the rightmost plotted point.
XRES(5,5)	Exponent p of a resistor whose resistance varies with time.
XRIGHT(20)	Right boundary of plot frame (in mks units).
XSCALE(20)	Scale of X variable (in mks units/inch).
XUNIT(2,20)	Units of the X variable.
YB	Y value of next to the bottom-most plot point.
YBOTTM(20)	Bottom boundary of plot frame (in mks units)
YSCALE(20)	Scale of Y variable (in mks units/inch).
YT	Y value of the next-to-the-topmost plot point.
YTOP(20)	Top boundary of plot frame (in mks units).
YUNIT(2,20)	Units of Y variable.
ZV(20)	Negligible $\dot{\phi}$ value of a core.

2. MTRAC's Local Variables

The following local variables have the same meaning in several READ-WRITE and computation circuit-element subroutines:

CC	Carriage control character for printing: blank - single space; 0 - double space.
F1	Type of Terminal a: blank - floating; S - Source, G - Ground.
F3	Alphanumeric blank = 6Hbbbbbb, where b represents a blank.
KO	Temporary storage for K(N): 1 - V_a and V_b are unknown; 2 - $V_a = V_{sa}$; 3 - $V_b = V_{sb}$.
KO1	Temporary storage for the index number of Terminal a (also Terminal B of a transistor base).
KO2	Temporary storage for the index number of Terminal b (also Terminal C of a transistor collector).
N	Index number of a circuit element.
R2	Type of Terminal b: blank - floating; S - Source; G - Ground.

The glossary of local variables is given as follows for each subprogram separately.

(1) Main Program

DTIM	Length of a cycle of a mode time (value of CT(J-1,2) when CT(J,1) < 0).
ERR	Parameter of Subroutine SEQSOL which indicates a singular matrix if ERR ≠ 0.
FETCH(2)	Storage for name of MTRAC segment in memory.
GENDA1	Alphanumeric "PLOTS" (segment name).
GETOFF	Logical indicator which is set to *TRUE* if the run should terminate after plotting.
JH	Index number of the previous break point of a variable source [= JG(J)].
JJ	Index of inner loop during which one Δt solution is obtained. The loop will repeat NB times for each plot point.
JJJ	Index of outer loop during which NB plot values are stored. The loop will repeat NBB times for each print-out point.

KDTIM	0 - until end of first cycle; 1 - otherwise.
KNL	Total number of nonlinear circuit elements = 2 · NTRANS + NDIODE + $\sum_{I=1}^{NCORE}$ NW(I).
LABEL	A temporary storage value for monitoring.
NI	Number of break points of a variable source.
RWCOR1	Alphanumeric "RWCORS" (Segment name).
RWELM1	Alphanumeric "RWDATA" (Segment name).
SOFAR	A temporary storage logical variable.
SOLVE1	Alphanumeric "SOLVE" (Segment name).
TEMP	Temporary storage.
TERMNT	A temporary storage logical variable.
TE1	Difference between t and the next break-point time of a variable source (negative, except when starting a new mode).
TOK	Difference between t and the beginning time of a Δt-specification period, used to adjust t and Δt before starting a new period.
T1	Difference between t and the next break-point time of a variable source.

(2) Subroutine HOLLER(,)

No special local variables.

(3) Subroutine RWRUNC

A1	Alphanumeric for termination conditions: A - Absolute value; otherwise - algebraic value.
A2	Alphanumeric for termination conditions: G - >: otherwise - $\leq$.
CKT	Alphanumeric name (up to 6 characters) of a circuit or a circuit mode.
HOL	Alphanumeric blank = 1Hb, where b represents a blank.
HOLABS	Alphanumeric "ABC" = 3HABC.
HOLBLN	Alphanumeric triple blank = 3Hbbb, where b represents a blank.
HOLGT	Alphanumeric "GT" = 2HGT.

HOLLA	Alphanumeric "A" = 1HA.
HOLLE	Alphanumeric "LE" = 2HLE.
HOLLG	Alphanumeric "G" = 1HG.
HOL0	Alphanumeric zero = 1H0.
LIMITT	Processor time limit (in seconds).
NAUX	Number of auxiliary variables (NZ = NV + NAUX).
NDTSPC	Number of Δt-specification periods.
NI	Number of break points of a variable source.
NI2	Indicator for variable sources: -1 - *dc*; 0 - hold last value; 1 - cycle.
Q1	Name of the *X* variable in plot.
Q2	Name of the *Y* variable in plot.
SCALE	Plot scales: 1 - specified by the user; 0 - determined on the basis of extreme values.
SOFAR	Logical variable used to decide whether the user is using any monitoring.
S1	Sign of worst-case deviation of values of a variable source.

(4) Subroutine PLAPR( )

I1	Subscript number of V( ) plotted as the *Y* value in a *V-vs.-V* plot.
I2	Subscript number of V( ) plotted as the *X* value in a *V-vs.-V* plot.
JVV	*JV*-2, the number of values in the phase-plane plot.
JV1	Subscript number of the *V* value being plotted *vs.* time.
KEND	-1 means plotting only, +1 means punch continued-run cards and plot, 0 means do entire subroutine.
SKIPFR	Logical variable which is *TRUE* if SKIPIT is *TRUE* and KPFIT = 2.
SKIPIT	Logical variable which is *TRUE* if the number of plotted points exceeds IPX, KPFIT > 0, and KONT = 0.
SKIPX	Logical variable which is *TRUE* if SKIPIT is *TRUE* and KPFIT = 1.
SKIPXY	Logical variable which is *TRUE* if SKIPIT is *TRUE* and KPFIT = 3.

TFJV	Recent time stored, TF(JV).
XLP	Smallest *X* value of a set of plot points.
XRP	Largest *X* value of a set of plot points.
YBP	Smallest *Y* value of a set of plot points (YB is the next-to-the-smallest value).
YTP	Largest *Y* value of a set of plot points (YT is the next-to-the-largest value).

(5) Subroutine PRPLT(, , ,)

AASS	Symbol used for data points in print plot (set equal to PRSYMB).
ABLANK	Hollerith character blank - 1H*b*, where *b* represents a blank.
ADOT	Hollerith character dot or period = 1H..
APLUS	Hollerith character "+" = 1H+; used in print-plot for those points which represent two or more data values.
C	Floating-point value corresponding to index of location in array G.
CSX	Scale for *X*-axis.
CSY	Scale for *Y*-axis.
DC	Value of a tic mark on the *Y*-axis.
G(81,4)	Storage for symbols to be print-plotted.
ID	Subscript number of the *V* value being plotted as the *Y* value.
II	Intermediate value in converting floating *X* value into column index.
IG	Index number of a value in the array *X*.
IL	Either zero or, in a V-*vs.*-V plot, the subscript number of the *V* value being plotted as the *X* value.
JI	Intermediate value in converting floating *Y* value into row index.
KI	Counter (0,1,2,3,4,5) used to indicate a line of the print-plot on which a *Y*-axis tic mark will occur and whose *Y* value will be printed.
NLC	-1 for V-*vs.*-*t* plot; 0 for phase-plane plot; +1 for V-*vs.*-V plot.
NN	Number of values to be plotted.

SX	Range of *X* values.
SY	Range of *Y* values.
TT	Inverse of scaling factor (temporary storage).
TX	Log_{10} (SX/7) + 1000, used to determine appropriate *X*-scaling.
TY	Log_{10} (SY/7) + 1000, used to determine appropriate *Y*-scaling.
X(14)	Array for storing the values represented by the tic marks on the *X*-axis.
Z	Fractional portion of TX or TY.

(6) Subroutine PLT(, , ,)*

BOTTOM	*Y* value of bottom of frame.
FIXX	Scale for *X*-axis.
FIXY	Scale for *Y*-axis.
IAR2(2)	Thirteen-character alphanumeric of plot units and a right parenthesis.
IBUF(1024)	1024-word buffer storage, used for plot tape.
ICOUNT	Number of tic marks on an axis.
IPC	Alphanumeric "PIC" = 3RPIC; used for scaling ($\times 10^{12}$) if units are PICO--.
ITEM	Temporary storage for Hollerith of first three letters of unit.
JV1	Subscript number of the *V* value being plotted as the *Y* value.
JV2	Either zero, or, in a V-*vs.*-V plot, the subscript number of the *V* value being plotted as the *X* value.
J2	-1 for V-*vs.* *t* plot, 0 for phase-plane plot, +1 for V-*vs.*-V plot.
MAKS	Alphanumeric for "MAX" = 3RMAX; used for scaling ($\times 10^{8}$) if units are *MAXWELLS*.
MIC	Alphanumeric for "MIC" = 3RMIC; used for scaling ($\times 10^{6}$) if units are MICRO--.

* Applicable at SRI computation center.

MIL	Alphanumeric for "MIL" = 3RMIL; used for scaling ($\times 10^3$) if units are MILLI--.
N	Number of values to be plotted.
NAN	Alphanumeric for "NAN" = 3RNAN; used for scaling ($\times 10^9$) if units are NANO--.
ONE	Value = 1.0.
PAREN1	Hollerith character for plotting left parenthesis = 1H(.
RATIO	Scaling factor based upon units specified by the user.
RIGHT	*X* value of right side of frame.
TAR1(2)	Name of plotted variable.
TAR4(15)	Storage of values of tic marks encoded for plotting.
TAR5(32)	Storage of values of tic marks before encoding.
TEMP	Temporary storage (e.g., value of final tic mark).
TEMP2	Temporary storage for various locations in plot.
TIC1	Value in inches of a tic mark.
TIME	Alphanumeric "TIME" = 6HTIME*bb*, where *b* represents a blank.
TMARK1	1 - if last plotted point is inside frame; 0 - otherwise.
TOP	*Y* value of top of frame.
VSPT	Hollerith 6H*bb*VS.*b*. where *b* represents a blank.
X	Temporary storage of numbers which will be used to label tic marks on *X*-axis.
XLEFT1	*X* value of left side of frame.
XMARK	Grid size (between tic marks) on the *X*-axis in specified units.
XTIC	Distance (in inch) between *X*-axis tic marks.
XVALUE	Equivalent inches of *X* value of a plot point.
XZERO	Equivalent inches of *X* = 0.0.
Y	Temporary storage of number which will be used to label tic marks on *Y*-axis.
YMARK	Grid size (between tic marks) on the *Y*-axis in specified units.
YTIC	Distance (in inch) between *Y*-axis tic marks.
YVALUE	Equivalent in inches of *Y* value of plot point.

YZERO	Equivalent in inches of $Y = 0.0$.
ZERO	Value = 0.0.

(7) Subroutine PAREN(,)*

A(10)	Set of masks to pick out one character from a word.
I	Output parameter giving the number of characters to be plotted.
IT	Temporary storage for a portion of a string of characters.
ITEM(2)	Array parameter in which string of characters is stored in order to put a parenthesis to the right of the rightmost nonblank character.
II	Storage for a single character out of a string; compared to see if character is blank.
PARENS	A set of ten characters (all right-parentheses).

(8) Subroutine RWIVS

A1	Type of Terminal *a*: blank - floating; *S* - Source; *G* - Ground.
A2	Type of Terminal *b*: blank - floating; *S* - Source; *G* - Ground.
S1	Sign of worst-case deviation of magnitude of a dc source.
S2	Sign of worst-case deviation of R_s of a voltage source.

(9) Subroutine RWRES

L	Index number of an $R(t)$ mode.
S1	Sign of worst-case deviation of resistance R (or R_f).

(10) Subroutine RWCAP

CI0	Initial capacitor current.
S1	Sign of worst-case deviation of capacitance C.
S2	Sign of worst-case deviation of series resistance R_C.
S3	Sign of worst-case deviation of shunt resistance R_{CSH}.

---

* Applicable at SRI computation center.

(11) Subroutine RWIND

CIL0	Initial inductor current.
S1	Sign of worst-case deviation of inductance L.
S2	Sign of worst-case deviation of series resistance R_L.
S3	Sign of worst-case deviation of shunt resistance R_{LSH}.
S4	Sign of worst-case deviation of $1/I_{con}$.

(12) Subroutine RWZEN

S1	Sign of worst-case deviation of R_z.
S2	Sign of worst-case deviation of R_{zb}.
S3	Sign of worst-case deviation of E_{zb}.

(13) Subroutine RWDIOD

K3	Diode parameters: 0 - specified; N - same as those of Nth diode.
QT	$11,590/T°K$.
S1	Sign of worst-case deviation of I_{sd}
S2	Sign of worst-case deviation of m_d.
S3	Sign of worst-case deviation of $R\ell_d$.
S4	Sign of worst-case deviation of C_{j0d}.
S5	Sign of worst-case deviation of $V_{\varphi d}$.
S7	Sign of worst-case deviation of T_d.

(14) Subroutine RWTRAN

BASE	Type of Terminal B (Base): blank - floating; S - Source; G - Ground.
COLC	Type of Terminal C (Collector): blank - floating; S - Source; G - Ground.
EMTR	Type of Terminal E (Emitter): blank - floating; S - Source; G - Ground.
ESSES(14)	Signs of worst-case deviations of transistor parameters.
K5	Transistor parameters: 0 - specified; N - same as those of Nth transistor.

NTO1	Counter (4,7,10,13...) used to print data of only three transistors per page.
QT	$11,590/T°K$.

(15) Subroutine RWCORE

PARBH	Alphanumeric "BH" = *6HBHbbbb*, where *b* represents a blank.
PARIN	Parameter input-data indicator: PHIF of blank - ϕF data (Call Subroutine RWCORF); BH - BH data (Call Subroutine RWCORH).

(16) Subroutine RWCORF

CI0	Initial current of a core winding.
CORMAT(20)	Core material; *T* - Tape-wound; otherwise - Ferrite.
ESSES (24)	Signs of worst-case deviations of core parameters.
F1C	Type of Terminal *a* (bottom): blank - floating; *S* Source; *G* - Ground.
HA(20)	H_a.
HAD	H_a (temporary storage).
HDMIN	H_d^{min}.
HN(20)	H_n.
HND	H_n (temporary storage).
HOL	Alphanumeric blank = 1H*b*, where *b* represents a blank.
HOLA	Alphanumeric "A" = 1HA.
HOLB	Alphanumeric "B" = 1HB.
HOLC	Alphanumeric "C" = 1HC
HOLD	Alphanumeric "D" = 1HD.
HOLE	Alphanumeric "E" = 1HE.
HOLF	Alphanumeric "F" = 1HF.
HOLG	Alphanumeric "G" = 1HG.
HQ(20)	H_q.
HQD	H_q (temporary storage).
HS	H_s.
IXNT	Number of turns of a winding (temporary storage).

KC1	Parameter values: 0 - specified; N - same as those of the Nth core.
M	Index number of a winding.
NUDT	Number of time steps above which the run is terminated if SWSTEX = 1 and flux switching is completed.
NWSUBN	Number of windings (temporary storage).
PHID1S	$\phi_{d1s} = \phi_d$ @ $F = F_{d1}$ in Region 1.
PHID2(20)	ϕ_{d2}.
PHID2D	ϕ_{d2} (temporary storage).
PHID3	ϕ_{d3}.
P0	$p_0 = (\phi_s + \phi_r)H_q/[(l_o - l_i)H_n]$
Q1	$p_1 \ell n[(F_{d1} - p_2)/(F_{d1} - p_3)]$.
R2C	Type of Terminal b (top): blank - floating; S - Source; G - Ground.
TYPE(20)	$\phi_d(F)$ - curve type: G, A, B, C, D, E, or F.
T1	Alphanumeric "NEGLEC" = 6HNEGLEC or "INCLUD" = 6HINCLUD.
T2	Alphanumeric "TED" = 6HTEDbbb or "ED" = 6HEDbbbb, where b represents a blank.
U0(20)	Initial value of ϕ/ϕ_r.
XID(20)	Inside core diameter (in inch).
XLI	Inside core circumference (in meter).
XLO	Outside core circumference (in meter).
XOD(20)	Outside core diameter (in inch).

(17) Subroutine RWCORH

A	Cross-sectional area of a core (in meter^2).
AIN(20)	Cross-sectional areas of a core (in cm^2)
BD2	B_{d2} (in weber/meter^2).
BD2IN(20)	B_{d2} (in gausses).
BD3	B_{d3} (in weber/meter^2).
BD3IN(20)	B_{d3} (in gausses).
BM	B_m (in weber/meter^2).
BMIN(20)	B_m (in gausses).

BR	B_r (in weber/meter^2).
BRIN(20)	B_r (in gausses).
BS	B_s (in weber/meter^2).
CI0	Initial current of a core winding.
CORMAT(20)	Core material: F - Ferrite; otherwise - Tape-wound.
EL	Average core circumference, $l = (l_i + l_o)/2$.
ESSES(12)	Signs of worst-case deviations of core parameters.
FREQ(20,10)	Frequency (in Hz) of a dynamic $B(H)$ curve.
F1C	Type of Terminal a (bottom): blank - floating; S - Source; G - Ground.
HAD	H_a (in ampere-turns/meter).
HDMIN	H_d^{min} (in ampere-turns/meter).
HDZ	H_{dz} (in ampere-turns/meter).
HDZIN(20)	H_{dz} (in oersted).
HD1	H_{d1} (in ampere-turns/meter).
HD1IN(20)	H_{d1} (in oersted).
HD2	H_{d2} (in ampere-turns/meter).
HD2IN(20)	H_{d2} (in oersted).
HD3	H_{d3} (in ampere-turns/meter).
HD3IN(20)	H_{d3} (in oersted).
HM	H_m (in ampere-turns/meter).
HMIN(20)	H_m (in oersted).
HND	H_n (in ampere-turns/meter).
HOL	Alphanumeric blank = 1Hb, where b represents a blank.
HOLA	Alphanumeric "A" = 1HA.
HOLB	Alphanumeric "B" = 1HB.
HOLC	Alphanumeric "C" = 1HC.
HOLD	Alphanumeric "D" = 1HD.
HOLE	Alphanumeric "E" = 1HE.
HOLF	Alphanumeric "F" = 1HF.
HOLG	Alphanumeric "G" = 1HG.
HOLT	Alphanumeric "T" = 1HT.

HQD	H_q (in ampere-turns/meter).
HS	H_s (in ampere-turns/meter).
H0B	H_{0B} (in ampere-turns/meter).
H0BIN(20,10)	H_{0B} (in oersted).
IXNT	Number of turns of a winding (temporary storage).
KC1	Parameter values: 0 - specified; N - same as those of the Nth core.
M	Index number of a winding.
NDYNP(20)	Number of dynamic $B(H)$ curves of different frequencies.
NDYNPT	Temporary storage of NDYNP(N).
NUDT	Number of time steps above which the run is terminated if STSWEX = 1 and flux switching is completed.
NWSUBN	Number of windings (temporary storage).
PHID1S	$\phi_{d1s} = \phi_d$ @ $F = F_{d1}$ in Region 1.
PHID2D	ϕ_{d2}.
PHID3	ϕ_{d3}.
P0	$p_0 = (\phi_s + \phi_r)H_q/[(l_o - l_i)H_n]$.
Q1	$q_1 = p_1 \ell n[(F_{d1} - p_2)/(F_{d1} - p_3)]$.
R2C	Type of Terminal b (top): blank - floating; S - Source; G - Ground.
TYPE(20)	$B_d(H)$ - curve type: G, A, B, C D, E, or F.
T1	Alphanumeric "NEGLEC" = 6HNEGLEC or "INCLUD" = 6HINCLUD.
T2	Alphanumeric "TED" = 6HTEDbbb or "ED" = 6HEDbbbb, where b represents a blank.
U0(20)	Initial value of B/B_r.
XID(20)	Inside core diameter (in inch).
XLI	Inside core circumference (in meter).
XLO	Outside core circumference (in meter).
XOD(20)	Outside core diameter (in inch).
Z1	$z_1 = \sum_j f_{(j)}$.
Z2	$z_2 = \sum_j H_{0B(j)}$
Z3	$z_3 = \sum_j f_{(j)} H_{0B(j)}$.

Z4 $z_4 = \sum_j H^2_{0B(j)}$.

Z5 $z_5 = (n_{dynp} z_3 - z_1 z_2)/(n_{dynp} z_4 - z_2^2)$.

(18) Subroutine RWCND

No special local variables.

(19) Subroutine SEQSOL( )

ID Size of an array column.

NX Column pointer, indicating which column is being processed [H(NX) ≡ H(1,N*ID)].

Q Ratio of two matrix elements of the same column, *e.g.*, H(3,5)/H(1,5).

(20) Subroutine INTIAL

ERR Parameter of Subroutine SEQSOL which indicates a singular matrix if ERR ≠ 0.

(21) Subroutine ELEM

No special local variables.

(22) Subroutine IVS

B Temporary storage for VI(N) or EI(K5) or 1/RVS(N).

C Temporary storage for VI(N)/RVS(N) or EI(K5)/RVS(N).

K5 Index number of a variable source.

(23) Subroutine RES

B $\Delta H = 1/R$.

L Index number of an $R(t)$ mode.

(24) Subroutine CAP

A1 $a_1 = \Delta t/R_{CSH}$.

A2 $a_2 = 2C + a_1$.

A3 $a_3 = 2C - a_1$.

A4 $a_4 = 1/(\Delta t + R_C a_2)$.

B1(20) $b_1 = a_3 a_4$.

B2(20) $b_2 = [(\Delta t/a_3) - R_C] i_{ab(-1)}$.

HAA(20) $\Delta H = h_{aa} = a_2 a_4$.

HAAJ $\Delta H = h_{aa}$ (temporary storage).

TA(20) ΔT.

(25) Subroutine IND

B $b = 1/[2L + R_L(\Delta t + c)]$; in solution of initial conditions, $B = 1/R_L$.

B1(20) $b_1 = b(c - \Delta t)$.

B2(20) $b_2 = \{[2L/(\Delta t - c)] - R_L\} i_{ab(-1)}$.

C $c = 2L/R_{LSH}$.

HAA(20) $\Delta H = h_{aa} = b(c + \Delta t)$.

HAAJ $\Delta H = h_{aa}$ (temporary storage).

TA(20) ΔT.

(26) Subroutine ZEN

GZ(20) ΔH.

GZD ΔH (temporary storage).

VZ V_{ab}.

ZI ΔT.

(27) Subroutine DIOD

B $s[1 + (T_d/\Delta t)] + 1/R_{\ell d}$, where s = DR(N) = slope of i_{fd} vs. V_d.

BV $V_{d(-1)}$.

C $V_{d<-1>} \cdot s[1 + (T_d/\Delta t)] - i_{fd<-1>} + PPID - (T_d/\Delta t)[i_{fd<-1>} - i_{fd(-1)}]$, where s = DR(N) = slope of i_{fd} vs. V_d.

CI i_{fd}.

D $C_{jd}/\Delta t = (C_{j0d}/\Delta t) \cdot [1 - (DV2/V_{\varphi d})]^{-0.5}$.

DDI $i_{fd<-1>} - i_{fd(-1)}$.

DEXP $V_{d(-1)}/\theta_{md}$.

DIP(20)	$i_{fd(-1)}$.
DV2	min ($V_{d<-1>}$, 0.9 $V_{\varphi d}$).
HACH	$\Delta H = G$.
LOOP	Count of a possible looping in Part (1) at first iteration.
SI1	I_{sd}.
TACH	$\Delta T = A$.

(28) Subroutine TRAN

BC	$V_{c(-1)}$.
BCC(20)	$(C_{j0c}/\Delta t)\{1 - [\min (V_{c<-1>}, 0.9\ V_{\varphi c})/V_{\varphi c}]\}^{-0.5}$
BCIP(20)	$i_{fc(-1)}$.
BCV2	min ($V_{c<-1>}$, 0.9 $V_{\varphi c}$).
BE	$V_{e(-1)}$.
BEC(20)	$(C_{j0e}/\Delta t)\{1 - [\min (V_{e<-1>}, 0.9\ V_{\varphi e})/V_{\varphi e}]\}^{-0.5}$.
BEIP(20)	$i_{fe(-1)}$.
BEV2	min ($V_{e<-1>}$, 0.9 $V_{\varphi e}$).
CI	i_{fc}.
CIO1	I_{sc}.
CO	$c_{2c} = 1/\theta_{mc} = 11,590/(Tm_c)$.
DBCI(20)	$[i_{fc<-1>} - i_{fc(-1)}]d_{t1c} = [i_{fc<-1>} - i_{fc(-1)}]T_c/\Delta t$..
DBEI(20)	$[i_{fe<-1>} - i_{fe(-1)}]d_{t1e} = [i_{fe<-1>} - i_{fe(-1)}]T_e/\Delta t$.
DEXP	$V_{c(-1)}/\theta_{mc}$; $V_{e(-1)}/\theta_{me}$.
EIO1	I_{se}.
EO	$c_{2e} = 1/\theta_{mc} = 11,590/(Tm_e)$.
EZ	i_{fe}.
HACH22	$\Delta H_{(cc)} = G_c$.
HACH23	$\Delta H_{ce} = S_c$.
HACH32	$\Delta H_{ec} = S_e$.
HACH33	$\Delta H_{(ee)} = G_e$.
KO3	Temporary storage of the index number of Terminal *E* (Emitter).

LOOP	Count of a possible looping in Part (1) at first iteration.
TACH2	$\Delta T_{(c)} = A_c$.
TACH3	$\Delta T_{(e)} = A_e$
TCR	$d_{t2c}(i_{fc<-1>} + I_{sc})c_{2c} = [1 + (T_c/\Delta t)](i_{fc<-1>} + I_{sc})/\theta_{mc}$.
TER	$d_{t2e}(i_{fe<-1>} + I_{se})c_{2e} = [1 + (T_e/\Delta t)](i_{fe<-1>} + I_{se})/\theta_{me}$.

(29) Subroutine CORE

AITKEN	Boolean variable: *FALSE* every odd iteration; *TRUE* every even iteration.
BV	$V_a - V_b$
CIJM1(20,10)	$i_{j<-1>}$.
CIJM2(20,10)	$i_{j<-2>}$.
CIW	$i_{ab} = G_c(V_a - V_b - V_c)$.
DELTAV	$S_t/3 = -[3\Delta t + 2\Delta t_{(-1)} + \Delta t_{(-2)}]/3$, if $\Delta t < 10^{-10}$ second; DELTAV = -TE/NDELT, otherwise.
DLPDTE	$\delta\dot{\phi}_\epsilon$.
EPS	ϵ.
EPSP	ϵ'.
ETA	$\eta = 1 - [(2s\varphi + \varphi_s - \varphi_d)/(\varphi_s + \varphi_d)]^2$.
FDOTAV	$\overline{\dot{F}}$.
FLXSW	0 - All cores have terminated flux switching; 1 - otherwise ($\lvert\varphi_i\rvert$ > ZV for at least one core).
FMM	MMF (temporary storage).
GC(20,10)	$\Delta H = G_c = (N^2\dot{\varphi}'_{<-1>} + R_w)^{-1}$.
GCNM	G_c (temporary storage).
GCVC	$G_c V_c$.
GW	$1/R_w$.
M	Index number of a core winding.
NWSUBN	Number of windings (temporary storage)-
PHDTEP(20)	$\dot{\varphi}'_\epsilon$.
PHDTI	$\dot{\varphi}_i$.
PHDTIP	$\dot{\varphi}'_i$.

PHDTP	$\dot{\Phi}_p$.
PHDTPP	$\dot{\Phi}'_p$.
PHDTPR	$\dot{\Phi}'$.
PHIDP	Φ'_d.
PHIDS	Φ_{ds}.
Q	Limiting boundary on $\dot{\Phi}_\epsilon$.
Q1	q_1.
Q2	q_2.
Q3	q_3.
Q4	q_4.
Q5	q_5.
Q6	q_6.
SF	$s \cdot F = \|F\|$, where s = sign (F).
SIGNF	1 if $F > 0$; -1 if $F \leq 0$.
SMFTN	$S_{Ftn} = S_F S_t - 3S_{Ft}$.
SMTSMT	S_t^2.
SPHI	$s \cdot \Phi$, where s = sign (F).
SUMF	$S_F = F_{(-1)} + F_{(-2)} + F_{(-3)}$.
SUMFT	$S_{Ft} = -[F_{(-1)}\Delta t_{(-1)} + F_{(-2)}t_{(-2)} + F_{(-3)}t_{(-3)}]$.
SUMT	$S_t = -[3\Delta t + 2\Delta t_{(-1)} + \Delta t_{(-2)}]$.
SUMTSQ	$S_{tsq} = S_t^2 - 3S_{tt}$.
SUMTT	$\Delta t^2 + [\Delta t + \Delta t_{(-1)}]^2 + [\Delta t + \Delta t_{(-1)} + \Delta t_{(-2)}]^2$.
S2	$1/(\|F\| + p_2)$.
S3	$1/(\|F\| + p_3)$.
TM1	$t_{(-1)} = -\Delta t$.
TM2	$t_{(-2)} = -\Delta t - \Delta t_{(-1)}$.
TM3	$t_{(-3)} = -\Delta t - \Delta t_{(-1)} - \Delta t_{(-2)}$.
VC(20,10)	$V_c = N\dot{\Phi}_{<-1>} - N^2\dot{\Phi}'_{<-1>}i_{ab<-1>}$.
VCNM	V_c (temporary storage).

(30) Subroutine AUX( )

K — Control indicator: ≤ 0 - computes functions for Part (1) in Subroutine ELEM; 1 - computes auxiliary variables for Part (2) in Subroutine ELEM.

B. MTRAC Program Listing

The listing of the MTRAC program is given next. Comments are scattered throughout the program in order to explain the functions of different program sections.

The main program is listed on pp. 91-99. The subroutines are listed in the following order:

```
C      MTRAC - MAIN PROGRAM
       COMMON AAAA(10),ABC,ABSER,ABSERR,ABTEMP,AF(20),AR(20),BCDT,BCDT1,
      2    BCI(20),BCR(20),BCV(20),BEI(20),BER(20),BEV(20),BVAL(6,6),
      3    BVALF(6,6),CAPC(20),CC1(20),CC2(20),CC3(20),CC4(20),CDT1(20),
      4    CDT2(20),CDT3(20),CIBASE(20),CIC(20),CICOLC(20),CICORE(20,10),
      5    CICOR1(20,10),CIDIOD(20),CIEMTR(20),CIIVS(20),CIL(20),CIR(40),
      6    CIZ(20),CT(11,2),C1(20),C2(20),C3(20),C4(20),DELT,DI(20),DMAX,
      7          DR(20),DTM1,DTM2,DT1(20),DT2(20),DT3(20),DV(20),
      X    E(19,16,2),EC1(20),EC2(20),EC3(20),EC4(20),EDT1(20),EDT2(20),
      8           EDT3(20),EI(20),EP1(20),
      9    EZB(20),E2(19),F(20),FB(20),FB1(20),FDB(20),FDZ(20),FD1(20)
       COMMON FD2(20),FD3(20),FD4(20),FM1(20),FM2(20),FM3(20),
      2    FO(20),FOPP(20),FO1(20),H(60,60),IAC(6),IACF(6),IBC,ICS(6,6),
      3    ICSF(6,6),IGRAPH,INCIL(20),INDL(20),          IOC,IOCF,IPX
      4       ,ITRMAX,ITRTST,IX(5),JCAT,JG(19),JKKJ,JKLJ,JSJ,
      5 JV,KCAP(20),KCORE(20,10),KDIOD(20),KFAIL,KFMAX,KIND(20),KIVS(20),
      6     KONT,KONVRG,KPFIT,KPLOT,KPRPLT,KPUNCH,KRES(40),KTRAN(20),KXT,
      7    KZEN(20),K1CAP(20),K1CORE(20,10),K1DIOD(20),K1IND(20),K1IVS(20)
      8   ,K1RES(40),K1TRAN(20),K1ZEN(20),K2CAP(20),K2CORE(20,10),
      9    K2DIOD(20),K2IND(20),K2IVS(20),K2RES(40),K2TRAN(20),K2ZEN(20)
       COMMON K3IVS(20),K3TRAN(20),K4IVS(20),K4TRAN(20),LASTIC,
      2              MAXMIN,MONC,MONCOR,MOND,MONELM,MONIV,MONL,MONMA,
      3    MONR,MONS,MONT,MONTOR,MONZEN,MVALUE(14),NAW,NB,NBB,NC,NCORE,ND,
      4    NDELT,NDIODE,NDTMAX,NDTMIN,ND1,NG,NITER,NITRSM,NIV,NL,NOEXIT,
      5    NOVAR(6,6),NOVARF(6,6),NP(20,10),NPLTVT,NPLTVV,NPP(10),
      6    NP1(10,2),NR,NRP(40),NS(20,10),NSVIL,NSVPHI,NSVVAR,NTRANS,NV,
      7    NW(20),NZ,NZEN,N1B,N3B,PDTEM1(20),PHDTE(20),PHDTM1(20),
      8    PHDTM2(20),PHI(20),PHID(20),PHIDCT(20),PHIM1(20),PHIM2(20),
      9    PHIM3(20),PHIR(20),PHIS(20),PHIO(20),PPIC(20),PPIC1(20)
       COMMON PPID(20),PPID1(20),PPIE(20),PPIE1(20),PRSYMB,PSTEP,
      2    PWCR(20),PWTR(20),P1(20),P2(20),P3(20),P4(20),P5(20),P6(20),
      3    P7(20),P8(20),P9(20),P10(20),P11(20),P12(20),P13(20),P14(20),
      4    P15(20),P16(20),P17(20),P18(20),P19(20),P20(20),QCR(20,7),
      5    QTAN(20,16),R(40),RC(20),RCSH(20),RELER,RELERR,RF(5,5),RL(20),
      6    RLSH(20),RMAG,RNUDT,ROP(20),ROP1(20),RVS(20),RW(20,10),PZ(20),
      7RZB(20),S(20),SAVEXL(10,3),SAVEXR(10,3),SAVEYB(10,3),SAVEYT(10,3),
      8           ST(19,2),STARTS,STSWEX,T(60),TE,TF(354),TFR(5,5),TI,
      9    TIMEIN,TITLE(12),TLIMIT,TOO(19),TOOLD,TO1,TTOOLD,V(70),VI(20)
       COMMON VICON(20),VP(10,350),VZB(20),VZF(20),V1(70),V2(70),WSQSOL,
      2    XL,XLAMDA(20),XLEFT(20),XLMDAD(20),XNT(20,10),XNU(20),XNUD(20)
      3   ,XP(713),XR,XRES(5,5),XRIGHT(20),XSCALE(20),XUNIT(2,20),YB,
      4     YBOTTM(20),YSCALE(20),YT,YTOP(20),YUNIT(2,20),ZV(20)
       LOGICAL   TERMNT,SOFAR,MONTOR,MONST,MONMAT,MONELT,WSQSLT,MONIVT ,      0030
      2         MONRT,MONCT,MONLT,MONDT,MONTT,MONCRT,MONZNT ,                 0040
      3         MONS,MONMA,MONELM,WSQSCL,MONIV,MONR,MONC,MONL,MOND,MONT ,     0050
      4         MONCOR,MONZEN,GETOFF                                          0060
       DIMENSION      FETCH(2)                                                0070
       EQUIVALENCE(MVALUE(1),MONMAT),(MVALUE(2),MONST),(MVALUE(3),MONELT)     0080
      2     , (MVALUE( 4),WSQSLT),(MVALUE( 5),MONIVT),(MVALUE( 6),MONRT )     0090
      3     , (MVALUE( 7),MONCT ),(MVALUE( 8),MONLT ),(MVALUE( 9),MONZNT)     0100
      4     , (MVALUE(10),MONDT ),(MVALUE(11),MONTT ),(MVALUE(12),MONCRT)     0110
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
       GETOFF = .FALSE.                                                       0120
  959  IF (FETCH(1) .EQ. GENDA1) GO TO 7722                                   0130
       FETCH(1) = GENDA1                                                      0140
C      -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
C      READ IN AND PRINT OUT RUN-CONTROL DATA
C      -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
 7722  CALL RWRUNC                                                            0150
       FETCH(1) = RWFLM1                                                      0160
```

```
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     READ IN AND PRINT OUT CIRCUIT-ELEMENT DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      CALL RWIVS                                                    0170
      CALL RWRES                                                    0180
      CALL RWCAP                                                    0190
      CALL RWIND                                                    0200
      CALL RWZEN                                                    0210
      CALL RWDIOD                                                   0220
      CALL RWTRAN                                                   0230
      FETCH(1) = RWCOR1                                             0240
      CALL RWCORE                                                   0250
      WRITE (6,33)                                                  0260
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     INITIAL CONDITIONS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      KNL = NTRANS + NTRANS + NDIODE                                0270
        IF (NCORE .EQ. 0) GO TO 1115                                0280
       DO 1116 I = 1,NCORE                                          0290
 1116 KNL = KNL + NW(I)                                             0300
 1115 IF ((ITRTST .NE. 0) .AND. (MAXMIN .NE. 0) .AND. (KFMAX .NE. 0))  0310
     2GO TO 1114                                                    0320
      IF (ITRTST .EQ. 0) ITRTST = 10 + KNL                          0330
      IF (MAXMIN .EQ. 0) MAXMIN = 2*(10 + KNL)                      0340
        ITRMAX = MAXMIN                                             0350
      IF (KFMAX .EQ. 0) KFMAX = 3                                   0360
      WRITE(6,7777) ITRTST,MAXMIN,KFMAX                             0370
 1114 IF (KONT .EQ. 1) GO TO 2102                                   0380
      FETCH(1) = SOLVE1                                             0390
      IF  (NSVVAR .EQ. 1) GO TO 2102                                0400
      NAW = 0                                                       0410
      IBC = 0                                                       0420
      JSJ = 0                                                       0430
      CALL DIOD                                                     0440
      CALL TRAN                                                     0450
      IF (NG .NE. -1) CALL CORE                                     0460
 2102 JSJ = 1                                                       0470
      KDTIM = 0                                                     0480
      IF (NG) 2330,1701,2101                                        0490
 2330 IF (KONT   .EQ. 1) GO TO 2101                                 0500
      CALL INTIAL                                                   0510
 1701 K = NV + 1                                                    0520
      IF (NG .EQ. 0) CALL AUX (1)                                   0530
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PRINT INITIAL VALUES
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DO 1702 J = K,NZ                                              0540
 1702 V1(J) = V(J)                                                  0550
 2101 NG=0                                                          0560
  152 ND1=ND-1                                                      0570
      IF (JKLJ .GT. 0) WRITE (6,7414) JKLJ,ABC                      0580
      WRITE (6,34)   (J,EI(J),J=1,ND1)                              0590
      WRITE (6,3434) (J,V1(J),J=1,NZ)                               0600
      WRITE (6,8556) RMAG                                           0610
      WRITE (6,32)   TI,TE,CT(1,2)                                  0620
      CALL SECOND(TEMP)                                             0630
      WRITE (6,7975) TEMP                                           0640
      IF (N38) 959,2601,2600                                        0650
 2601 IF(JKKJ .GT. 0) CALL EXIT                                     0660
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     READ IN AND PRINT OUT CONTINUED-RUN DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 2600 IF(KONT .NE. 1) GO TO 2301                                    0670
      CALL RWCND                                                    0680
      FETCH(1) = SOLVE1                                             0690
      GO TO 2602                                                    0700
```

```
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C    INITIALIZE DELT AND T. STORE INITIAL PLOTTED VARIABLES IN PLOT ARRAY
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 2301 DELT = TI                                                          0710
      IF (NSVVAR .NE. 0) GO TO 2602                                      0720
      TO1=DELT                                                           0730
      DTM1= DELT                                                         0740
      DTM2= DELT                                                         0750
 2602 DO 1003 I=1,NPLTVT                                                 0760
      JJ       = IABS(NPP(I))                                            0770
 1003 VP(I,JV) = V1(JJ)                                                  0780
      TF(JV)=  TE                                                        0790
      TE    = TE + DELT                                                  0800
      WRITE (6,33)                                                       0810
      JSJ = -1                                                           0820
  108 IF (MONMA) WRITE (6,6662)        ND1,   NC,          ABC,TO1,DELT,  0830
     2       TE                                                          0840
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     START OF TRANSIENT LOOP
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DO 76 JJJ=1,NBR                                                    0850
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     EACH NEW JJJ LOOP GIVES A NEW SET OF PRINT VALUES
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      JV=JV+1                                                            0860
      DO 131 JJ=1,N8                                                     0870
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     EACH NEW  JJ LOOP GIVES A NEW SET OF PLOT VALUES
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      NAW   = 0                                                          0880
      TO1   = DELT                                                       0890
 2222 NITER = 1                                                          0900
      IF (MONMA) WRITE (6,6663) JJJ,JV,JJ,JSJ,TO1                        0910
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE VARIABLE-SOURCE VALUES
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF(ND .LE. 0) GO TO 119                                            0920
      TE1=0.0                                                            0930
      DO 121 J=1,ND                                                      0940
      IF(ST(J,2) .LT.0.0)GO TO 121                                       0950
      JH =JG(J)                                                          0960
      T1 =TE-E(J,JH+1,2)-TOO(J)                                          0970
      IF(TE1 .GE.       T1) GO TO 121                                    0980
      IF(T1  .GE. 0.999*DELT) GO TO 121                                  0990
      TE1 = T1                                                           1000
  121 CONTINUE                                                           1010
      IF(TE1 .LE.0.0)GO TO 3000                                          1020
      TE=TE-TE1                                                          1030
      DELT = DELT - TE1                                                  1040
 3000 DO 3124 J=1,ND                                                     1050
      IF(ST(J,2) .LT.0.0)GO TO 124                                       1060
      JH=JG(J)                                                           1070
      IF(J .GT. 1) GO TO 8504                                            1080
      K=19                                                               1090
      GO TO 8505                                                         1100
 8504 K =J-1                                                             1110
 8505 TE1=TE-E(J,JH+1,2)-TOO(J)                                          1120
      IF(TE1 .LE. 0.01*DELT) GO TO 1071                                  1130
      JG(J)=JG(J)+1                                                      1140
      NI    =ST(J,1)                                                     1150
      IF(JG(J) .GE. NI) GO TO 1074                                       1160
      E2(J)=(E(J,JH+2,1)-E(J,JH+1,1))/(E(J,JH+2,2)-E(J,JH+1,2))          1170
 1071 EI(K)=E(J,JH+1,1)+E2(J)*TE1                                        1180
      GO TO 124                                                          1190
 1074 IF(ST(J,2) .GT.0.0)GO TO 1081                                      1200
      ST(J,2)=-1.0                                                       1210
```

```
      EI(K)  =E(J,NI,1)                                                        1220
      GO TO 124                                                               1230
 1081 JG (J)=1                                                                1240
      TOO(J)=TOO(J)+E(J,NI,2)                                                 1250
      E2 (J)=(E(J,2,1)-E(J,NI,1))/E(J,2,2)                                    1260
      EI (K)=E(J,NI,1)                                                        1270
  124 IF(MONMA)WRITE(6,6665)J,JH,K,JG(J),NI,TE1,E2(J),EI(J),EI(K),TOO(J)      1280
 3124 CONTINUE                                                                1290
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK MONITOR SET UP
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  119 SOFAR = MONMA                                                           1300
      IF (NDELT .LT. NDTMIN) GO TO 7726                                       1310
      IF (NDTMAX .LT. NDELT) GO TO 7726                                       1320
      IF (MONTOR) GO TO 701                                                   1330
      MONTOR = .TRUE.                                                         1340
      MONMA  = MONMAT                                                         1350
      MONS   = MONST                                                          1360
      MONELM = MONELT                                                         1370
      WSQSOL = WSQSLT                                                         1380
      MONIV  = MONIVT                                                         1390
      MONR   = MONRT                                                          1400
      MONC   = MONCT                                                          1410
      MONL   = MONLT                                                          1420
      MOND   = MONDT                                                          1430
      MONT   = MONTT                                                          1440
      MONCOR = MONCRT                                                         1450
      MONZEN = MONZNT                                                         1460
      IF (MONMA ) GO TO 3333                                                  1470
      IF (MONS  ) GO TO 3333                                                  1480
      IF (MONELM) GO TO 3333                                                  1490
      IF (WSQSOL) GO TO 3333                                                  1500
      IF (MONIV ) GO TO 3333                                                  1510
      IF (MONR  ) GO TO 3333                                                  1520
      IF (MONC  ) GO TO 3333                                                  1530
      IF (MONL  ) GO TO 3333                                                  1540
      IF (MOND  ) GO TO 3333                                                  1550
      IF (MONT  ) GO TO 3333                                                  1560
      IF (MONCOR) GO TO 3333                                                  1570
      IF (MONZEN) GO TO 3333                                                  1580
      GO TO 701                                                               1590
 3333 WRITE (6,6671) (MVALUE (I),I=1,12)                                      1600
      GO TO 701                                                               1610
 7726 IF (.NOT. MONTOR) GO TO 701                                             1620
      IF (MONMA) WRITE (6,6673)                                               1630
      MONTOR = .FALSE.                                                        1640
      MONMA  = .FALSE.                                                        1650
      MONS   = .FALSE.                                                        1660
      MONELM = .FALSE.                                                        1670
      WSQSOL = .FALSE.                                                        1680
      MONIV  = .FALSE.                                                        1690
      MONR   = .FALSE.                                                        1700
      MONC   = .FALSE.                                                        1710
      MONL   = .FALSE.                                                        1720
      MOND   = .FALSE.                                                        1730
      MONT   = .FALSE.                                                        1740
      MONCOR = .FALSE.                                                        1750
      MONZEN = .FALSE.                                                        1760
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     BEGINNING OF ITERATION LOOP
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  701 IF(SOFAR.OR.MONMA)WRITE(6,6666)NITER,TE,DELT                             1770
      CALL ELEM                                                               1780
      IF (.NOT. WSQSOL) GO TO 9992                                            1790
      WRITE (6,9999)                                                          1800
      DO 9993 J = 1,NV                                                        1810
```

```
 9993 WRITE (6,9997) (H(J,K),K=1,NV)                                    1820
      WRITE (6,9996) (T(J),J=1,NV)                                      1830
 9992 CALL SEQSOL (ERR)                                                 1840
      IF (ERR .EQ. 0.0) GO TO 8100                                      1850
         IF (.NOT. WSQSOL) GO TO 9979                                   1860
         DO 9978 J=1,NV                                                 1870
 9978    WRITE (6,9997) (H(J,K),K=1,NV)                                 1880
 9979    CALL ELEM                                                      1890
      WRITE (6,2002) ((JACK,JILL,H(JACK,JILL),JILL=1,NV),JACK=1,NV)     1900
         CALL EXIT                                                      1910
 8100 IF (.NOT. WSQSOL) GO TO 2008                                      1920
         CALL AUX(1)                                                    1930
         WRITE (6,9970) (V(J),J=1,NZ)                                   1940
 2008 TOL    = -TOL                                                     1950
      CALL ELEM                                                         1960
      NITER = NITER + 1                                                 1970
      NITRSM = NITRSM + 1                                               1980
      IF(ABC .LE. 0.0) GO TO 5000                                       1990
      IF(NAW .NE. 0) GO TO 701                                          2000
      NAW=-1                                                            2010
      GO TO 701                                                         2020
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     END OF ITERATION LOOP
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 5000 IF (MONMA) WRITE (6,7500) JCAT,DELT,ABC, CT(JCAT,2)               2030
      NITER = NITER - 1                                                 2040
      IF (ABC .NE. -3.0) GO TO 5001                                     2050
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DID NOT CONVERGE IN ITRMAX ITERATIONS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (NAW .LT. 0) NAW = 0                                           2060
      NAW = NAW + 1                                                     2070
      IF (NAW .LT.  6) GO TO 2222                                       2080
      WRITE (6,5003)                                                    2090
      GO TO 1000                                                        2100
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK FOR NORMAL-TERMINATION CONDITIONS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 5001 TERMNT = .FALSE.                                                  2110
      IF (IOC .LE. 0) GO TO 7503                                        2120
      DO 7729 I = 1, IOC                                                2130
          SOFAR = .TRUE.                                                2140
          J3    = IAC(I)                                                2150
          DO 7730 J1 = 1,J3                                             2160
               J4   = NOVAR(I,J1)                                       2170
               TEMP = V(J4)                                             2180
               I1   = ICS(I,J1)                                         2190
               IF (I1 .LT. 0) TEMP = ABS(TEMP)                          2200
               IF(IABS(I1) .NE.  2) GO TO  7731                         2210
               IF (TEMP-BVAL(I,J1)) 7733,7733,7730                      2220
 7731          IF (TEMP-BVAL(I,J1)) 7730,7733,7733                      2230
 7733          SOFAR = .FALSE.                                          2240
 7730     CONTINUE                                                      2250
          TERMNT = TERMNT .OR. SOFAR                                    2260
 7729 CONTINUE                                                          2270
      IF (TERMNT) JCAT = 10                                             2280
      TERMNT = .FALSE.                                                  2290
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK FOR FAILURE-TERMINATION CONDITIONS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 7503 IF (IOCF .LE. 0) GO TO 7505                                       2300
      DO 7734 I = 1,IOCF                                                2310
          SOFAR = .TRUE.                                                2320
          J3    = IACF(I)                                               2330
          DO 7735 J1 = 1,J3                                             2340
               J4   = NOVARF(I,J1)                                      2350
```

```
                TEMP = V(J4)                                                    2360
                I1   = ICSF(I,J1)                                               2370
                IF (I1 .LT. 0) TEMP = ABS(TEMP)                                 2380
                IF (IABS(I1) .NE.  2) GO TO 7736                                2390
                IF (TEMP-BVALF(I,J1)) 7738,7738, 7735                           2400
 7736           IF (TEMP-BVALF(I,J1)) 7735,7738,7738                            2410
 7738           SOFAR = .FALSE.                                                 2420
 7735       CONTINUE                                                            2430
            TERMNT = TERMNT .OR. SOFAR                                          2440
 7734 CONTINUE                                                                  2450
      IF (TERMNT) JCAT = 10                                                     2460
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     UPDATE VARIABLES FOR THE NEXT TIME STEP
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 7505 TE=TE + DELT                                                          2470
      DO 3015 J=1,NZ                                                        2480
 3015 V1(J)=V(J)                                                            2490
      IF(ND  .LE. 0) GO TO 131                                              2500
      DO 1253 J = 1, ND1                                                    2510
 1253 EP1(J) = EI(J)                                                        2520
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK FOR END OF DELT MODE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (TE .LT. CT (JCAT,2)) GO TO 131                                    2530
      IF (TE .LT. (CT(JCAT,2)+0.999* DELT))GO TO 8561                       2540
      JCAT=JCAT+1                                                           2550
      IF (CT(JCAT,1) )   82,1200,84                                         2560
   82 CT(JCAT,1)= CT(JCAT,1)+1.0                                            2570
      JCAT      = JCAT - 1                                                  2580
      IF (KDTIM  .GT. 0) GO TO 4360                                         2590
      DTIM =CT(JCAT,2)                                                      2600
      KDTIM= 1                                                              2610
 4360 DO 83  J=1,JCAT                                                       2620
   83 CT(J,2)= CT(J,2)+DTIM                                                 2630
      JCAT   = 1                                                            2640
   84 TI     = CT(JCAT,1)                                                   2650
      TE=TE-DELT                                                            2660
      IF (TI .GT. (DELT+DELT)) GO TO 9234                                   2670
      TE=TE+TI                                                              2680
      DELT = TI                                                             2690
      GO TO 321                                                             2700
 9234 DELT = DELT + DELT                                                    2710
      TE=TE+DELT                                                            2720
  321 WRITE (6,32)TI,CT(JCAT-1,2),CT(JCAT,2)                                2730
      GO TO 131                                                             2740
 8561 TOK=TE-CT(JCAT,2)                                                     2750
      TE=TE-TOK                                                             2760
      DELT = DELT - TOK                                                     2770
  131 CONTINUE                                                              2780
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     END OF  JJ LOOP
C     STORE PLOT DATA IN ARRAYS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      TE=TE-DELT                                                            2790
      DO 205 I=1,NPLTVT                                                 D   2800
      JJ      =IABS(NPP(I))                                             A   2810
  205 VP(I,JV)=V(JJ)                                                    T   2820
 7107 IF (RMAG .NE.0.0)GO TO 8765                                       A   2830
      XP(JV+IPX) = 1.0                                                      2840
      GO TO 8766                                                        F   2850
 8765 XP(JV+IPX)=EI(19)/RMAG                                            O   2860
 8766 TF(JV)=TE                                                         R   2870
      TE    =TE+DELT                                                        2880
      IF (JV .LT. IPX) GO TO 76                                         P   2890
      IF (NPLTVT  .EQ. 0) GO TO 75                                      L   2900
      KPRPLT = 1                                                        O   2910
```

```
      GO TO 1200                                                          T  2920
   75 JV = 1                                                                 2930
   76 CONTINUE                                                               2940
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     END OF JJJ LOOP
C     PRINT SOURCE MAGNITUDES AND RESULTING NODAL VOLTAGES
C                    AND AUXILIARY VARIABLES
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (ND .LE. 0) GO TO 653                                                 2950
      IF (EI(19) .NE. 0.0) WRITE (6,69) EI(19)                                 2960
      WRITE (6,70)        (L,EI(L),L=1,ND1)                                    2970
  653 TE    = TE-DELT                                                       P  2980
      LABEL = 653                                                         R  2990
      IF (MONMA) WRITE (6,7501) LABEL                                     I  3000
      WRITE (6,50)TE,(I,V(I),I=1,NZ)                                      N  3010
      WRITE (6,7727) DELT,NITER,NDELT,NITRSM                              T  3020
      TE=TE+DELT                                                               3030
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK FOR COMPUTER PROCESSOR TIME
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      CALL SECOND(TEMP)                                                        3040
      IF (TEMP-STARTS .LE. TLIMIT) GO TO 108                                   3050
      WRITE (6,7767)                                                           3060
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     ERROR - EITHER 6 ITRMAX FAILURES OR COMPLETION OF ALLOTTED
C          COMPUTER PROCESSOR TIME
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 1000 JCAT = 11                                                                3070
      GETOFF = .TRUE.                                                          3080
      IF (NAW .LT. 6) KPUNCH = 1                                               3090
      IF (NPLTVT .GT. 0) KPRPLT = 1                                            3100
      GO TO 109                                                                3110
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PRINT SOLUTION VALUES FOR FINAL DELT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 1200 IF (ND .LE. 0) GO TO 7000                                                3120
      IF (EI(19) .NE. 0.0) WRITE (6,69) EI(19)                                 3130
      WRITE (6,70)        (L,EI(L),L=1,ND1)                                    3140
 7000 TE    =TE-DELT                                                          3150
      TF(JV)=TF                                                                3160
      LABEL = 7000                                                             3170
      IF (MONMA) WRITE (6,7501) LABEL                                          3180
      WRITE (6,50)TE,(II,V(II),II=1,NZ)                                        3190
      WRITE (6,7727) DELT,NITER,NDELT,NITRSM                                   3200
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK AND PRINT IF TERMINATION CONDITIONS WERE FULFILLED
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (IOC .LE. 0 ) GO TO 7507                                              3210
      TERMNT = .FALSE.                                                         3220
      DO 7739 I = 1,IOC                                                        3230
          SOFAR = .TRUE.                                                       3240
          J3    = IAC(I)                                                       3250
          DO 7740 J1 = 1,J3                                                    3260
               J4   = NOVAR(I,J1)                                              3270
               TEMP = V(J4)                                                    3280
               I1   = ICS(I,J1)                                                3290
               IF (I1 .LT. 0) TEMP = ABS(TEMP)                                 3300
               IF (IABS(I1) .NE. 2) GO TO 7741                                 3310
               IF (TEMP-BVAL(I,J1)) 7743,7743,7740                             3320
 7741          IF (TEMP-BVAL(I,J1)) 7740,7743,7743                             3330
 7743          SOFAR = .FALSE.                                                 3340
 7740     CONTINUE                                                             3350
          IF (SOFAR) WRITE(6,7749) I                                           3360
          TERMNT = TERMNT .OR. SOFAR                                           3370
 7739 CONTINUE                                                                 3380
      IF (TERMNT) WRITE(6,7750)                                                3390
```

```
 7507 IF (IOCF .LE. 0) GO TO 7002                                          3400
      TERMNT = .FALSE.                                                     3410
      DO 7744 I = 1,IOCF                                                   3420
           SOFAR = .TRUE.                                                  3430
           J3    = IACF(I)                                                 3440
           DO 7745 J1 = 1,J3                                               3450
                J4   = NOVARF(I,J1)                                        3460
                TEMP = V(J4)                                               3470
                I1   = ICSF(I,J1)                                          3480
                IF (I1 .LT. 0) TEMP = ABS(TEMP)                            3490
                IF (IABS (I1) .NE. 2 ) GO TC 7746                          3500
                IF (TEMP-BVALF(I,J1)) 7748,7748,7745                       3510
 7746           IF (TEMP-BVALF(I,J1)) 7745,7748,7748                       3520
 7748           SOFAR = .FALSE.                                            3530
 7745      CONTINUE                                                        3540
           IF (SOFAR) WRITE(6,7749) I                                      3550
           TERMNT = TERMNT .OR. SOFAR                                      3560
 7744 CONTINUE                                                             3570
      IF (TERMNT) WRITE (6,7977)                                           3580
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CALL PLOT ROUTINES
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 7002 TE=TE+DELT                                                           3590
  109 FETCH(1) = GENDA1                                                    3600
      IF (CT(JCAT, 1) .GT.0.0)GO TO 85                                     3610
      IF (KPLOT .EQ. 1.OR.KPUNCH .EQ. 1.CR.KPRPLT .EQ. 1) CALL PLAPR(1)    3620
      CALL SECOND(TEMP)                                              P     3630
      WRITE (6,7975) TEMP                                            L     3640
      IF (GETOFF) CALL EXIT                                          O     3650
      GO TO 959                                                      T     3660
   85 CALL PLAPR ( -1 )                                                    3670
      FETCH(1) = SOLVE1                                                    3680
      GO TO 108                                                            3690
C  . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   32 FORMAT (    /12HODELTA TIME=E12.5,3X,12H START TIME=E12.5,3X,10H E   3700
     2ND TIME=E12.5)                                                       3710
   33 FORMAT (1H1)                                                         3720
   34 FORMAT (    15HOINITIAL VALUES /7H SOURCE /6(I4,E13.5))              3730
   50 FORMAT (    6HOTIME=,E12.5,/,9H UNKNOWNS, /(6(I4,E13.5)))           3740
   69 FORMAT (    17HOIONIZING PULSE =,E13.5)                              3750
   70 FORMAT (                              7H SOURCE,/(6(I4,E13.5)))      3760
 2002 FORMAT(26H1SINGULAR MATRIX EQUATION ,/,1HO,/,      20H    H MATRIX   3770
     1ELEMENTS/(I4,I4,E13.5))                                              3780
 3434 FORMAT (  9HOUNKNOWNS,/(6(I4,E13.5))   )                             3790
 5003 FORMAT (21H CONVERGENCE FAILURE    )                                 3800
 6662 FORMAT (34H NDI,   NG,      ABC,TOI,DELT,  TE        ,2I7,          3810
     2     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )                   3820
 6663 FORMAT (33H JJJ,JV,JJ,JSJ,       TOI                    ,4I7,      3830
     2     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )                   3840
 6665 FORMAT(42H J,JH,K,JG(),NI,TE1,E2(),EI(),EI(K),TOO()  , 5I7,/,1H ,    3850
     2    10E12.5)                                                         3860
 6666 FORMAT(50(2H *),10HITERATION ,I5,          /                         3870
     2     23H                TE,DELT              ,/,5E24.15)             3880
 6671 FORMAT (15H MONITORS ON.. ,12L4)                                     3890
 6673 FORMAT (29H MONITORS TURNED OFF AT 7726    )                         3900
 7414 FORMAT (1H   , I5, 30H INITIAL CONDITION ITERATIONS                  3910
     2     ,/, 5H ABC=   , F5.0)                                           3920
 7500 FORMAT (22H JCAT,DELT,ABC,CT(,2)       ,1I6,3E12.5)                  3930
 7501 FORMAT ( 7H LABEL    ,I5)                                            3940
 7727 FORMAT (8HODELT = ,E12.5,15X,I4,14H ITERATION(S) ,15X,8HNDELT =      3950
     2     ,I5,15X,13H ITER. SUM = ,I6,/)                                  3960
 7749 FORMAT (15HOCONDITION NO. ,I2,13H IS SATISFIED  )                    3970
 7750 FORMAT (20HONORMAL TERMINATION  )                                    3980
 7767 FORMAT (24H TIME-LIMIT TERMINATION    )                              3990
 7777 FORMAT(44HOCONVERGENCE DATA DETERMINED BY THE PROGRAM    ,/,         4000
     1       (10H ITRTST  =,I3,102H (NUMBER CF TEST ITERATIONS BELOW WHI   4010
```

```
     2CH DELT IS MULTIPLIED BY 2 AND ABOVE WHICH DELT IS DIVIDED BY 2)      4020
     3  ,/,10H MAXMIN  = ,I3,99H (INITIAL MAXIMUM NUMBER OF ITERATIONS B   4030
     4EFORE VARIABLE VALUES ARE RESET AND DELT IS DIVIDED BY 4)       ,/,    4040
     5 10H KFMAX    =,I3,54H (MAXIMUM NUMBER OF TRIALS BEFORE ITRMAX IS C  4050
     6HANGED)    )                                                         4060
7975 FORMAT (13HOPRO. TIME =  ,E12.3)                                      4070
7977 FORMAT (21HOFAILURE TERMINATION  )                                    4080
8556 FORMAT (    30HOIONIZING PULSE SCALE FACTOR =E14.5)                   4090
9970 FORMAT (16H SEQSOL RESULTS ,(/,1H ,5E24.15))                          4100
9996 FORMAT (2H T ,(/,1H ,10E12.5))                                        4110
9997 FORMAT (2H H ,(  1H ,10E12.5))                                        4120
9999 FORMAT (8H SEQSOL  )                                                  4130
     END                                                                   4140
```

```
SUBROUTINE HOLLER(SOFTLY, SWEET)                                    00010
SWEET = SOFTLY                                                      00020
RETURN                                                              00030
END                                                                 00040
```

```
      SUBROUTINE RWRUNC                                                 RRU 0010
C     RWRUNC READS IN AND PRINTS OUT RUN-CONTROL DATA
```

COMMON VARIABLES

```
      LOGICAL                  MONTOR,MONST,MONMAT,MONELT,WSQSLT,MONIVT  RRU 0020
      LOGICAL MONRT,MONCT,MONLT,MONDT,MONTT,MONCRT,MONZNT               RRU 0030
      LOGICAL MONS,MONMA,MONELM,WSQSOL,MONIV,MONR,MONC,MONL,MOND,MONT   RRU 0040
      LOGICAL MONCOR,MONZEN,MONEY(14),SOFAR                             RRU 0050
      DIMENSION EMVALU(14)                                              RRU 0060
      EQUIVALENCE(MVALUE(1),MONMAT),(MVALUE(2),MONST),(MVALUE(3),MONELT)RRU 0070
     2    ,(MVALUE( 4),WSQSLT),(MVALUE( 5),MONIVT),(MVALUE( 6),MONRT )  RRU 0080
     3    ,(MVALUE( 7),MONCT ),(MVALUE( 8),MONLT ),(MVALUE( 9),MONZNT)  RRU 0090
     4    ,(MVALUE(10),MOND  ),(MVALUE(11),MONTT ),(MVALUE(12),MONCRT)  RRU 0100
     5    ,(MVALUE,EMVALU,MONEY)                                        RRU 0110
C  . . . . . .
      CALL HOLLER(1H ,HOL )                                             RRU 0120
      CALL HOLLER(1HO,HOLO)                                             RRU 0130
      CALL SECOND(STARTS)                                               RRU 0140
      WRITE(6,7975) STARTS                                              RRU 0150
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     READ-WRITE OVERALL-RUN DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ  (5,7751) CKT,LIMITT,KONT,DMAX                               RRU 0160
      IF (EOF    ,5) 5551,5552
 5551 WRITE(6,5553)
      IF (KPLOT .NE. 1) CALL EXIT
      CALL PLOT (0,0,999)
      CALL EXIT
 5552 WRITE (6,7755) CKT                                                RRU 0170
      TITLE(4) = CKT                                                    RRU 0180
      WRITE (6,7753) LIMITT,KONT                                        RRU 0190
      TLIMIT = LIMITT - 20                                              RRU 0200
      IF (DMAX .GT.0.0)WRITE (6,4446) DMAX                              RRU 0210
C     READ-WRITE MONITOR SPECIFICATIONS
      READ (5,7752) (EMVALU(I),I=1,12),NDTMIN,NDTMAX                    RRU 0220
      MONTOR = .FALSE.                                                  RRU 0230
      SOFAR = .FALSE.                                                   RRU 0240
      DO 5550 I = 1,12                                                  RRU 0250
      MONEY (I) = .NOT. ((EMVALU(I).EQ.HOL ) .OR. (EMVALU(I).EQ.HOLO))  RRU 0260
 5550 SOFAR = SOFAR .OR. MONEY(I)                                       RRU 0270
      IF (SOFAR) WRITE (6,1111) NDTMIN,NDTMAX                           RRU 0280
      NDELT = 0                                                         RRU 0290
      NITRSM= 0                                                         RRU 0300
      JCAT  = 1                                                         RRU 0310
      JKKJ  = 0                                                         RRU 0320
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     READ-WRITE INSTRUCTIONS FOR SAVING FINAL VARIABLE VALUES FROM A
C     PREVIOUS MODE TO BE USED AS INITIAL VALUES
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ (5, 10 )          NSVVAR,NSVPHI,NSVIL                        RRU 0330
      IF (NSVVAR .EQ. 1) NSVPHI = 1                                     RRU 0340
      IF (NSVVAR .EQ. 1) NSVIL  = 1                                     RRU 0350
      WRITE(6,7756) NSVVAR, NSVPHI,NSVIL                                RRU 0360
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     READ-WRITE CONVERGENCE CONTROL DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
      READ (5,11)   ITRTST,MAXMIN,KFMAX,RELEP,ABSER,N38                  RRU 0370
      WRITE(6,7777) ITRTST,MAXMIN,KFMAX                                   RRU 0380
      ITRMAX = MAXMIN                                                     RRU 0390
      IF(RELER .LE.0.0)RELER=1.0E-6                                       RRU 0400
      IF(ABSER .LE.0.0)ABSER=1.0E-6                                       RRU 0410
      WRITE(6,800) RELER,ABSER                                            RRU 0420
      IF ((NSVVAR .EQ. 1) .OR.  (KONT .EQ. 1)) GO TO 7718                 RRU 0430
      DO 5117 I = 1,19                                                    RRU 0440
 5117 TOO(I)=0.0                                                          RRU 0450
      TE    = 0.0                                                         RRU 0460
      JV    = 1                                                           RRU 0470
      GO TO 7720                                                          RRU 0480
 7718 TE     = TE - DELT                                                  RRU 0490
 7720 TIMEIN = TE                                                         RRU 0500
      JKLJ   = 0                                                          RRU 0510
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK MONITOR SET UP
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (NDELT .LT. NDTMIN) GO TO 7426                                   RRU 0520
      IF (NDTMAX .LT. NDELT) GO TO 7426                                   RRU 0530
      IF (MONTOR) GO TO 1112                                              RRU 0540
      MONTOR = .TRUE.                                                     RRU 0550
      MONMA  = MONMAT                                                     RRU 0560
      MONS   = MONST                                                      RRU 0570
      MONELM = MONELT                                                     RRU 0580
      WSQSOL = WSQSLT                                                     RRU 0590
      MONIV  = MONIVT                                                     RRU 0600
      MONR   = MONRT                                                      RRU 0610
      MONC   = MONCT                                                      RRU 0620
      MONL   = MONLT                                                      RRU 0630
      MOND   = MONDT                                                      RRU 0640
      MONT   = MONTT                                                      RRU 0650
      MONCOR = MONCRT                                                     RRU 0660
      MONZEN = MONZNT                                                     RRU 0670
      IF (MONMA ) GO TO 3333                                              RRU 0680
      IF (MONS  ) GO TO 3333                                              RRU 0690
      IF (MONELM) GO TO 3333                                              RRU 0700
      IF (WSQSOL) GO TO 3333                                              RRU 0710
      IF (MONIV ) GO TO 3333                                              RRU 0720
      IF (MONR  ) GO TO 3333                                              RRU 0730
      IF (MONC  ) GO TO 3333                                              RRU 0740
      IF (MONL  ) GO TO 3333                                              RRU 0750
      IF (MOND  ) GO TO 3333                                              RRU 0760
      IF (MONT  ) GO TO 3333                                              RRU 0770
      IF (MONCOR) GO TO 3333                                              RRU 0780
      IF (MONZEN) GO TO 3333                                              RRU 0790
      GO TO 1112                                                          RRU 0800
 3333 WRITE (6,6671) (MVALUE (I),I=1,12)                                  RRU 0810
      GO TO 1112                                                          RRU 0820
 7426 IF (.NOT. MONTOR) GO TO 1112                                        RRU 0830
      IF (MONMA) WRITE (6,6672)                                           RRU 0840
      MONTOR = .FALSE.                                                    RRU 0850
      MONMA  = .FALSE.                                                    RRU 0860
      MONS   = .FALSE.                                                    RRU 0870
      MONELM = .FALSE.                                                    RRU 0880
      WSQSOL = .FALSE.                                                    RRU 0890
      MONIV  = .FALSE.                                                    RRU 0900
      MONR   = .FALSE.                                                    RRU 0910
      MONC   = .FALSE.                                                    RRU 0920
      MONL   = .FALSE.                                                    RRU 0930
      MOND   = .FALSE.                                                    RRU 0940
      MONT   = .FALSE.                                                    RRU 0950
      MONCOR = .FALSE.                                                    RRU 0960
      MONZEN = .FALSE.                                                    RRU 0970
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     READ-WRITE GENERAL RUN CONTROLS
```

```
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 1112 READ  (5,10) NV,NAUX,NG,ND,NB,NBB                                RRU 0980
      WRITE (6,31) NV,NAUX,NG,N3B,ND,NB,NBB                            RRU 0990
      NZ = NV + NAUX                                                   RRU 1000
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     READ-WRITE PLOT AND PRINT-PLOT SPECIFICATIONS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ (5,10) NPLTVT,N1B,NPLTVV                                    RRU 1010
      IF (NPLTVT .EQ. 0) GO TO 8990                                    RRU 1020
      READ  (5,836) KPLOT,KPRPLT,SCALE,IPX,PRSYMB,KPFIT                RRU 1030
      IF (KPRPLT .EQ.  1 ) GO TO 1113                                  RRU 1040
      IF (KPLOT  .NE.  1 ) GO TO 1113                                  RRU 1050
      IF (SCALE  .NE. 0.0) GO TO 1113                                  RRU 1060
        KPRPLT = 1                                                     RRU 1070
 1113 WRITE (6,34) SCALE , NPLTVT,NPLTVV,N1B,KPRPLT,KPLOT,IPX,KPFIT    RRU 1080
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PLOTS OF VARIABLE VS. TIME.  A VARIABLE NUMBER IS ENTERED NEGATIVE
C     IF IT IS TO BE PLOTTED ONLY VS. ANOTHER VARIABLE(S)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ  (5,5) (AAAA(J),NPP(J),J=1,NPLTVT)                          RRU 1090
      IF(N1B .GT. 0) GO TO 37                                          RRU 1100
      WRITE (6,30)(AAAA(J),NPP(J),J=1,NPLTVT)                          RRU 1110
      GO TO 38                                                         RRU 1120
   37 WRITE (6,35)(AAAA(J),NPP(J),J=1,NPLTVT)                          RRU 1130
   38 WRITE (6,33)                                                     RRU 1140
      IF (NPLTVV .EQ. 0) GO TO 39                                      RRU 1150
      READ  (5,10)(NP1(J,1),NP1(J,2),J=1,NPLTVV)                       RRU 1160
      WRITE (6,41)(NP1(J,1),NP1(J,2),J=1,NPLTVV)                       RRU 1170
      DO 2230 J=1,2                                                    RRU 1180
      DO 2230 K=1,NPLTVV                                               RRU 1190
      DO 2231 L=1,NPLTVT                                               RRU 1200
      IF ((IABS(NPP(L)) .EQ. NP1(K,J)) GO TO 2230                      RRU 1210
 2231 CONTINUE                                                         RRU 1220
      WRITE (6,2232)K                                                  RRU 1230
      CALL EXIT                                                        RRU 1240
 2230 NP1(K,J)=L                                                       RRU 1250
   39 K = NPLTVT* (N1B+1) + NPLTVV                                     RRU 1260
      IF (K .LE. 0) GO TO 8990                                         RRU 1270
      IF (SCALE  .EQ. 0.0) GO TO 8989                                  RRU 1280
      XSCALE(20) =  0.0                                                RRU 1290
      WRITE (6,7900)                                                   RRU 1300
      DO 7902 I = 1 , K                                                RRU 1310
       READ (5,7903) Q1,XUNIT(1,I),XUNIT(2,I),XSCALE(I),XLEFT(I),      RRU 1320
     2       XRIGHT(I),Q2,YUNIT(1,I),YUNIT(2,I),YSCALE(I),YBOTTM(I),   RRU 1330
     3          YTOP(I)                                               RRU 1340
      WRITE(6,7904)I,Q1,XUNIT(1,I),XUNIT(2,I),XSCALE(I),XLEFT(I),      RRU 1350
     2       XRIGHT(I),Q2,YUNIT(1,I),YUNIT(2,I),YSCALE(I),YBOTTM(I),   RRU 1360
     3          YTOP(I)                                               RRU 1370
 7902 CONTINUE                                                         RRU 1380
      GO TO 8990                                                       RRU 1390
 8989 XSCALE(20) = -1.0                                                RRU 1400
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK IF UPPER LIMITS OF NUMBER OF VARIABLES AND VARIABLE SOURCES
C          ARE EXCEEDED
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 8990 IF(NV .LE. 60) GO TO 3407                                        RRU 1410
      WRITE (6,3409)                                                   RRU 1420
      CALL EXIT                                                        RRU 1430
 3407 IF(NZ .LE. 70) GO TO 3414                                        RRU 1440
      WRITE (6,3413)                                                   RRU 1450
      CALL EXIT                                                        RRU 1460
 3414 IF(ND .LE. 18) GO TO 3417                                        RRU 1470
      WRITE (6,3419)                                                   RRU 1480
      CALL EXIT                                                        RRU 1490
 3417 IF (NSVVAR .NE. 0) GO TO 7722                                    RRU 1500
      DO 81 J = 1,NZ                                                   RRU 1510
```

```
      V (J)=0.0                                                          RRU 1520
   81 V1(J)=0.0                                                          RRU 1530
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     READ-WRITE TEMPERATURE AND DELT-MODE SPECIFICATIONS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 7722 READ  (5,25)  ABTEMP,NDTSPC                                        RRU 1540
      WRITE (6,61)  ABTEMP                                               RRU 1550
      IF (NDTSPC .EQ. 0) NDTSPC = 1                                      RRU 1560
      READ  (5,20)  (  CT(J,1),CT(J,2),J=1,NDTSPC)                       RRU 1570
      WRITE (6,5112)(J,CT(J,1),CT(J,2),J=1,NDTSPC)                       RRU 1580
      IF (NG .NE. 1) GO TO 134                                           RRU 1590
      READ  (5,20)(V1(J),J=1,NZ)                                         RRU 1600
  134 TI =CT(1,1)                                                        RRU 1610
      DELT = TI                                                          RRU 1620
      IF (NG .EQ. -1) DELT = 1.0E+20                                     RRU 1630
      TO1=TI                                                             RRU 1640
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     READ-WRITE VARIABLE-SOURCE DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      ND =ND+1                                                           RRU 1650
      CALL HOLLER(1HO,CC)                                                RRU 1660
      IF (DMAX .NE. 0.0) CALL HOLLER (1H ,CC)                            RRU 1670
      WRITE (6,33)                                                       RRU 1680
      DO 118 J=1,ND                                                      RRU 1690
      JG(J) = 1                                                          RRU 1700
      READ  (5,10)NI,NI2                                                 RRU 1710
      IF (NI .NE. 0) GO TO 500                                           RRU 1720
      IF (J .GT. 1) GO TO 500                                            RRU 1730
      NI = 2                                                             RRU 1740
      ST(1,1) = NI                                                       RRU 1750
      ST(1,2) = NI2                                                      RRU 1760
      E(1,1,1) = 0.0                                                     RRU 1770
      E(1,1,2) = 0.0                                                     RRU 1780
      E(1,2,1) = 0.0                                                     RRU 1790
      E(1,2,2) = 1.0E+10                                                 RRU 1800
      K = 19                                                             RRU 1810
      RMAG = 0                                                           RRU 1820
      GO TO 5122                                                         RRU 1830
  500 ST(J,1)=NI                                                         RRU 1840
      ST(J,2)=NI2                                                        RRU 1850
      READ  (5,20)(E(J,L,1),E(J,L,2),L=1,NI)                             RRU 1860
      IF (DMAX .EQ.0.0)GO TO 4444                                        RRU 1870
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     MODIFY NOMINAL SOURCE VALUES BY DMAX PERCENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ (5,4445) S1                                                   RRU 1880
      S1 = S1 * DMAX                                                     RRU 1890
      WRITE (6,4449) S1                                                  RRU 1900
      DO 4447 L=1,NI                                                     RRU 1910
 4447 E(J,L,1) = E(J,L,1) * (1.0 + S1 / 100.0)                           RRU 1920
 4444 IF(J .GT. 1) GO TO 8501                                            RRU 1930
  501 K=19                                                               RRU 1940
      DO 8555 L=1,NI                                                     RRU 1950
 8555 E(1,L,1)=E(1,L,1)*E(1,1,2)                                         RRU 1960
      RMAG    =E(1,1,2)                                                  RRU 1970
      E(1,1,2)=0.0                                                       RRU 1980
      WRITE (6,5116)RMAG,(E(1,L,1),E(1,L,2),L=1,NI)                      RRU 1990
      GO TO 5122                                                         RRU 2000
 8501 K=J-1                                                              RRU 2010
      IF(NI2)5118,5119,5120                                              RRU 2020
 5118 WRITE (6,5121) CC,K,E(J,1,1)                                       RRU 2030
      GO TO 5122                                                         RRU 2040
 5119 WRITE (6,5123) CC,K,(E(J,L,1),E(J,L,2),L=1,NI)                     RRU 2050
      GO TO 5122                                                         RRU 2060
 5120 WRITE (6,5124) CC,K,(E(J,L,1),E(J,L,2),L=1,NI)                     RRU 2070
 5122 IF (E(J,2,2) .NE.0.0)E2(J) = (E(J,2,1)-E(J,1,1)) / E(J,2,2)        RRU 2080
```

```
      EI (K)=E(J,1,1)                                                    RRU 2090
      EP1(K)=EI(K)                                                       RRU 2100
      IF (MONMA) WRITE (6,1125) J,K,NI,NI2,ST(J,1),ST(J,2),RMAG,EI(J)    RRU 2110
  118 CONTINUE                                                           RRU 2120
      IF(MONMA)WRITE (6,6661)       JCAT,TE,TIMEIN,TI,DELT,RMAG          RRU 2130
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     READ-WRITE NORMAL-TERMINATION CONDITIONS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      CALL     HOLLER(1HA,HOLLA)                                         RRU 2140
      CALL     HOLLER(1HG,HOLLG)                                         RRU 2150
      CALL     HOLLER(3HABS,HOLABS)                                      RRU 2160
      CALL     HOLLER(3H   ,HOLBLN)                                      RRU 2170
      CALL     HOLLER(2HLE,HOLLE)                                        RRU 2180
      CALL     HOLLER(2HGT,HOLGT)                                        RRU 2190
      READ (5, 10 ) IOC                                                  RRU 2200
      IF (IOC .LE. 0) GO TO 7926                                         RRU 2210
      WRITE (6,7928) IOC                                                 RRU 2220
      DO 7932 I=1,IOC                                                    RRU 2230
          READ (5, 10 ) JACK                                             RRU 2240
          IAC(I) = JACK                                                  RRU 2250
          WRITE (6,7930) I,JACK                                          RRU 2260
          DO 7931 J = 1,JACK                                             RRU 2270
               READ (5,7933) A1,NOVAR(I,J),A2,BVAL(I,J)                  RRU 2280
               IF (A1 .EQ. HOLLA) GO TO 7938                             RRU 2290
               A1 = HOLBLN                                               RRU 2300
               IF (A2 .EQ. HOLLG) GO TO 7937                             RRU 2310
               ICS(I,J) = 1                                              RRU 2320
               A2 = HOLLE                                                RRU 2330
               GO TO 7941                                                RRU 2340
 7937          ICS(I,J) = 2                                              RRU 2350
               GO TO 7841                                                RRU 2360
 7938          A1 = HOLABS                                               RRU 2370
               IF (A2 .EQ. HOLLG) GO TO 7940                             RRU 2380
               ICS(I,J) = -1                                             RRU 2390
               A2 = HOLLE                                                RRU 2400
               GO TO 7941                                                RRU 2410
 7940          ICS(I,J) = -2                                             RRU 2420
 7841          A2 = HOLGT                                                RRU 2430
 7941          WRITE (6,7934) A1,NOVAR(I,J),A2,BVAL(I,J)                 RRU 2440
 7931     CONTINUE                                                       RRU 2450
 7932 CONTINUE                                                           RRU 2460
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     READ-WRITE FAILURE-TERMINATION CONDITIONS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 7926 READ (5, 10 ) IOCF                                                 RRU 2470
      IF (IOCF .LE. 0) RETURN                                            RRU 2480
      WRITE (6,7948) IOCF                                                RRU 2490
      DO 7952 I = 1,IOCF                                                 RRU 2500
          READ (5, 10 ) JACK                                             RRU 2510
          IACF(I) = JACK                                                 RRU 2520
          WRITE (6,7950) I,JACK                                          RRU 2530
          DO 7951 J = 1,JACK                                             RRU 2540
               READ (5,7933) A1,NOVARF(I,J),A2,BVALF(I,J)                RRU 2550
               IF (A1 .EQ. HOLLA) GO TO 7958                             RRU 2560
               A1 = HOLBLN                                               RRU 2570
               IF (A2 .EQ. HOLLG) GO TO 7957                             RRU 2580
               ICSF(I,J) = 1                                             RRU 2590
               A2 = HOLLE                                                RRU 2600
               GO TO  7961                                               RRU 2610
 7957          ICSF(I,J) = 2                                             RRU 2620
               GO TO  7861                                               RRU 2630
 7958          A1 = (+HOLABS)                                            RRU 2640
               IF (A2 .EQ. HOLLG) GO TO 7960                             RRU 2650
               ICSF(I,J) = -1                                            RRU 2660
               A2 = HOLLE                                                RRU 2670
               GO TO  7961                                               RRU 2680
```

```
 7960           ICSF(I,J) = -2                                                  RRU 2690
 7861           A2 = HOLGT                                                      RRU 2700
 7961           WRITE (6,7934) A1,NOVARF(I,J),A2,BVALF(I,J)                     RRU 2710
 7951      CONTINUE                                                             RRU 2720
 7952  CONTINUE                                                                 RRU 2730
       RETURN                                                                   RRU 2740
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
    5  FORMAT (6(A6,I6))                                                       RRU 2750
   10  FORMAT (6I12)                                                           RRU 2760
   11  FORMAT (3I12,2F12.8,I12)                                                RRU 2770
   20  FORMAT (6F12.8)                                                         RRU 2780
   25  FORMAT (E12.5,I12)                                                      RRU 2790
   30  FORMAT (    32HOPLOTS OF VARIABLE VS. TIME     ,/,52H (SAVED BUT NRRU 2800
     2OT PLOTTED IF VARIABLE NO. IS NEGATIVE) ,/10H  VARIABLE ,/,            RRU 2810
     3     13H NAME     NO.  ,/,(1H ,A6,I4))                                  RRU 2820
   31  FORMAT (14HOGENERAL DATA                                                RRU 2830
     2 ,/ 6H NV   =,I3,38H   (NUMBER OF UNKNOWN NODAL VOLTAGES)  ,            RRU 2840
     3  / 6H NAUX=,I3,33H   (NUMBER OF AUXILIARY UNKNOWNS) ,                  RRU 2850
     4  / 5H NG =I4,54H   (-1=CALCULATE VALUES, 0=ZERO VALUES, 1=READ VARRU 2860
     5LUES)          ,                                                       RRU 2870
     8 /,5H N3B=,I4,3X,43H(-1 = INITIAL CONDITIONS ONLY, 0 = HALT IF     RRU 2880
     9     ,  54HERROR IN I.C., 1 = PRINT IF ERROR IN I.C. BUT PROCEED),RRU 2890
     5/5H ND =I4,31H   (NUMBER OF VARIABLE SOURCES)/5H NB =I4,49H   (RATRRU 2900
     6IO OF COMPUTED POINTS TO PLOTTED POINTS)  /5H NBB=I4,46H   (RATIO RRU 2910
     7OF PLOTTED POINTS TO PRINTED POINTS)   )                               RRU 2920
   33  FORMAT (1H1)                                                            RRU 2930
   34  FORMAT (25HOPLOT AND PRINT-PLOT DATA  ,/,                               RRU 2940
     1       (9H SCALE  =,F3.0,34H  (1 = READ SCALE FROM CARDS, 0 =      RRU 2950
     2  11HOTHERWISE)  ,/,9H NPLTVT = ,I3,38H  (NUMBER OF VARIABLES TO BRRU 2960
     XE PLOTTED)                                                             RRU 2970
     3  ,/,9H NPLTVV = ,I3,28H  (NUMBER OF V-VS.-V PLOTS)  ,/,  9H N1B  RRU 2980
     4  = ,I3,2X,50H(1 = PLOT VDOT VS. V PHASE PLANES, 0 = OTHERWISE)  /RRU 2990
     5    9H KPRPLT =  ,I3,41H  (1 = PRINT PLOT, 0 = DO NOT PRINT PLOT  RRU 3000
     X      16HAT NORMAL EXIT)     ,/,                                      RRU 3010
     6    9H KPLOT  =  ,I3,31H  (1 = PLOT, 0 = DO NOT PLOT)  ,/,          RRU 3020
     7    9H IPX    =  ,I3,39H  (NUMBER OF PLOTTED POINTS PER FRAME) ,/,RRU 3030
     8    9H KPFIT  =  ,I3,84H  (0=INDEPENDENT SCALES AND BOUNDARIES FORRRU 3040
     9 FRAMES, 1=SAVE Y SCALES AND Y BOUNDARIES, ,/,15X,79H 2=SAVE X SCARRU 3050
     XLES, Y SCALES, AND Y BOUNDARIES, 3=SAVE ALL SCALES AND BOUNDARIES)RRU 3060
     X)                                                                      RRU 3070
   35  FORMAT (    66HOPLOTS OF VARIABLE VS. TIME AND VARIABLE DERIVITIVERRU 3080
     2 VS. VARIABLE  ,/,52H (SAVED BUT NOT PLOTTED IF VARIABLE NO. IS NERRU 3090
     3GATIVE) ,        /9H VARIABLE/12H  NAME   NO. /(X,A6,I4))          RRU 3100
   41  FORMAT (    32HOPLOTS OF VARIABLE VS. VARIABLE /26H VARIABLE X    RRU 3110
     2VARIABLE Y  /(I7,6X,I7))                                               RRU 3120
   61  FORMAT(1HO,/23HOABSOLUTE TEMPERATURE=  ,E13.5)                          RRU 3130
  600  FORMAT (67HO*** THE PLOT ROUTINE IS CALLED FOR BUT NO PLOTS ARE SPRRU 3140
     2ECIFIED ***    )                                                       RRU 3150
  800  FORMAT (                          7H RELER=,E12.5,9H ABSER = ,E12.5,RRU 3160
     2       24H FOR INITIAL CONDITIONS     )                               RRU 3170
  836  FORMAT (2I12,F12.0,I12,11X,A1,I12)                                      RRU 3180
 1111  FORMAT (35H MONITORING SPECIFIED FROM NDELT = ,I4,12H  TO NDELT = RRU 3190
     2    ,I4)                                                               RRU 3200
 1125  FORMAT (38H J,K,NI,NI2,ST(J,1),ST(J,2),RMAG,EI(J) ,4I10,/,4E24.15)RRU 3210
 2232  FORMAT (    41HODATA MISSING FOR UNKNOWN VS UNKNOWN PLOT 13)        RRU 3220
 3409  FORMAT (    28HONO OF NODES GREATER THAN 60)                        RRU 3230
 3413  FORMAT (    31HONO OF UNKNOWNS GREATER THAN 70)                     RRU 3240
 3419  FORMAT (    30HONO OF SOURCES GREATER THAN 18 )                     RRU 3250
 4445  FORMAT (F12.0)                                                          RRU 3260
 4446  FORMAT (31HOW O R S T   C A S E   F O R  ,F2.0,  10H PERCENT     RRU 3270
     2   58H D E V I A T I O N   O F   P A R A M E T E R   V A L U E S/)RRU 3280
 4449  FORMAT (1HO,F3.0,24H PERCENT DEVIATION FOR     )                        RRU 3290
 5112  FORMAT (1HO/26H NO.   DELT      END TIME   /(I3,2E12.4))                RRU 3300
 5116  FORMAT (    12H IPP SOURCE  /15H SCALE FACTOR = E12.4 /20H MAGNITURRU 3310
     2DE      TIME /(2E12.4))                                                RRU 3320
 5121  FORMAT (A1,    13HDC SOURCE NO. ,I3,/,12H MAGNITUDE =  ,E12.4)     RRU 3330
```

```
5123 FORMAT (A1,  19HVARIABLE SOURCE NO. ,I3,/,20H LAST MAGNITUDE HELD RRU 3340
    2    ,/,21H MAGNITUDE      TIME     ,/,(2E12.4))                  RRU 3350
5124 FORMAT (A1,  19HVARIABLE SOURCE NO. I3 / 6H CYCLE /20H MAGNITUDE  RRU 3360
    2    TIME /(2E12.4))                                              RRU 3370
5553 FORMAT (17HOEND OF FILE READ)                                    RRU 3380
6661 FORMAT (48H            JCAT, TE, TIMEIN, TI, DELT, RMAG     , I7, RRU 3390
    2     /5E24.15)                                                   RRU 3400
6671 FORMAT (15H MONITORS ON.. ,12L4)                                 RRU 3410
6672 FORMAT (29H MONITORS TURNED OFF AT 7426    )                     RRU 3420
7751 FORMAT (A6,6X,2I12,F12.0)                                        RRU 3430
7752 FORMAT (12(A1,1X),2I12)                                          RRU 3440
7753 FORMAT ( 1X,17HOVERALL-RUN DATA , /,8H LIMITT= ,I3,              RRU 3450
    2    50H (COMPUTER-RUN PROCESS TIME = LIMITT-20  SECONDS) ,/,     RRU 3460
    3     7H KONT = ,I3,36H (1 = CONTINUED RUN, 0 = OTHERWISE) )      RRU 3470
7755 FORMAT (1H1,35X,16H C I R C U I T    ,A6,//)                     RRU 3480
7756 FORMAT ( 20H SAVE-VARIABLE DATA ,/, 11H NSVVAR  = ,I3,           RRU 3490
    2        53H (1 = SAVE ALL TIME VARIABLES FOR THE FOLLOWING MODE,, RRU 3500
    3        16H 0 = OTHERWISE) ,/, 11H NSVPHI  = ,I3,               RRU 3510
    4        50H (1 = SAVE ALL FLUX VALUES FOR THE FOLLOWING MODE,,  RRU 3520
    5        16H 0 = OTHERWISE) ,/, 11H NSVIL   = ,I3,               RRU 3530
    6        58H (1 = SAVE INDUCTOR-CURRENT VALUES FOR THE FOLLOWING MORRU 3540
    7DE,     16H 0 = OTHERWISE) )                                    RRU 3550
7777 FORMAT(17HOCONVERGENCE DATA  ,/,                                 RRU 3560
    1       (10H ITRTST  =,I3,102H (NUMBER OF TEST ITERATIONS BELOW WHIRRU 3570
    2CH DELT IS MULTIPLIED BY 2 AND ABOVE WHICH DELT IS DIVIDED BY 2)  RRU 3580
    3  ,/,10H MAXMIN  = ,I3,99H (INITIAL MAXIMUM NUMBER OF ITERATIONS BRRU 3590
    4EFORE VARIABLE VALUES ARE RESET AND DELT IS DIVIDED BY 4)       ,/RRU 3600
    5 10H KFMAX   =,I3,54H (MAXIMUM NUMBER OF TRIALS BEFORE ITRMAX IS CRRU 3610
    6HANGED)    )                                                     RRU 3620
7900 FORMAT (      1HO,  52X,18HP L O T   D A T A ,/,1HO,             RRU 3630
    2          ,/,121H PLOT NO.  X       X-UNITS        X-SCALE     LERRU 3640
    3FT        RIGHT    Y       Y-UNITS        Y-SCALE      BOTTOM    RRU 3650
    4  TOP     ,/)                                                   RRU 3660
C7903 FORMAT (A6,6X,2A6,3E12.5)                                       RRU 3670
 7903 FORMAT (A6,6X,2R6,3E12.5)                                       RRU 3670
 7904 FORMAT (1X,I4,4X,A6,2X,2R6,E12.3,2E11.2,3X,A6,2X,2R6,E12.3,2E11.2)RRU 3680
C7904 FORMAT (1X,I4,4X,A6,2X,2A6,E12.3,2E11.2,3X,A6,2X,2A6,E12.3,2E11.2)RRU 3680
 7928 FORMAT (      35H1NORMAL-TERMINATION SPECIFICATIONS  //,         RRU 3690
    2        5HOIOC= ,I2,/)                                          RRU 3700
 7930 FORMAT (4HOAC( ,I2,3H)=  ,I2)                                   RRU 3710
 7933 FORMAT (A1,8X,I3,A1,11X,E12.5)                                  RRU 3720
 7934 FORMAT (1H ,A3,1X,I2,1X,A2,E12.5)                               RRU 3730
 7948 FORMAT (/     36HOFAILURE-TERMINATION SPECIFICATIONS  /,         RRU 3740
    2        6HOIOCF=  ,I2,/)                                        RRU 3750
 7950 FORMAT (5HOACF( ,I2,3H)=  ,I2)                                  RRU 3760
 7975 FORMAT (13HOPRO. TIME =  ,E12.3)                                RRU 3770
      END                                                             RRU 3780
```

```
      SUBROUTINE PLAPR (KEND)                                          PLA 0010
C     PLAPR CONTROLS PLOTTING (CALLS SUBROUTINE PLT),
C     PRINT-PLOTTING (CALLS PRPLT), AND CARD PUNCHING
```

COMMON VARIABLES

```
      LOGICAL MONMA, SKIPIT,SKIPX,SKIPFR,SKIPXY
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
      IF (MONMA ) WRITE (6,9999)                                       PLA 0020
      IGRAPH = 0                                                       PLA 0030
      IF (KEND) 310,150,30                                             PLA 0040
  150 IF (KXT .NE. 0) RETURN                                           PLA 0050
      KXT =  1                                                         PLA 0060
      JV  =  JV - 1                                                    PLA 0070
      WRITE (6,5)JV,TO1                                                PLA 0080
      IF (ND .LE. 0) GO TO 700                                         PLA 0090
      ND1=ND-1                                                         PLA 0100
      WRITE (6,70)EP1(19),(I,EP1(I),I=1,ND1)                           PLA 0110
  700 TE= TE-DELT                                                      PLA 0120
      WRITE (6,50)TE,(I ,V1( I), I=1,NZ)                               PLA 0130
   30 IF (KPUNCH .NE. 1) GO TO 310                                     PLA 0140
      TE = TE - DELT                                                   PLA 0150
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PUNCH CARDS FOR CONTINUED RUN
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      WRITE (6,11) NDELT,TE,DELT,DTM1,DTM2,NITRSM                      PLA 0160
      PUNCH    11, NDELT,TE,DELT,DTM1,DTM2,NITRSM                      PLA 0170
      TE = TE + DELT                                                   PLA 0180
      WRITE (6,10) (TOO(I),I = 1, 19)                                  PLA 0190
      PUNCH  10  , (TOO(I),I = 1, 19)                                  PLA 0200
      WRITE (6,10) (EP1(I),I = 1,ND1)                                  PLA 0210
      PUNCH  10  , (EP1(I),I = 1,ND1)                                  PLA 0220
      WRITE (6,10) (V1( I), I=1,NZ)                                    PLA 0230
      PUNCH 10,(V1( I), I=1,NZ)                                        PLA 0240
      IF (NC .LE. 0) GO TO 330                                         PLA 0250
      DO 51 I=1,NC                                                     PLA 0260
      WRITE (6,10) CIC(I)                                              PLA 0270
   51 PUNCH 10 , CIC(I)                                                PLA 0280
  330 IF (NL .LE. 0) GO TO 170                                         PLA 0290
      DO 52 I=1,NL                                                     PLA 0300
      WRITE (6,10) CIL(I)                                              PLA 0310
   52 PUNCH 10 ,CIL(I)                                                 PLA 0320
  170 IF (NDIODE .LE. 0) GO TO 171                                     PLA 0330
      DO 177 I=1,NDIODE                                                PLA 0340
      WRITE (6,10) DV(I),DR(I),DI(I)                                   PLA 0350
  177 PUNCH 10    ,DV(I),DR(I),DI(I)                                   PLA 0360
  171 IF (NTRANS .LE. 0) GO TO 173                                     PLA 0370
      DO 179 I=1,NTRANS                                                PLA 0380
      WRITE (6,10) BCV(I),BCR(I),BCI(I),BEV(I),BER(I),BEI(I)           PLA 0390
  179 PUNCH 10    ,BCV(I),BCR(I),BCI(I),BEV(I),BFR(I),BEI(I)           PLA 0400
  173 IF (NCORE .LE. 0) GO TO 310                                      PLA 0410
      DO 181 I=1,NCORE                                                 PLA 0420
      WRITE(6, 10 )   FM3(I),FM2(I),FM1(I),PHIM3(I),PHIM2(I),PHIM1(I), PLA 0430
     2     PDTEM1(I),PHDTM2(I),PHDTM1(I)                               PLA 0440
      PUNCH   10    , FM3(I),FM2(I),FM1(I),PHIM3(I),PHIM2(I),PHIM1(I), PLA 0450
     2     PDTEM1(I),PHDTM2(I),PHDTM1(I)                               PLA 0460
      K = NW(I)                                                        PLA 0470
      DO 182 J  = 1,K                                                  PLA 0480
      WRITE (6,10)             CICORE(I,J)                             PLA 0490
  182 PUNCH 10                ,CICORE(I,J)                             PLA 0500
  181 CONTINUE                                                         PLA 0510
  310 CALL  AUX (-3)                                                   PLA 0520
      IF (KPRPLT .EQ. 1) GO TO 555                                     PLA 0530
      IF (KPLOT .NE. 1) RETURN                                         PLA 0540
  555 IF (JV .LT. 2) RETURN                                            PLA 0550
```

```
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PLOTTING AND PRINT PLOTTING
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      TFJV = TF(JV)                                                    PLA 0560
      SKIPIT = (NDELT .GT. IPX*NB) .AND. (KPFIT .GT. 0).AND.(KONT.EQ.0) PLA 0570
      SKIPX  = SKIPIT .AND. (KPFIT .EQ. 1)                             PLA 0580
      SKIPFR = SKIPIT .AND. (KPFIT .EQ. 2)                             PLA 0590
      SKIPXY = SKIPIT .AND. (KPFIT .EQ. 3)                             PLA 0600
      IF (NPLTVT .EQ. 0) GO TO 206                                     PLA 0610
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     V-VS.-T PLOTS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      CALL HOLLER(3H      ,TITLE(2))                                   PLA 0620
  109 DO 306 I=1,NPLTVT                                                PLA 0630
      JV1=NPP(I)                                                       PLA 0640
      K = IABS(JV1)                                                    PLA 0650
      VP(I,JV)=V( K )                                                  PLA 0660
      IF (JV1 .LT. 0) GO TO 306                                        PLA 0670
      BCDT=AAAA(I)                                                     PLA 0680
      TITLE(1) = BCDT                                                  PLA 0690
      IF (SKIPXY) GO TO 71                                             PLA 0700
      XL = +1.0E30                                                     PLA 0710
      XR = -1.0E30                                                     PLA 0720
      YT = -1.0E30                                                     PLA 0730
      YTP= -1.0E30                                                     PLA 0740
      YB = +1.0E30                                                     PLA 0750
      YBP= +1.0E30                                                     PLA 0760
   71 DO 307 K=1,JV                                                    PLA 0770
      XP(K)=VP(I,K)                                                    PLA 0780
      IF (SKIPXY) GO TO 307                                            PLA 0790
      XL=AMIN1(XL,TF(K))                                               PLA 0800
      XR=AMAX1(XR,TF(K))                                               PLA 0810
      IF (XP(K) .GT. YB ) GO TO 716                                    PLA 0820
      IF (XP(K) .LT. YBP) GO TO 715                                    PLA 0830
         YB  = XP(K)                                                   PLA 0840
         GO TO 716                                                     PLA 0850
  715 YB  = YBP                                                        PLA 0860
      YBP = XP(K)                                                      PLA 0870
  716 IF (XP(K) .LT. YT ) GO TO 307                                    PLA 0880
      IF (XP(K) .GT. YTP) GO TO 717                                    PLA 0890
         YT  = XP(K)                                                   PLA 0900
         GO TO 307                                                     PLA 0910
  717 YT  = YTP                                                        PLA 0920
      YTP = XP(K)                                                      PLA 0930
  307 CONTINUE                                                         PLA 0940
      IF (SKIPXY) GO TO 718                                            PLA 0950
      IF (SKIPFR) GO TO 817                                            PLA 0960
      IF (SKIPX ) GO TO 818                                            PLA 0970
         SAVEXL(I,1) = XL                                              PLA 0980
         SAVEXR(I,1) = XR                                              PLA 0990
         SAVEYB(I,1) = YB                                              PLA 1000
         SAVEYT(I,1) = YT                                              PLA 1010
         GO TO 719                                                     PLA 1020
  817 IF (XR .LE. SAVEXR(I,1)) GO TO 718                               PLA 1030
      XRP = (SAVEXR(I,1) - SAVEXL(I,1)) / 8.0                          PLA 1040
      XLP = XRP * AMIN1(7.0,AINT((XR-SAVEXR(I,1))/XRP)) + XRP          PLA 1050
      XL = SAVEXL(I,1) + XLP                                           PLA 1060
      XR = SAVEXR(I,1) + XLP                                           PLA 1070
      GO TO 818                                                        PLA 1080
  718 XL = SAVEXL(I,1)                                                 PLA 1090
      XR = SAVEXR(I,1)                                                 PLA 1100
  818 YB = SAVEYB(I,1)                                                 PLA 1110
      YT = SAVEYT(I,1)                                                 PLA 1120
  719 IF (KPRPLT .EQ. 1) CALL PPPLT (JV,-1,JV1,0)                      PLA 1130
      IF (KPLOT  .EQ. 1) CALL PLT   (JV,-1,JV1,0)                      PLA 1140
  306 CONTINUE                                                         PLA 1150
      IF(N1B .LE. 0) GO TO 2202                                        PLA 1160
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     VDOT-VS.-V (PHASE PLANE) PLOTS
```

```
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      JVV=JV-2                                                          PLA 1170
      CALL HOLLER(3HDOT   ,TITLE(2))                                    PLA 1180
      M = 0                                                             PLA 1190
      DO 2201 I=1,NPLTVT                                                PLA 1200
      JV1=NPP(I)                                                        PLA 1210
      IF (JV1 .LT. 0) GO TO 2201                                        PLA 1220
      BCDT=AAAA(I)                                                      PLA 1230
      BCDT1 = BCDT                                                      PLA 1240
      TITLE(1) = BCDT                                                   PLA 1250
      IF (SKIPIT) GO TO 72                                              PLA 1260
      YT = -1.0E30                                                      PLA 1270
      YTP= -1.0E30                                                      PLA 1280
      YB = +1.0E30                                                      PLA 1290
      YBP= +1.0E30                                                      PLA 1300
   72 DO 2200 K=1,JVV                                                   PLA 1310
      IF (M .GT. 0) GO TO 3406                                          PLA 1320
      XP(K)=(VP(I,K+2)-VP(I,K))/(TF(K+2)-TF(K))                         PLA 1330
      GO TO 3407                                                        PLA 1340
 3406 XP(K)=(VP(I,K+2)-VP(I,K))*XP(K)/(VP(M,K+2)-VP(M,K))               PLA 1350
 3407 TF(K)=VP(I,K+1)                                                   PLA 1360
      IF (SKIPIT) GO TO 2200                                            PLA 1370
      IF (XP(K) .GT. YB ) GO TO 726                                     PLA 1380
      IF (XP(K) .LT. YBP) GO TO 725                                     PLA 1390
         YB  = XP(K)                                                    PLA 1400
         GO TO 726                                                      PLA 1410
  725 YB  = YBP                                                         PLA 1420
      YBP = XP(K)                                                       PLA 1430
  726 IF (XP(K) .LT. YT ) GO TO 2200                                    PLA 1440
      IF (XP(K) .GT. YTP) GO TO 727                                     PLA 1450
         YT  = XP(K)                                                    PLA 1460
         GO TO 2200                                                     PLA 1470
  727 YT  = YTP                                                         PLA 1480
      YTP = XP(K)                                                       PLA 1490
 2200 CONTINUE                                                          PLA 1500
      IF (SKIPIT) GO TO 728                                             PLA 1510
         XL          = SAVEYB(I,1)                                      PLA 1520
         SAVEXL(I,2) = XL                                               PLA 1530
         XR          = SAVEYT(I,1)                                      PLA 1540
         SAVEXR(I,2) = XR                                               PLA 1550
         SAVEYB(I,2) = YB                                               PLA 1560
         SAVEYT(I,2) = YT                                               PLA 1570
         GO TO 729                                                      PLA 1580
  728 XL = SAVEXL(I,2)                                                  PLA 1590
      XR = SAVEXR(I,2)                                                  PLA 1600
      YB = SAVEYB(I,2)                                                  PLA 1610
      YT = SAVEYT(I,2)                                                  PLA 1620
  729 IF (KPRPLT .EQ. 1) CALL PRPLT(JVV,0, JV1,0)                       PLA 1630
      IF (KPLOT  .EQ. 1) CALL PLT  (JVV,0, JV1,0)                       PLA 1640
      M = I                                                             PLA 1650
 2201 CONTINUE                                                          PLA 1660
 2202 IF (NPLTVV .LE. 0) GO TO 206                                      PLA 1670
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     V-VS.-V PLOTS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      CALL HOLLER(3H      ,TITLE(2))                                    PLA 1680
      DO 2205 I=1,NPLTVV                                                PLA 1690
      I1=NP1(I,1)                                                       PLA 1700
      I2=NP1(I,2)                                                       PLA 1710
      BCDT=AAAA(I1)                                                     PLA 1720
      BCDT1=AAAA(I2)                                                    PLA 1730
      TITLE(1) = BCDT1                                                  PLA 1740
      IF (SKIPIT) GO TO 73                                              PLA 1750
      XL = +1.0E30                                                      PLA 1760
      XLP= +1.0E30                                                      PLA 1770
      XR = -1.0E30                                                      PLA 1780
      XRP= -1.0E30                                                      PLA 1790
      YT = -1.0E30                                                      PLA 1800
      YTP= -1.0E30                                                      PLA 1810
      YB = +1.0E30                                                      PLA 1820
      YBP= +1.0E30                                                      PLA 1830
```

```
 73 DO 2206 K=1,JV                                                      PLA 1840
    TF(K)=VP(I1,K)                                                      PLA 1850
    XP(K)=VP(I2,K)                                                      PLA 1860
    IF (SKIPIT) GO TO 2206                                              PLA 1870
    IF (TF(K) .GT. XL ) GO TO 732                                       PLA 1880
    IF (TF(K) .LT. XLP) GO TO 731                                       PLA 1890
       XL  = TF(K)                                                      PLA 1900
       GO TO 732                                                        PLA 1910
731 XL  = XLP                                                           PLA 1920
    XLP = TF(K)                                                         PLA 1930
732 IF (TF(K) .LT. XR ) GO TO 734                                       PLA 1940
    IF (TF(K) .GT. XRP) GO TO 733                                       PLA 1950
       XR  = TF(K)                                                      PLA 1960
       GO TO 734                                                        PLA 1970
733 XR  = XRP                                                           PLA 1980
    XRP = TF(K)                                                         PLA 1990
734 IF (XP(K) .GT. YB ) GO TO 736                                       PLA 2000
    IF (XP(K) .LT. YBP) GO TO 735                                       PLA 2010
       YB  = XP(K)                                                      PLA 2020
       GO TO 736                                                        PLA 2030
735 YB  = YBP                                                           PLA 2040
    YBP = XP(K)                                                         PLA 2050
736 IF (XP(K) .LT. YT ) GO TO 2206                                      PLA 2060
    IF (XP(K) .GT. YTP) GO TO 737                                       PLA 2070
       YT  = XP(K)                                                      PLA 2080
       GO TO 2206                                                       PLA 2090
737 YT  = YTP                                                           PLA 2100
    YTP = XP(K)                                                         PLA 2110
2206 CONTINUE                                                           PLA 2120
    IF (SKIPIT) GO TO 738                                               PLA 2130
       SAVEXL(I,3) = XL                                                 PLA 2140
       SAVEXR(I,3) = XR                                                 PLA 2150
       SAVEYB(I,3) = YB                                                 PLA 2160
       SAVEYT(I,3) = YT                                                 PLA 2170
       GO TO 739                                                        PLA 2180
 738 XL = SAVEXL(I,3)                                                   PLA 2190
     XR = SAVEXR(I,3)                                                   PLA 2200
     YB = SAVEYB(I,3)                                                   PLA 2210
     YT = SAVEYT(I,3)                                                   PLA 2220
 739 I1=NPP(I1)                                                         PLA 2230
     I2=NPP(I2)                                                         PLA 2240
     IF (KPRPLT .EQ. 1) CALL PRPLT(JV, 1, I1,I2)                        PLA 2250
     IF (KPLOT  .EQ. 1) CALL PLT  (JV, 1, I1,I2)                        PLA 2260
2205 CONTINUE                                                           PLA 2270
 206 WRITE (6,5)                                                        PLA 2280
     IF (JV .LT. IPX) GO TO 1206                                        PLA 2290
     DO 1205 I = 1,NPLTVT                                               PLA 2300
        VP(I,1) = VP(I,JV)                                              PLA 2310
1205 CONTINUE                                                           PLA 2320
     TF(1) = TFJV                                                       PLA 2330
     JV = 1                                                             PLA 2340
     RETURN                                                             PLA 2350
1206 JV = 0                                                             PLA 2360
     RETURN                                                             PLA 2370
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
   5 FORMAT (1H1/I6,E13.5)                                              PLA 2380
  10 FORMAT     (6E12.5   )                                             PLA 2390
  11 FORMAT (I12,4E12.5,I12)                                            PLA 2400
  50 FORMAT (1H /6H TIME=E12.5/9H UNKNOWNS/6(I4,E13.5))                 PLA 2410
  70 FORMAT (1H /17H IONIZING PULSE = E13.5/7H SOURCE/6(I4,F13.5))       PLA 2420
9999 FORMAT (8H PLAPR   )                                               PLA 2430
     END                                                                PLA 2440
```

```
      SUBROUTINE PRPLT                    (NN,NLC,ID,IL)                PRP 0010
C     PRPLT HANDLES PRINT PLOTTING
```

COMMON VARIABLES

```
      EQUIVALENCE (H,G)                                                 PRP 0020
      DIMENSION G(81,41)                                                PRP 0030
      DIMENSION X(14)                                                   PRP 0040
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
C     CHECK PLOT RANGE
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
      SX = XR - XL                                                      PRP 0050
      IF (SX .LE. 0.0) GO TO 31                                         PRP 0060
      SY = YT - YB                                                      PRP 0070
      IF (SY .GT. 0.0) GO TO 33                                         PRP 0080
   31 WRITE (6,32)  TITLE(1),TITLE(2)                                   PRP 0090
      GO TO 35                                                          PRP 0100
   33 CALL HOLLER(1H ,ABLANK)                                           PRP 0110
      CALL HOLLER(1H.,ADOT)                                             PRP 0120
      IF (PRSYMB .EQ. ABLANK) CALL HOLLER(1H*,PRSYMB)                   PRP 0130
      AASS = PRSYMB                                                     PRP 0140
      CALL HOLLER(1H+,APLUS)                                            PRP 0150
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
C     BLANK ARRAY AND FILL IN VERTICAL AND HORIZONTAL GRID LINES
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
      DO 50 J = 1,41                                                    PRP 0160
      DO 50 I = 1,81                                                    PRP 0170
       G(I,J) = ABLANK                                                  PRP 0180
      IF (I .EQ. 81) GO TO 160                                          PRP 0190
      IF (I .EQ.  1) GO TO 160                                          PRP 0200
      IF (J .EQ. 41) GO TO 160                                          PRP 0210
      IF (J .EQ.  1) GO TO 160                                          PRP 0220
      IF ( MOD (I,10) .NE. 1) GO TO 159                                 PRP 0230
      IF (J .EQ.  2) GO TO 160                                          PRP 0240
      IF (J .EQ. 40) GO TO 160                                          PRP 0250
  159 IF ( MOD (J,5) .NE. 1) GO TO 50                                   PRP 0260
      IF (I .EQ.  2) GO TO 160                                          PRP 0270
      IF (I .LT. 80) GO TO 50                                           PRP 0280
  160  G(I,J)= ADOT                                                     PRP 0290
   50 CONTINUE                                                          PRP 0300
      WRITE (6,401)                                                     PRP 0310
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
C     DETERMINE BOUNDS AND TIC SPACING FOR GRAPH
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
      TX = ALOG10(SX/7.0) + 1000.0                                      PRP 0320
      Z = TX -AINT(TX)                                                  PRP 0330
      IF (Z .EQ. 0.0) GO TO 89                                          PRP 0340
      IF (Z .GT. 0.30103) GO TO 81                                      PRP 0350
      TX =AINT(TX) + 1.0299956E-3 + 0.3                                 PRP 0360
      GO TO 89                                                          PRP 0370
   81 IF (Z .GT. 0.69898) GO TO 82                                      PRP 0380
      TX =AINT(TX) - 1.0299956E-3 + 0.7                                 PRP 0390
      GO TO 89                                                          PRP 0400
   82 TX =AINT(TX+1.0)                                                  PRP 0410
   89 SX = 10.0 ** (1001.0-TX)                                          PRP 0420
      TT = 10.0 / SX                                                    PRP 0430
      TX =AINT(XL/TT) * TT                                              PRP 0440
      IF (XL .LT. TX) TX = TX - TT                                      PRP 0450
      XL = TX                                                           PRP 0460
      TY = ALOG10(SY/7.0) + 1000.0                                      PRP 0470
                                                                        PRP 0480
```

```
      IF (Z .EQ. 0.0) GO TO 79                                          PRP 0490
      IF (Z .GT. 0.30103) GO TO 71                                      PRP 0500
      TY =AINT(TY) + 1.0299956E-3 + 0.3                                 PRP 0510
      GO TO 79                                                          PRP 0520
   71 IF (Z .GT. 0.69898) GO TO 72                                      PRP 0530
      TY =AINT(TY) - 1.0299956E-3 + 0.7                                 PRP 0540
      GO TO 79                                                          PRP 0550
   72 TY =AINT(TY+1.0)                                                  PRP 0560
   79 SY = 10.0** (1000.7-TY-1.0299956E-3)                              PRP 0570
      TT = 5.0 / SY                                                     PRP 0580
      TY =AINT(YB/TT) * TT                                              PRP 0590
      IF (YB .LT. TY) TY = TY - TT                                      PRP 0600
      YB = TY                                                           PRP 0610
      IF (KPLOT .EQ. 0) GO TO 106                                       PRP 0620
      IF (XSCALE(20) .GT. -0.5) GO TO 106                               PRP 0630
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     SET SCALES AND FRAME FOR PLT SUBROUTINE (IF SCALE = 1)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      K = IGRAPH + 1                                                    PRP 0640
      XLEFT (K)  = XL                                                   PRP 0650
      XRIGHT(K)  = XL   + 80.0 / SX                                     PRP 0660
      XSCALE(K)  =        (XRIGHT(K) - XL) / 8.0                        PRP 0670
      IF (NLC .NE. -1) GO TO 90                                         PRP 0680
      CALL HOLLER(6RSECOND,XUNIT(1,K))                                  PRP 0690
      CALL HOLLER(6RS     ,XUNIT(2,K))                                  PRP 0700
      GO TO 92                                                          PRP 0710
   90 CALL HOLLER (6RMKS UN,XUNIT(1,K))                                 PRP 0720
      CALL HOLLER (6RITS   ,XUNIT(2,K))                                 PRP 0730
   92 YBOTTM(K)  = YB                                                   PRP 0740
      YTOP  (K)  = YB + 40.0 / SY                                       PRP 0750
      YSCALE(K)  =        (YTOP(K) - YB) / 8.0                          PRP 0760
      CALL HOLLER (6RMKS UN,YUNIT(1,K))                                 PRP 0770
      CALL HOLLER (6RITS   ,YUNIT(2,K))                                 PRP 0780
  106 CSX = SX * 10.0                                                   PRP 0790
      CSY = SY * 20.0                                                   PRP 0800
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     TRANSFORM X AND Y DATA FROM TF(K) AND XP(K) ARRAYS INTO
C          G(I,J) SYMBOL ARRAY.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DO 206 K = 1, NN                                                  PRP 0810
      II =(TF(K) -  XL ) * CSX + .5                                     PRP 0820
      JI =(XP(K) -  YB ) * CSY + .5                                     PRP 0830
      I = 1 + (II + 5) / 10                                             PRP 0840
      J = 1 + (JI + 10) / 20                                            PRP 0850
      IF (I .LT. 1) I = 1                                               PRP 0860
      IF (I .GT. 81) I = 81                                             PRP 0870
      IF (J .GT. 41) J = 41                                             PRP 0880
      J = 42 - J                                                        PRP 0890
      IF (J .GT. 41) J = 41                                             PRP 0900
      IF (G(I,J) .EQ. APLUS) GO TO 206                                  PRP 0910
      IF (G(I,J) .NE. AASS) GO TO 180                                   PRP 0920
      G(I,J) = APLUS                                                    PRP 0930
      GO TO 206                                                         PRP 0940
  180  G(I,J)= AASS                                                     PRP 0950
  206 CONTINUE                                                          PRP 0960
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PRINT-PLOT SYMBOL ARRAY  G(I,J)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      KI = 1                                                            PRP 0970
      DO 407 I = 1,40                                                   PRP 0980
      IF (KI .NE. 1) GO TO 307                                          PRP 0990
      C = 41 - I                                                        PRP 1000
```

```
      DC = YB + C / SY                                                    PRP 1010
      IF (ABS(DC) .LE. 1.0E-4*ABS(YB)) DC = 0.0                           PRP 1020
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     LABEL Y AXIS IN LINES 21(VARIABLE), 23(UNITS), AND 26(V(NO.))
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (I .NE. 21) GO TO 306                                            PRP 1030
      WRITE (6,120) TITLE(1),TITLE(2),CC,(G(L,I),L=1,81)                  PRP 1040
      GO TO 207                                                           PRP 1050
  306 IF (I .NE. 26) GO TO 406                                            PRP 1060
      IF (NLC .EQ. 0) GO TO 406                                           PRP 1070
      IF (NLC .EQ. -1) GO TO 404                                          PRP 1080
      J = IABS(IL)                                                        PRP 1090
      GO TO 405                                                           PRP 1100
  404 J = IABS(ID)                                                        PRP 1110
  405 WRITE (6,121) J ,DC,(G(L,I),L=1,81)                                 PRP 1120
      GO TO 207                                                           PRP 1130
  406 WRITE (6, 20)    DC,(G(L,I),L=1,81)                                 PRP 1140
      GO TO 207                                                           PRP 1150
  307 IF (I .NE. 23) GO TO 308                                            PRP 1160
      WRITE (6, 23)       (G(L,I),L=1,81)                                 PRP 1170
      GO TO 207                                                           PRP 1180
  308 WRITE (6, 22)       (G(L,I),L=1,81)                                 PRP 1190
      IF (KI .LT. 5) GO TO 207                                            PRP 1200
      KI= 0                                                               PRP 1210
  207 KI=KI+1                                                             PRP 1220
  407 CONTINUE                                                            PRP 1230
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     LABEL X-AXIS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      WRITE (6,20) YB,(G(L,41),L=1,81)                                    PRP 1240
      IG = 0                                                              PRP 1250
      DO 208 I = 1,81, 10                                                 PRP 1260
      IG = IG + 1                                                         PRP 1270
      C = I - 1                                                           PRP 1280
      X(IG) = XL + C / SX                                                 PRP 1290
      IF (ABS(X(IG)) .LE. 1.0E-4*ABS(XL)) X(IG)= 0.0                      PRP 1300
  208 CONTINUE                                                            PRP 1310
      WRITE              (6, 21) (X(L),   L = 1, IG)                      PRP 1320
   35 IF (NLC)            3,4,4                                          PRP 1330
    3 WRITE(6,96)                                                         PRP 1340
      WRITE(6, 6) TITLE(4)                                                PRP 1350
      RETURN                                                              PRP 1360
    4 J = IABS(ID)                                                        PRP 1370
      WRITE(6,98) BCDT,J                                                  PRP 1380
      WRITE(6, 6) TITLE(4)                                                PRP 1390
      RETURN                                                              PRP 1400
C . . . . . .
    6 FORMAT (1HO,52X,8HCIRCUIT ,A6)                                      PRP 1410
   20 FORMAT ( 4X, E15.2, 2X, 81A1  )                                     PRP 1420
   21 FORMAT  (15X, 9(10.2)                                               PRP 1430
   22 FORMAT (21X81A1)                                                    PRP 1440
   23 FORMAT (13H (MKS UNITS) ,7X,81A1)                                   PRP 1450
   32 FORMAT (16H-NO PRINTER PLOT,5X,A6,A3)                               PRP 1460
   96 FORMAT (1HO,54X,              ,10HTIME (SEC)  ,/1HO)                PRP 1470
   98 FORMAT (1HO,46X,A6,20H (MKS UNITS),       2HV(, I2, 1H),/,1HO)     PRP 1480
  120 FORMAT (1H , A6,A3         , E9.2,2X,81A1)                         PRP 1490
  121 FORMAT (3H V(, I2,1H),E14.2,2X,81A1)                               PRP 1500
  401 FORMAT(1H1/1H ,30X12A6)                                            PRP 1510
      END                                                                 PRP 1520
```

```
      SUBROUTINE PLT (N,J2,JV1,JV2)                                      PLT 0010
C     PLT HANDLES PLOTTING
```

COMMON VARIABLES

```
      INTEGER XUNIT,YUNIT                                                PLT 0020
      LOGICAL BULL                                                       PLT 0030
      DIMENSION TAR1(2),TAR4(15),TAR5(32),IAR2(2)                        PLT 0040
      DIMENSION IBUF(1024)                                               PLT 0050
      DATA DOT /3HDOT/                                                   PLT 0060
      DATA PAREN1/6H(     /                                              PLT 0070
      DATA TIME /6HTIME  /                                               PLT 0080
      DATA  VSPT /6H  VS. /                                              PLT 0090
      DATA MIC/3RMIC/                                                    PLT 0100
      DATA MIL/3RMIL/                                                    PLT 0110
      DATA MAKS/3RMAX/                                                   PLT 0120
      DATA NAN/3RNAN/                                                    PLT 0130
      DATA IPC/3RPIC/                                                    PLT 0140
C     . . . . . .
      IGRAPH = IGRAPH + 1                                                PLT 0150
      IF (CT(11,2) .NE. 123.456) CALL PLCTS(IBUF,1024,7)                 PLT 0160
      CT(11,2) = 123.456                                                 PLT 0170
      ZERO = 0.0                                                         PLT 0180
      ONE = 1.0                                                          PLT 0190
      XLEFT1 = XLEFT(IGRAPH)                                             PLT 0200
      RIGHT  = XRIGHT(IGRAPH)                                            PLT 0210
      TOP = YTOP(IGRAPH)                                                 PLT 0220
      BOTTOM = YBOTTM(IGRAPH)                                            PLT 0230
      FIXX = 1.0 / XSCALE(IGRAPH)                                        PLT 0240
      FIXY = 1.0 / YSCALE(IGRAPH)                                        PLT 0250
      XZERO = -XLEFT1 * FIXX                                             PLT 0260
      YZERO = ONE - BOTTOM * FIXY                                        PLT 0270
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DRAW FRAME
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      CALL PLOT (ZERO,ONE,3)                                             PLT 0280
      XTIC = RIGHT * FIXX + XZERO                                        PLT 0290
      CALL PLOT (XTIC,ONE,2)                                             PLT 0300
      YTIC = TOP * FIXY + YZERO                                          PLT 0310
      CALL PLOT (XTIC,YTIC,2)                                            PLT 0320
      CALL PLOT (ZERO,YTIC,2)                                            PLT 0330
      CALL PLOT (ZERO,ONE,2)                                             PLT 0340
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DETERMINE SCALE OF X-AXIS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      ITEM = XUNIT(1,IGRAPH) / 10000008                                  PLT 0350
      IF (ITEM - MIC) 10, 9,10                                           PLT 0360
    9 RATIO = 1.0E6                                                      PLT 0370
      GO TO 30                                                           PLT 0380
   10 IF (ITEM - MIL) 12,11,12                                           PLT 0390
   11 RATIO = 1.0E3                                                      PLT 0400
      GO TO 30                                                           PLT 0410
   12 IF (ITEM - MAKS)14,13,14                                           PLT 0420
   13 RATIO = 1.0E8                                                      PLT 0430
      GO TO 30                                                           PLT 0440
   14 IF (ITEM - NAN) 16,15,16                                           PLT 0450
   15 RATIO = 1.0E9                                                      PLT 0460
```

```
      GO TO 30                                                          PLT 0470
   16 IF (ITEM - IPC ) 18,17,18                                         PLT 0480
   17 RATIO = 1.0E12                                                    PLT 0490
      GO TO 30                                                          PLT 0500
   18 RATIO = 1.0                                                       PLT 0510
   30 TEMP = XSCALE(IGRAPH) * RATIO                                     PLT 0520
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DETERMINE TIC SPACING FOR X-AXIS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      XMARK=ALOG10(TEMP)                                                PLT 0530
      IF (INT(XMARK) .EQ. XMARK) GO TO 21                               PLT 0540
      IF (XMARK) 19,29,21                                               PLT 0550
   19 XMARK = 10.0 ** (INT(XMARK) - 1.0)                                PLT 0560
      GO TO 20                                                          PLT 0570
   21 XMARK = 10.0 **  INT(XMARK)                                       PLT 0580
   20 XTIC = ABS(XMARK/TEMP)                                            PLT 0590
      IF (XTIC - 0.2828427712475) 22,23,23                              PLT 0600
   22 IF (XTIC - 0.1414213562371 24,25,25                               PLT 0610
   24 XTIC = 10.0 * XTIC                                                PLT 0620
      XMARK = 10.0 * XMARK                                              PLT 0630
      GO TO 29                                                          PLT 0640
   25 XTIC = 5.0 * XTIC                                                 PLT 0650
      XMARK = 5.0 * XMARK                                               PLT 0660
   23 IF (XTIC - 0.707106781191 26,29,29                                PLT 0670
   26 XTIC = XTIC + XTIC                                                PLT 0680
      XMARK = XMARK + XMARK                                             PLT 0690
   29 ICOUNT = (RIGHT - XLEFT1) * FIXX / XTIC + 2.00000001              PLT 0700
      TEMP = XLEFT1 * FIXX / XTIC + 0.999999999                         PLT 0710
      IF (INT(TEMP) .EQ. TEMP) GO TO 28                                 PLT 0720
      IF (TEMP) 27,28,28                                                PLT 0730
   27 TEMP = TEMP - 1.0                                                 PLT 0740
   28 TEMP = INT(TEMP)                                                  PLT 0750
      YTIC = TEMP * XTIC - XTIC + XZERO                                 PLT 0760
      TIC1 = YTIC                                                       PLT 0770
      TMARK1 = TEMP * XMARK                                             PLT 0780
      X =TMARK1                                                         PLT 0790
      DO 31 I = 1,ICOUNT                                                PLT 0800
      IF (ABS(10000.0*X) .LT. XMARK) X = 0.0                            PLT 0810
      TAR5(I) = X                                                       PLT 0820
   31 X = X + XMARK                                                     PLT 0830
      DO 110 J =1,ICOUNT,2                                              PLT 0840
  110 ENCODE(20,32,TAR4(J)) TAR5(J),TAR5(J+1)                           PLT 0850
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     WRITE NUMBERS ON X-AXIS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      TEMP = RIGHT * FIXX + XZERO + 0.010                               PLT 0860
      TIC1 = XTIC + TIC1                                                PLT 0870
      BULL = .TRUE.                                                     PLT 0880
      DO 35 I = 1,1000                                                  PLT 0890
          TAR4(I) = TAR4(I)                                             PLT 0900
      BULL = .NOT. BULL                                                 PLT 0910
      IF (BULL) GO TO 34                                                PLT 0920
      TEMP2 = TIC1-0.48                                                 PLT 0930
      CALL SYMBOL(TEMP2,0.73,0.14,TAR4,0.0,8)                           PLT 0940
   34     CALL PLOT(TIC1,ONE,3)                                         PLT 0950
          CALL PLOT(TIC1,1.1,2)                                         PLT 0960
          TIC1 = TIC1 + XTIC                                            PLT 0970
          IF (TIC1 - TEMP) 35,36,36                                     PLT 0980
   35 CONTINUE                                                          PLT 0990
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PLOT TICS ON TOP AND BOTTOM FRAME
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   36 TIC1 = YTIC                                                       PLT 1000
      YTIC = TOP * FIXY + YZERO                                         PLT 1010
      DO 37 I = 1,1000                                                  PLT 1020
          TIC1 = XTIC + TIC1                                            PLT 1030
```

```
      CALL PLOT (TIC1,YTIC,3)                                          PLT 1040
      TEMP2 = YTIC - 0.1                                               PLT 1050
      CALL PLOT (TIC1,TEMP2,2)                                         PLT 1060
      IF (TIC1 + XTIC - TEMP) 37,38,38                                 PLT 1070
   37 CONTINUE                                                         PLT 1080
   38 TEMP = (XLEFT1 + RIGHT) * 0.5 * FIXX + XZERO                     PLT 1090
      IF (J2) 41,40,40                                                 PLT 1100
   40 TAR1(1) = BCDT                                                   PLT 1110
      GO TO 42                                                         PLT 1120
   41 TAR1(1) = TIME                                                   PLT 1130
   42 TEMP2 = TEMP - 1.34                                              PLT 1140
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     WRITE TITLES AND DIMENSIONS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      CALL SYMBOL (TEMP-2.01,0.11,0.21,TAR1,0.0,6)                     PLT 1150
      CALL SYMBOL (TEMP-0.75,0.11,0.21,PAREN1,0.0,1)                   PLT 1160
      IAR2(1) = XUNIT(1,IGRAPH)                                        PLT 1170
      IAR2(2) = XUNIT(2,IGRAPH)                                        PLT 1180
      CALL PAREN(IAR2,I)                                               PLT 1190
      CALL SYMBOL (TEMP-0.54,0.11,0.21,IAR2,0.0,I)                     PLT 1200
      CALL SYMBOL (TEMP-0.72,-0.46,0.28,VSPT,0.0,6)                    PLT 1210
      CALL SYMBOL (TEMP+1.00,-0.46,0.28,TAR1,0.0,6)                    PLT 1220
      IF (J2) 44,45,43                                                 PLT 1230
   43 BCDT = BCDT1                                                     PLT 1240
   44 TAR1(1) = BCDT                                                   PLT 1250
      CALL SYMBOL (TEMP-2.12,-0.46,0.28,TAR1,0.0,6)                    PLT 1260
      TEMP = (TOP + BOTTOM) * 0.5 * FIXY + YZERO                       PLT 1270
      CALL SYMBOL (-1.50,TEMP-2.01,0.21,TAR1,90.0,6)                   PLT 1280
      GO TO 46                                                         PLT 1290
   45 TAR1(1) = BCDT                                                   PLT 1300
      CALL SYMBOL (TEMP-3.08,-0.46,0.28,TAR1,0.0,6)                    PLT 1310
      CALL SYMBOL (TEMP-1.40,-0.46,0.28,DOT,0.0,3)                     PLT 1320
      TEMP=(TOP+BOTTOM)*0.5*FIXY+YZERO                                 PLT 1330
      CALL SYMBOL (-1.50,TEMP-2.73,0.21,TAR1,90.0,6)                   PLT 1340
      CALL SYMBOL (-1.50,TEMP-1.47,0.21,DOT,90.0,3)                    PLT 1350
   46 CALL SYMBOL (-1.50,TEMP-0.75,0.21,PAREN1,90.0,1)                 PLT 1360
      IAR2(1) = YUNIT(1,IGRAPH)                                        PLT 1370
      IAR2(2) = YUNIT(2,IGRAPH)                                        PLT 1380
      CALL PAREN(IAR2,I)                                               PLT 1390
      CALL SYMBOL (-1.50,TEMP-0.54,0.21,IAR2,90.0,I)                   PLT 1400
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DETERMINE SCALE OF Y-AXIS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      ITEM = YUNIT(1,IGRAPH) / 1000000B                                PLT 1410
      IF (ITEM - MIC) 50,49,50                                         PLT 1420
   49 RATIO = 1.0E6                                                    PLT 1430
      GO TO 60                                                         PLT 1440
   50 IF (ITEM - MIL) 52,51,52                                         PLT 1450
   51 RATIO = 1.0E3                                                    PLT 1460
      GO TO 60                                                         PLT 1470
   52 IF (ITEM - MAKS)54,53,54                                         PLT 1480
   53 RATIO = 1.0E8                                                    PLT 1490
      GO TO 60                                                         PLT 1500
   54 IF (ITEM - NAN) 56,55,56                                         PLT 1510
   55 RATIO = 1.0E9                                                    PLT 1520
      GO TO 60                                                         PLT 1530
   56 IF (ITEM - IPC) 58,57,58                                         PLT 1540
   57 RATIO = 1.0E12                                                   PLT 1550
      GO TO 60                                                         PLT 1560
   58 RATIO = 1.0                                                      PLT 1570
   60 TEMP = YSCALE(IGRAPH) * RATIO                                    PLT 1580
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DETERMINE TIC SPACING FOR Y-AXIS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      YMARK = ALOG10(TEMP)                                             PLT 1590
      IF (INT(YMARK) .EQ. YMARK) GO TO 61                              PLT 1600
```

```
      IF (YMARK.LT.0.0) YMARK = YMARK - 1.0                              PLT 1610
   61 YMARK = 10.0 ** INT(YMARK)                                         PLT 1620
      YTIC = ABS(YMARK / TEMP          )                                 PLT 1630
      IF (YTIC - 0.282842712475) 62,63,63                               PLT 1640
   62 IF (YTIC - 0.141421356237) 64,65,65                               PLT 1650
   64 YTIC = 10.0 * YTIC                                                PLT 1660
      YMARK = 10.0 * YMARK                                              PLT 1670
      GO TO 69                                                          PLT 1680
   65 YTIC = 5.0 * YTIC                                                 PLT 1690
      YMARK = 5.0 * YMARK                                               PLT 1700
   63 IF (YTIC - 0.707106781191 66,69,69                                PLT 1710
   66 YTIC = YTIC + YTIC                                                PLT 1720
      YMARK = YMARK + YMARK                                             PLT 1730
   69 ICOUNT = (TOP - BOTTOM) * FIXY / YTIC + 2.000000001               PLT 1740
      TEMP =       BOTTOM * FIXY / YTIC + 0.999999999                   PLT 1750
      IF (TEMP .EQ. INT(TEMP)) GO TO 200                                PLT 1760
      IF (TEMP .LT. 0.0) TEMP= TEMP -1.0                                PLT 1770
      TEMP= INT(TEMP)                                                   PLT 1780
  200 XTIC = TEMP * YTIC - YTIC + YZERO                                 PLT 1790
      TIC1 = XTIC                                                       PLT 1800
      TMARK1= TEMP * YMARK                                              PLT 1810
      Y = TMARK1                                                        PLT 1820
      DO 70 I = 1,ICOUNT                                                PLT 1830
      IF (ABS(10000.0*Y) .LT. YMARK) Y = 0.0                            PLT 1840
      TAR5(I) = Y                                                       PLT 1850
   70 Y = Y + YMARK                                                     PLT 1860
      DO 111 J =1,ICOUNT,2                                              PLT 1870
  111 ENCODE(20,32,TAR4(J)) TAR5(J),TAR5(J+1)                           PLT 1880
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     WRITE NUMBERS ON Y-AXIS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      TEMP = TOP * FIXY + YZERO + 0.010                                 PLT 1890
      DO 71 I = 1,1000                                                  PLT 1900
          TIC1 = YTIC + TIC1                                            PLT 1910
          TAR4(1) = TAR4(I)                                             PLT 1920
      TEMP2 = TIC1 - 0.07                                               PLT 1930
      CALL SYMBOL(-1.25,TEMP2,0.14,TAR4,0.0,8)                          PLT 1940
          CALL PLOT(0.0,TIC1,3)                                         PLT 1950
          CALL PLOT(0.1,TIC1,2)                                         PLT 1960
          IF (TIC1 + YTIC - TEMP) 71,72,72                              PLT 1970
   71 CONTINUE                                                          PLT 1980
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PLOT TICS ON RIGHT AND LEFT FRAME
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   72 TIC1 = XTIC                                                       PLT 1990
      XTIC = RIGHT * FIXX + XZERO                                       PLT 2000
      TEMP2 = XTIC - 0.1                                                PLT 2010
      DO 73 I = 1,1000                                                  PLT 2020
          TIC1 = YTIC + TIC1                                            PLT 2030
          CALL PLOT (XTIC,TIC1, 3)                                      PLT 2040
          CALL PLOT (TEMP2,TIC1, 2)                                     PLT 2050
          IF (TIC1+YTIC-TEMP) 73,733,733                                PLT 2060
   73 CONTINUE                                                          PLT 2070
  733 TMARK1 = 0                                                        PLT 2080
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PLOT DATA POINTS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DO 1100 I = 1,N                                                   PLT 2090
          IF (TF(I)-XLEFT1) 74,77,75                                    PLT 2100
   74     TMARK1 = 0                                                    PLT 2110
          GO TO 100                                                     PLT 2120
   75     IF (TF(I)-RIGHT) 77,77,76                                     PLT 2130
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     HANDLE DATA POINTS THAT ARE OUT OF FRAME AT RIGHT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   76     IF (TMARK1) 78,100,78                                         PLT 2140
```

```
   78     YVALUE = (XP(I) - XP(I-1))*(RIGHT - TF(I-1))                    PLT 2150
    2          / (TF(I) - TF(I-1)) + XP(I-1)                             PLT 2160
          IF (YVALUE - BOTTOM) 80,81,81                                  PLT 2170
   80     XVALUE =((RIGHT - TF(I-1)) * (BOTTOM - XP(I-1))                PLT 2180
    2          / (YVALUE - XP(I-1)) + TF(I-1)) * FIXX + XZERO            PLT 2190
          YVALUE = BOTTOM                                                PLT 2200
          GO TO 85                                                       PLT 2210
   81     IF (YVALUE - TOP) 83,83,82                                     PLT 2220
   82     XVALUE =((RIGHT - TF(I-1)) * (TOP - XP(I-1))                   PLT 2230
    2          / (YVALUE - XP(I-1)) + TF(I-1)) * FIXX + XZERO            PLT 2240
          YVALUE = TOP                                                   PLT 2250
          GO TO 85                                                       PLT 2260
   83     XVALUE =  RIGHT * FIXX + XZERO                                 PLT 2270
   85     YVALUE = YVALUE * FIXY + YZERO                                 PLT 2280
          TMARK1 = 0                                                     PLT 2290
          GO TO 99                                                       PLT 2300
   77     IF (XP(I) - BOTTOM) 86,91,87                                   PLT 2310
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     HANDLE DATA POINTS THAT ARE OUT OF FRAME AT BOTTOM
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   86     IF (TMARK1) 88,100,88                                          PLT 2320
   88     XVALUE = ((TF(I) - TF(I-1)) * (BOTTOM - XP(I-1))               PLT 2330
    2          / (XP(I) - XP(I-1)) + TF(I-1)) * FIXX + XZERO             PLT 2340
          YVALUE = BOTTOM * FIXY + YZERO                                 PLT 2350
          TMARK1 = 0                                                     PLT 2360
          GO TO 99                                                       PLT 2370
   87     IF (XP(I) - TOP) 91,91,90                                      PLT 2380
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     HANDLE DATA POINTS THAT ARE OUT OF FRAME AT TOP
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   90     IF (TMARK1) 92,100,92                                          PLT 2390
   92     XVALUE = ((TF(I) - TF(I-1)) * (TOP - XP(I-1))                  PLT 2400
    2          / (XP(I) - XP(I-1)) + TF(I-1)) * FIXX + XZERO             PLT 2410
          YVALUE = TOP * FIXY + YZERO                                    PLT 2420
          TMARK1 = 0                                                     PLT 2430
          GO TO 99                                                       PLT 2440
   91     IF (TMARK1) 98,94,98                                           PLT 2450
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     FIRST IN-FRAME VALUE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   94     TMARK1= 1                                                      PLT 2460
          IF (I - 1) 102,102,103                                         PLT 2470
  103     IF (XP(I-1) - BOTTOM) 95,102,96                                PLT 2480
   95     XVALUE = ((TF(I) - TF(I-1)) * (BOTTOM - XP(I-1))               PLT 2490
    2          / (XP(I) - XP(I-1)) + TF(I-1)) * FIXX + XZERO             PLT 2500
          YVALUE = BOTTOM * FIXY + YZERO                                 PLT 2510
          GO TO 101                                                      PLT 2520
   96     IF (XP(I-1) - TOP) 102,102,97                                  PLT 2530
   97     XVALUE = ((TF(I) - TF(I-1)) * (TOP - XP(I-1))                  PLT 2540
    2          / (XP(I) - XP(I-1)) + TF(I-1)) * FIXX + XZERO             PLT 2550
          YVALUE = TOP * FIXY + YZERO                                    PLT 2560
  101     CALL PLOT (XVALUE,YVALUE, 3)                                   PLT 2570
          GO TO 98                                                       PLT 2580
  102     YVALUE = XP(I) * FIXY + YZERO                                  PLT 2590
          XVALUE = TF(I) * FIXX + XZERO                                  PLT 2600
          CALL PLOT (XVALUE,YVALUE, 3)                                   PLT 2610
          GO TO 100                                                      PLT 2620
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PLOT IN-FRAME DATA POINTS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   98     XVALUE = TF(I) * FIXX + XZERO                                  PLT 2630
          YVALUE = XP(I) * FIXY + YZERO                                  PLT 2640
   99     CALL PLOT (XVALUE,YVALUE, 2)                                   PLT 2650
  100 CONTINUE                                                           PLT 2660
 1100 CONTINUE                                                           PLT 2670
      XVALUE = RIGHT * FIXX + XZERO + 7.0                                PLT 2680
      CALL PLOT (XVALUE,0.0,-3)                                          PLT 2690
      RETURN                                                             PLT 2700
C  . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   32 FORMAT(15(E8.1,2H  ))                                              PLT 2710
      END                                                                PLT 2720
```

```
      SUBROUTINE PAREN(ITEM,I)                                          PAR 0010
C     PAREN HANDLES PLOTTING PARENTHESES AROUND VARIABLE UNITS
      DIMENSION ITEM(3) ,A(10)                                          PAR 0020
      DATA (A(J),J=1,10)/77000000000000000000B,7700000000000000000B,    PAR 0030
     2     7700000000000000B,770000000000B,77000000000B,7700000000B,    PAR 0040
     3     77000000B,7700000B,77000B,7700B/                             PAR 0050
      DATA PARENS/10H))))))))))/                                        PAR 0060
C . . . . . .                                                           PAR 0070
      I = 13                                                            PAR 0070
      IT = ITEM(2)                                                      PAR 0080
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK FOR NON-BLANK IN SECOND SIX CHARACTERS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DO 15 J = 1,6                                                     PAR 0090
          I1 = IT .AND. 77B                                             PAR 0100
          IF (I1 .NE. 1R  ) GO TO 30                                    PAR 0110
          I = I - 1                                                     PAR 0120
          IT = IT / 100B                                                PAR 0130
   15 CONTINUE                                                          PAR 0140
      IT = ITEM(1)                                                      PAR 0150
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK FOR NON-BLANK IN FIRST  SIX CHARACTERS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DO 25 J = 1,5                                                     PAR 0160
          I1 = IT .AND. 77B                                             PAR 0170
          IF (I1 .NE. 1R  ) GO TO 30                                    PAR 0180
          I = I - 1                                                     PAR 0190
          IT = IT / 100B                                                PAR 0200
   25 CONTINUE                                                          PAR 0210
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHANGE 2R6 FORMAT TO A10,A4 FORMAT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   30 ENCODE(13, 32, ITEM(1)) ITEM(1), ITEM(2)                          PAR 0220
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DELETE A BLANK AND INSERT A RIGHT PARENTHESIS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (I .GT. 10)GO TO 40                                            PAR 0230
      IT= ITEM(1) .AND. .NOT. A(I)                                      PAR 0240
      ITEM(1)= IT .OR. (PARENS .AND. A(I))                              PAR 0250
      I = I + 1                                                         PAR 0260
      RETURN                                                            PAR 0270
   40 IT= ITEM(2) .AND. .NOT. A(I-10)                                   PAR 0280
      ITEM(2)= IT .OR. (PARENS .AND. A(I-10))                           PAR 0290
      RETURN                                                            PAR 0300
C . . . . . .
   32 FORMAT(2R6, 1H )                                                  PAR 0310
      END                                                               PAR 0320
```

```
      SUBROUTINE RWIVS                                                  RIV 0010
C     RWIVS READS IN AND PRINTS OUT DATA OF CURRENT AND VOLTAGE SOURCES
```

COMMON VARIABLES

```
      EQUIVALENCE (KIVS,K),(K1IVS,K1),(K2IVS,K2),(K3IVS,K3),(K4IVS,K4)  RIV 0020
      DIMENSION K(20), K1(20), K2(20), K3(20), K4(20)                   RIV 0030
C     . . . . . .                                                       
      READ (5,10)NIV                                                    RIV 0040
      IF(NIV.LE. 0) RETURN                                              RIV 0050
      IF(NIV.LE. 20) GO TO 2101                                         RIV 0060
      WRITE (6,2103)                                                    RIV 0070
      CALL EXIT                                                         RIV 0080
 2101 CALL HOLLER (6H      ,F3)                                         RIV 0090
      CALL HOLLER(1HO,CC)                                               RIV 0100
      IF (DMAX .NE. 0.0) CALL HOLLER (1H ,CC)                           RIV 0110
      WRITE (6,2100)                                                    RIV 0120
      DO 2202 N = 1,NIV                                                 RIV 0130
      READ(5,20)K4(N),K3(N),A1,K1(N),A2,K2(N),VI(N),RVS(N)              RIV 0140
      IF (RVS(N) .EQ. 0.0) RVS(N) = 0.1                                 RIV 0150
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DETERMINE TYPE OF TERMINALS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF(A1 .NE. F3) GO TO 2112                                         RIV 0160
      IF(A2 .NE. F3) GO TO 2114                                         RIV 0170
      K (N)=1                                                           RIV 0180
      GO TO 2202                                                        RIV 0190
 2112 K (N)=2                                                           RIV 0200
      GO TO 2202                                                        RIV 0210
 2114 K (N)=3                                                           RIV 0220
 2202 CONTINUE                                                          RIV 0230
      DO 5 N=1,NIV                                                      RIV 0240
      A1 = VI(N)                                                        RIV 0250
      A2 = RVS(N)                                                       RIV 0260
      IF (DMAX .EQ.0.0)GO TO 4444                                       RIV 0270
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     MODIFY NOMINAL PARAMETER VALUES BY DMAX PERCENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ (5,4445) S1,S2                                               RIV 0280
      S1 = S1 * DMAX                                                    RIV 0290
      S2 = S2 * DMAX                                                    RIV 0300
      A1 = A1 * (1.0 + S1 / 100.0)                                      RIV 0310
      A2 = A2 * (1.0 + S2 / 100.0)                                      RIV 0320
      VI(N) = A1                                                        RIV 0330
      RVS(N) = A2                                                       RIV 0340
 4444 KO = K (N)                                                        RIV 0350
      IF(K4(N) .LE. 0) GO TO 2200                                       RIV 0360
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PRINT CURRENT-SOURCES DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      WRITE (6,29)N                                                     RIV 0370
 2221 GO TO (9,2117,2118),KO                                            RIV 0380
 2117 IF(K1(N) .GT. 0) GO TO 2119                                       RIV 0390
      WRITE (6,32)K2(N)                                                 RIV 0400
      K1(N)=20                                                          RIV 0410
      GO TO 11                                                          RIV 0420
 2119 WRITE (6,34)K1(N),K2(N)                                           RIV 0430
```

```
      GO TO 11                                                          RIV 0440
 2118 IF(K2(N) .GT. 0) GO TO 2220                                       RIV 0450
      WRITE (6,33)K1(N)                                                 RIV 0460
      K2(N)=20                                                          RIV 0470
      GO TO 11                                                          RIV 0480
    9 WRITE (6,31)K1(N),K2(N)                                           RIV 0490
      GO TO 11                                                          RIV 0500
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PRINT VOLTAGE-SOURCES DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 2220 WRITE (6,35)K1(N),K2(N)                                           RIV 0510
   11 IF(K4(N) .LE. 0) GO TO 16                                         RIV 0520
      IF(K3(N) .LE. 0) GO TO 100                                        RIV 0530
      WRITE (6,53)K3(N)                                                 RIV 0540
      GO TO 5                                                           RIV 0550
  100 WRITE (6,40)                                                      RIV 0560
      IF (DMAX .GT. 0.0) WRITE (6,4446) S1                              RIV 0570
      WRITE (6,50) CC,A1                                                RIV 0580
      GO TO 5                                                           RIV 0590
 2200 WRITE (6,30)N                                                     RIV 0600
      GO TO 2221                                                        RIV 0610
   16 IF(K3(N) .LE. 0) GO TO 102                                        RIV 0620
      WRITE (6,53)K3(N)                                                 RIV 0630
      GO TO 104                                                         RIV 0640
  102 WRITE (6,40)                                                      RIV 0650
      IF (DMAX .GT. 0.0) WRITE (6,4446) S1                              RIV 0660
      WRITE (6,51) CC,A1                                                RIV 0670
  104 IF (DMAX .GT. 0.0) WRITE (6,4446) S2                              RIV 0680
      WRITE (6,52) CC,A2                                                RIV 0690
    5 CONTINUE                                                          RIV 0700
      RETURN                                                            RIV 0710
C   . . . . . .
   10 FORMAT (6I12)                                                     RIV 0720
   20 FORMAT (2I12,A6,I6,A6,I6,2E12.8)                                  RIV 0730
   29 FORMAT (1H /15H CURRENT SOURCE/4H NO=I3/15H NODE A  NODE B)       RIV 0740
   30 FORMAT (1H /15H VOLTAGE SOURCE/4H NO=I3/15H NODE A  NODE B)       RIV 0750
   31 FORMAT (2X,I3,5X,I3)                                              RIV 0760
   32 FORMAT (7H GROUND,3X,I3)                                          RIV 0770
   33 FORMAT (2X,I3,3X,7H GROUND)                                       RIV 0780
   34 FORMAT (3H  S12,5X,I3)                                            RIV 0790
   35 FORMAT (2X,I3,3X,3H  S,I2)                                        RIV 0800
   40 FORMAT (    24HO                    NOMINAL)                      RIV 0810
   50 FORMAT (A1,10HCURRENT   ,3X,E12.4)                                RIV 0820
   51 FORMAT (A1,10HVOLTAGE   ,3X,E12.4)                                RIV 0830
   52 FORMAT (A1,10HRESISTANCE,3X,E12.4)                                RIV 0840
   53 FORMAT (     7HOSOURCE I3)                                        RIV 0850
 2100 FORMAT (1H1)                                                      RIV 0860
 2103 FORMAT (    21HOMORE THAN 20 SOURCES)                             RIV 0870
 4445 FORMAT (48X,2F12.0)                                               RIV 0880
 4446 FORMAT (14X,F3.0,8H PERCENT  )                                    RIV 0890
      END                                                               RIV 0900
```

```
      SUBROUTINE RWRES                                                  RRE 0010
C     RWRES READS IN AND PRINTS OUT DATA OF RESISTORS
```

COMMON VARIABLES

```
      EQUIVALENCE (KRES,K),(K1RES,K1),(K2RES,K2)                        RRE 0020
      DIMENSION K(40), K1(40), K2(40)                                   RRE 0030
C     . . . . . .
      READ (5,10)NR                                                     RRE 0040
      IF(NR .LE. 0) RETURN                                              RRE 0050
      IF(NR .LE. 40) GO TO 2101                                         RRE 0060
      WRITE (6,2103)                                                    RRE 0070
      CALL EXIT                                                         RRE 0080
 2101 WRITE (6,2100)                                                    RRE 0090
      L = 0                                                             RRE 0100
      CALL HOLLER (6H      ,F3)                                         RRE 0110
      CALL HOLLER(1H0,CC)                                               RRE 0120
      IF (DMAX .NE. 0.0) CALL HOLLER (1H ,CC)                           RRE 0130
      DO 2202 N=1,NR                                                    RRE 0140
      READ (5,20) F1,K1 (N),R2,K2 (N),R(N) ,NRP(N)                      RRE 0150
      IF (R(N) .EQ. 0.0) R(N) = 0.001                                   RRE 0160
      IF (NRP( N) .LE. 1) GO TO 3002                                    RRE 0170
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     VARIABLE R-VS.-T DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      L = L + 1                                                         RRE 0180
      RF(L,1) = R(N)                                                    RRE 0190
      TFR(L,1) = TE                                                     RRE 0200
      IX(L) = 2                                                         RRE 0210
      KO = NRP(N)                                                       RRE 0220
      READ (5,3004) (RF(L,J),TFR(L,J),XRES(L,J),J=2,KO)                 RRE 0230
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DETERMINE TYPE OF TERMINALS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 3002 IF(F1 .EQ. F3) GO TO 2201                                         RRE 0240
      K(N)=3                                                            RRE 0250
      GO TO 2202                                                        RRE 0260
 2201 IF(R2 .EQ. F3) GO TO 2204                                         RRE 0270
      K(N)=2                                                            RRE 0280
      GO TO 2202                                                        RRE 0290
 2204 K(N)=1                                                            RRE 0300
 2202 CONTINUE                                                          RRE 0310
      L = 0                                                             RRE 0320
      DO 115 N=1,NR                                                     RRE 0330
      IF (NRP(N) .LE. 1) GO TO 4449                                     RRE 0340
      L = L + 1                                                         RRE 0350
 4449 IF (DMAX .EQ.0.0)GO TO 4444                                       RRE 0360
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     MODIFY NOMINAL PARAMETER VALUES BY DMAX PERCENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ (5,4445) S1                                                  RRE 0370
      S1 = S1 * DMAX                                                    RRE 0380
      WRITE (6,4446) S1                                                 RRE 0390
      R(N) = R(N) * (1.0 + S1/ 100.0)                                   RRE 0400
      KO = NRP(N)                                                       RRE 0410
      DO 4447 J = 1,KO                                                  RRE 0420
 4447 RF(L,J) = RF(L,J) * (1.0 + S1 / 100.0)                            RRE 0430
```

```
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PRINT RESISTORS DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 4444 KO=K(N)                                                          RRE 0440
      GO TO (6,7,8),KO                                                 RRE 0450
    6 WRITE (6,31) CC,N,K1(N) ,K2(N) , R(N)                            RRE 0460
      GO TO 5                                                          RRE 0470
    7 IF(K2(N) .GT. 0) GO TO 23                                        RRE 0480
      K2(N)=20                                                         RRE 0490
      WRITE (6,35) CC,N,K1(N),    R(N)                                 RRE 0500
      GO TO 5                                                          RRE 0510
   23 WRITE (6,33) CC,N,K1(N),K2(N),R(N)                               RRE 0520
      GO TO 5                                                          RRE 0530
    8 IF(K1(N) .GT. 0) GO TO 12                                        RRE 0540
      K1(N)=20                                                         RRE 0550
      WRITE (6,34) CC,N, K2(N),   R(N)                                 RRE 0560
      GO TO 5                                                          RRE 0570
   12 WRITE (6,32) CC,N,K1(N),K2(N),   R(N)                            RRE 0580
    5 IF (NRP(N) .LT. 2) GO TO 115                                     RRE 0590
      KO = NRP(N)                                                      RRE 0600
      WRITE (6,3040) (J,RF(L,J),TFR(L,J),XRES(L,J),J=1,KO)             RRE 0610
  115 CONTINUE                                                         RRE 0620
      RETURN                                                           RRE 0630
C  . . . . . .
   10 FORMAT (6I12)                                                    RRE 0640
   20 FORMAT (A6,I6,A6,I6, E12.8,I12)                                  RRE 0650
   31 FORMAT (A1, I3,3X,I3,5X,I3,6X,8X,E13.5)                          RRE 0660
   32 FORMAT (A1,I3,3H  S,I3,5X,I3,6X,8X,E13.5)                        RRE 0670
   33 FORMAT (A1,I3,3X,I3,2X,3H  S,I3,6X,8X,E13.5)                     RRE 0680
   34 FORMAT (A1,I3,7H GROUND,3X,I3,6X,8X,E13.5)                       RRE 0690
   35 FORMAT (A1,I3,3X,I3,3X,7H GROUND,4X,8X,E13.5)                    RRE 0700
 2100 FORMAT (1H1/10H RESISTORS/32H PART                             /44HRRE 0710
     1 30  NODE A  NODE B                RESISTANCE)                   RRE 0720
 2103 FORMAT (    23HOMORE THAN 40 RESISTORS)                         RRE 0730
 3004 FORMAT (3E12.5)                                                  RRE 0740
 3040 FORMAT (20H WITH TIME STATES..  ,/,                              RRE 0750
     2     43H POINT   RESISTANCE     TIME     EXPONENT    ,/,        RRE 0760
     3     5(1H  ,I5,3E13.4,/))                                        RRE 0770
 4445 FORMAT (24X, F12.0)                                              RRE 0780
 4446 FORMAT (1HO,31X,F3.0,8H PERCENT )                                RRE 0790
      END                                                              RRE 0800
```

```
      SUBROUTINE RWCAP                                                  RCA 0010
C     RWCAP READS IN AND PRINTS OUT DATA OF CAPACITORS
```

COMMON VARIABLES

```
      EQUIVALENCE (KCAP,K),(K1CAP,K1),(K2CAP,K2)                        RCA 0020
      DIMENSION K(20), K1(20), K2(20)                                   RCA 0030
C     . . . . . .
      READ  (5,10)NC                                                    RCA 0040
      IF( NC .LE. 0 ) RETURN                                            RCA 0050
      IF(NC .LE. 20) GO TO 2101                                         RCA 0060
      WRITE (6,2103)                                                    RCA 0070
      CALL EXIT                                                         RCA 0080
 2101 WRITE (6,2100)                                                    RCA 0090
      CALL HOLLER (6H       ,F3)                                        RCA 0100
      CALL HOLLER(1HO,CC)                                               RCA 0110
      IF (DMAX .NE. 0.0) CALL HOLLER (1H ,CC)                           RCA 0120
      DO 2202 N = 1,NC                                                  RCA 0130
      READ  (5,20)F1,K1(N),R2,K2(N),CAPC(N),RC(N),RCSH(N),CIO           RCA 0140
      IF (NSVVAR .EQ. 0) CIC(N) = CIO                                   RCA 0150
      IF (RC(N) .LE. 0.0) RC(N)= 0.1                                    RCA 0160
      IF (RCSH(N) .LE. 0.0) RCSH(N) = 1.0E+9                            RCA 0170
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DETERMINE TYPE OF TERMINALS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (F1 .EQ. F3) GO TO 2201                                        RCA 0180
      K(N)=3                                                            RCA 0190
      GO TO 2202                                                        RCA 0200
 2201 IF(R2 .EQ. F3) GO TO 2204                                         RCA 0210
      K(N)=2                                                            RCA 0220
      GO TO 2202                                                        RCA 0230
 2204 K(N)=1                                                            RCA 0240
 2202 CONTINUE                                                          RCA 0250
      DO 5 N=1,NC                                                       RCA 0260
      IF (DMAX .EQ.0.0)GO TO 4444                                       RCA 0270
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     MODIFY NOMINAL PARAMETER VALUES BY DMAX PERCENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ (5,4445) S1,S2,S3                                            RCA 0280
      S1 = S1 * DMAX                                                    RCA 0290
      S2 = S2 * DMAX                                                    RCA 0300
      S3 = S3 * DMAX                                                    RCA 0310
      WRITE (6,4446) S1,S2,S3                                           RCA 0320
      CAPC(N)=CAPC(N)*(1.0+S1 / 100.0)                                  RCA 0330
      RC(N)=RC(N) * (1.0 + S2 / 100.0)                                  RCA 0340
      RCSH(N)=RCSH(N)*(1.0+S3 / 100.0)                                  RCA 0350
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PRINT CAPACITORS DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 4444 KO=K(N)                                                           RCA 0360
      GO TO (6,7,8),KO                                                  RCA 0370
    6 WRITE (6,31) CC,N,K1(N),K2(N),CAPC(N),RC(N),RCSH(N),CIC(N)        RCA 0380
      GO TO 5                                                           RCA 0390
    7 IF(K2(N) .GT. 0) GO TO 23                                         RCA 0400
      K2(N)=20                                                          RCA 0410
      WRITE (6,35) CC,N,K1(N),      CAPC(N),RC(N),RCSH(N),CIC(N)        RCA 0420
      GO TO 5                                                           RCA 0430
```

```
   23 WRITE (6,33) CC,N,K1(N),K2(N),CAPC(N),RC(N),RCSH(N),CIC(N)          RCA 0440
      GO TO 5                                                            RCA 0450
    8 IF(K1(N) .GT. 0) GO TO 12                                          RCA 0460
      K1(N)=20                                                           RCA 0470
      WRITE (6,34) CC,N,       K2(N),CAPC(N),RC(N),RCSH(N),CIC(N)        RCA 0480
      GO TO 5                                                            RCA 0490
   12 WRITE (6,32) CC,N,K1(N),K2(N),CAPC(N),RC(N),RCSH(N),CIC(N)         RCA 0500
    5 CONTINUE                                                           RCA 0510
      RETURN                                                             RCA 0520
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
   10 FORMAT (6I12)                                                      RCA 0530
   20 FORMAT (A6,I6,A6,I6,4E12.8)                                        RCA 0540
   31 FORMAT (A1 ,I4,3X,I3,5X,I3,6X,    8X,4E14.5)                       RCA 0550
   32 FORMAT (A1 ,I4,3H  S,I3,5X,I3,6X,    8X,4E14.5)                    RCA 0560
   33 FORMAT (A1 ,I4,3X,I3,2X, 3H  S I3,6X,    8X,4E14.5)                RCA 0570
   34 FORMAT (A1 ,I4,8H  GROUND,3X,I3,6X,    8X,4E14.5)                 RCA 0580
   35 FORMAT (A1 ,I4,3X,I3,3X,7H GROUND,4X,    8X,4E14.5)                RCA 0590
 2100 FORMAT                        (1H1/14H   CAPACITORS       /7H  PART  RCA 0600
     1   45X,21HSERIES          SHUNT   ,/,                          101RCA 0610
     2  H   NO  NODE A  NODE B                CAPACITANCE   RESISTANCE   RCA 0620
     3  RESISTANCE    CURRENT                 )                          RCA 0630
 2103 FORMAT (    24HOMORE THAN 20 CAPACITORS)                          RCA 0640
 4445 FORMAT (24X,3F12.0)                                              RCA 0650
 4446 FORMAT (1H0,33X,4(F3.0,11H PERCENT                 ))            RCA 0660
      END                                                               RCA 0670
```

```
      SUBROUTINE RWIND                                                  RIN 0010
C     RWIND READS IN AND PRINTS OUT DATA OF INDUCTORS
```

COMMON VARIABLES

```
      REAL INDL                                                         RIN 0020
      LOGICAL MONL                                                      RIN 0030
      EQUIVALENCE (KIND,K),(K1IND,K1),(K2IND,K2)                        RIN 0040
      DIMENSION K(20), K1(20), K2(20)                                   RIN 0050
C     . . . . . .
      READ  (5,10)NL                                                    RIN 0060
      IF(NL .LE.  0) RETURN                                             RIN 0070
      IF(NL .LE. 20) GO TO 1000                                         RIN 0080
      WRITE (6,1002)                                                    RIN 0090
      CALL EXIT                                                         RIN 0100
 1000 WRITE (6,201)                                                     RIN 0110
      CALL HOLLER (6H       ,F3)                                        RIN 0120
      CALL HOLLER(1HO,CC)                                               RIN 0130
      IF (DMAX .NE. 0.0) CALL HOLLER (1H ,CC)                           RIN 0140
      DO 2202 N = 1,NL                                                  RIN 0150
      READ (5,20) F1,K1(N),R2,K2(N),INDL(N),RL(N),RLSH(N),VICON(N)      RIN 0160
      READ (5,1903) CILO, INCIL(N)                                      RIN 0170
      IF (RL(N)    .LE.0.0)RL(N) = 1.0                                  RIN 0180
      IF (RLSH(N) .LE.0.0)RLSH(N) = 1.0E+9                              RIN 0190
      IF (INCIL(N).EQ. 1) GO TO 1901                                    RIN 0200
      IF (NSVIL   .EQ. 1) GO TO 1902                                    RIN 0210
      IF (NSVVAR  .EQ. 1) GO TO 1902                                    RIN 0220
 1901 CIL(N) = CILO                                                     RIN 0230
 1902 IF (VICON(N).EQ.0.0)S(N) = INDL(N)                                RIN 0240
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DETERMINE TYPE OF TERMINALS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF(F1 .EQ. F3) GO TO 2201                                         RIN 0250
      K(N)=3                                                            RIN 0260
      GO TO 2202                                                        RIN 0270
 2201 IF(R2 .EQ. F3) GO TO 2204                                         RIN 0280
      K(N)=2                                                            RIN 0290
      GO TO 2202                                                        RIN 0300
 2204 K(N)=1                                                            RIN 0310
 2202 CONTINUE                                                          RIN 0320
      DO 115 N = 1,NL                                                   RIN 0330
      IF (DMAX .EQ.0.0)GO TO 4444                                       RIN 0340
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     MODIFY NOMINAL PARAMETER VALUES BY DMAX PERCENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ (5,4445) S1,S2,S3,S4                                         RIN 0350
      S1 = S1 * DMAX                                                    RIN 0360
      S2 = S2 * DMAX                                                    RIN 0370
      S3 = S3 * DMAX                                                    RIN 0380
      S4 = S4 * DMAX                                                    RIN 0390
      WRITE (6,4446) S1,S2,S3,S4                                        RIN 0400
      INDL(N)=INDL(N)*(1.0+S1 / 100.0)                                  RIN 0410
      RL(N)  = RL(N)  * (1.0 + S2 / 100.0)                              RIN 0420
      RLSH(N) = RLSH(N) * (1.0 + S3 / 100.0)                            RIN 0430
      VICON(N)=VICON(N)* (1.0 + S4 / 100.0)                             RIN 0440
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PRINT INDUCTORS DATA
```

```
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 4444 KO=K(N)                                                          RIN 0450
      GO TO (6,7,8),KO                                                 RIN 0460
    6 WRITE (6,31)CC,N,K1(N),K2(N),INDL(N),RL(N),RLSH(N),VICON(N),CIL(N)RIN 0470
      GO TO 5                                                          RIN 0480
    7 IF(K2(N) .GT. 0) GO TO 23                                        RIN 0490
      K2(N)=20                                                         RIN 0500
      WRITE (6,35)CC,N,K1(N),      INDL(N),RL(N),RLSH(N),VICON(N),CIL(N)RIN 0510
      GO TO 5                                                          RIN 0520
   23 WRITE (6,33)CC,N,K1(N),K2(N),INDL(N),RL(N),RLSH(N),VICON(N),CIL(N)RIN 0530
      GO TO 5                                                          RIN 0540
    8 IF(K1(N) .GT. 0) GO TO 12                                        RIN 0550
      K1(N)=20                                                         RIN 0560
      WRITE (6,34)CC,N,      K2(N),INDL(N),RL(N),RLSH(N),VICON(N),CIL(N)RIN 0570
      GO TO 5                                                          RIN 0580
   12 WRITE (6,32)CC,N,K1(N),K2(N),INDL(N),RL(N),RLSH(N),VICON(N),CIL(N)RIN 0590
    5 IF (MONL) WRITE (6,1950) S(N),CILO,INCIL(N)                      RIN 0600
  115 CONTINUE                                                         RIN 0610
      RETURN                                                           RIN 0620
C     . . . . . .
   10 FORMAT (I12)                                                     RIN 0630
   20 FORMAT (A6,I6,A6,I6,4E12.8)                                      RIN 0640
   31 FORMAT (A1 ,I4,3X,I3,5X,I3,6X,    8X,5E14.5)                     RIN 0650
   32 FORMAT (A1 ,I4,3H  S,I3,5X,I3,6X,    8X,5E14.5)                  RIN 0660
   33 FORMAT (A1 ,I4,3X,I3,2X,3H  S,I3,6X,    8X,5E14.5)              RIN 0670
   34 FORMAT (A1 ,I4,8H  GROUND,3X,I3,6X,    8X,5E14.5)               RIN 0680
   35 FORMAT (A1 ,I4,3X,I3,3X,7H GROUND,4X,    8X,5E14.5)              RIN 0690
  201 FORMAT                    (1H1/12H    INDUCTORS         /7H  PART  RIN 0700
     1   45X,21HSERIES          SHUNT    ,/,                      101RIN 0710
     2  H    NO  NODE A  NODE B                INDUCTANCE     RESISTANCE  RIN 0720
     3  RESISTANCE     INVICON        CURRENT    )                      RIN 0730
 1002 FORMAT (1H /23H MORE THAN 20 INDUCTORS)                          RIN 0740
 1903 FORMAT (F12.5,I12)                                               RIN 0750
 1950 FORMAT (16H S(),CILO,INCIL ,2E12.5,I12)                          RIN 0760
 4445 FORMAT (24X,4F12.0)                                              RIN 0770
 4446 FORMAT (1H0,33X,4(F3.0,11H PERCENT                 ))            RIN 0780
      END                                                              RIN 0790
```

```
      SUBROUTINE RWZEN                                                  RZE 0010
C     RWZEN READS IN AND PRINTS OUT DATA OF ZENER DIODES
```

COMMON VARIABLES

```
      EQUIVALENCE (KZEN,K),(K1ZEN,K1),(K2ZEN,K2)                        RZE 0020
      DIMENSION K(20), K1(20), K2(20)                                   RZE 0030
C     . . . . . .
      READ  (5,10)NZEN                                                  RZE 0040
      IF (NZEN .LE. 0) RETURN                                           RZE 0050
      IF (NZEN .LE.20) GO TO 2101                                       RZE 0060
      WRITE (6,2103)                                                    RZE 0070
      CALL EXIT                                                         RZE 0080
 2101 WRITE (6,2100)                                                    RZE 0090
      CALL HOLLER (6H       ,F3)                                        RZE 0100
      CALL HOLLER(1H0,CC)                                               RZE 0110
      IF (DMAX .NE. 0.0) CALL HOLLER (1H ,CC)                           RZE 0120
      DO 2202 N=1,NZEN                                                  RZE 0130
      READ  (5,20)F1,K1(N),R2,K2(N),RZ(N),RZB(N),EZB(N)                 RZE 0140
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DETERMINE TYPE OF TERMINALS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF(F1 .EQ. F3) GO TO 2201                                         RZE 0150
      K(N)=3                                                            RZE 0160
      GO TO 2202                                                        RZE 0170
 2201 IF(R2 .EQ. F3) GO TO 2204                                         RZE 0180
      K(N)=2                                                            RZE 0190
      GO TO 2202                                                        RZE 0200
 2204 K(N)=1                                                            RZE 0210
 2202 CONTINUE                                                          RZE 0220
      DO 5 N=1,NZEN                                                     RZE 0230
      IF (DMAX .EQ.0.0)GO TO 4444                                       RZE 0240
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     MODIFY NOMINAL PARAMETER VALUES BY DMAX PERCENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ (5,4445) S1,S2,S3                                            RZE 0250
      S1 = S1 * DMAX                                                    RZE 0260
      S2 = S2 * DMAX                                                    RZE 0270
      S3 = S3 * DMAX                                                    RZE 0280
      WRITE (6,4446) S1,S2,S3                                           RZE 0290
      RZ(N) = RZ(N) * (1.0 + S1 / 100.0)                                RZE 0300
      RZB(N)=RZB(N) * (1.0 + S2 / 100.0)                                RZE 0310
      EZB(N)=EZB(N) * (1.0 + S3 / 100.0)                                RZE 0320
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PRINT ZENER-DIODES DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 4444 KO=K(N)                                                           RZE 0330
      VZB(N) = EZB(N) * RZ(N) / (RZ(N) - RZB(N))                        RZE 0340
      VZF(N) = 0.7 * RZ(N) / (RZ(N) - 1.0)                              RZE 0350
      GO TO (6,7,8),KO                                                  RZE 0360
    6 WRITE (6,31) CC,N,K1(N),K2(N),RZ(N),RZB(N),EZB(N),VZB(N)          RZE 0370
      GO TO 5                                                           RZE 0380
    7 IF(K2(N) .GT. 0) GO TO 23                                         RZE 0390
      K2(N)=20                                                          RZE 0400
      WRITE (6,35) CC,N,K1(N),      RZ(N),RZB(N),EZB(N),VZB(N)          RZE 0410
      GO TO 5                                                           RZE 0420
   23 WRITE (6,33) CC,N,K1(N),K2(N),RZ(N),RZB(N),EZB(N),VZB(N)          RZE 0430
```

```
      GO TO 5                                                                  RZE 0440
    8 IF(K1(N) .GT. 0) GO TO 12                                                RZE 0450
      K1(N)=20                                                                 RZE 0460
      WRITE (6,34) CC,N,      K2(N),RZ(N),RZB(N),EZB(N),VZB(N)                 RZE 0470
      GO TO 5                                                                  RZE 0480
   12 WRITE (6,32) CC,N,K1(N),K2(N),RZ(N),RZB(N),EZB(N),VZB(N)                 RZE 0490
    5 CONTINUE                                                                 RZE 0500
      RETURN                                                                   RZE 0510
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
   10 FORMAT (6I12)                                                            RZE 0520
   20 FORMAT (A6,I6,A6,I6,4E12.8)                                              RZE 0530
   31 FORMAT (A1 ,I3,3X,I3,5X,I3, 13X  , 4F14.5)                               RZE 0540
   32 FORMAT (A1 ,I3,3H  S13,5X,I3,13X,    4F14.5)                             RZE 0550
   33 FORMAT (A1 ,I3,3X,I3,2X,3H  S,I3,13X,   4E14.5)                          RZE 0560
   34 FORMAT (A1 ,I3,8H  GROUND,3X,I3,13X,    4E14.5)                          RZE 0570
   35 FORMAT (A1 ,I3,3X,I3,3X,7H GROUND,11X,    4E14.5)                       RZE 0580
 2100 FORMAT (1H1/14H ZENER DIODES                                             RZE 0590
     2                          /32H PART                                /21HRZE 0600
     3  NO  NODE A  NODE B ,16X,2HRZ,12X,3HRZB,11X,3HEZB,11X,3HVZB )          RZE 0610
 2103 FORMAT (    27HOMORE THAN 20 ZENER DIODES  )                           RZE 0620
 4445 FORMAT (24X,3F12.0)                                                      RZE 0630
 4446 FORMAT (1HO,31X,3(F3.0,11H PERCENT   ))                                  RZE 0640
      END                                                                      RZE 0650
```

```
      SUBROUTINE RWDIOD                                                 RDI 0010
C     RWDIOD READS IN AND PRINTS OUT DATA OF DIODES
```

COMMON VARIABLES

```
      EQUIVALENCE (KDIOD,K),(K1DIOD,K1),(K2DIOD,K2)                     RDI 0020
      DIMENSION K(20), K1(20), K2(20)                                   RDI 0030
C     . . . . . .                                                       
      READ (5,10)NDIODE                                                 RDI 0040
      IF(NDIODE .LE. 0) RETURN                                          RDI 0050
      IF(NDIODE .LE. 20) GO TO 8601                                     RDI 0060
      WRITE (6,8600)                                                    RDI 0070
      CALL EXIT                                                         RDI 0080
 8601 WRITE (6,30)                                                      RDI 0090
      QT = 1.159E+4 / ABTEMP                                            RDI 0100
      TOOLD=-7777777.777                                                RDI 0110
      CALL HOLLER (6H      ,F3)                                         RDI 0120
      CALL HOLLER(1HO,CC)                                               RDI 0130
      IF (DMAX .NE. 0.0) CALL HOLLER (1H ,CC)                           RDI 0140
      DO 8503 N=1,NDIODE                                                RDI 0150
      READ (5,2199) F1,K1(N),R2,K2(N),K3,QCR(N,6)                       RDI 0160
C     - - - - - - - - - - - - - - - - - - - - - - - -
C     DETERMINE TYPE OF TERMINALS
C     - - - - - - - - - - - - - - - - - - - - - - - -
      IF(F1 .EQ. F3) GO TO 2201                                         RDI 0170
      K(N)=3                                                            RDI 0180
      GO TO 2202                                                        RDI 0190
 2201 IF(R2 .EQ. F3) GO TO 2204                                         RDI 0200
      K(N)=2                                                            RDI 0210
      GO TO 2202                                                        RDI 0220
 2204 K(N)=1                                                            RDI 0230
 2202 IF(K3 .LE. 0) GO TO 8500                                          RDI 0240
C     - - - - - - - - - - - - - - - - - - - - - - - -
C     COPY THE PARAMETER VALUES OF DIODE NO. K3
C     - - - - - - - - - - - - - - - - - - - - - - - -
      DO 8502 I=1,7                                                     RDI 0250
 8502 QCR(N,I) = QCR(K3,I)                                              RDI 0260
      GO TO 8503                                                        RDI 0270
 8500 READ (5,20) QCR(N,1),QCR(N,2),QCR(N,3),QCR(N,4),QCR(N,5),QCR(N,7) RDI 0280
 8503 CONTINUE                                                          RDI 0290
      DO 5 N=1,NDIODE                                                   RDI 0300
      IF (DMAX .EQ.0.0)GO TO 4444                                       RDI 0310
C     - - - - - - - - - - - - - - - - - - - - - - - -
C     MODIFY NOMINAL PARAMETER VALUES BY DMAX PERCENT
C     - - - - - - - - - - - - - - - - - - - - - - - -
      READ (5,4445) S1,S2,S3,S4,S5,S7                                   RDI 0320
      S1 = S1 * DMAX                                                    RDI 0330
      S2 = S2 * DMAX                                                    RDI 0340
      S3 = S3 * DMAX                                                    RDI 0350
      S4 = S4 * DMAX                                                    RDI 0360
      S5 = S5 * DMAX                                                    RDI 0370
      S7 = S7 * DMAX                                                    RDI 0380
      WRITE (6,4446) S1,S2,S3,S4,S5,S7                                  RDI 0390
      QCR(N,1) = QCR(N,1) * (1.0 + S1 / 100.0)                          RDI 0400
      QCR(N,2) = QCR(N,2) * (1.0 + S2 / 100.0)                          RDI 0410
      QCR(N,3) = QCR(N,3) * (1.0 + S3 / 100.0)                          RDI 0420
      QCR(N,4) = QCR(N,4) * (1.0 + S4 / 100.0)                          RDI 0430
```

```
      QCR(N,5) = QCR(N,5) * (1.0 + S5 / 100.0)                          RDI 0440
      QCR(N,7) = QCR(N,7) * (1.0 + S7 / 100.0)                          RDI 0450
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PRINT DIODES DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 4444 C1(N) = 0.9 * QCR(N,5)                                            RDI 0460
      C2(N) = QT / QCR(N,2)                                             RDI 0470
      C3(N) = 1.0 / QCR(N,3)                                            RDI 0480
      C4(N) = 1.0 / QCR(N,5)                                            RDI 0490
      KO=K(N)                                                           RDI 0500
      GO TO (6,7,8),KO                                                  RDI 0510
    6 WRITE (6,31) CC,N,K1(N),K2(N), (QCR(N,I),I =1,5),QCR(N,7),QCR(N,6)RDI 0520
      GO TO 5                                                           RDI 0530
    7 IF(K2(N) .GT. 0) GO TO 23                                         RDI 0540
      K2(N)=20                                                          RDI 0550
      WRITE (6,35) CC,N,K1(N),  (QCR(N,I), I= 1, 5),QCR(N,7),QCR(N,6)   RDI 0560
      GO TO 5                                                           RDI 0570
   23 WRITE (6,33) CC,N,K1(N),K2(N), (QCR(N,I),I =1,5),QCR(N,7),QCR(N,6)RDI 0580
      GO TO 5                                                           RDI 0590
    8 IF(K1(N) .GT. 0) GO TO 12                                         RDI 0600
      K1(N)=20                                                          RDI 0610
      WRITE (6,34) CC,N, K2(N),(QCR(N,I),I =1,5),QCR(N,7),QCR( N,6)     RDI 0620
      GO TO 5                                                           RDI 0630
   12 WRITE (6,32) CC,N,K1(N),K2(N),(QCR(N,I), I=1, 5),QCR(N,7),QCR(N,6)RDI 0640
    5 CONTINUE                                                          RDI 0650
      RETURN                                                            RDI 0660
C . . . . . .
   10 FORMAT (6I12)                                                     RDI 0670
   20 FORMAT(6E12.8)                                                    RDI 0680
   30 FORMAT(1H1/7H DIODES/99H NO  NODE P  NODE N      ISD          MD   RDI 0690
     1      RLD          COD        VPOTD          T D          IPP)   RDI 0700
   31 FORMAT (A1 ,I2,3X,I3,5X,I3,E13.3,6E12.3)                          RDI 0710
   32 FORMAT (A1 ,I2,3H  S I3,5X,I3,E13.3,6E12.3)                       RDI 0720
   33 FORMAT (A1 ,I2,3X,I3,2X,3H  S I3,E13.3,6E12.3)                    RDI 0730
   34 FORMAT (A1 ,I2,8H  GROUND,3X,I3,E13.3,6E12.3)                     RDI 0740
   35 FORMAT (A1 ,I2,3X,I3,4X,6HGROUND,E11.3,6E12.3)                    RDI 0750
 2199 FORMAT (A6,I6,A6,I6,I12,3E12.8)                                   RDI 0760
 4445 FORMAT (6F12.0)                                                   RDI 0770
 4446 FORMAT (1H0,18X,6(F3.0, 9H PERCENT     ))                         RDI 0780
 8600 FORMAT (     20HOMORE THAN 20 DIODES)                             RDI 0790
      END                                                               RDI 0800
```

```
      SUBROUTINE RWTRAN                                                 RTR 0010
C     RWTRAN READS IN AND PRINTS OUT DATA OF TRANSISTORS
```

COMMON VARIABLES

```
      LOGICAL MONT                                                      RTR 0020
      EQUIVALENCE (KTRAN,K),(K1TRAN,K1),(K2TRAN,K2),(K3TRAN,K3),        RTR 0030
     2    (K4TRAN,K4)                                                  RTR 0040
      DIMENSION K(20), K1(20), K2(20), K3(20), K4(20)                   RTR 0050
      DIMENSION ESSES(14)                                               RTR 0060
C     . . . . . .
      READ (5,10)NTRANS                                                 RTR 0070
      IF(NTRANS .LE. 0) RETURN                                          RTR 0080
      IF(NTRANS .LE. 20) GO TO 8601                                     RTR 0090
      WRITE (6,8600)                                                    RTR 0100
      CALL EXIT                                                         RTR 0110
 8601 QT = 1.159E+4/ABTEMP                                              RTR 0120
      TTOOLD=-7777777.777                                               RTR 0130
      CALL HOLLER (6H       ,F3)                                        RTR 0140
      CALL HOLLER(1HO,CC)                                               RTR 0150
      IF (DMAX .NE. 0.0) CALL HOLLER (1H ,CC)                           RTR 0160
      DO 8003 N=1,NTRANS                                                RTR 0170
      READ (5,2199)K1(N),BASE,K2(N),COLC,K3(N),EMTR,K4(N),K5            RTR 0180
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DETERMINE TYPE OF TERMINALS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (BASE .EQ. F3) GO TO 2201                                      RTR 0190
      K (N)=2                                                           RTR 0200
      GO TO 2202                                                        RTR 0210
 2201 IF (COLC .EQ. F3) GO TO 2204                                      RTR 0220
      K (N)=3                                                           RTR 0230
      GO TO 2202                                                        RTR 0240
 2204 IF (EMTR .EQ. F3) GO TO 2206                                      RTR 0250
      K (N)=4                                                           RTR 0260
      GO TO 2202                                                        RTR 0270
 2206 K (N)=1                                                           RTR 0280
 2202 IF(K5 .LE. 0) GO TO 8500                                          RTR 0290
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COPY THE PARAMETER VALUES OF TRANSISTOR NO. K5
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DO 8502 I=1,16                                                    RTR 0300
 8502 QTAN(N,I) = QTAN(K5,I)                                            RTR 0310
      GO TO 8003                                                        RTR 0320
 8500 READ (5,20) (QTAN(N,I),I = 1,16)                                  RTR 0330
 8003 CONTINUE                                                          RTR 0340
      NTO1=4                                                            RTR 0350
      WRITE (6,2500)                                                    RTR 0360
      DO 3 N=1,NTRANS                                                   RTR 0370
      IF(N .NE. NTO1) GO TO 2501                                        RTR 0380
      WRITE (6,2500)                                                    RTR 0390
      NTO1=NTO1+3                                                       RTR 0400
 2501 IF(K1(N).GT. 0) GO TO 5                                           RTR 0410
      WRITE (6,30)N                                                     RTR 0420
      GO TO 6                                                           RTR 0430
    5 WRITE (6,31)N                                                     RTR 0440
    6 KO=K (N)                                                          RTR 0450
      WRITE (6,75)                                                      RTR 0460
```

```
      GO TO (100,101,102,103),KO                                        RTR 0470
  100 WRITE (6,35)K2(N),K3(N),K4(N)                                     RTR 0480
      GO TO 11                                                          RTR 0490
  101 IF(K2(N) .GT. 0) GO TO 108                                        RTR 0500
      K2(N)=20                                                          RTR 0510
      WRITE (6,36)K3(N),K4(N)                                           RTR 0520
      GO TO 11                                                          RTR 0530
  108 WRITE (6,37)K2(N),K3(N),K4(N)                                     RTR 0540
      GO TO 11                                                          RTR 0550
  102 IF(K3(N) .GT. 0) GO TO 110                                        RTR 0560
      K3(N)=20                                                          RTR 0570
      WRITE (6,38)K2(N),K4(N)                                           RTR 0580
      GO TO 11                                                          RTR 0590
  110 WRITE (6,39)K2(N),K3(N),K4(N)                                     RTR 0600
      GO TO 11                                                          RTR 0610
  103 IF(K4(N) .GT. 0) GO TO 112                                        RTR 0620
      K4(N)=20                                                          RTR 0630
      WRITE (6,40)K2(N),K3(N)                                           RTR 0640
      GO TO 11                                                          RTR 0650
  112 WRITE (6,41)K2(N),K3(N),K4(N)                                     RTR 0660
   11 IF (DMAX .EQ.0.0)GO TO 4444                                       RTR 0670
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     MODIFY NOMINAL PARAMETER VALUES BY DMAX PERCENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ (5,4445) (ESSES(I) ,I=1,14)                                  RTR 0680
      DO 4447 I = 1,14                                                  RTR 0690
      ESSES(I) = ESSES(I) * DMAX                                        RTR 0700
 4447 QTAN (N,I) = QTAN(N,I) * (1.0 + ESSES(I) / 100.0)                 RTR 0710
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PRINT TRANSISTORS DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      WRITE (6,4446) (ESSES(I),I=1,4)                                   RTR 0720
 4444 WRITE (6,32) CC, (QTAN (N,I), I = 1, 4)                           RTR 0730
      IF (DMAX .GT.0.0)WRITE (6,4446) (ESSES(I),I=5,9)                  RTR 0740
      WRITE (6,4448) (QTAN(N,I), I = 5, 9)                              RTR 0750
      IF (DMAX .GT.0.0)WRITE (6,4446) (ESSES(I),I=10,14)                RTR 0760
      WRITE (6,4449) (QTAN(N,I), I = 10,14)                             RTR 0770
      CC1(N) = 0.9* QTAN(N,8)                                           RTR 0780
      CC2(N) = QT / QTAN(N,6)                                           RTR 0790
      CC3 (N) = 1.0/ QTAN(N,9)                                          RTR 0800
      CC4(N) = 1.0 / QTAN(N,8)                                          RTR 0810
      EC1(N) =0.9* QTAN(N,13)                                           RTR 0820
      EC2 (N) = QT / QTAN (N,11)                                        RTR 0830
      EC3 (N) = 1.0 / QTAN(N,14)                                        RTR 0840
      EC4(N) = 1.0 / QTAN(N,13)                                         RTR 0850
      AF(N) = QTAN(N,1) / (1.0+QTAN(N,1))                               RTR 0860
      AR(N) = QTAN(N,2) / (1.0+QTAN(N,2))                               RTR 0870
      IF (MONT) WRITE (6,865) CC1(N),CC2(N),CC3(N),CC4(N),AF(N),        RTR 0880
     2     EC1(N),EC2(N),EC3(N),EC4(N),AR(N)                            RTR 0890
      IF(QTAN(N,15) .EQ. -QTAN(N,16)) GO TO 3                           RTR 0900
      WRITE (6,58)QTAN(N,15),QTAN(N,16)                                 RTR 0910
      IF(K1(N).GT. 0) GO TO 3                                           RTR 0920
      QTAN(N,16) = -QTAN(N,16)                                          RTR 0930
      QTAN(N,15) = -QTAN(N,15)                                          RTR 0940
    3 CONTINUE                                                          RTR 0950
      RETURN                                                            RTR 0960
C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   10 FORMAT (6I12)                                                     RTR 0970
   20 FORMAT (6E12.8)                                                   RTR 0980
   30 FORMAT (1HO/15H NPN TRANSISTOR/4H NO=I3)                          RTR 0990
   31 FORMAT (1HO/15H PNP TRANSISTOR/4H NO=I3)                          RTR 1000
   32 FORMAT (A1 ,7H  BETANE11.4,7H  BETAIE11.4,7H    TE E11.4,7H    TC RTR 1010
     1E11.4)                                                            RTR 1020
   35 FORMAT (2X,I3,5X,I3,5X,I3)                                        RTR 1030
   36 FORMAT (7H GROUND,3X,I3,5X,I3)                                    RTR 1040
   37 FORMAT (3H  SI2,5X,I3,5X,I3)                                      RTR 1050
```

```
  38 FORMAT (2X,I3,3X,7H GROUND,3X,I3)                                    RTR 1060
  39 FORMAT (2X,I3,3X,3H  SI2,5X,I3)                                      RTR 1070
  40 FORMAT (2X,I3,5X,I3,3X,7H GROUND)                                    RTR 1080
  41 FORMAT (2X,I3,5X,I3,3X,3H  SI2)                                      RTR 1090
  58 FORMAT (    7H   IPPCE11.4,7H   IPPEE11.4)                           RTR 1100
  75 FORMAT (    23HONODE B  NODE C  NODE E)                              RTR 1110
 865 FORMAT (60H CC1(),CC2(),CC3(),CC4(),AF(),EC1(),EC2(),EC3(),EC4(),AR  RTR 1120
    2R()     ,2(/,5F24.15))                                               RTR 1130
2199 FORMAT (I12,A6,I6,A6,I6,A6,I6,2I12)                                  RTR 1140
2500 FORMAT (1H1)                                                         RTR 1150
4445 FORMAT (6F12.0)                                                      RTR 1160
4446 FORMAT (/6X,5(F3.0,8H PERCENT ,7X))                                  RTR 1170
4448 FORMAT                                                               RTR 1180
    1    (  6H   ISCE11.4, 6H    MCE11.4,7X, 3HCOCE11.4,7H  VPOTEE11.4,  RTR 1190
    27H    RLCE11.4)                                                      RTR 1200
4449 FORMAT         ( 6H   ISEE11.4, 6H    MEE11.4,7X, 3HCOEE11.4,7H  VPR  RTR 1210
    30TEE11.4,7H    RLEE11.4)                                             RTR 1220
8600 FORMAT (    25HOMORE THAN 20 TRANSISTORS)                            RTR 1230
     END                                                                  RTR 1240
```

```
      SUBROUTINE RWCORE                                                 RCE 0010
C     RWCORE CALLS SUBROUTINE RWCORF FOR PHI-F DATA
C                OR SUBROUTINE RWCORH FOR B-H DATA
```

COMMON VARIABLES

```
C  . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
      READ  (5,500) NCORE, PARIN                                        RCE 0020
      IF (NCORE .LE. 0) RETURN                                          RCE 0030
      IF (NCORE .LE. 20) GO TO 5                                        RCE 0040
      WRITE             (6,502)                                         RCE 0050
      CALL EXIT                                                         RCE 0060
    5 CALL HOLLER(6HBH    ,PARBH)                                       RCE 0070
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHOOSE BETWEEN SUBROUTINES RWCORF AND RWCORH WHICH READ
C          PHI-F AND B-H DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (PARIN  .EQ. PARBH) GO TO 7                                    RCE 0080
      CALL RWCORF                                                       RCE 0090
      RETURN                                                            RCE 0100
    7 CALL RWCORH                                                       RCE 0110
      RETURN                                                            RCE 0120
C  . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
  502 FORMAT(20X,39HTHE MAXIMUM NUMBER OF CORES IS EXCEEDED//)          RCE 0130
  500 FORMAT (I12,A6)                                                   RCE 0140
      END                                                               RCE 0150
```

```
      SUBROUTINE RWCORF                                                 RCF 0010
C     RWCORF READS IN AND PRINTS OUT PHI-F DATA OF CORES
```

COMMON VARIABLES

```
      LOGICAL MONCOR                                                    RCF 0020
      REAL NS,NP                                                        RCF 0030
      INTEGER STSWEX                                                    RCF 0040
      EQUIVALENCE (KCORE,KC),(K1CORE,K1C),(K2CORE,K2C)                  RCF 0050
      DIMENSION KC(20,10), K1C(20,10), K2C(20,10)                       RCF 0060
      DIMENSION ESSES(24), HA(20),          HN(20), HQ(20), XID(20),    RCF 0070
     2   XOD(20)   ,TYPE(20), PHID2(20) , CORMAT(20),U0(20)             RCF 0080
C     . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
C     READ IN DATA COMMON TO ALL CORES.
C     SET HOLLERITH CONSTANTS FOR PHID(F) TYPES, BLANKS,
C        AND CARRIAGE CONTROL.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ  (5,531) RELERR,ABSERR,PSTEP  ,STSWEX,NUDT,LASTIC            RCF 0090
      IF (RELERR .EQ. 0.0) RELERR = 0.0001                              RCF 0100
      IF (ABSERR .EQ. 0.0) ABSERR = 0.0001                              RCF 0110
      CALL HOLLER(1H ,HOL )                                             RCF 0120
      CALL HOLLER(1HA,HOLA)                                             RCF 0130
      CALL HOLLER(1HB,HOLB)                                             RCF 0140
      CALL HOLLER(1HC,HOLC)                                             RCF 0150
      CALL HOLLER(1HD,HOLD)                                             RCF 0160
      CALL HOLLER(1HE,HOLE)                                             RCF 0170
      CALL HOLLER(1HF,HOLF)                                             RCF 0180
      CALL HOLLER(1HG,HOLG)                                             RCF 0190
      CALL HOLLER (6H      ,F3)                                         RCF 0200
      CALL HOLLER(1HO,CC)                                               RCF 0210
      IF (OMAX .NE. 0.0) CALL HOLLER (1H ,CC)                           RCF 0220
      IF (NSVPHI .NE. 1) GO TO 7757                                     RCF 0230
      IF (NSVVAR .EQ. 1) GO TO 7757                                     RCF 0240
      IF (KONT   .EQ. 1) GO TO 7757                                     RCF 0250
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     SUBTRACT ELASTIC FLUX CHANGE FROM PHI OF PREVIOUS MODE,
C        UNLESS NSVVAR=1 OR KONT=1.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DO 7759 N = 1, NCORE                                              RCF 0260
       PHI(N)=PHI(N)-P1(N)*F(N)*ALOG((ABS(F(N))+P2(N))/(ABS(F(N))+P3(RCF 0270
     2    N)))                                                         RCF 0280
       IF (ABS(PHI(N)) .LE. PHIR(N)) GO TO 7759                         RCF 0290
       IF (PHI(N)) 7761,7759,7762                                       RCF 0300
 7761  PHI(N) = -PHIR(N)                                                RCF 0310
       GO TO 7759                                                       RCF 0320
 7762  PHI(N) =  PHIR(N)                                                RCF 0330
 7759 CONTINUE                                                          RCF 0340
 7757 WRITE (6,9789)                                                    RCF 0350
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK AND WRITE
C        (1) WHETHER ELASTIC SWITCHING IS INCLUDED.
C        (2) WHETHER TO TERMINATE MODE RUN WHEN SWITCHING
C           OF ALL CORES IS COMPLETED.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (LASTIC .GE. 1) GO TO 477                                      RCF 0360
      CALL HOLLER(6HNEGLEC,T1)                                          RCF 0370
      CALL HOLLER(6HTED   ,T2)                                          RCF 0380
```

```
      GO TO 478                                                          RCF 0390
  477 CALL HOLLER(6HINCLUD,T1)                                           RCF 0400
      CALL HOLLER(6HED    ,T2)                                           RCF 0410
  478 WRITE (6,150) T1,T2                                                RCF 0420
      RNUDT = NUDT                                                       RCF 0430
      IF (STSWEX .NE.0.0)GO TO 8                                         RCF 0440
      WRITE             (6,528)                                          RCF 0450
      GO TO 9                                                            RCF 0460
    8 WRITE             (6,529)NUDT                                      RCF 0470
    9 NOEXIT = 1                                                         RCF 0480
      WRITE             (6, 533) RELERR, ABSERR                          RCF 0490
      WRITE(6,7778) PSTEP                                                RCF 0500
      DO 2202 N=1,NCORE                                                  RCF 0510
      READ  (5,503) NWSUBN, KC1, UO(N)                                   RCF 0520
      NW(N) = NWSUBN                                                     RCF 0530
      IF (NWSUBN .GT. 0) GO TO 6                                         RCF 0540
      WRITE             (6,524)     N                                    RCF 0550
      CALL EXIT                                                          RCF 0560
    6 IF (NWSUBN .LE. 10) GO TO 7                                        RCF 0570
      WRITE             (6,523)N                                         RCF 0580
      CALL EXIT                                                          RCF 0590
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     READ IN WINDING DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
    7 DO 13 M = 1,NWSUBN                                                 RCF 0600
      READ (5,504) F1C,K1C(N,M),R2C,K2C(N,M),XNT(N,M),RW(N,M),I,J,CIO    RCF 0610
      IF (I .LE. 0) I = 1                                                RCF 0620
      IF (J .LE. 0) J = 1                                                RCF 0630
      NP(N,M) = I                                                        RCF 0640
      NS(N,M) = J                                                        RCF 0650
      IF ((NSVVAR .NE. 1) .AND. (KONT .NE. 1)) CICORE(N,M) = CIO         RCF 0660
      IF (RW(N,M) .LE.0.0) RW(N,M) = 0.1                                 RCF 0670
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DETERMINE TYPE OF TERMINALS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (F1C    .NE. F3) GO TO 12                                       RCF 0680
      IF (R2C    .NE. F3)GO TO 11                                        RCF 0690
      KC(N,M)=1                                                          RCF 0700
      GO TO 13                                                           RCF 0710
   11 KC(N,M)=2                                                          RCF 0720
      GO TO 13                                                           RCF 0730
   12 KC(N,M)=3                                                          RCF 0740
   13 CONTINUE                                                           RCF 0750
      IF (KC1 .GT. 0) GO TO 15                                           RCF 0760
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     UP TO THIS POINT, SUBROUTINES RWCORF AND RWCORH ARE IDENTICAL.
C     READ IN CORE-PARAMETER DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ  (5,501) XOD(N),XID(N),PHIS(N),PHIR(N),TYPE(N),CORMAT(N)      RCF 0770
      IF (TYPE(N) .EQ. HOL ) TYPE(N) = HOLB                              RCF 0780
      IF (CORMAT(N) .EQ. HOL ) CORMAT(N) = HOLF                          RCF 0790
      IF (PHIS(N) .EQ. 0.0) PHIS(N) = 1.1 * PHIR(N)                      RCF 0800
      READ  (5,505) FD1(N),FDZ(N),FD2(N),FD3(N),PHID2(N)                 RCF 0810
      READ  (5,505) HA(N),HQ(N),HN(N)                                    RCF 0820
      READ  (5,505) XLMDAD(N),XNUD(N),FD8(N),FOPP(N),XLAMDA(N),XNU(N),   RCF 0830
     2              FB(N),FO(N),ROP(N),FB1(N),FO1(N),ROP1(N)             RCF 0840
      IF (FB1(N) .EQ. 0.0)FB1(N) = 1.0E30                                RCF 0850
      IF (FOPP(N) .LT. FD1(N)) FOPP(N) = FD1(N)                          RCF 0860
      IF (XNU(N)  .EQ.   0.0 ) XNU(N) =(FB(N)-FOPP(N)) / (FB(N)-FO(N))   RCF 0870
      IF (XLAMDA(N) .EQ. 0.0) XLAMDA(N)=ROP(N)/(XNU(N)*(FB(N)-FOPP(N))   RCF 0880
     2     ** (XNU(N)-1.0))                                              RCF 0890
      IF (FD8(N)  .EQ.   0.0 ) FD8(N)  = 1.15 * FOPP(N)                  RCF 0900
      IF (XNUD(N) .EQ.   0.0 ) XNUD(N) = XNU(N)*(FD8(N)-FD1(N))          RCF 0910
     2     / (FD8(N)-FOPP(N))                                            RCF 0920
      IF (XLMDAD(N) .EQ. 0.0 ) XLMDAD(N)=XLAMDA(N)*(XNU(N)/XNUD(N))**    RCF 0930
     2     XNU(N) / (FD8(N)-FD1(N))**(XNUD(N)-XNU(N))                    RCF 0940
```

```
      GO TO 2202                                                        RCF 0950
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COPY THE PARAMETER VALUES OF CORE NO. KC1.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   15 XOD    (N)= XOD    (KC1)                                          RCF 0960
      XID    (N)= XID    (KC1)                                          RCF 0970
      PHIS   (N)= PHIS   (KC1)                                          RCF 0980
      PHIR   (N)= PHIR   (KC1)                                          RCF 0990
      CORMAT(N) = CORMAT(KC1)                                           RCF 1000
      TYPE   (N)= TYPE   (KC1)                                          RCF 1010
      FD1    (N)= FD1    (KC1)                                          RCF 1020
      FD2    (N)= FD2    (KC1)                                          RCF 1030
      FD2    (N)= FD2    (KC1)                                          RCF 1040
      FD3    (N)= FD3    (KC1)                                          RCF 1050
      PHID2  (N)= PHID2  (KC1)                                          RCF 1060
      HA     (N)= HA     (KC1)                                          RCF 1070
      HQ     (N)= HQ     (KC1)                                          RCF 1080
      HN     (N)= HN     (KC1)                                          RCF 1090
      XLMDAD(N)= XLMDAD(KC1)                                            RCF 1100
      XNUD   (N)= XNUD   (KC1)                                          RCF 1110
      FDB    (N)= FDB    (KC1)                                          RCF 1120
      FOPP   (N)= FOPP   (KC1)                                          RCF 1130
      XLAMDA(N)= XLAMDA(KC1)                                            RCF 1140
      XNU    (N)= XNU    (KC1)                                          RCF 1150
      FB     (N)= FB     (KC1)                                          RCF 1160
      FO     (N)= FO     (KC1)                                          RCF 1170
      ROP    (N)= ROP    (KC1)                                          RCF 1180
      FB1    (N)= FB1    (KC1)                                          RCF 1190
      FO1    (N)= FO1    (KC1)                                          RCF 1200
      ROP1   (N)= ROP1   (KC1)                                          RCF 1210
 2202 CONTINUE                                                          RCF 1220
      DO 25 N = 1,NCORE                                                 RCF 1230
      IF (DMAX .EQ. 0.0) GO TO 4444                                     RCF 1240
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     MODIFY NOMINAL PARAMETER VALUES BY DMAX PERCENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ  (5,4445) (ESSES(I),I=1,24)                                  RCF 1250
      DO 4447 I = 1,24                                                  RCF 1260
 4447 ESSES(I) = ESSES(I) * DMAX                                        RCF 1270
      XOD    (N)= XOD    (N) * (1.0 + ESSES( 1) / 100.0)                RCF 1280
      XID    (N)= XID    (N) * (1.0 + ESSES( 2) / 100.0)                RCF 1290
      PHIS   (N)= PHIS   (N) * (1.0 + ESSES( 3) / 100.0)                RCF 1300
      PHIR   (N)= PHIR   (N) * (1.0 + ESSES( 4) / 100.0)                RCF 1310
      FD1    (N)= FD1    (N) * (1.0 + ESSES( 5) / 100.0)                RCF 1320
      FD2    (N)= FD2    (N) * (1.0 + ESSES( 6) / 100.0)                RCF 1330
      FD2    (N)= FD2    (N) * (1.0 + ESSES( 7) / 100.0)                RCF 1340
      FD3    (N)= FD3    (N) * (1.0 + ESSES( 8) / 100.0)                RCF 1350
      PHID2  (N)= PHID2  (N) * (1.0 + ESSES( 9) / 100.0)                RCF 1360
      HA     (N)= HA     (N) * (1.0 + ESSES(10) / 100.0)                RCF 1370
      HQ     (N)= HQ     (N) * (1.0 + ESSES(11) / 100.0)                RCF 1380
      HN     (N)= HN     (N) * (1.0 + ESSES(12) / 100.0)                RCF 1390
      XLMDAD(N)= XLMDAD(N) * (1.0 + ESSES(13) / 100.0)                  RCF 1400
      XNUD   (N)= XNUD   (N) * (1.0 + ESSES(14) / 100.0)                RCF 1410
      FDB    (N)= FDB    (N) * (1.0 + ESSES(15) / 100.0)                RCF 1420
      FOPP   (N)= FOPP   (N) * (1.0 + ESSES(16) / 100.0)                RCF 1430
      XLAMDA(N)= XLAMDA(N) * (1.0 + ESSES(17) / 100.0)                  RCF 1440
      XNU    (N)= XNU    (N) * (1.0 + ESSES(18) / 100.0)                RCF 1450
      FB     (N)= FB     (N) * (1.0 + ESSES(19) / 100.0)                RCF 1460
      FO     (N)= FO     (N) * (1.0 + ESSES(20) / 100.0)                RCF 1470
      ROP    (N)= ROP    (N) * (1.0 + ESSES(21) / 100.0)                RCF 1480
      FB1    (N)= FB1    (N) * (1.0 + ESSES(22) / 100.0)                RCF 1490
      FO1    (N)= FO1    (N) * (1.0 + ESSES(23) / 100.0)                RCF 1500
      ROP1   (N)= ROP1   (N) * (1.0 + ESSES(24) / 100.0)                RCF 1510
 4444 CONTINUE                                                          RCF 1520
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PRELIMINARY COMPUTATION OF CERTAIN PARAMETERS.
```

```
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      PHIR(N) = U0(N) * PHIR(N)                                          RCF 1530
      PHID2D = PHID2(N)                                                  RCF 1540
      HAD    = HA(N)                                                     RCF 1550
      HQD    = HQ(N)                                                     RCF 1560
      HND    = HN(N)                                                     RCF 1570
      XLO = 3.1415927 * 0.0254 * XOD(N)                                  RCF 1580
      XLI = 3.1415927 * 0.0254 * XID(N)                                  RCF 1590
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTATION OF AUXILIARY STATIC PARAMETERS P1(N) THROUGH P20(N).
C     FROM HERE UNTIL THE END OF COMPUTATION OF AUXILIARY STATIC-
C        PARAMETERS, SUBROUTINES RWCORF AND RWCORH ARE IDENTICAL.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      P1(N)  = (PHIS(N) - PHIR(N))/((XLO - XLI) * HAD   )                RCF 1600
      P2(N) = HAD  *XLO                                                  RCF 1610
      P3(N) = HAD  *XLI                                                  RCF 1620
      P4 (N) = P1(N) * (P2(N)-P3(N))                                     RCF 1630
      IF (TYPE(N) .NE. HOLF) GO TO 200                                   RCF 1640
          IF (FD2(N) .EQ. 0.0) FD2(N) = FD1(N)                           RCF 1650
          PHID2D   = P1(N) * ALOG((FD2(N)+P2(N))/(FD2(N)+P3(N))) *       RCF 1660
     2        FD2(N) + PHIR(N)                                            RCF 1670
          GO TO 201                                                      RCF 1680
  200 IF (TYPE(N) .EQ. HOLG) GO TO 201                                   RCF 1690
      IF (TYPE(N) .EQ. HOLB) GO TO 201                                   RCF 1700
      IF (TYPE(N) .NE. HOLC) GO TO 206                                   RCF 1710
  201 IF (FD2 (N) .EQ.  0.0  ) GO TO 202                                 RCF 1720
      IF (FD2 (N) .EQ. FD1(N)) GO TO 202                                 RCF 1730
      IF (FD2(N)*2.0 .NE. FD1(N)+FD2(N)) GO TO 205                       RCF 1740
  202     IF (FD2 (N) .EQ.  0.0  ) GO TO 203                             RCF 1750
          IF (FD2 (N) .NE. FD1(N)) GO TO 204                             RCF 1760
  203         FD2(N) = FD1(N)                                            RCF 1770
              FD1(N) = 0.95 * FD1(N)                                     RCF 1780
  204     FD2(N) = 0.5 * (FD1(N) + FD2(N))                               RCF 1790
          P5(N) = P1(N) * ALOG((FD1(N)-P2(N))/(FD1(N)-P3(N))) * FD1(N)   RCF 1800
     2        - PHIR(N)                                                   RCF 1810
          P6(N) =(PHID2D   - P5(N)) / (FD2(N) - FD1(N))                  RCF 1820
          P7(N) = 1.0                                                    RCF 1830
          P8(N) = 0.0                                                    RCF 1840
          P9(N) = 0.0                                                    RCF 1850
      GO TO 206                                                          RCF 1860
  205 Q1 = P1(N) * ALOG((FD1(N) - P2(N)) / (FD1(N) - P3(N)))             RCF 1870
      P5(N) = Q1 * FD1(N) - PHIR(N)                                      RCF 1880
      P6(N)=Q1+ P4(N)*FD1(N)/((FD1(N)-P2(N))*(FD1(N)-P3(N)))             RCF 1890
      P7(N) =ALOG((PHID2D  -P5(N)-P6(N)*(FD2(N)-FD1(N)))/(0.5*(PHID2D  -RCF 1900
     2P5(N))-P6(N)*(FD2(N)-FD1(N))))                                     RCF 1910
     3                              / ALOG((FD2(N)-FD1(N))/(FD2(N)-FD1(N)))RCF 1920
      P8(N) =(PHID2D-P5(N)-P6(N)*(FD2(N)-FD1(N)))/(FD2(N)-FD1(N))**P7(N)RCF 1930
      P9(N) = P7(N) - 1.0                                                RCF 1940
  206 IF (TYPE(N) .EQ. HOLG) GO TO 207                                   RCF 1950
      IF (TYPE(N) .EQ. HOLA) GO TO 207                                   RCF 1960
      IF (TYPE(N) .EQ. HOLB) GO TO 207                                   RCF 1970
      IF (TYPE(N) .NE. HOLD) GO TO 208                                   RCF 1980
  207 HS = HAD   + HQD   + HND   + PHIR(N)*(HAD  +HQD  -HND   )/PHIS(N)  RCF 1990
      HDMIN = 0.25 * (HS   -SQRT(HS**2-8.0*HAD  *HQD                     RCF 2000
     2    *(1.0+PHIR(N)/PHIS(N))))                                       RCF 2010
      P0    = (PHIS(N)+PHIR(N))*HQD   /((XLO-XLI)*HND   )                RCF 2020
      P12(N)= P0 / HDMIN                                                 RCF 2030
      P13(N)= P0 * (1.0/HND  -1.0/HQD  )                                 RCF 2040
      P14(N)= 1.0 - HND  /HDMIN                                          RCF 2050
      P15(N)= HND   * XLI                                                RCF 2060
      P16(N)= PHIR(N) + P0*XLI                                           RCF 2070
      P17(N)= HND   * XLO                                                RCF 2080
      P18(N)= P16(N) - P0*XLO                                            RCF 2090
      P19(N)= P13(N) * P15(N)                                            RCF 2100
      P20(N)= P13(N) * (P17(N)-P15(N))                                   RCF 2110
      IF (TYPE(N) .EQ. HOLD) FD1(N) = HDMIN * XLI                        RCF 2120
```

```
      FD4(N) = HDMIN * XLO                                                  RCF 2130
      GO TO 289                                                             RCF 2140
  208 P13(N)= P1(N)                                                         RCF 2150
      P15(N)= -P3(N)                                                        RCF 2160
      P17(N)= -P2(N)                                                        RCF 2170
      P18(N)= -PHIR(N)                                                      RCF 2180
      P20(N)= -P4(N)                                                        RCF 2190
  289 IF (TYPE(N) .EQ. HOLA) GO TO 210                                      RCF 2200
      IF (TYPE(N) .EQ. HOLG) GO TO 210                                      RCF 2210
      IF (TYPE(N) .EQ. HOLC) GO TO 209                                      RCF 2220
      IF (TYPE(N) .NE. HOLE) GO TO 216                                      RCF 2230
  209 PHID3 = P1(N)*ALOG((FD3(N)+P2(N))/(FD3(N)+P3(N))) + PHIR(N)           RCF 2240
      GO TO 212                                                             RCF 2250
  210 IF (FD3(N) .LT. FD4(N)) GO TO 211                                     RCF 2260
      PHID3 = P13(N)*ALOG((FD3(N)-P17(N))/(FD3(N)-P15(N)))*FD3(N)-P18(N)RCF 2270
      GO TO 212                                                             RCF 2280
  211 PHID3 = (P12(N) + P13(N)*ALOG(P14(N)*FD3(N)/(FD3(N)-P15(N))) )        RCF 2290
     2    * FD3(N) - P16(N)                                                RCF 2300
  212 IF (TYPE(N) .EQ. HOLC) GO TO 213                                      RCF 2310
      IF (TYPE(N) .NE. HOLG) GO TO 214                                      RCF 2320
  213 P10(N)= (PHID3 - PHID2D  ) / (FD3(N) - FD2(N))                        RCF 2330
      GO TO 215                                                             RCF 2340
  214 PHID1S= P1(N)*ALOG((FD1(N)-P2(N))/(FD1(N)-P3(N)))*FD1(N) - PHIR(N)RCF 2350
      P10(N)= (PHID3 - PHID1S) / (FD3(N) - FD1(N))                          RCF 2360
  215 P11(N)= P10(N) * FD3(N) - PHID3                                       RCF 2370
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     END OF COMPUTATION OF AUXILIARY STATIC PARAMETERS.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  216 ZV(N)  =0.003* XLAMDA(N) * (0.3*FOPP(N))**XNU(N)                      RCF 2380
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     WRITE CORE-WINDINGS DATA.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      WRITE      (6,506)  N                                                RCF 2390
      WRITE               (6,507)                                          RCF 2400
      WRITE               (6,508)                                          RCF 2410
      NWSUBN = NW(N)                                                       RCF 2420
      DO 31 M = 1,NWSUBN                                                   RCF 2430
      KO = KC(N,M)                                                         RCF 2440
      IXNT = XNT(N,M)                                                      RCF 2450
      GO TO (17,18,21),KO                                                  RCF 2460
   17 WRITE (6,509) M,K1C(N,M),K2C(N,M),IXNT,RW(N,M),CICORE(N,M)           RCF 2470
      GO TO 24                                                             RCF 2480
   18 IF(K2C(N,M) .LE. 0) GO TO 20                                         RCF 2490
      WRITE (6,510) M,K1C(N,M),K2C(N,M),IXNT,RW(N,M),CICORE(N,M)           RCF 2500
      GO TO 24                                                             RCF 2510
   20 K2C(N,M) = 20                                                        RCF 2520
      WRITE (6,511) M,K1C(N,M),          IXNT,RW(N,M),CICORE(N,M)          RCF 2530
      GO TO 24                                                             RCF 2540
   21 IF (K1C(N,M) .LE. 0) GO TO 23                                        RCF 2550
      WRITE (6,512) M,K1C(N,M),K2C(N,M),IXNT,RW(N,M),CICORE(N,M)           RCF 2560
      GO TO 24                                                             RCF 2570
   23 K1C(N,M) = 20                                                        RCF 2580
      WRITE (6,513) M,          K2C(N,M),IXNT,RW(N,M),CICORE(N,M)          RCF 2590
   24 I = NS(N,M)                                                          RCF 2600
      IF (I .GT. 1) WRITE (6,7906) I                                       RCF 2610
      J = NP(N,M)                                                          RCF 2620
      IF (J .GT. 1) WRITE (6,7905) J                                       RCF 2630
   31 CONTINUE                                                             RCF 2640
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     WRITE CORE PARAMETERS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      WRITE               (6,514)                                          RCF 2650
      CALL HOLLER(6HFERRIT,T1)                                             RCF 2660
      CALL HOLLER(6HE     ,T2)                                             RCF 2670
      IF (CORMAT(N) .EQ. HOLF) GO TO 1111                                  RCF 2680
          CALL HOLLER(6HTAPE-W,T1)                                         RCF 2690
```

```
      CALL HOLLER(6HOUND  ,T2)                                          RCF 2700
1111  IF (DMAX .GT.0.0)WRITE (6,4446) ESSES(1),ESSES(2)                 RCF 2710
      WRITE          (6,515)CC,XOD(N),XID(N),TYPE(N),T1,T2              RCF 2720
      IF (DMAX .GT.0.0)WRITE (6,4446) (ESSES(I),I=3,7)                  RCF 2730
      WRITE          (6,516)CC,PHIS(N),PHIR(N),FD1(N),FDZ(N),FD2(N)     RCF 2740
      IF (DMAX .GT.0.0)WRITE (6,4446) (ESSES(I),I=8,12)                 RCF 2750
      WRITE          (6,517)CC,FD3(N),PHID2D  ,HAD  ,HQD  ,HND          RCF 2760
      IF (DMAX .GT. 0) WRITE (6,4446) (ESSES(I),I=13,17)                RCF 2770
      WRITE          (6,518)CC,XLMDAD(N),XNUD(N),FDB(N),FOPP(N),XLAMDA(N)RCF 2780
      IF (DMAX .GT.0.0)WRITE (6,4446) (ESSES(I),I=18,21)                RCF 2790
      WRITE          (6,519)CC,XNU(N),FB(N),FO(N),ROP(N)                RCF 2800
      IF (DMAX .GT.0.0)WRITE (6,4446) (ESSES(I),I=22,24)                RCF 2810
      WRITE          (6,520)CC,FB1(N),FO1(N),ROP1(N)                    RCF 2820
      WRITE            (6,525)                                          RCF 2830
      IF (MONCOR) WRITE (6,3167) XLO,XLI,HS,HDMIN,PO,PHID3,PHID1S       RCF 2840
      WRITE            (6,527)  P1(N), P2(N), P3(N), P4 (N), P5 (N),    RCF 2850
     2           P6(N), P7(N), P8(N), P9(N), P10(N), P11(N), P12(N) ,   RCF 2860
     3   P13(N), P14(N), P15(N),P16(N),P17(N),P18(N),P19(N),P20(N),ZV(N)RCF 2870
      IF (NSVVAR .NE. 0) GO TO 404                                      RCF 2880
      IF (NSVPHI .NE. 0) GO TO 404                                      RCF 2890
      WRITE            (6, 532) PHIO(N)                                 RCF 2900
      GO TO 25                                                          RCF 2910
404   WRITE            (6, 532) PHI(N)                                  RCF 2920
 25   CONTINUE                                                          RCF 2930
      RETURN                                                            RCF 2940
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
 150  FORMAT (16HOELASTIC PHIDOT  ,2A6)                                 RCF 2950
 501  FORMAT (4E12.5,11X,2A1)                                           RCF 2960
 503  FORMAT (2I12,E12.5)                                               RCF 2970
 504  FORMAT(A6,I6,A6,I6,F12.0,E12.5,2I6,E12.5)                         RCF 2980
 505  FORMAT(6E12.5)                                                    RCF 2990
 506  FORMAT(1H1,/,43X,14HC O R E  N O.  ,I2,/)                         RCF 3000
 507  FORMAT(47X,15HW I N D I N G S,/ )                                 RCF 3010
 508  FORMAT(7X,11HWINDING NO.,8X,6HNODE B,11X,6HNODE T,10X,5HTURNS,8X, RCF 3020
     1 10HRESISTANCE,9X,15HINITIAL CURRENT,//)                          RCF 3030
 509  FORMAT(1HO,5X,I6,11X,      I6,11X,      I6,5X,  I12,2(8X,E12.5))  RCF 3040
 510  FORMAT(1HO,5X,I6,11X,      I6,10X,1HS,  I6,5X,  I12,2(8X,E12.5))  RCF 3050
 511  FORMAT(1HO,5X,I6,11X,      I6,14X,6HGROUND,2X,  I12,2(8X,E12.5))  RCF 3060
 512  FORMAT(1HO,5X,I6,10X,1HS,  I6,11X,      I6,5X,  I12,2(8X,E12.5))  RCF 3070
 513  FORMAT(1HO,5X,I6,14X,6HGROUND, 8X,      I6,5X,  I12,2(8X,E12.5))  RCF 3080
 514  FORMAT(//,43X,29HC O R E   P A R A M E T E R S,  )                RCF 3090
 515  FORMAT(A1 ,4X,3HOD=,E12.5,8X,3HID=,E12.5,6X,5HTYPE-  ,A1,13X,     RCF 3100
     2    10HMATERIAL-  ,2A6)                                           RCF 3110
 516  FORMAT(A1 ,2X,5HPHIS=,E12.5,6X,5HPHIR=,E12.5,7X,4HFD1=,E12.5,7X,  RCF 3120
     2      4HFDZ=,E12.5,7X,4HFD2=,E12.5)                              RCF 3130
 517  FORMAT(A1 ,3X,4HFD3=,E12.5,5X,6HPHID2=,E12.5,8X,3HHA=,E12.5,8X,3HHRCF 3140
     1Q=,E12.5, 8X,3HHN=,E12.5)                                         RCF 3150
 518  FORMAT(A1,                                                        RCF 3160
     2       7HLAMDAD=,E12.5,7X,4HNUD=,E12.5,7X,4HFDB=,E12.5,6X,5HFOPP=RCF 3170
     1,E12.5,5X,6HLAMDA=,E12.5)                                         RCF 3180
 519  FORMAT(A1 ,4X,3HNU=,E12.5,8X,3HFB=,E12.5,8X,3HFO=,E12.5,7X,4HROP=,RCF 3190
     1E12.5)                                                            RCF 3200
 520  FORMAT(A1 ,3X,4HFB1=,E12.5,7X,4HFO1=,E12.5,6X,5HROP1=,E12.5)      RCF 3210
 523  FORMAT(16X,39HTHE MAXIMUM NUMBER OF WINDINGS FOR CORE,I3,11HIS EXCRCF 3220
     1EEDED)                                                            RCF 3230
 524  FORMAT(16X,4HCORE,I3,16H HAS NO WINDINGS)                         RCF 3240
 525  FORMAT(//22X,57HC O M P U T E D   A U X I L I A R Y   P A R A M E RCF 3250
     1T E R S)                                                          RCF 3260
 527  FORMAT(/6X,3HP1=,E12.5,8X,3HP2=,E12.5,7X,3HP3=,E12.5,7X,4H P4=,   RCF 3270
     1  E12.5,7X,4H P5=,E12.5//6X,3HP6=,E12.5,8X,3HP7=,E12.5,7X,3HP8=,  RCF 3280
     2  E12.5,8X,3HP9=,E12.5,7X,4HP10=,E12.5//5X,4HP11=,E12.5,7X,4HP12=,RCF 3290
     3  E12.5,6X,4HP13=,E12.5,7X,4HP14=,E12.5,7X,4HP15=,E12.5,//,5X,    RCF 3300
     4    4HP16=,E12.5,7X,4HP17=,E12.5,6X,4HP18=,E12.5,7X,4HP19=,E12.5, RCF 3310
     5    7X,4HP20=,E12.5,//,5X,4HZV = ,E12.5)                          RCF 3320
 528  FORMAT(//38H DO NOT EXIT WHEN FLUX SWITCHING STOPS)               RCF 3330
 529  FORMAT(//49H EXIT WHEN FLUX SWITCHING STOPS IF T GREATER THAN,I4, RCF 3340
```

```
     15H*DELT)                                                                  RCF 3350
531   FORMAT (3E12.5,F12.0,2I12)                                                RCF 3360
532   FORMAT(///6X,16H INITIAL FLUX = ,E12.5)                                   RCF 3370
533   FORMAT (// 10H RELERR = ,F10.3, 8X,  9HABSERR = ,E10.3//)                 RCF 3380
770   FORMAT ( 7H HS,PO  ,2E22.15)                                              RCF 3390
4445 FORMAT (4F12.0,/,5F12.0,/,3F12.0,2(/,6F12.0))                              RCF 3400
4446 FORMAT (1H0,6X,5(F3.0,8H PERCENT ,12X))                                    RCF 3410
7778 FORMAT (10HOPSTEP   = ,E12.5)                                              RCF 3420
7905 FORMAT (14H REPRESENTING ,I2,32H IDENTICALLY SWITCHING CORES IN            RCF 3430
     2      ,9HPARALLEL  )                                                      RCF 3440
7906 FORMAT (14H REPRESENTING ,I2,32H IDENTICALLY SWITCHING CORES IN            RCF 3450
     2      ,7HSERIES  )                                                        RCF 3460
3167 FORMAT (34H XLO,XL1,HS,HDMIN,PO,PHID3,PHID1S  ,2(/,5E24.15))               RCF 3470
9789 FORMAT (16H1MAGNETIC CORES   ////)                                         RCF 3480
     END                                                                        RCF 3490
```

```
      SUBROUTINE RWCORH                                                 RCH 0010
C     RWCORH READS IN AND PRINTS OUT B-H DATA OF CORES
```

COMMON VARIABLES

```
      LOGICAL MONCOR                                                    RCH 0020
      REAL NS,NP                                                        RCH 0030
      INTEGER STSWEX                                                    RCH 0040
      EQUIVALENCE (KCORE,KC),(K1CORE,K1C),(K2CORE,K2C)                  RCH 0050
      DIMENSION  A1N(20),   BD2IN(20),BD3IN(20) ,  BMIN(20),  BRIN(20), RCH 0060
     2           CORMAT(20),ESSES(12),FREQ(20,10), HD1IN(20), HD2IN(20),RCH 0070
     3  HD3IN(20),HDZIN(20),HMIN(20),HOBIN(20,10),NDYNP(20),  TYPE(20), RCH 0080
     4   IO(20), XID(20), XOD(20)                                       RCH 0090
      DIMENSION KC(20,10), K1C(20,10), K2C(20,10)                       RCH 0100
C     . . . . . .
C     READ IN DATA COMMON TO ALL CORES.
C     SET HOLLERITH CONSTANTS FOR PHID(F) TYPES, BLANKS,
C        AND CARRIAGE CONTROL.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ  (5,531) RELERR,ABSERR,PSTEP  ,STSWEX,NUDT,LASTIC            RCH 0110
      IF (RELERR .EQ. 0.0) RELERR = 0.0001                              RCH 0120
      IF (ABSERR .EQ. 0.0) ABSERR = 0.0001                              RCH 0130
      CALL HOLLER(1H ,HOL )                                             RCH 0140
      CALL HOLLER(1HA,HOLA)                                             RCH 0150
      CALL HOLLER(1HB,HOLB)                                             RCH 0160
      CALL HOLLER(1HC,HOLC)                                             RCH 0170
      CALL HOLLER(1HD,HOLD)                                             RCH 0180
      CALL HOLLER(1HE,HOLE)                                             RCH 0190
      CALL HOLLER(1HF,HOLF)                                             RCH 0200
      CALL HOLLER(1HG,HOLG)                                             RCH 0210
      CALL HOLLER(1HT,HOLT)                                             RCH 0220
      CALL HOLLER (6H      ,F3)                                         RCH 0230
      CALL HOLLER(1HO,CC)                                               RCH 0240
      IF (DMAX .NE. 0.0) CALL HOLLER (1H ,CC)                           RCH 0250
      IF (NSVPHI .NE. 1) GO TO 7757                                     RCH 0260
      IF (NSVVAR .EQ. 1) GO TO 7757                                     RCH 0270
      IF (KONT   .EQ. 1) GO TO 7757                                     RCH 0280
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     SUBTRACT ELASTIC FLUX CHANGE FROM PHI OF PREVIOUS MODE,
C        UNLESS NSVVAR=1 OR KONT=1.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DO 7759 N = 1, NCORE                                              RCH 0290
         PHI(N)=PHI(N)-P1(N)*F(N)*ALOG((ABS(F(N))+P2(N))/(ABS(F(N))+P3(RCH 0300
     2      N))                                                        RCH 0310
         IF (ABS(PHI(N)) .LE. PHIR(N)) GO TO 7759                       RCH 0320
         IF (PHI(N)) 7761,7759,7762                                     RCH 0330
 7761    PHI(N) = -PHIR(N)                                              RCH 0340
         GO TO 7759                                                     RCH 0350
 7762    PHI(N) =  PHIR(N)                                              RCH 0360
 7759 CONTINUE                                                          RCH 0370
 7757 WRITE (6,9789)                                                    RCH 0380
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK AND WRITE
C        (1) WHETHER ELASTIC SWITCHING IS INCLUDED.
C        (2) WHETHER TO TERMINATE MODE RUN WHEN SWITCHING
C           OF ALL CORES IS COMPLETED.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
      IF (LASTIC .GE. 1) GO TO 477                                      RCH 0390
      CALL HOLLER(6HNEGLEC,T1)                                          RCH 0400
      CALL HOLLER(6HTED   ,T2)                                          RCH 0410
      GO TO 478                                                         RCH 0420
  477 CALL HOLLER(6HINCLUD,T1)                                          RCH 0430
      CALL HOLLER(6HED    ,T2)                                          RCH 0440
  478 WRITE (6,150) T1,T2                                               RCH 0450
      RNUDT = NUDT                                                      RCH 0460
      IF (STSWEX .NE.0.0)GO TO 8                                        RCH 0470
      WRITE             (6,528)                                         RCH 0480
      GO TO 9                                                           RCH 0490
    8 WRITE             (6,529)NUDT                                     RCH 0500
    9 NOEXIT = 1                                                        RCH 0510
      WRITE             (6, 533) RELERR, ABSERR                         RCH 0520
      WRITE(6,7778) PSTEP                                               RCH 0530
      DO 2202 N=1,NCORE                                                 RCH 0540
      READ  (5,503) NWSUBN, KC1, UO(N)                                  RCH 0550
      NW(N) = NWSUBN                                                    RCH 0560
      IF (NWSUBN .GT. 0) GO TO 6                                        RCH 0570
      WRITE             (6,524)    N                                    RCH 0580
      CALL EXIT                                                         RCH 0590
    6 IF (NWSUBN .LE. 10) GO TO 7                                       RCH 0600
      WRITE             (6,523)N                                        RCH 0610
      CALL EXIT                                                         RCH 0620
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     READ IN WINDING DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
    7 DO 13 M = 1,NWSUBN                                                RCH 0630
      READ (5,504) F1C,K1C(N,M),R2C,K2C(N,M),XNT(N,M),RW(N,M),I,J,CIO   RCH 0640
      IF (I .LE. 0) I = 1                                               RCH 0650
      IF (J .LE. 0) J = 1                                               RCH 0660
      NP(N,M) = I                                                       RCH 0670
      NS(N,M) = J                                                       RCH 0680
      IF ((NSVVAR .NE. 1) .AND. (KONT .NE. 1)) CICORE(N,M) = CIO        RCH 0690
      IF (RW(N,M) .LE.0.0) RW(N,M) = 0.1                                RCH 0700
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     DETERMINE TYPE OF TERMINALS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (F1C    .NE. F3) GO TO 12                                      RCH 0710
      IF (R2C    .NE. F3)GO TO 11                                       RCH 0720
      KC(N,M)=1                                                         RCH 0730
      GO TO 13                                                          RCH 0740
   11 KC(N,M)=2                                                         RCH 0750
      GO TO 13                                                          RCH 0760
   12 KC(N,M)=3                                                         RCH 0770
   13 CONTINUE                                                          RCH 0780
      IF (KC1 .GT.0) GO TO 15                                           RCH 0790
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     UP TO THIS POINT, SUBROUTINES RWCORF AND RWCORH ARE IDENTICAL.
C     READ IN CORE-PARAMETER DATA
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ  (5,501) XOD(N),XID(N),AIN(N),BRIN(N),TYPE(N),CORMAT(N)      RCH 0800
      IF (TYPF(N) .EQ. HOL ) TYPE(N) = HCLC                             RCH 0810
      IF (CORMAT(N) .EQ. HOL ) CORMAT(N) = HCLT                         RCH 0820
      READ  (5,505) HD1IN(N),HD2IN(N),HD2IN(N),HD3IN(N),HMIN(N)         RCH 0830
      READ  (5,505) BD2IN(N),BD3IN(N),BMIN(N)                           RCH 0840
      READ (5,503) NDYNPT                                               RCH 0850
      NDYNP(N)=NDYNPT                                                   RCH 0860
      DO  14 J=1,NDYNPT                                                 RCH 0870
   14 READ (5,505) FREQ(N,J),HOBIN(N,J)                                 RCH 0880
      GO TO 2202                                                        RCH 0890
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COPY THE PARAMETER VALUES OF CORE NO. KC1.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   15 XOD    (N) = XOD    (KC1)                                          RCH 0900
      XID    (N) = XID    (KC1)                                          RCH 0910
```

```
      AIN   (N) = AIN   (KC1)                                                  RCH 0920
      BRIN  (N) = BRIN  (KC1)                                                  RCH 0930
      TYPE  (N) = TYPE  (KC1)                                                  RCH 0940
      CORMAT(N) = CORMAT(KC1)                                                  RCH 0950
      HD1IN (N) = HD1IN (KC1)                                                  RCH 0960
      HDZIN (N) = HDZIN (KC1)                                                  RCH 0970
      HD2IN (N) = HD2IN (KC1)                                                  RCH 0980
      HD3IN (N) = HD3IN (KC1)                                                  RCH 0990
      HMIN  (N) = HMIN  (KC1)                                                  RCH 1000
      BD2IN (N) = BD2IN (KC1)                                                  RCH 1010
      BD3IN (N) = BD3IN (KC1)                                                  RCH 1020
      BMIN  (N) = BMIN  (KC1)                                                  RCH 1030
      NDYNPT    = NDYNP (KC1)                                                  RCH 1040
      NDYNP (N) = NDYNPT                                                       RCH 1050
      DO 650 J = 1,NDYNPT                                                      RCH 1060
      FREQ(N,J)=FREQ(KC1,J)                                                    RCH 1070
  650 HOBIN(N,J)=HOBIN(KC1,J)                                                  RCH 1080
 2202 CONTINUE                                                                 RCH 1090
      DO 25 N=1,NCORE                                                          RCH 1100
      NDYNPT=NDYNP(N)                                                          RCH 1110
      IF (DMAX .EQ.0.0)GO TO 4444                                              RCH 1120
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     MODIFY NOMINAL PARAMETER VALUES BY DMAX PERCENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      READ (5,4445)(ESSES(I),I=1,13)                                           RCH 1130
      DO 4447 I = 1,13                                                         RCH 1140
 4447 ESSES(I) = FSSES(I) * DMAX                                               RCH 1150
      XOD   (N)=XOD   (N)* (1.0 + ESSES(1) / 100.0)                            RCH 1160
      XID   (N)=XID   (N)* (1.0 + ESSES(2) / 100.0)                            RCH 1170
      AIN   (N)=AIN   (N)* (1.0 + ESSES(3) / 100.0)                            RCH 1180
      BRIN  (N)=BRIN  (N)* (1.0 + ESSES(4) / 100.0)                            RCH 1190
      HD1IN (N)=HD1IN (N)* (1.0 + ESSES(5) / 100.0)                            RCH 1200
      HDZIN (N)=HDZIN (N)* (1.0 + ESSES(6) / 100.0)                            RCH 1210
      HD2IN (N)=HD2IN (N)* (1.0 + ESSES(7) / 100.0                             RCH 1220
      HD3IN (N)=HD3IN (N)* (1.0 + ESSES(8) / 100.0)                            RCH 1230
      HMIN  (N)=HMIN  (N)* (1.0 + ESSES(9) / 100.0)                            RCH 1240
      BD2IN (N)=BD2IN (N)* (1.0 + ESSES(10)/ 100.0)                            RCH 1250
      BD3IN (N)=BD3IN (N)* (1.0 + ESSES(11)/ 100.0)                            RCH 1260
      BMIN  (N)=BMIN  (N)* (1.0 + ESSES(12)/ 100.0)                            RCH 1270
      DO 651 I=1,NDYNPT                                                        RCH 1280
  651 HOBIN(N,I)=HOBIN(N,I)*(1.0+ ESSES(13)/ 100.0)                            RCH 1290
 4444 CONTINUE                                                                 RCH 1300
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CONVERT INPUT CORE-PARAMETER VALUES TO MKS UNITS.
C     COMPUTE STATIC PHI(F) PARAMETERS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      XLO = 3.1415927 * 0.0254 * XOD(N)                                        RCH 1310
      XLI = 3.1415927 * 0.0254 * XID(N)                                        RCH 1320
      A   = AIN  (N) * 1.0E-4                                                  RCH 1330
      BR  = BRIN (N) * 1.0E-4                                                  RCH 1340
      HD1 = HD1IN(N) * 79.577                                                  RCH 1350
      HDZ = HDZIN(N) * 79.577                                                  RCH 1360
      HD2 = HD2IN(N) * 79.577                                                  RCH 1370
      HD3 = HD3IN(N) * 79.577                                                  RCH 1380
      HM  = HMIN (N) * 79.577                                                  RCH 1390
      BD2 = BD2IN(N) * 1.0E-4                                                  RCH 1400
      BD3 = BD3IN(N) * 1.0E-4                                                  RCH 1410
      BM  = BMIN (N) * 1.0E-4                                                  RCH 1420
      IF (TYPE(N) .EQ. HOLC) GO TO 1625                                        RCH 1430
      IF (TYPE(N) .EQ. HOLE) GO TO 1625                                        RCH 1440
      IF (TYPE(N) .EQ. HOLF) GO TO 1625                                        RCH 1450
          BS = BM + BM - BR                                                    RCH 1460
          GO TO 1630                                                           RCH 1470
 1625 BS = 1.1 * BR                                                            RCH 1480
 1630 PHIS(N)=BS * A                                                           RCH 1490
      PHIR(N)=BR * A                                                           RCH 1500
```

```
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PRELIMINARY COMPUTATION OF CERTAIN PARAMETERS
C        IN ORDER TO MATCH WITH SUBROUTINE RWCORF.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      PHIO(N) = UO(N) * PHIR(N)                                        RCH 1510
      PHID2D= BD2 * A                                                  RCH 1520
      FD1(N)= HD1 * XLI                                                RCH 1530
      FDZ(N)= HDZ * XLI                                                RCH 1540
      FD2(N)= HD2 * XLI                                                RCH 1550
      FD3(N)= HD3 * XLO                                                RCH 1560
      IF (TYPE(N) .NE. HOLF) GO TO 171                                 RCH 1570
          FDZ(N) = 0.5 * HDZ * (XLO+XLI)                               RCH 1580
          FD2(N) = HD2 * XLO                                           RCH 1590
          HD3    = HD2                                                 RCH 1600
  171 HAD    = HM                                                      RCH 1610
      IF (TYPE(N) .EQ. HOLC) GO TO 172                                 RCH 1620
      IF (TYPE(N) .EQ. HOLE) GO TO 172                                 RCH 1630
      IF (TYPE(N) .NE. HOLF) GO TO 173                                 RCH 1640
  172     HAD = 4.0 * HD3                                              RCH 1650
          GO TO 175                                                    RCH 1660
  173 IF (TYPE(N) .NE. HOLB) GO TO 174                                 RCH 1670
          HD3 = HD2                                                    RCH 1680
          BD3 = BD2                                                    RCH 1690
  174 HND    = (HD3 * (BS-BD3) - HM * (BM-BR)) / (BM-BD3)              RCH 1700
      HQD    = HD3 - (BD3+BR) * (HD3-HND) / (BM+BM)                    RCH 1710
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTATION OF AUXILIARY STATIC PARAMETERS P1(N) THROUGH P20(N).
C     FROM HERE UNTIL THE END OF COMPUTATION OF AUXILIARY STATIC-
C        PARAMETERS, SUBROUTINES RWCORF AND RWCCRH ARE IDENTICAL.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  175 P1(N)  = (PHIS(N) - PHIR(N))/((XLO - XLI) * HAD    )             RCH 1720
      P2(N) = HAD  *XLO                                                RCH 1730
      P3(N) = HAD  *XLI                                                RCH 1740
      P4 (N) = P1(N) * (P2(N)-P3(N))                                   RCH 1750
      IF (TYPE(N) .NE. HOLF) GO TO 200                                 RCH 1760
          IF (FD2(N) .EQ. 0.0) FD2(N) = FD1(N)                         RCH 1770
          PHID2D   = P1(N) * ALOG((FD2(N)+P2(N))/(FD2(N)+P3(N))) *     RCH 1780
     2         FD2(N) + PHIR(N)                                        RCH 1790
      GO TO 201                                                        RCH 1800
  200 IF (TYPE(N) .EQ. HOLG) GO TO 201                                 RCH 1810
      IF (TYPE(N) .EQ. HOLB) GO TO 201                                 RCH 1820
      IF (TYPE(N) .NE. HOLC) GO TO 206                                 RCH 1830
  201 IF (FDZ (N) .EQ.  0.0  ) GO TO 202                               RCH 1840
      IF (FDZ (N) .EQ. FD1(N)) GO TO 202                               RCH 1850
      IF (FDZ(N)*2.0 .NE. FD1(N)+FD2(N)) GO TO 205                     RCH 1860
  202     IF (FD2 (N) .EQ.  0.0  ) GO TO 203                           RCH 1870
          IF (FD2 (N) .NE. FD1(N)) GO TO 204                           RCH 1880
  203         FD2(N) = FD1(N)                                          RCH 1890
              FD1(N) = 0.95 * FD1(N)                                   RCH 1900
  204     FDZ(N) = 0.5 * (FD1(N) + FD2(N))                             RCH 1910
          P5(N) = P1(N) * ALOG((FD1(N)-P2(N))/(FD1(N)-P3(N))) * FD1(N) RCH 1920
     2         - PHIR(N)                                               RCH 1930
          P6(N) =(PHID2D   - P5(N)) / (FD2(N) - FD1(N))                RCH 1940
          P7(N) = 1.0                                                  RCH 1950
          P8(N) = 0.0                                                  RCH 1960
          P9(N) = 0.0                                                  RCH 1970
      GO TO 206                                                        RCH 1980
  205 Q1 = P1(N) * ALOG((FD1(N) - P2(N)) / (FD1(N) - P3(N)))           RCH 1990
      P5(N) = Q1 * FD1(N) - PHIR(N)                                    RCH 2000
      P6(N)=Q1+ P4(N)*FD1(N)/((FD1(N)-P2(N))*(FD1(N)-P3(N)))           RCH 2010
      P7(N) =ALOG((PHID2D  -P5(N)-P6(N)*(FD2(N)-FD1(N)))/(0.5*(PHID2D  -RCH 2020
     2P5(N))-P6(N)*(FDZ(N)-FD1(N))))                                   RCH 2030
     3                          / ALOG((FD2(N)-FD1(N))/(FDZ(N)-FD1(N)))RCH 2040
      P8(N) =(PHID2D-P5(N)-P6(N)*(FD2(N)-FD1(N)))/(FD2(N)-FD1(N))**P7(N)RCH 2050
      P9(N) = P7(N) - 1.0                                              RCH 2060
  206 IF (TYPE(N) .EQ. HOLG) GO TO 207                                 RCH 2070
```

```
      IF (TYPE(N) .EQ. HOLA) GO TO 207                                  RCH 2080
      IF (TYPE(N) .EQ. HOLB) GO TO 207                                  RCH 2090
      IF (TYPE(N) .NE. HOLD) GO TO 208                                  RCH 2100
  207 HS = HAD   + HQD   + HND   + PHIR(N)*(HAD  +HQD  -HND  )/PHIS(N)  RCH 2110
      HDMIN = 0.25 * (HS   -SQRT(HS**2-8.0*HAD  *HQD                    RCH 2120
     2     *(1.0+PHIR(N)/PHIS(N))))                                     RCH 2130
      P0    = (PHIS(N)+PHIR(N))*HQD   /((XLO-XLI)*HND  )                RCH 2140
      P12(N)= P0 / HDMIN                                                RCH 2150
      P13(N)= P0 * (1.0/HND  -1.0/HQD  )                                RCH 2160
      P14(N)= 1.0 - HND  /HDMIN                                         RCH 2170
      P15(N)= HND   * XLI                                               RCH 2180
      P16(N)= PHIR(N) + P0*XLI                                          RCH 2190
      P17(N)= HND   * XLO                                               RCH 2200
      P18(N)= P16(N) - P0*XLO                                           RCH 2210
      P19(N)= P13(N) * P15(N)                                           RCH 2220
      P20(N)= P13(N) * (P17(N)-P15(N))                                  RCH 2230
      IF (TYPE(N) .EQ. HOLD) FD1(N) = HDMIN * XLI                       RCH 2240
      FD4(N) = HDMIN * XLO                                              RCH 2250
      GO TO 289                                                         RCH 2260
  208 P13(N)= P1(N)                                                     RCH 2270
      P15(N)= -P3(N)                                                    RCH 2280
      P17(N)= -P2(N)                                                    RCH 2290
      P18(N)= -PHIR(N)                                                  RCH 2300
      P20(N)= -P4(N)                                                    RCH 2310
  289 IF (TYPE(N) .EQ. HOLA) GO TO 210                                  RCH 2320
      IF (TYPE(N) .EQ. HOLG) GO TO 210                                  RCH 2330
      IF (TYPE(N) .EQ. HOLC) GO TO 209                                  RCH 2340
      IF (TYPE(N) .NE. HOLE) GO TO 216                                  RCH 2350
  209 PHID3 = P1(N)*ALOG((FD3(N)+P2(N))/(FD3(N)+P3(N))) + PHIR(N)       RCH 2360
      GO TO 212                                                         RCH 2370
  210 IF (FD3(N) .LT. FD4(N)) GO TO 211                                 RCH 2380
      PHID3 = P13(N)*ALOG((FD3(N)-P17(N))/(FD3(N)-P15(N)))*FD3(N)-P18(N)RCH 2390
      GO TO 212                                                         RCH 2400
  211 PHID3 = (P12(N) + P13(N)*ALOG(P14(N)*FD3(N)/(FD3(N)-P15(N))) )    RCH 2410
     2     * FD3(N) - P16(N)                                            RCH 2420
  212 IF (TYPE(N) .EQ. HOLC) GO TO 213                                  RCH 2430
      IF (TYPE(N) .NE. HOLG) GO TO 214                                  RCH 2440
  213 P10(N)= (PHID3 - PHID2D  ) / (FD3(N) - FD2(N))                    RCH 2450
      GO TO 215                                                         RCH 2460
  214 PHID1S= P1(N)*ALOG((FD1(N)-P2(N))/(FD1(N)-P3(N)))*FD1(N) - PHIR(N)RCH 2470
      P10(N)= (PHID3 - PHID1S) / (FD3(N) - FD1(N))                      RCH 2480
  215 P11(N)= P10(N) * FD3(N) - PHID3                                   RCH 2490
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     END OF COMPUTATION OF AUXILIARY STATIC PARAMETERS.
C     COMPUTATION OF DYNAMIC PARAMETERS.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  216 Z1=0.0                                                            RCH 2500
      Z2=0.0                                                            RCH 2510
      Z3=0.0                                                            RCH 2520
      Z4=0.0                                                            RCH 2530
      DO 603 J=1,  NDYNPT                                               RCH 2540
      HOB   =HOBIN(N,J)*79.577                                          RCH 2550
      Z1=Z1+FRFQ(N,J)                                                   RCH 2560
      Z2=Z2+HOB                                                         RCH 2570
      Z3=Z3+FREQ(N,J)*HOB                                               RCH 2580
      Z4=Z4+HOB   **2                                                   RCH 2590
  603 CONTINUE                                                          RCH 2600
      T1 = NDYNPT                                                       RCH 2610
      Z5=( T1   *Z3-Z1*Z2)/( T1   *Z4-Z2**2)                           RCH 2620
      EL=0.5*(XLI+XLO)                                                  RCH 2630
      ROP(N)=Z5*3.1415927*(PHIR(N)+PHIS(N))/EL                          RCH 2640
      F0(N)=EL*(Z2-Z1/Z5)/T1                                            RCH 2650
      IF (CORMAT(N) .EQ. HOLF) GO TO 831                                RCH 2660
      FB(N)=F0(N)*(1.33-0.0007*(ABTEMP - 298.0))                        RCH 2670
      GO TO 832                                                         RCH 2680
  831 FB(N)=F0(N)*(2.0+0.008*(ABTEMP - 298.0))                          RCH 2690
```

```
  832 XNU(N)=(FB(N)-FD1(N))/(FB(N)-FO(N))                                  RCH 2700
      XLAMDA(N)=ROP(N)/(XNU(N)*(FB(N)-FD1(N))**(XNU(N)-1.0))               RCH 2710
      FDB(N) = 0.99 * FD1(N)                                               RCH 2720
      FOPP(N) = FD1(N)                                                     RCH 2730
      FB1(N) = 1.0E+30                                                     RCH 2740
      ZV(N)  =0.003* XLAMDA(N) * (0.3*FOPP(N))**XNU(N)                     RCH 2750
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     WRITE CORE-WINDINGS DATA.
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      WRITE      (6,506)  N                                                RCH 2760
      IF (MONCOR) WRITE (6,770) Q1,P0,EL,Z1,Z2,Z3,Z4,Z5                    RCH 2770
      WRITE             (6,507)                                            RCH 2780
      WRITE             (6,508)                                            RCH 2790
      NWSUBN = NW(N)                                                       RCH 2800
      DO 31 M = 1,NWSUBN                                                   RCH 2810
      KO = KC(N,M)                                                         RCH 2820
      IXNT = XNT(N,M)                                                      RCH 2830
      GO TO (17,18,21),KO                                                  RCH 2840
   17 WRITE (6,509) M,K1C(N,M),K2C(N,M),IXNT,RW(N,M),CICORE(N,M)           RCH 2850
      GO TO 24                                                             RCH 2860
   18 IF(K2C(N,M) .LE. 0) GO TO 20                                         RCH 2870
      WRITE (6,510) M,K1C(N,M),K2C(N,M),IXNT,RW(N,M),CICORE(N,M)           RCH 2880
      GO TO 24                                                             RCH 2890
   20 K2C(N,M) = 20                                                        RCH 2900
      WRITE (6,511) M,K1C(N,M),          IXNT,RW(N,M),CICORE(N,M)          RCH 2910
      GO TO 24                                                             RCH 2920
   21 IF (K1C(N,M) .LE. 0) GO TO 23                                        RCH 2930
      WRITE (6,512) M,K1C(N,M),K2C(N,M),IXNT,RW(N,M),CICORE(N,M)           RCH 2940
      GO TO 24                                                             RCH 2950
   23 K1C(N,M) = 20                                                        RCH 2960
      WRITE (6,513) M,          K2C(N,M),IXNT,RW(N,M),CICORE(N,M)          RCH 2970
   24 I = NS(N,M)                                                          RCH 2980
      IF (I .GT. 1) WRITE (6,7906) I                                       RCH 2990
      J = NP(N,M)                                                          RCH 3000
      IF (J .GT. 1) WRITE (6,7905) J                                       RCH 3010
   31 CONTINUE                                                             RCH 3020
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     WRITE CORE PARAMETERS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      WRITE(6,799)                                                         RCH 3030
      WRITE(6,901)                                                         RCH 3040
      CALL HOLLER(6HFERRIT,T1)                                             RCH 3050
      CALL HOLLER(6HE     ,T2)                                             RCH 3060
      IF (CORMAT(N) .EQ. HOLF) GO TO 1111                                  RCH 3070
          CALL HOLLER(6HTAPE-W,T1)                                         RCH 3080
          CALL HOLLER(6HOUND  ,T2)                                         RCH 3090
 1111 IF (DMAX .GT.0.0)WRITE (6,4446) ESSES(1),ESSES(2),ESSES(3)           RCH 3100
      WRITE (6,902)CC,XOD(N),XID(N),AIN(N)                                 RCH 3110
      WRITE(6,803)                                                         RCH 3120
      IF (DMAX .GT. 0.0) WRITE (6,4466) ESSES(4) ,ESSES(5)                 RCH 3130
      WRITE (6,900) CC,BRIN(N),TYPE(N),T1,T2,HD1IN(N)                      RCH 3140
      IF (DMAX .GT.0.0)WRITE (6,4446) (ESSES(I),I=6,9)                     RCH 3150
      WRITE (6,804) CC,                HDZIN(N),HD2IN(N),HD3IN(N),HMIN(N)RCH 3160
      IF (DMAX .GT.0.0)WRITE (6,4449) (ESSES(I),I=10,12)                   RCH 3170
      WRITE (6,820) CC,BD2IN(N),BD3IN(N),BMIN(N)                           RCH 3180
      WRITE(6,805)                                                         RCH 3190
      IF(DMAX.GT.0.0)WRITE(6,821) ESSES(13)                                RCH 3200
      DO 807 J=1,NDYNPT                                                    RCH 3210
      WRITE(6,806) J,FREQ(N,J),HOBIN(N,J)                                  RCH 3220
  807 CONTINUE                                                             RCH 3230
      WRITE             (6,526)                                            RCH 3240
      WRITE(6,808)A,XLI,XLO,BR,BS,BD2,BD3,BM,HM,HS,HDMIN,HD1,HDZ,HD2,      RCH 3250
     2    HD3,FD4(N),PHID1S,PHID3                                          RCH 3260
      WRITE (6,514)                                                        RCH 3270
      WRITE          (6,515)   XOD(N),XID(N),TYPE(N),T1,T2                 RCH 3280
      WRITE          (6,516)   PHIS(N),PHIR(N),FD1(N),FDZ(N),FD2(N)        RCH 3290
```

```
      WRITE          (6,517)   FD3(N),PHID2D   ,HAD   ,HQD   ,HND            RCH 3300
      WRITE          (6,518)   XLMDAD(N),XNUD(N),FDB(N),FOPP(N),XLAMDA(N)RCH 3310
      WRITE          (6,519)   XNU(N),FB(N),FO(N),ROP(N)                     RCH 3320
      WRITE          (6,520)   FB1(N),FO1(N),ROP1(N)                        RCH 3330
      WRITE             (6,525)                                             RCH 3340
      WRITE             (6,527)  P1(N), P2(N), P3(N), P4 (N), P5 (N),       RCH 3350
     2              P6(N), P7(N), P8(N), P9(N), P10(N), P11(N), P12(N) ,    RCH 3360
     3   P13(N), P14(N), P15(N),P16(N),P17(N),P18(N),P19(N),P20(N),ZV(N)RCH 3370
      IF (NSVVAR .NE. 0) GO TO 404                                           RCH 3380
      IF (NSVPHI .NE. 0) GO TO 404                                           RCH 3390
      WRITE             (6, 532) PHIO(N)                                    RCH 3400
      GO TO 25                                                              RCH 3410
404   WRITE             (6, 532) PHI(N)                                     RCH 3420
 25   CONTINUE                                                              RCH 3430
      RETURN                                                                RCH 3440
C . . . . . .
 150  FORMAT (16HOELASTIC PHIDOT  ,2A6)                                     RCH 3450
 501  FORMAT (4E12.5,11X,2A1)                                               RCH 3460
 503  FORMAT (2I12,E12.5)                                                   RCH 3470
 504  FORMAT(A6,I6,A6,I6,F12.0,E12.5,2I6,E12.5)                             RCH 3480
 505  FORMAT(6E12.5)                                                        RCH 3490
 506  FORMAT(1H1,/,51X,14HC O R E  N O.  ,I2,/)                              RCH 3500
 507  FORMAT(47X,15HW I N D I N G S,/ )                                     RCH 3510
 508  FORMAT(7X,11HWINDING NO.,8X,6HNODE B,11X,6HNODE T,10X,5HTURNS,8X,    RCH 3520
     110HRESISTANCE,9X,15HINITIAL CURRENT,//)                               RCH 3530
 509  FORMAT(1HO,5X,I6,11X,      I6,11X,      I6,5X,  I12,2(8X,E12.5))      RCH 3540
 510  FORMAT(1HO,5X,I6,11X,      I6,10X,1HS,  I6,5X,  I12,2(8X,E12.5))      RCH 3550
 511  FORMAT(1HO,5X,I6,11X,      I6,14X,6HGROUND,2X,  I12,2(8X,E12.5))      RCH 3560
 512  FORMAT(1HO,5X,I6,10X,1HS,  I6,11X,      I6,5X,  I12,2(8X,E12.5))      RCH 3570
 513  FORMAT(1HO,5X,I6,14X,6HGROUND, 8X,      I6,5X,  I12,2(8X,E12.5))      RCH 3580
 514  FORMAT(1H1,25X,47HC O M P U T E D   C C R E   P A R A M E T E R S)RCH 3590
 515  FORMAT(1HO,4X,3HOD=,E12.5,8X,3HID=,E12.5,5X,5HTYPE-  ,A1,18X,         RCH 3600
     2     10HMATERIAL-  ,2A6)                                              RCH 3610
 516  FORMAT(1HO,2X,5HPHIS=,E12.5,6X,5HPHIR=,E12.5,6X,4HFD1=,E12.5,8X,      RCH 3620
     2       4HFDZ=,E12.5,9X,4HFD2=,E12.5)                                  RCH 3630
 517  FORMAT(1HO,3X,4HFD3=,E12.5,5X,6HPHIC2=,E12.5,7X,3HHA=,E12.5,9X,3HHRCH 3640
     1Q=,E12.5,10X,3HHN=,E12.5)                                             RCH 3650
 518  FORMAT(8HOLAMDAD=,E12.5,7X,4HNUD=,E12.5,6X,4HFDB=,E12.5,7X,5HFOPP=RCH 3660
     1,E12.5,7X,6HLAMDA=,E12.5)                                             RCH 3670
 519  FORMAT(1HO,4X,3HNU=,E12.5,8X,3HFB=,E12.5,7X,3HFO=,E12.5,8X,4HROP=,RCH 3680
     1E12.5)                                                                RCH 3690
 520  FORMAT(1HO,3X,4HFB1=,E12.5,7X,4HFO1=,E12.5,5X,5HROP1=,E12.5)          RCH 3700
 523  FORMAT(16X,39HTHE MAXIMUM NUMBER OF WINDINGS FOR CORE,I3,11HIS EXCRCH 3710
     1EEDED)                                                                RCH 3720
 524  FORMAT(16X,4HCORE,I3,16H HAS NO WINDINGS)                              RCH 3730
 525  FORMAT(//22X,57HC O M P U T E D   A U X I L I A R Y   P A R A M E RCH 3740
     1T E R S)                                                              RCH 3750
 526  FORMAT (/5(HOCONVERSION TO MKS UNITS AND PRELIMINARY COMPUTATION )RCH 3760
 527  FORMAT(/6X,3HP1=,E12.5,8X,3HP2=,E12.5,7X,3HP3=,E12.5,7X,4H P4=,       RCH 3770
     1  E12.5,7X,4H P5=,E12.5//6X,3HP6=,E12.5,8X,3HP7=,E12.5,7X,3HP8=,      RCH 3780
     2  E12.5,8X,3HP9=,E12.5,7X,4HP10=,E12.5//5X,4HP11=,E12.5,7X,4HP12=,RCH 3790
     3  E12.5,6X,4HP13=,E12.5,7X,4HP14=,E12.5,7X,4HP15=,E12.5,//,5X,       RCH 3800
     4    4HP16=,E12.5,7X,4HP17=,E12.5,6X,4HP18=,E12.5,7X,4HP19=,E12.5,    RCH 3810
     5    7X,4HP20=,E12.5,//,5X,4HZV = ,E12.5)                             RCH 3820
 528  FORMAT(//38H DO NOT EXIT WHEN FLUX SWITCHING STOPS)                    RCH 3830
 529  FORMAT(//49H EXIT WHEN FLUX SWITCHING STOPS IF T GREATER THAN,I4,      RCH 3840
     15H*DELT)                                                              RCH 3850
 531  FORMAT (3E12.5,F12.0,2I12)                                            RCH 3860
 532  FORMAT(///6X,16H INITIAL FLUX = ,E12.5)                               RCH 3870
 533  FORMAT (// 10H RELERR = ,E10.3, 8X,  9HABSERR = ,E10.3//)             RCH 3880
 770  FORMAT (25H Q1,PO,EL,Z1,Z2,Z3,Z4,Z5 ,3E24.15,/,1H ,5E24.15)           RCH 3890
 799  FORMAT(//,20X,48HS T A T I C   A N D   D Y N A M I C   I N P U T     RCH 3900
     2     ,34H   C O R E   P A R A M E T E R S     )                       RCH 3910
 803 FORMAT(/9X,22HSTATIC B(H)-CURVE DATA)                                  RCH 3920
 804 FORMAT (A1,13HHDZ(OERSTED)= ,E12.5,5X,13HHD2(OERSTED)= ,E12.5,5X,      RCH 3930
     2     13HHD3(OERSTED)= ,E12.5,5X,13HHM (OERSTED)= ,E12.5)              RCH 3940
```

```
  805 FORMAT(/9X,23HDYNAMIC B(H)-CURVE DATA,/10X,30HFREQUENCY   H(OERSTERCH 3950
     1DS) AT B=0/)                                                      RCH 3960
  806 FORMAT(5X,I3,2E14.5)                                              RCH 3970
  808 FORMAT(/6X,2HA=,E12.5,5X,3HLI=,E12.5,5X,3HLO=,E12.5,5X,3HBR=,E12.5RCH 3980
     2,5X,3HBS=,E12.5/4X,4HBD2=,E12.5,4X,4HBD3=,E12.5,5X,3HBM=,E12.5,5X,RCH 3990
     33HHM=,E12.5,5X,3HHS=,E12.5,2X,6HHDMIN=,E12.5/4X,4HHD1=,E12.5,4X,  RCH 4000
     44HHD2=,E12.5,4X,4HHD2=,E12.5,4X,4HHD3=,E12.5,4X,4HFD4=,E12.5,     RCH 4010
     5    /1X,7HPHID1S=,E12.5,2X,6HPHID3=,E12.5)                        RCH 4020
  820 FORMAT (A1,30X,13HBD2(GAUSSES)= ,E12.5,5X,13HBD3(GAUSSES)= ,E12.5,RCH 4030
     2    5X,13HBM (GAUSSES)= ,E12.5)                                   RCH 4040
  821 FORMAT(23X,F3.0,8H PERCENT)                                       RCH 4050
  900 FORMAT (A1,13HBR (GAUSSES)= ,E12.5,13X,                           RCH 4060
     2    6HTYPE-  ,A1,15X,14HCORE MATERIAL-  ,2A6,4X,13HHD1(OERSTED)= ,RCH 4070
     3    E12.5)                                                        RCH 4080
  901 FORMAT(/9X,10HDIMENSIONS)                                         RCH 4090
  902 FORMAT (A1,3X,10HOD (INCH)= ,E12.5,8X,10HID (INCH)= ,E12.5,7X,    RCH 4100
     2    11HA (CM.SQ.)= ,E12.5)                                        RCH 4110
 4445 FORMAT (4F12.0/5F12.0/3F12.0/12X,F12.0)                           RCH 4120
 4446 FORMAT (1HO,12X,4(F3.0,8H PERCENT ,19X))                          RCH 4130
 4449 FORMAT (1HO,42X,F3.0,8H PERCENT ,2(19X,F3.0,8H PERCENT ))         RCH 4140
 4466 FORMAT (1HO,12X,F3.0,8H PERCENT ,79X,F3.0,8H PERCENT )            RCH 4150
 7778 FORMAT (10HOPSTEP   = ,E12.5)                                     RCH 4160
 7905 FORMAT (14H REPRESENTING ,I2,32H IDENTICALLY SWITCHING CORES IN   RCH 4170
     2      ,9HPARALLEL  )                                             RCH 4180
 7906 FORMAT (14H REPRESENTING ,I2,32H IDENTICALLY SWITCHING CORES IN   RCH 4190
     2      ,7HSERIES  )                                               RCH 4200
 9789 FORMAT (16H1MAGNETIC CORES   ////)                                RCH 4210
      END                                                               RCH 4220
```

```
      SUBROUTINE RWCND                                                  RCN 0010
C     RWCND READS IN AND PRINTS OUT CONTINUED-RUN DATA

                    COMMON  VARIABLES

C . . . . . .
      WRITE(6,2303)                                                     RCN 0020
      READ (5,2304) NDELT,TF,DELT,DTM1,DTM2,NITRSM                      RCN 0030
      WRITE(6,2305) NDELT,TF,DELT,DTM1,DTM2,NITRSM                      RCN 0040
      WRITE(6,2349)                                                     RCN 0050
      READ (5,2300) (  TOD(I),I=1, 19)                                  RCN 0060
      WRITE(6,2350) (I,TOD(I),I=1, 19)                                  RCN 0070
      READ (5,2300)  (  EP1(I),I=1,ND1)                                 RCN 0080
      WRITE(6,2350)  (I,EP1(I),I=1,ND1)                                 RCN 0090
      WRITE(6,2306)                                                     RCN 0100
      READ (5,2300) (  V1(J),J=1,NZ)                                    RCN 0110
      WRITE(6,3434) (J,V1(J),J=1,NZ)                                    RCN 0120
      IF (NC.LE.0) GO TO 2308                                           RCN 0130
      WRITE(6,2307)                                                     RCN 0140
      DO 2309 I=1,NC                                                    RCN 0150
      READ (5,2300)    CIC(I)                                           RCN 0160
 2309 WRITE(6,2310)  I,CIC(I)                                           RCN 0170
 2308 CONTINUE                                                          RCN 0180
      IF (NL.LE.0) GO TO 2312                                           RCN 0190
      WRITE(6,2311)                                                     RCN 0200
      DO 2313 I=1,NL                                                    RCN 0210
      READ (5,2300)    CIL(I)                                           RCN 0220
 2313 WRITE(6,2310)  I,CIL(I)                                           RCN 0230
 2312 CONTINUE                                                          RCN 0240
      IF (NDIODE.LE.0) GO TO 2315                                       RCN 0250
      WRITE(6,2314)                                                     RCN 0260
      DO 2316 I=1,NDIODE                                                RCN 0270
      READ (5,2300)  DV(I),DR(I),DI(I)                                  RCN 0280
 2316 WRITE(6,2310)I,DV(I),DR(I),DI(I)                                  RCN 0290
 2315 CONTINUE                                                          RCN 0300
      IF (NTRANS.LE.0) GO TO 2318                                       RCN 0310
      WRITE(6,2317)                                                     RCN 0320
      DO 2319 I=1,NTRANS                                                RCN 0330
      READ (5,2300)   BCV(I),BCR(I),BCI(I),BEV(I),BER(I),BEI(I)         RCN 0340
 2319 WRITE(6,2310)I, BCV(I),BCR(I),BCI(I),BEV(I),BER(I),BEI(I)         RCN 0350
 2318 CONTINUE                                                          RCN 0360
      IF (NCORE.LE.0) RETURN                                            RCN 0370
      WRITE(6,2320)                                                     RCN 0380
      DO 2322 I=1,NCORE                                                 RCN 0390
      WRITE(6,2323)   I                                                 RCN 0400
      READ (5,2300)   FM3(I),FM2(I),FM1(I),PHIM3(I),PHIM2(I),PHIM1(I),  RCN 0410
     2     PDTEM1(I),PHDTM2(I),PHDTM1(I)                                RCN 0420
      WRITE(6,2324)   FM3(I),FM2(I),FM1(I),PHIM3(I),PHIM2(I),PHIM1(I),  RCN 0430
     2     PDTEM1(I),PHDTM2(I),PHDTM1(I)                                RCN 0440
      K = NW(I)                                                         RCN 0450
      WRITE(6,2325)                                                     RCN 0460
      DO 2326 J=1,K                                                     RCN 0470
      READ (5,2300)    CICOR1(I,J)                                      RCN 0480
 2326 WRITE(6,2310)  J,CICOR1(I,J)                                      RCN 0490
 2322 CONTINUE                                                          RCN 0500
      RETURN                                                            RCN 0510
C . . . . . .
```

```
2300 FORMAT (6E12.5)                                                    RCN 0520
2303 FORMAT (1H1,30X,71HC O N T I N U E D   R U N - - - L A S T  U P D RCN 0530
    2A T E D  R E S U L T S  ,/ ,                                       RCN 0540
    3   59H0   NDELT        TE          CELT          DTM1          DTM2 RCN 0550
    4    ,7X,6HNITRSM     )                                             RCN 0560
2304 FORMAT     (I12,4E12.5,I12)                                        RCN 0570
2305 FORMAT (1H ,I7,5X,4E12.5,I8)                                       RCN 0580
2306 FORMAT (1H0,30X,34HNODAL AND AUXILIARY V1() VOLTAGES   )           RCN 0590
2307 FORMAT ( 20 HOCAPACITOR CURRENTS   )                               RCN 0600
2310 FORMAT ( 1H ,I5,3X,6E12.5)                                         RCN 0610
2311 FORMAT ( 20HOINDUCTOR  CURRENTS    )                               RCN 0620
2314 FORMAT ( 18HODIODE VARIABLES     ,/,       5X,                     RCN 0630
    2    35H NO.     DV          DR          DI     )                  RCN 0640
2317 FORMAT ( 22HOTRANSISTOR VARIABLES     ,/,  5X,                     RCN 0650
    2    72H NO.    BCV         BCR         BCI         BEV          BERRCN 0660
    3         BEI             )                                         RCN 0670
2320 FORMAT (18HOCORE VARIABLES         )                               RCN 0680
2323 FORMAT (13H CORE NUMBER   ,I3,/,120H      FM3          FM2        RCN 0690
    2  FM1          PHIM3        PHIM2        PHIM1        PHDTEM1     RCN 0700
    3PHDTM2       PHDTM1         )                                      RCN 0710
2324 FORMAT (1H 9E13.5)                                                 RCN 0720
2325 FORMAT(30H WINDING    CICOR1              )                        RCN 0730
2349 FORMAT (23HOVARIABLE-SOURCE DATA          )                        RCN 0740
2350  FORMAT (6(I4,E13.5))                                              RCN 0750
3434 FORMAT (9HOUNKNOWNS (  /6(I4,E13.5)))                              RCN 0760
     END                                                                RCN 0770
```

```
      SUBROUTINE SEQSOL (ERROR)                                          SEQ 0010
C     SEQSOL SOLVES MATRIX EQUATIONS BY GAUSSIAN ELIMINATION
C                 AND BACK SUBSTITUTION
```

COMMON VARIABLES

```
      EQUIVALENCE (H,G)                                                  SEQ 0020
      DIMENSION G(3600)                                                  SEQ 0030
      DATA ID/60/                                                        SEQ 0040
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
      ERROR = 0.0                                                        SEQ 0050
      NM1 = NV- 1                                                        SEQ 0060
      NX = 0                                                             SEQ 0070
      DO 19 I4 = 1,NM1                                                   SEQ 0080
         I3 = NX + I4                                                    SEQ 0090
         T1 = G(I3)                                                      SEQ 0100
C        - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C        CHECK IF DIAGONAL ELEMENT H(I4,I4) IS ZERO
C        - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
         IF (T1 .NE.0.0)GO TO 7                                          SEQ 0110
         ERROR = -1.0                                                    SEQ 0120
         RETURN                                                          SEQ 0130
    7    I4P1 = I4 + 1                                                   SEQ 0140
         DO 18 I1 = I4P1,NV                                              SEQ 0150
            I3 = NX + I1                                                 SEQ 0160
            T2 = G(I3)                                                   SEQ 0170
C           - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C           COMPUTE THE RATIO H(I1,I4)/H(I4,I4)
C           - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
            IF (T2 .EQ.0.0)GO TO 18                                      SEQ 0180
            Q = -T2 / T1                                                 SEQ 0190
            I3 = NX + ID + I4                                            SEQ 0200
            I5 = I3 + I1 - I4                                            SEQ 0210
            DO 16 I2 = I4P1,NV                                           SEQ 0220
C           - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C           ASSUME THAT THE FIRST I4 ELEMENTS (LEFT OF THE DIAGONAL) OF
C              ROW I1 ARE ZERO
C              - - - - - - - - - - - - - - - - - - - - - - - - - - - -
               T3 = G(I3)                                                SEQ 0230
               IF (T3 .NE.0.0) G(I5) = G(I5) + Q * T3                    SEQ 0240
C              - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C              SUBTRACT (H(I1,I4)/H(I4,I4)) * ROW I4 FROM ROW I1
C              - - - - - - - - - - - - - - - - - - - - - - - - - - - -
               I3 = I3 + ID                                              SEQ 0250
               I5 = I5 + ID                                              SEQ 0260
   16       CONTINUE                                                     SEQ 0270
            T(I1) = T(I1) + Q * T(I4)                                    SEQ 0280
C           - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C           REPLACE T(I1) BY T(I1) - T(I4)*H(I1,I4)/H(I4,I4)
C        - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   18    CONTINUE                                                        SEQ 0290
         NX = NX + ID                                                    SEQ 0300
   19 CONTINUE                                                           SEQ 0310
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     BACK SUBSTITUTE IN TRIANGULARIZED MATRIX
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DO 33 I2 = 1,NV                                                    SEQ 0320
```

```
      I4 = NV+ 1 - I2                                                   SEQ 0330
      I3 = NX + I4                                                      SEQ 0340
      T1 = G(I3)                                                        SEQ 0350
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK IF DIAGONAL ELEMENT H(I4,I4) OF TRIANGULARIZED MATRIX IS
C     ZERO
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (T1 .NE.0.0)GO TO 26                                           SEQ 0360
      ERROR = 1.0                                                       SEQ 0370
      RETURN                                                            SEQ 0380
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     SOLVE FOR V ARRAY
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   26 Q = -T(I4) / T1                                                   SEQ 0390
      IF (Q .EQ.0.0)GO TO 32                                            SEQ 0400
      IF (I4 .LE. 1) GO TO 32                                           SEQ 0410
      I4M1=I4 - 1                                                       SEQ 0420
      DO 31 I1 = 1,I4M1                                                 SEQ 0430
C  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C        SUBSTITUTE VALUE V(I4) INTO ROWS 1 TO (I4-1)
C  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
         I5 = NX + I1                                                   SEQ 0440
         T2 = G(I5)                                                     SEQ 0450
         IF (T2 .NE.0.0) T(I1) = T(I1) + Q * T2                         SEQ 0460
   31 CONTINUE                                                          SEQ 0470
   32 V(I4) = -Q                                                        SEQ 0480
      NX = NX - ID                                                      SEQ 0490
   33 CONTINUE                                                          SEQ 0500
      RETURN                                                            SEQ 0510
C  . . . . . .
      END                                                               SEQ 0520
```

```
      SUBROUTINE INTIAL                                                  INT 0010
C     INTIAL SOLVES INITIAL CONDITIONS
```

COMMON VARIABLES

```
      LOGICAL MONMA,WSQSOL                                               INT 0020
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
      IF (NSVVAR .EQ. 0) GO TO 400                                       INT 0030
      WRITE (6,9237)                                                     INT 0040
         CALL EXIT                                                       INT 0050
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
C     START OF ITERATION LOOP
C     FILL H AND T ARRAYS
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
  400 CALL ELEM                                                          INT 0060
      IF (.NOT. WSQSOL) GO TO 8872                                       INT 0070
      WRITE (6,9999)                                                     INT 0080
      DO 9995 J = 1,NV                                                   INT 0090
 9995 WRITE (6,9997) (H(J,K),K=1,NV)                                     INT 0100
      WRITE (6,9996) (T(J),J=1,NV)                                       INT 0110
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
C     SOLVE MATRIX EQUATION
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
 8872 CALL SEQSOL (ERR)                                                  INT 0120
      IF(ERR .EQ.0.0)GO TO 2001                                          INT 0130
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
C        PRINT OUT THE CONTENTS OF A SINGULAR MATRIX AND EXIT
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
         IF (.NOT. WSQSOL) GO TO 9980                                    INT 0140
         DO 9981 J = 1,NV                                                INT 0150
 9981    WRITE (6,9997) (H(J,K),K=1,NV)                                  INT 0160
 9980    CALL ELEM                                                       INT 0170
         WRITE (6,2002) ((JACK,JILL,H(JACK,JILL),JILL=1,NV),JACK=1,NV)   INT 0180
         CALL EXIT                                                       INT 0190
 2001 IF (.NOT. WSQSOL) GO TO 9982                                       INT 0200
         CALL AUX(1)                                                     INT 0210
         WRITE (6,9970) (V(J),J=1,NZ)                                    INT 0220
 9982 TOL = -TOL                                                         INT 0230
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
C     TEST CONVERGENCE OF VOLTAGES
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
      DO 173 J = 1,NV                                                    INT 0240
      IF(ABS(V1(J)-V(J)) .LE. (RELER*ABS(V1(J))+ABSER)) GO TO 173        INT 0250
      JKKJ = 1                                                           INT 0260
      GO TO 174                                                          INT 0270
  173 CONTINUE                                                           INT 0280
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
C     COMPUTE CURRENTS AND TEST FOR CONVERGENCE
C     -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -
  174 CALL ELEM                                                          INT 0290
      DO 175 J=1,NV                                                      INT 0300
  175 V1(J)=V(J)                                                         INT 0310
      IF (MONMA) WRITE (6,9994) JKKJ                                     INT 0320
      IF(JKKJ .GT. 0) GO TO 6011                                         INT 0330
      IF(ABC .LE.0.0) RETURN                                             INT 0340
 6011 JKKJ=0                                                             INT 0350
      JKLJ=JKLJ+1                                                        INT 0360
```

```
      IF(JKLJ-200)400,410,411                                          INT 0370
  410 DO 412 J=1,NZ                                                    INT 0380
  412 V2(J)=V1(J)                                                      INT 0390
      GO TO 400                                                        INT 0400
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     END OF ITERATION LOOP
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  411 DO 413 J=1,NZ                                                    INT 0410
      IF (V1(J) .NE.0.0)GO TO 7413                                     INT 0420
      V2(J) = V2(J) * 1.0E10                                           INT 0430
      GO TO 413                                                        INT 0440
 7413 V2(J)=((V1(J)-V2(J))/V1(J))*100.0                                INT 0450
  413 CONTINUE                                                         INT 0460
      WRITE (6,414)                                                    INT 0470
      WRITE (6,415)(J,V2(J),J=1,NZ)                                    INT 0480
      JKKJ=1                                                           INT 0490
      RETURN                                                           INT 0500
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
  414 FORMAT (    42H1PERCENT ERROR FOR NODE VALUE CALCULATION )       INT 0510
  415 FORMAT(1H 6(I4,F10.5))                                           INT 0520
 2002 FORMAT(26H1SINGULAR MATRIX EQUATION ,/,1H0,/,      20H   H MATRIX INT 0530
     1ELEMENTS/(I4,I4,E13.5))                                          INT 0540
 9237 FORMAT (49H INCONSISTENT SPECIFICATION OF SAVE-VARIABLE AND      INT 0550
     2     32HINITIAL-CONDITION SOLUTION (NG)    )                    INT 0560
 9970 FORMAT (16H SEQSOL RESULTS ,/( 1H ,5E24.15))                     INT 0570
 9994 FORMAT (6H JKKJ   ,I4)                                           INT 0580
 9996 FORMAT (2H T ,(   1H ,10E12.5))                                  INT 0590
 9997 FORMAT (2H H ,(   1H ,10E12.5))                                  INT 0600
 9999 FORMAT (8H SEQSOL   )                                            INT 0610
      END                                                              INT 0620
```

```
      SUBROUTINE ELEM                                                   ELE 0010
C     ELEM CALLS PARTS (1) AND (2) OF EACH ELEMENT SUBROUTINE
C          AND CONTROLS CONVERGENCE MEANS
```

COMMON VARIABLES

```
      LOGICAL MONELM                                                    ELE 0020
C     . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
      IF (MONELM) WRITE (6,9999)                                        ELE 0030
      IF(ABC .LT. 0.0) ABC = 0.0                                        ELE 0040
      IF(TO1 .LE.0.0)GO TO 2                                            ELE 0050
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PART (1) - FILL H AND T ARRAYS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DO 1 J=1,NV                                                       ELE 0060
      T(J)=0.0                                                          ELE 0070
      DO 1 K=1,NV                                                       ELE 0080
    1 H(J,K)=0.0                                                        ELE 0090
      IBC=0                                                             ELE 0100
      CALL  AUX(0)                                                      ELE 0110
      CALL IVS                                                          ELE 0120
      CALL RES                                                          ELE 0130
      CALL CAP                                                          ELE 0140
      CALL IND                                                          ELE 0150
      CALL ZEN                                                          ELE 0160
      CALL DIOD                                                         ELE 0170
      CALL TRAN                                                         ELE 0180
      CALL CORE                                                         ELE 0190
      IF (MONELM) WRITE (6,7900) ABC,DELT,IBC                           ELE 0200
      RETURN                                                            ELE 0210
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PART (2) - COMPUTE CURRENTS AND TEST FOR CONVERGENCE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
    2 TO1=-TO1                                                          ELE 0220
      ABC=-1.0                                                          ELE 0230
      IBC=1                                                             ELE 0240
      CALL DIOD                                                         ELE 0250
      CALL TRAN                                                         ELE 0260
      CALL CORE                                                         ELE 0270
      IF (ABC .LE. 0.0) GO TO 5                                         ELE 0280
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CONVERGENCE NOT ACHIEVED
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF(TE .LE. TIMEIN) GO TO 100                                      ELE 0290
      IF (NITER .LT. ITRMAX) GO TO 100                                  ELE 0300
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     FAILED TO CONVERGE WITHIN ITRMAX ITERATIONS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      KONVRG = 0                                                        ELE 0310
      KFAIL = KFAIL + 1                                                 ELE 0320
      IF (KFAIL .LT. KFMAX) GO TO 200                                   ELE 0330
      KFAIL = 0                                                         ELE 0340
      ITRMAX = ITRMAX + ITRMAX                                          ELE 0350
      GO TO 100                                                         ELE 0360
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     RESET VARIABLES AND DIVIDE DELT BY 4
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
  200 TE = TE - DELT                                                    ELE 0370
      DELT = 0.25*DELT                                                  ELE 0380
      TE=TE+DELT                                                        ELE 0390
      WRITE (6,775) DELT,NITER                                          ELE 0400
      DO 104 J=1,NV                                                     ELE 0410
  104 V(J)=V1(J)                                                        ELE 0420
      ABC = -3.0                                                        ELE 0430
  100 IF (MONELM) WRITE (6,9180) TE,ABC,DELT,ITRMAX,KFAIL,KONVRG        ELE 0440
      RETURN                                                            ELE 0450
    5 IF (NG  .NE. -1) GO TO 6                                          ELE 0460
      IF (JKKJ.LE.  0) GO TO 6                                          ELE 0470
      IF (JKLJ.LE.200) RETURN                                           ELE 0480
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CONVERGENCE ACHIEVED. COMPUTE CURRENTS OF LINEAR ELEMENTS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
    6 CALL IVS                                                          ELE 0490
      CALL RES                                                          ELE 0500
      CALL CAP                                                          ELE 0510
      CALL IND                                                          ELE 0520
      CALL ZEN                                                          ELE 0530
      CALL  AUX(1)                                                      ELE 0540
      IF (TE .LE. TIMEIN) GO TO 180                                     ELE 0550
      DTM2 = DTM1                                                       ELE 0560
      DTM1 = DELT                                                       ELE 0570
      NDELT = NDELT + 1                                                 ELE 0580
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     ADJUST DELT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (NITER-ITRTST) 201,180,140                                     ELE 0590
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  201 IF (TI .GT. (DELT+DELT)) GO TO  9236                              ELE 0600
      DELT = TI                                                         ELE 0610
      GO TO 1016                                                        ELE 0620
 9236 DELT = DELT+DELT                                                  ELE 0630
 1016 KFAIL = 0                                                         ELE 0640
      KONVRG = KONVRG + 1                                               ELE 0650
      IF (KONVRG .LT. KFMAX) GO TO 180                                  ELE 0660
      KONVRG = 0                                                        ELE 0670
      IF (ITRMAX .GT. MAXMIN) ITRMAX = ITRMAX / 2                       ELE 0680
      GO TO 180                                                         ELE 0690
  140 DELT = 0.5 * DELT                                                 ELE 0700
  180 IF (MONELM) WRITE (6,9180) TE, ABC, DELT, ITRMAX,KFAIL,KONVRG     ELE 0710
      RETURN                                                            ELE 0720
C . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
  775 FORMAT (8HODELT = ,E12.5,18X,I4,13H ITERATION(S)  )               ELE 0730
 7900 FORMAT (14H ABC,DELT,IBC  ,2E24.15,I12)                           ELE 0740
 9180 FORMAT (34H TE,ABC,DELT,ITRMAX,KFAIL,KONVRG     ,                 ELE 0750
     2     /3E24.15,3I12)                                               ELE 0760
 9999 FORMAT (8H ELMTS   )                                              ELE 0770
      END                                                               ELE 0780
```

```
      SUBROUTINE IVS                                                    IVS 0010
C     IVS COMPUTES CONTRIBUTIONS TO H AND T ARRAYS AND CURRENTS
C          OF CURRENT AND VOLTAGE SOURCES
```

COMMON VARIABLES

```
      LOGICAL MONIV                                                     IVS 0020
      EQUIVALENCE (KIVS,K),(K1IVS,K1),(K2IVS,K2),(K3IVS,K3),(K4IVS,K4)  IVS 0030
      DIMENSION K(20), K1(20), K2(20), K3(20), K4(20)                   IVS 0040
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
      IF (NIV .LE. 0) RETURN                                            IVS 0050
      IF (MONIV) WRITE (6,9999)                                         IVS 0060
      DO 117 N = 1,NIV                                                  IVS 0070
      KO1=K1(N)                                                         IVS 0080
      KO2=K2(N)                                                         IVS 0090
      KO=K (N)                                                          IVS 0100
      IF(K4(N).LE. 0) GO TO 18                                          IVS 0110
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CURRENT SOURCE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF(K3(N) .GT. 0) GO TO 106                                        IVS 0120
      B=VI(N)                                                           IVS 0130
      GO TO 107                                                         IVS 0140
  106 K5=K3(N)                                                          IVS 0150
      B=EI(K5)                                                          IVS 0160
  107 CIIVS(N) = B                                                      IVS 0170
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE CONTRIBUTION TO T ARRAY OF A CURRENT SOURCE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      GO TO (24,21,23),KO                                               IVS 0180
   24 T(KO1)=T(KO1)-B                                                   IVS 0190
   21 T(KO2)=T(KO2)+B                                                   IVS 0200
      GO TO 17                                                          IVS 0210
   23 T(KO1)=T(KO1)-B                                                   IVS 0220
      GO TO 17                                                          IVS 0230
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     VOLTAGE SOURCE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   18 B=1.0/RVS(N)                                                      IVS 0240
      IF(K3(N) .LE. 0) GO TO 108                                        IVS 0250
      K5=K3(N)                                                          IVS 0260
      C=EI(K5)*B                                                        IVS 0270
      GO TO 110                                                         IVS 0280
  108 C=VI(N) *B                                                        IVS 0290
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE CONTRIBUTION TO H AND T ARRAYS OF A VOLTAGE SOURCE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  110 IF (IBC .GT. 0) GO TO 3333                                        IVS 0300
      GO TO (28,26,27),KO                                               IVS 0310
   28 H(KO1,KO1)=H(KO1,KO1)+B                                           IVS 0320
      H(KO1,KO2)=H(KO1,KO2)-B                                           IVS 0330
      T(KO1)=T(KO1)+C                                                   IVS 0340
      H(KO2,KO1)=H(KO2,KO1)-B                                           IVS 0350
      H(KO2,KO2)=H(KO2,KO2)+B                                           IVS 0360
      T(KO2)=T(KO2)-C                                                   IVS 0370
      GO TO 17                                                          IVS 0380
   26 H(KO2,KO2)=H(KO2,KO2)+B                                           IVS 0390
```

```
      T(KO2)=T(KO2)-C+B*EI(KO1)                                          IVS 0400
      GO TO 17                                                           IVS 0410
   27 H(KO1,KO1)=H(KO1,KO1)+B                                            IVS 0420
      T(KO1)=T(KO1)+C+B*EI(KO2)                                          IVS 0430
      GO TO 17                                                           IVS 0440
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE VOLTAGE-SOURCE CURRENTS (AFTER CONVERGENCE HAS BEEN
C     ACHIEVED)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 3333 GO TO (300,301,302),KO                                             IVS 0450
  300 CIIVS(N) = (V (KO1) - V (KO2)) * B - C                             IVS 0460
      GO TO 17                                                           IVS 0470
  301 CIIVS(N) = (V (KO1) - EI(KO2)) * B - C                             IVS 0480
      GO TO 17                                                           IVS 0490
  302 CIIVS(N) = (EI(KO1) - V (KO2)) * B - C                             IVS 0500
   17 IF (MONIV) WRITE (6,9117) K5,KO,B,C,T(KO1),T(KO2),H(KO1,KO1),      IVS 0510
     2    H(KO2,KO2),H(KO1,KO2)                                         IVS 0520
  117 CONTINUE                                                           IVS 0530
      RETURN                                                             IVS 0540
C     . . . . . .
 9117 FORMAT (46H K3(),K4(),B,C,T(1),T(2),H(1,1),H(2,2),H(1,2) ,2I6,/,   IVS 0550
     1    5(  /,  1H ,5E24.15 )   )                                     IVS 0560
 9999 FORMAT (13H IV SOURCES   )                                         IVS 0570
      END                                                                IVS 0580
```

```
      SUBROUTINE RES                                                    RES 0010
C     RES  COMPUTES CONTRIBUTIONS TO H AND T ARRAYS AND CURRENTS
C               OF RESISTORS

                    COMMON VARIABLES

      LOGICAL MONR                                                      RES 0020
      EQUIVALENCE (KRES,K),(K1RES,K1),(K2RES,K2)                        RES 0030
      DIMENSION K(40), K1(40), K2(40)                                   RES 0040
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
      IF (NR .LE. 0) RETURN                                             RES 0050
      IF (MONR)WRITE (6,9999)                                           RES 0060
      IF (IBC .GT. 0) GO TO 3333                                        RES 0070
      L = 0                                                             RES 0080
      DO 114 N = 1,NR                                                   RES 0090
      IF (NRP(N) .LE. 1) GO TO 3005                                     RES 0100
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     R VARIES WITH TIME
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      L = L + 1                                                         RES 0110
      J1= IX(L)                                                         RES 0120
      IF (TE .LE. TFR(L,J1)) GO TO 3007                                 RES 0130
      J1= J1+ 1                                                         RES 0140
      IX(L) = J1                                                        RES 0150
      R(N) = RF(L,J1-1)                                                 RES 0160
 3007 IF (XRES(L,J1) .NE. 0.0)                                          RES 0170
     2R(N) =(RF(L,J1  )-RF(L,J1-1))*((TE-TFR(L,J1-1))                   RES 0180
     3     /(TFR(L,J1)-TFR(L,J1-1))) ** XRES(L,J1) + RF(L,J1-1)         RES 0190
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE CONTRIBUTION TO H AND T ARRAYS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 3005 B=1.0/R(N)                                                        RES 0200
      KO=K(N)                                                           RES 0210
      KO1=K1(N)                                                         RES 0220
      KO2=K2(N)                                                         RES 0230
      GO TO (100,101,102),KO                                            RES 0240
  100 H(KO1,KO1)=H(KO1,KO1)+B                                           RES 0250
      H(KO1,KO2)=H(KO1,KO2)-B                                           RES 0260
      H(KO2,KO1)=H(KO2,KO1)-B                                           RES 0270
      H(KO2,KO2)=H(KO2,KO2)+B                                           RES 0280
      GO TO 14                                                          RES 0290
  101 H(KO1,KO1)=H(KO1,KO1)+B                                           RES 0300
      T(KO1)=T(KO1)+B*EI(KO2)                                           RES 0310
      GO TO 14                                                          RES 0320
  102 H(KO2,KO2)=H(KO2,KO2)+B                                           RES 0330
      T(KO2)=T(KO2)+B*EI(KO1)                                           RES 0340
   14 IF (MONR) WRITE (6,9114)   B,T(KO1),T(KO2),H(KO1,KO1),H(KO2,KO2), RES 0350
     2    H(KO1,KO2)                                                   RES 0360
  114 CONTINUE                                                          RES 0370
      RETURN                                                            RES 0380
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE RESISTOR CURRENTS (AFTER CONVERGENCE HAS BEEN ACHIEVED)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 3333 DO 303 N = 1,NR                                                   RES 0390
      KO=K(N)                                                           RES 0400
      KO1=K1(N)                                                         RES 0410
      KO2=K2(N)                                                         RES 0420
      GO TO (300,301,302),KO                                            RES 0430
  300 CIR(N) = (V (KO1) - V (KO2)) / R(N)                               RES 0440
      GO TO 303                                                         RES 0450
  301 CIR(N) = (V (KO1) - EI(KO2)) / R(N)                               RES 0460
      GO TO 303                                                         RES 0470
  302 CIR(N) = (EI(KO1) - V (KO2)) / R(N)                               RES 0480
  303 CONTINUE                                                          RES 0490
      RETURN                                                            RES 0500
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
 9114 FORMAT (36H   B,T(1),T(2),H(1,1),H(2,2),H(1,2) ,                  RES 0510
     1     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )                RES 0520
 9999 FORMAT (13H RESISTORS    )                                        RES 0530
      END                                                               RES 0540
```

```
      SUBROUTINE CAP                                                    CAP 0010
C     CAP  COMPUTES CONTRIBUTIONS TO H AND T ARRAYS AND CURRENTS
C               OF CAPACITORS

                    COMMON VARIABLES

      LOGICAL MONC                                                      CAP 0020
      EQUIVALENCE (KCAP,K),(K1CAP,K1),(K2CAP,K2)                        CAP 0030
      DIMENSION K(20), K1(20), K2(20)                                   CAP 0040
      DIMENSION B1(20), B2(20), HAA(20), TA(20)                         CAP 0050
C     . . . . . .
      IF (NC .LE. 0) RETURN                                             CAP 0060
      IF (MONC)WRITE (6,9999)                                           CAP 0070
      DO 114 N = 1,NC                                                   CAP 0080
      KO=K(N)                                                           CAP 0090
      KO1=K1(N)                                                         CAP 0100
      KO2=K2(N)                                                         CAP 0110
      IF (IBC .GT. 0) GO TO 105                                         CAP 0120
      IF(ABC .GT.0.0)GO TO 53                                           CAP 0130
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     AT THE FIRST ITERATION, COMPUTE VALUES THAT ARE CONSTANT FOR A
C     GIVEN DELT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      A1=DELT/RCSH(N)                                                   CAP 0140
      A2=2.0*CAPC(N)+A1                                                 CAP 0150
      A3=2.0*CAPC(N)-A1                                                 CAP 0160
      A4=1.0/(DELT+RC(N)*A2)                                            CAP 0170
      HAA(N)=A2*A4                                                      CAP 0180
      B1(N)=A3*A4                                                       CAP 0190
      B2(N)=CIC(N)*(DELT/A3-RC(N))                                      CAP 0200
      GO TO (61,62,63),KO                                               CAP 0210
   61 TA(N)=B1(N)*(V1(KO1)-V1(KO2)+B2(N))                               CAP 0220
      GO TO 53                                                          CAP 0230
   62 TA(N)=B1(N)*(V1(KO1)-EP1(KO2)+B2(N))+HAA(N)*EI(KO2)               CAP 0240
      GO TO 53                                                          CAP 0250
   63 TA(N)=-B1(N)*(EP1(KO1)-V1(KO2)+B2(N))+HAA(N)*EI(KO1)              CAP 0260
   53 HAAJ = HAA(N)                                                     CAP 0270
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     ADD THE CONTRIBUTIONS TO THE H AND T ARRAYS (AT EVERY ITERATION)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      GO TO (100,101,102),KO                                            CAP 0280
  100 H(KO1,KO1)=H(KO1,KO1)+HAAJ                                        CAP 0290
      H(KO1,KO2)=H(KO1,KO2)-HAAJ                                        CAP 0300
      T(KO1)=T(KO1)+TA(N)                                               CAP 0310
      H(KO2,KO1)=H(KO2,KO1)-HAAJ                                        CAP 0320
      H(KO2,KO2)=H(KO2,KO2)+HAAJ                                        CAP 0330
      T(KO2)=T(KO2)-TA(N)                                               CAP 0340
      GO TO 14                                                          CAP 0350
  101 H(KO1,KO1)=H(KO1,KO1)+HAAJ                                        CAP 0360
      T(KO1)=T(KO1)+TA(N)                                               CAP 0370
      GO TO 14                                                          CAP 0380
  102 H(KO2,KO2)=H(KO2,KO2)+HAAJ                                        CAP 0390
      T(KO2)=T(KO2)+TA(N)                                               CAP 0400
      GO TO 14                                                          CAP 0410
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE CAPACITOR CURRENTS (AFTER CONVERGENCE HAS BEEN ACHIEVED)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  105 GO TO  (103,106,108), KO                                          CAP 0420
  103 CIC(N)=HAA(N)*(V(KO1)-V(KO2))-B1(N)*(V1(KO1)-V1(KO2)+B2(N))       CAP 0430
      GO TO 14                                                          CAP 0440
  106 CIC(N)=HAA(N)*(V(KO1)-EI(KO2))-B1(N)*(V1(KO1)-EP1(KO2)+B2(N))     CAP 0450
      GO TO 14                                                          CAP 0460
  108 CIC(N)=HAA(N)*(EI(KO1)-V(KO2))-B1(N)*(EP1(KO1)-V1(KO2)+B2(N))     CAP 0470
   14 IF (MONC) WRITE (6,3899) KO1,KO2,A1,A2,A3,A4,HAA(N),B1(N),B2(N),  CAP 0480
     2   TA(N),CIC(N),T(KO1),T(KO2),H(KO1,KO1),H(KO2,KO2),H(KO1,KO2)    CAP 0490
  114 CONTINUE                                                          CAP 0500
      RETURN                                                            CAP 0510
C     . . . . . .
 3899 FORMAT (83H KO1,KO2,A1,A2,A3,A4,HAA( ),B1( ),B2( ),TA( ),CIC(),T(1CAP 0520
     2),T(2),H(1,1),H(2,2),H(1,2)    ,213,/,5E24.15,/,5E24.15,/,5E24.15) CAP 0530
 9999 FORMAT (13H CAPACITORS    )                                       CAP 0540
      END                                                               CAP 0550
```

```
      SUBROUTINE IND                                                    IND 0010
C     IND  COMPUTES CONTRIBUTIONS TO H AND T ARRAYS AND CURRENTS
C               OF INDUCTORS
```

COMMON VARIABLES

```
      REAL INDL                                                         IND 0020
      LOGICAL MONL                                                      IND 0030
      EQUIVALENCE (KIND,K),(K1IND,K1),(K2IND,K2)                        IND 0040
      DIMENSION K(20), K1(20), K2(20)                                   IND 0050
      DIMENSION B1(20), B2(20), HAA(20), TA(20)                         IND 0060
C  .  . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
      IF (NL .LE. 0) RETURN                                             IND 0070
      IF (MONL)WRITE (6,9999)                                           IND 0080
      IF (NG .LT. 0) GO TO 202                                          IND 0090
      DO 114 N=1,NL                                                     IND 0100
      KO=K(N)                                                           IND 0110
      KO1=K1(N)                                                         IND 0120
      KO2=K2(N)                                                         IND 0130
      IF (IBC .GT. 0) GO TO 105                                         IND 0140
      IF(ABC .GT.0.0)GO TO 53                                           IND 0150
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     AT THE FIRST ITERATION, COMPUTE VALUES THAT ARE CONSTANT FOR A
C     GIVEN DELT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (VICON(N) .GT.0.0)  S(N)=INDL(N)*EXP(-ABS(                     IND 0160
     2   (CIL(N)*(RLSH(N)+RL(N)) - (V(KO1)-V(KO2)))/RLSH(N)) * VICON(N))IND 0170
      C =2.0*S(N)/RLSH(N)                                               IND 0180
      B=1.0/(2.0*S(N)+RL(N)*(DELT+C))                                   IND 0190
      HAA(N)=B*(DELT+C)                                                 IND 0200
      B1(N)=B*(C-DELT)                                                  IND 0210
      B2(N)=CIL(N)*(2.0*S(N)/(DELT-C)-RL(N))                            IND 0220
      GO TO (61,62,63),KO                                               IND 0230
   61 TA(N)=B1(N)*(V1(KO1)-V1(KO2)+B2(N))                               IND 0240
      GO TO 53                                                          IND 0250
   62 TA(N)=B1(N)*(V1(KO1)-EP1(KO2)+B2(N))+HAA(N)*EI(KO2)               IND 0260
      GO TO 53                                                          IND 0270
   63 TA(N)=-B1(N)*(EP1(KO1)-V1(KO2)+B2(N))+HAA(N)*EI(KO1)              IND 0280
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     ADD THE CONTRIBUTIONS TO THE H AND T ARRAYS (AT EVERY ITERATION)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   53 HAAJ = HAA(N)                                                     IND 0290
      GO TO (100,101,102),KO                                            IND 0300
  100 H(KO1,KO1)=H(KO1,KO1)+HAAJ                                        IND 0310
      H(KO1,KO2)=H(KO1,KO2)-HAAJ                                        IND 0320
      T(KO1)=T(KO1)+TA(N)                                               IND 0330
      H(KO2,KO1)=H(KO2,KO1)-HAAJ                                        IND 0340
      H(KO2,KO2)=H(KO2,KO2)+HAAJ                                        IND 0350
      T(KO2)=T(KO2)-TA(N)                                               IND 0360
      GO TO 14                                                          IND 0370
  101 H(KO1,KO1)=H(KO1,KO1)+HAAJ                                        IND 0380
      T(KO1)=T(KO1)+TA(N)                                               IND 0390
      GO TO 14                                                          IND 0400
  102 H(KO2,KO2)=H(KO2,KO2)+HAAJ                                        IND 0410
      T(KO2)=T(KO2)+TA(N)                                               IND 0420
      GO TO 14                                                          IND 0430
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE INDUCTOR CURRENTS (AFTER CONVERGENCE HAS BEEN ACHIEVED)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  105 GO TO  (103,106,108), KO                                          IND 0440
  103 CIL(N)=HAA(N)*(V(KO1)-V(KO2))-B1(N)*(V1(KO1)-V1(KO2)+B2(N))       IND 0450
      GO TO 14                                                          IND 0460
  106 CIL(N)=HAA(N)*(V(KO1)-EI(KO2))-B1(N)*(V1(KO1)-EP1(KO2)+B2(N))     IND 0470
      GO TO 14                                                          IND 0480
```

```
  108 CIL(N)=HAA(N)*(EI(KO1)-V(KO2))-B1(N)*(EP1(KO1)-V1(KO2)+B2(N))      IND 0490
   14 IF (MONL) WRITE (6,3914) CIL(N),S(N),C,B,HAA(N),B1(N),B2(N),TA(N),IND 0500
     1T(KO1),T(KO2),H(KO1,KO1),H(KO2,KO2),H(KO1,KO2)                    IND 0510
  114 CONTINUE                                                          IND 0520
      RETURN                                                            IND 0530
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTATION OF INITIAL CONDITIONS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  202 DO 203 N=1,NL                                                     IND 0540
      KO=K(N)                                                           IND 0550
      KO1=K1(N)                                                         IND 0560
      KO2=K2(N)                                                         IND 0570
      IF (IBC .GT. 0) GO TO 4007                                        IND 0580
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE CONTRIBUTIONS TO H AND T ARRAYS (INITIAL CONDITIONS, EVERY
C       ITERATION)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (NSVIL    .GT. 0) GO TO 4015                                   IND 0590
      IF (INCIL(N) .LE. 0) GO TO 2213                                   IND 0600
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     EQUIVALENT CIRCUIT FOR INITIAL CONDITIONS IS A CURRENT SOURCE OF
C     FIXED VALUE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 4015 GO TO (4004,4005,4006),KO                                         IND 0610
 4005 T(KO1) = T(KO1) - CIL(N)                                          IND 0620
      GO TO 1203                                                        IND 0630
 4004 T(KO1) = T(KO1) - CIL(N)                                          IND 0640
 4006 T(KO2) = T(KO2) + CIL(N)                                          IND 0650
      GO TO 1203                                                        IND 0660
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     EQUIVALENT CIRCUIT FOR INITIAL CONDITICNS IS A RESISTANCE RL
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 2213 B=1.0/RL(N)                                                       IND 0670
      GO TO (204,205,206),KO                                            IND 0680
  204 H(KO1,KO1)=H(KO1,KO1)+B                                           IND 0690
      H(KO1,KO2)=H(KO1,KO2)-B                                           IND 0700
      H(KO2,KO1)=H(KO2,KO1)-B                                           IND 0710
      H(KO2,KO2)=H(KO2,KO2)+B                                           IND 0720
      GO TO 1203                                                        IND 0730
  205 H(KO1,KO1)=H(KO1,KO1)+B                                           IND 0740
      T(KO1)=T(KO1)+B*EI(KO2)                                           IND 0750
      GO TO 1203                                                        IND 0760
  206 H(KO2,KO2)=H(KO2,KO2)+B                                           IND 0770
      T(KO2)=T(KO2)+B*EI(KO1)                                           IND 0780
      GO TO 1203                                                        IND 0790
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE INDUCTOR CURRENTS (INITIAL CONDITIONS, AFTER ACHIEVING
C       CONVERGENCE)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 4007 IF (NSVIL    .GT. 0) GO TO 1203                                  IND 0800
      IF (INCIL(N) .GT. 0) GO TO 1203                                   IND 0810
      GO TO(208,209,210),KO                                             IND 0820
  208 CIL(N)=(V (KO1)-V (KO2))/RL(N)                                    IND 0830
      GO TO  1203                                                       IND 0840
  209 CIL(N)=(V (KO1)-EI(KO2))/RL(N)                                    IND 0850
      GO TO 1203                                                        IND 0860
  210 CIL(N)=(EI(KO1)-V (KO2))/RL(N)                                    IND 0870
 1203 IF (MONL) WRITE (6,1990) B,T(KO1),T(KO2),H(KO1,KO1),H(KO2,KO2),   IND 0880
     2    H(KO1,KO2)                                                   IND 0890
  203 CONTINUE                                                          IND 0900
      RETURN                                                            IND 0910
C     . . . . . .
 1990 FORMAT (34H B,T(1),T(2),H(1,1),H(2,2),H(1,2) ,                    IND 0920
     1     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )                IND 0930
 3914 FORMAT(41HCIL( ),S( ),C,B,HAA( ),B1( ),B2( ),TA( )  ,             IND 0940
     1                        31HT(1),T(2),H(1,1),H(2,2),H(1,2) , /,    IND 0950
     2 4(1H ,5E24.15,/)  )                                              IND 0960
 9999 FORMAT (13H INDUCTORS    )                                        IND 0970
      END                                                               IND 0980
```

```
      SUBROUTINE ZEN                                                    ZEN 0010
C     ZEN  COMPUTES CONTRIBUTIONS TO H AND T ARRAYS AND CURRENTS
C               OF ZENER DIODES
```

COMMON VARIABLES

```
      LOGICAL MONZEN                                                    ZEN 0020
      EQUIVALENCE (KZEN,K),(K1ZEN,K1),(K2ZEN,K2)                        ZEN 0030
      DIMENSION K(20), K1(20), K2(20), GZ(20), ZI(20)                   ZEN 0040
C  . . . . . .
      IF (NZEN .LE. 0) RETURN                                           ZEN 0050
      IF (MONZEN) WRITE (6,9999)                                        ZEN 0060
      DO 114 N = 1,NZEN                                                 ZEN 0070
      KO=K(N)                                                           ZEN 0080
      KO1=K1(N)                                                         ZEN 0090
      KO2=K2(N)                                                         ZEN 0100
      IF (IBC .GT. 0) GO TO 502                                         ZEN 0110
      IF(ABC .GT. 0.0) GO TO 208                                        ZEN 0120
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     AT THE FIRST ITERATION, COMPUTE THE ZENER-DIODE VOLTAGE VZ AND
C     DETERMINE THE EFFECTIVE REGION. COMPUTE TERMS THAT WILL REMAIN
C     CONSTANT FOR A GIVEN DELT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      GO TO (200,201,202),KO                                            ZEN 0130
  200 VZ = V(KO1) - V(KO2)                                              ZEN 0140
      GO TO 203                                                         ZEN 0150
  201 VZ = V(KO1) - EI(KO2)                                             ZEN 0160
      GO TO 203                                                         ZEN 0170
  202 VZ = EI(KO1) - V(KO2)                                             ZEN 0180
  203 IF (VZ .GT. VZB(N)) GO TO 205                                     ZEN 0190
      GZ(N) = 1.0 / RZB(N)                                              ZEN 0200
      ZI(N) = EZB(N) * GZ(N)                                            ZEN 0210
      GO TO 208                                                         ZEN 0220
  205 IF (VZ .GT. VZF(N)) GO TO 207                                     ZEN 0230
      GZ(N) = 1.0 / RZ(N)                                               ZEN 0240
      ZI(N) = 0.0                                                       ZEN 0250
      GO TO 208                                                         ZEN 0260
  207 GZ(N) = 1.0                                                       ZEN 0270
      ZI(N) = 0.7                                                       ZEN 0280
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     ADD CONTRIBUTIONS TO H AND T ARRAYS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  208 GZD = GZ(N)                                                       ZEN 0290
      GO TO (100,101,102),KO                                            ZEN 0300
  100 H(KO1,KO1)=H(KO1,KO1)+GZD                                         ZEN 0310
      H(KO1,KO2)=H(KO1,KO2)-GZD                                         ZEN 0320
      T(KO1) = T(KO1) + ZI(N)                                           ZEN 0330
      H(KO2,KO1)=H(KO2,KO1)-GZD                                         ZEN 0340
      H(KO2,KO2)=H(KO2,KO2)+GZD                                         ZEN 0350
      T(KO2) = T(KO2) - ZI(N)                                           ZEN 0360
      GO TO 14                                                          ZEN 0370
  101 H(KO1,KO1)=H(KO1,KO1)+GZD                                         ZEN 0380
      T(KO1) = T(KO1) +ZI(N) + GZD * EI(KO2)                            ZEN 0390
      GO TO 14                                                          ZEN 0400
  102 H(KO2,KO2)=H(KO2,KO2)+GZD                                         ZEN 0410
      T(KO2) = T(KO2) -ZI(N)+ GZD * EI(KC1)                             ZEN 0420
      GO TO 14                                                          ZEN 0430
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE ZENER-DIODE CURRENTS (AFTER CONVERGENCE HAS BEEN ACHIEVED)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  502 GO TO (504,506,508),KO                                            ZEN 0440
  504 CIZ(N) = (V(KO1)-V(KO2)) * GZ(N) - ZI(N)                          ZEN 0450
      GO TO 14                                                          ZEN 0460
  506 CIZ(N) = (V(KO1)-EI(KO2)) * GZ(N) - ZI(N)                         ZEN 0470
      GO TO 14                                                          ZEN 0480
  508 CIZ(N) = (EI(KO1) - V(KO2)) * GZ(N) - ZI(N)                       ZEN 0490
   14 IF(MONZEN)WRITE (6,9114) GZ(N),ZI(N),CIZ(N),                      ZEN 0500
     2      T(KO1),T(KO2),H(KO1,KO1),H(KC2,KO2),H(KO1,KO2)             ZEN 0510
  114 CONTINUE                                                          ZEN 0520
      RETURN                                                            ZEN 0530
C  . . . . . .
 9114 FORMAT (21H GZ( ),ZI( ),CIZ( ),     ,                          ZEN 0540
     2              32HT(1),T(2),H(1,1),H(2,2),H(1,2) ,                ZEN 0550
     1     5(/,1H ,E22.15,3E22.15,E22.15,E22.15) )                     ZEN 0560
 9999 FORMAT (13H ZENER          )                                      ZEN 0570
      END                                                               ZEN 0580
```

```
      SUBROUTINE DIOD                                                   DIO 0010
C     DIOD COMPUTES CONTRIBUTIONS TO H AND T ARRAYS AND CURRENTS
C               OF DIODES

                    COMMON  VARIABLES

      LOGICAL MOND                                                      DIO 0020
      EQUIVALENCE (KDIOD,K),(K1DIOD,K1),(K2DIOD,K2)                     DIO 0030
      DIMENSION K(20), K1(20), K2(20)                                   DIO 0040
      DIMENSION DIP(20)                                                 DIO 0050
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
      IF(NDIODE .LE. 0) RETURN                                          DIO 0060
      IF (MOND)WRITE (6,9999)                                           DIO 0070
      IF(IBC .GT. 0) GO TO 3000                                         DIO 0080
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C    * * * P A R T (1) - COMPUTE CONTRIBUTIONS TO H AND T ARRAYS * * *
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF(DELT.EQ.TOOLD) GO TO 7950                                      DIO 0090
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE DELT-DEPENDENT TERMS (AT FIRST ITERATION,IF DELT VALUE HAS
C     CHANGED)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      TOOLD=DELT                                                        DIO 0100
      DO 2212 N=1,NDIODE                                                DIO 0110
      DT1(N)=QCR(N,7)/DELT                                              DIO 0120
      DT2(N)= 1.+DT1(N)                                                 DIO 0130
      DT3(N) = QCR(N,4) / DELT                                          DIO 0140
      IF (MOND) WRITE (6,2272) TOOLD,C1(N),C2(N),C3(N),C4(N),           DIO 0150
     2     DT1(N),DT2(N),DT3(N)                                         DIO 0160
 2212 CONTINUE                                                          DIO 0170
 7950 DO 116 N = 1,NDIODE                                               DIO 0180
      KO=K(N)                                                           DIO 0190
      KO1=K1(N)                                                         DIO 0200
      KO2=K2(N)                                                         DIO 0210
      SI1=QCR(N,1)                                                      DIO 0220
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE DIODE VOLTAGE BV AT THE PREVIOUS DELT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      GO TO (100,101,102),KO                                            DIO 0230
  100 BV=V1(KO1)-V1(KO2)                                                DIO 0240
      GO TO 1000                                                        DIO 0250
  101 BV=V1(KO1)-EP1(KO2)                                               DIO 0260
      GO TO 1000                                                        DIO 0270
  102 BV=EP1(KO1)-V1(KO2)                                               DIO 0280
 1000 IF(ABC .GT.0.0)GO TO 8002                                         DIO 0290
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     AT THE FIRST ITERATION ONLY (UNTIL 8002)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      LOOP = 0                                                          DIO 0300
      IF (JSJ .LE. 0) GO TO 1001                                        DIO 0310
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     FOR INITIAL CONDITIONS (UNTIL 1001)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF(BV .GT. -1.0) GO TO 9001                                       DIO 0320
      DI(N)=-SI1                                                        DIO 0330
      DV(N)=BV                                                          DIO 0340
      GO TO 1003                                                        DIO 0350
```

```
 9001 IF(DV(N)-BV)1002,1001,1001                                        DIO 0360
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE DC CURRENT AND VOLTAGE (AT FIRST ITERATION)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
1001  DEXP=BV*C2(N)                                                    DIO 0370
      IF(DEXP .LT. 50.0) GO TO 1008                                    DIO 0380
      LOOP = LOOP + 1                                                  DIO 0390
      IF (LOOP .LE. 100) GO TO 1002                                    DIO 0400
      WRITE (6,9500) N                                                 DIO 0410
      CALL EXIT                                                        DIO 0420
 1002 DI(N)=(BV-DV(N))*DR(N)+DI(N)                                     DIO 0430
      IF(DI(N) .LE. -SI1) GO TO 1001                                   DIO 0440
      DV(N)=      ALOG(DI(N)/SI1+1.0)/C2(N)                            DIO 0450
      GO TO 1003                                                       DIO 0460
 1008 DI(N)=SI1*(EXP(DEXP)-1.0)                                        DIO 0470
      DV(N)=BV                                                         DIO 0480
1003  DR(N)=(DI(N)+SI1)*C2(N)                                          DIO 0490
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     UPDATE PREVIOUS DC CURRENT DIP  IF PRESENT VARIABLES ARE NOT RESET
C       DUE TO CONVERGENCE FAILURE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF(NAW .EQ. 0) DIP(N) = DI(N)                                    DIO 0500
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE EFFECT OF JUNCTION CAPACITANCE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 8002 DV2 = AMIN1(DV(N),C1(N))                                         DIO 0510
      D = DT3(N) / SQRT(1.0 - DV2 * C4(N))                             DIO 0520
      DDI    =DI(N)-DIP(N)                                             DIO 0530
      IF (DR(N) .GT. 10000.0) DR(N) = 10000.0                          DIO 0540
      B=DR(N)*DT2(N)+C3(N)                                             DIO 0550
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     SET PHOTO CURRENT AT PRESENT ANC PREVICUS DELT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF(NAW)3050,3150,3151                                            DIO 0560
 3150 PPID1(N)=PPID(N)                                                 DIO 0570
      GO TO  3051                                                      DIO 0580
 3151 PPID(N)=PPID1(N)                                                 DIO 0590
3051  PPID(N)=(EI(19)*QCR(N,6)+DT1(N)*PPID(N))/DT2(N)                  DIO 0600
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE CONTRIBUTIONS TO H AND T ARRAYS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
3050  C=DV(N)*DR(N)*DT2(N)-DI(N)+PPID(N)-DT1(N)*DDI                    DIO 0610
      HACH=B+D                                                         DIO 0620
      TACH=C+BV*D                                                      DIO 0630
      GO TO (300,301,302),KO                                           DIO 0640
300   H(KO1,KO1)=H(KO1,KO1)+HACH                                       DIO 0650
      H(KO1,KO2)=H(KO1,KO2)-HACH                                       DIO 0660
      T(KO1)=T(KO1)+TACH                                               DIO 0670
      H(KO2,KO1)=H(KO2,KO1)-HACH                                       DIO 0680
      H(KO2,KO2)=H(KO2,KO2)+HACH                                       DIO 0690
      T(KO2)=T(KO2)-TACH                                               DIO 0700
      GO TO 16                                                         DIO 0710
301   H(KO1,KO1)=H(KO1,KO1)+HACH                                       DIO 0720
      T(KO1)=T(KO1)+TACH+HACH*EI(KO2)                                  DIO 0730
      GO TO 16                                                         DIO 0740
302   H(KO2,KO2)=H(KO2,KO2)+HACH                                       DIO 0750
      T(KO2)=T(KO2)-TACH+HACH*EI(KO1)                                  DIO 0760
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE TOTAL DIODE CURRENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   16 CIDIOD(N) = HACH * DV(N) - TACH                                  DIO 0770
      IF (MOND) WRITE (6,9116) BV,DV(N),DV2,D,DI(N),DDI    ,DR(N),PPID(N)DIO 0780
     2    ,B,C,T(KO1),T(KO2),H(KO1,KO1),H(KO2,KO2),H(KO1,KO2),CIDIOD(N) DIO 0790
  116 CONTINUE                                                         DIO 0800
      RETURN                                                           DIO 0810
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
C     * * * P A R T (2) - COMPUTE DC DIODE CURRENTS AND TEST FOR
C     CONVERGENCE * * *
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 3000 DO 9102 N = 1,NDIODE                                          DIO 0820
      KO=K(N)                                                       DIO 0830
      KO1=K1(N)                                                     DIO 0840
      KO2=K2(N)                                                     DIO 0850
      SI1=QCR(N,1)                                                  DIO 0860
      GO TO (3003,3004,3005),KO                                     DIO 0870
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE DIODE VOLTAGE BV AT PRESENT DELT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 3003 BV=V(KO1)-V(KO2)                                              DIO 0880
      GO TO 3006                                                    DIO 0890
 3004 BV=V(KO1)-EI(KO2)                                             DIO 0900
      GO TO 3006                                                    DIO 0910
 3005 BV=EI(KO1)-V(KO2)                                             DIO 0920
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE NEW DC DIODE CURRENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 3006 CI=(BV-DV(N))*DR(N)+DI(N)                                     DIO 0930
      IF(BV*CI .GE. -1.0E-20) GO TO 3008                            DIO 0940
      DR(N)=DI(N)/DV(N)                                             DIO 0950
      ABC=1.0                                                       DIO 0960
      IF (MOND  ) WRITE (6,9998)                                    DIO 0970
      GO TO 3002                                                    DIO 0980
 3008 IF(CI .GT.(-SI1-SI1))GO TO 3010                               DIO 0990
      ABC=1.0                                                       DIO 1000
      IF (MOND  ) WRITE (6,9998)                                    DIO 1010
      DI(N)=-SI1                                                    DIO 1020
      DV(N)=BV                                                      DIO 1030
      DR(N)=0.0                                                     DIO 1040
      GO TO 3002                                                    DIO 1050
3010  IF (BV*C2(N) .LT. 1.0) GO TO 3002                             DIO 1060
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     TEST FOR CONVERGENCE OF DC CURRENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DEXP=BV*C2(N)                                                 DIO 1070
      IF(DEXP .GT. 60.0) GO TO 3009                                 DIO 1080
      DEXP=SI1*(EXP(DEXP)-1.0)                                      DIO 1090
      IF(ABS(CI-DEXP) .LT. 0.1*DEXP) GO TO 3002                     DIO 1100
 3009 ABC=1.0                                                       DIO 1110
      IF (MOND  ) WRITE (6,9998)                                    DIO 1120
      DI(N)=CI                                                      DIO 1130
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE NEW DIODE VOLTAGE AND I/V SLOPE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DV(N)=      ALOG(DI(N)/SI1+1.0)/C2(N)                         DIO 1140
      DR(N)=(DI(N)+SI1)*C2(N)                                       DIO 1150
 3002 PWCR(N)=DI(N)*DV(N)+BV**2*C3(N)                               DIO 1160
      IF (MOND) WRITE (6,9112) BV,CI,DR(N),DI(N),DV(N),DEXP,PWCR(N) DIO 1170
 9102 CONTINUE                                                      DIO 1180
      RETURN                                                        DIO 1190
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
 2272 FORMAT (45H TOOLD,C1(),C2(),C3(),C4(),DT1(),DT2(),DT3() ,     DIO 1200
     2     2(/,5E24.15))                                            DIO 1210
 9112 FORMAT (34H BV,CI,DR(),DI(),DV(),DEXP,PWCR() ,                DIO 1220
     1     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )            DIO 1230
 9500 FORMAT (18H LOOPING IN DIODE    ,I5)                          DIO 1240
 9116 FORMAT (42H BV,DV(),DV2,D,DI(),DDI(),DR(),PPID(),B,C ,        DIO 1250
     2    41H,T(1),T(2),H(1,1),H(2,2),H(1,2),CIDIOD()          ,    DIO 1260
     3     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )            DIO 1270
 9998 FORMAT (18H DID NOT CONVERGE    )                             DIO 1280
 9999 FORMAT (13H DIODES       )                                    DIO 1290
      END                                                           DIO 1300
```

```
      SUBROUTINE TRAN                                                    TRA 0010
C     TRAN COMPUTES CONTRIBUTIONS TO H AND T ARRAYS AND CURRENTS
C               OF TRANSISTORS
```

COMMON VARIABLES

```
      LOGICAL MONT                                                       TRA 0020
      EQUIVALENCE (KTRAN,K),(K1TRAN,K1),(K2TRAN,K2),(K3TRAN,K3),         TRA 0030
     2    (K4TRAN,K4)                                                   TRA 0040
      DIMENSION K(20), K1(20), K2(20), K3(20), K4(20)                    TRA 0050
      DIMENSION   BCC(20),  BCIP(20),  BEC(20),  BEIP(20),               TRA 0060
     2                     DBCI(20),  DBEI(20)                         TRA 0070
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
      IF(NTRANS .LE. 0) RETURN                                           TRA 0080
      IF (MONT)WRITE (6,9999)                                            TRA 0090
      IF(IBC .GT. 0) GO TO 9200                                          TRA 0100
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     * * * P A R T (1)- COMPUTE CONTRIBUTIONS TO H AND T ARRAYS * * *
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (DELT.EQ. TTOOLD ) GO TO 2213                                   TRA 0110
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE DELT-DEPENDENT TERMS (AT FIRST ITERATION,IF DELT VALUE HAS
C     CHANGED)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      TTOOLD=DELT                                                        TRA 0120
      DO 2212 N=1,NTRANS                                                 TRA 0130
      CDT1(N)=QTAN(N,4)/DELT                                             TRA 0140
      CDT2(N)=1.0+CDT1(N)                                                TRA 0150
      CDT3(N) = QTAN(N,7) / DELT                                         TRA 0160
      EDT1(N)=QTAN(N,3)/DELT                                             TRA 0170
      EDT2(N)=1.0+EDT1(N)                                                TRA 0180
      EDT3(N) = QTAN(N,12) / DELT                                        TRA 0190
      IF (MONT) WRITE (6,2272) TTOOLD,CDT1(N),CDT2(N),CDT3(N),           TRA 0200
     2     EDT1(N),EDT2(N),EDT3(N)                                       TRA 0210
 2212 CONTINUE                                                           TRA 0220
 2213 DO  909 N = 1,NTRANS                                               TRA 0230
      KO=K (N)                                                           TRA 0240
      KO1=K2(N)                                                          TRA 0250
      KO2=K3(N)                                                          TRA 0260
      KO3=K4(N)                                                          TRA 0270
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE BASE-COLLECTOR VOLTAGE BC AND BASE-EMITTER VOLTAGE BE AT
C       PREVIOUS DELT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      GO TO (201,202,203,204),KO                                         TRA 0280
  201 BC=V1(KO1)-V1(KO2)                                                 TRA 0290
      BE=V1(KO1)-V1(KO3)                                                 TRA 0300
      GO TO 300                                                          TRA 0310
  202 BC=EP1(KO1)-V1(KO2)                                                TRA 0320
      BE=EP1(KO1)-V1(KO3)                                                TRA 0330
      GO TO 300                                                          TRA 0340
  203 BC=V1(KO1)-EP1(KO2)                                                TRA 0350
      BE=V1(KO1)-V1(KO3)                                                 TRA 0360
      GO TO 300                                                          TRA 0370
  204 BC=V1(KO1)-V1(KO2)                                                 TRA 0380
      BE=V1(KO1)-EP1(KO3)                                                TRA 0390
  300 IF(K1(N).LE. 0) GO TO 301                                          TRA 0400
```

```
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PNP TRANSISTOR ONLY (UNTIL 301)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      BE=-BE                                                          TRA 0410
      BC=-BC                                                          TRA 0420
      BCV(N)=-BCV(N)                                                  TRA 0430
      BEV(N)=-BEV(N)                                                  TRA 0440
      BCI(N)=-BCI(N)                                                  TRA 0450
      BEI(N)=-BEI(N)                                                  TRA 0460
      BCIP(N)=-BCIP(N)                                                TRA 0470
      BEIP(N)=-BEIP(N)                                                TRA 0480
  301 CO=CC2(N)                                                       TRA 0490
      EO=EC2(N)                                                       TRA 0500
      CIO1=QTAN(N,5)                                                  TRA 0510
      EIO1=QTAN(N,10)                                                 TRA 0520
      IF(ABC .GT.0.0)GO TO 8002                                       TRA 0530
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     AT THE FIRST ITERATION ONLY (UNTIL 8002)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (DELT .LE. 0.0) GO TO 8002                                   TRA 0540
      LOOP = 0                                                        TRA 0550
      IF (JSJ .LE. 0) GO TO 307                                       TRA 0560
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     FOR INITIAL CONDITIONS (UNTIL 307)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF(BC .GT. -1.0) GO TO 9317                                     TRA 0570
      BCI(N)=-CIO1                                                    TRA 0580
      BCV(N)=BC                                                       TRA 0590
      GO TO 601                                                       TRA 0600
 9317 IF(BCV(N) .LT. BC) GO TO 308                                    TRA 0610
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE DC CURRENT AND VOLTAGE (AT FIRST ITERATION)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  307 DEXP=BC*CO                                                      TRA 0620
      IF(DEXP .LT. 50.0) GO TO 2308                                   TRA 0630
      LOOP = LOOP + 1                                                 TRA 0640
      IF (LOOP .LE. 100) GO TO 308                                    TRA 0650
      WRITE (6,9501) N                                                TRA 0660
      CALL EXIT                                                       TRA 0670
 2308 BCI(N)=CIO1*(EXP(DEXP)-1.0)                                     TRA 0680
      BCV(N)=BC                                                       TRA 0690
      GO TO 601                                                       TRA 0700
  308 BCI(N)=(BC-BCV(N))*BCR(N)+BCI(N)                                TRA 0710
      IF(BCI(N) .LE. -CIO1) GO TO  307                                TRA 0720
      BCV(N)=ALOG(BCI(N)/CIO1+1.0)/CO                                 TRA 0730
  601 BCR(N)=(BCI(N)+CIO1)*CO                                         TRA 0740
      BCV2=AMIN1(BCV(N),CC1(N))                                       TRA 0750
      IF (MONT) WRITE (6,9751) BC,BE,AF(N),AR(N),CO,EO,CIO1,EIO1,BCI(N),TRA 0760
     2     BCV2,BCR(N),BCV(N),DEXP                                    TRA 0770
      BCC(N) = COT3(N) / SQRT(1.0 - BCV2 * CC4(N))                    TRA 0780
      LOOP = 0                                                        TRA 0790
      IF (BER(N)*QTAN(N,14) .LE. 0.1) GO TO 313                       TRA 0800
      IF (JSJ .LE. 0) GO TO 313                                       TRA 0810
      IF(BE .GT. -1.0) GO TO 9313                                     TRA 0820
      BEI(N)=-EIO1                                                    TRA 0830
      BEV(N)=BE                                                       TRA 0840
      GO TO 602                                                       TRA 0850
 9313 IF (BEV(N) .LT. BE) GO TO 314                                   TRA 0860
  313 DEXP=BE*EO                                                      TRA 0870
      IF(DEXP .LT. 50.0) GO TO 2314                                   TRA 0880
      LOOP = LOOP + 1                                                 TRA 0890
      IF (LOOP .LE. 100) GO TO 314                                    TRA 0900
      WRITE (6,9502) N                                                TRA 0910
      CALL EXIT                                                       TRA 0920
 2314 BEI(N)=EIO1*(EXP(DEXP)-1.0)                                     TRA 0930
      BEV(N)=BE                                                       TRA 0940
```

```
      GO TO 602                                                          TRA 0950
  314 BEI(N)=(BE-BEV(N))*BER(N)+BEI(N)                                   TRA 0960
      IF(BEI(N) .LE. -EIO1)  GO TO 313                                   TRA 0970
      BEV(N)=ALOG(BEI(N)/EIO1+1.0)/EC                                    TRA 0980
  602 BER(N)=EO*(BEI(N)+EIO1)                                            TRA 0990
      BEV2=AMIN1(BEV(N),EC1(N))                                          TRA 1000
      BEC(N) = EDT3(N) / SQRT(1.0 - BEV2 * EC4(N))                       TRA 1010
      IF(NAW .NE. 0) GO TO 8002                                          TRA 1020
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     UPDATE PREVIOUS DC CURRENTS, BCIP AND BEIP, IF PRESENT VARIABLES
C       ARE NOT RESET DUE TO CONVERGENCE FAILURE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      BCIP(N)=BCI(N)                                                     TRA 1030
      BEIP(N)=BEI(N)                                                     TRA 1040
 8002 IF(K1(N).LE. 0) GO TO 208                                          TRA 1050
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PNP TRANSISTOR ONLY (UNTIL 208 )
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      BC=-BC                                                             TRA 1060
      BE=-BE                                                             TRA 1070
      BCV(N)=-BCV(N)                                                     TRA 1080
      BEV(N)=-BEV(N)                                                     TRA 1090
      BCI(N)=-BCI(N)                                                     TRA 1100
      BEI(N)=-BEI(N)                                                     TRA 1110
      BCIP(N)=-BCIP(N)                                                   TRA 1120
      BEIP(N)=-BEIP(N)                                                   TRA 1130
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE EFFECTS OF JUNCTION CAPACITANCES
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  208 BCR(N)=AMIN1(BCR(N),100.0)                                         TRA 1140
      BER(N)=AMIN1(BER(N),100.0)                                         TRA 1150
      DBCI(N)=(BCI(N)-BCIP(N))*CDT1(N)                                   TRA 1160
      DBEI(N)=(BEI(N)-BEIP(N))*EDT1(N)                                   TRA 1170
      TCR=BCR(N)*CDT2(N)                                                 TRA 1180
      TER=BER(N)*EDT2(N)                                                 TRA 1190
      IF (MONT) WRITE (6,9752) BCV2,BCC(N),BEI(N),BEV(N),BER(N),BEC(N),  TRA 1200
     2    DBCI(N),DBEI(N),TCR,TER,DEXP,BEV2                              TRA 1210
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     SET PHOTO CURRENTS AT PRESENT AND PREVICUS DELT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF(NAW)3050,3150,3151                                              TRA 1220
 3150 PPIE1(N)=PPIE(N)                                                   TRA 1230
      PPIC1(N)=PPIC(N)                                                   TRA 1240
      GO TO 3051                                                         TRA 1250
 3151 PPIE(N)=PPIE1(N)                                                   TRA 1260
      PPIC(N)=PPIC1(N)                                                   TRA 1270
 3051 PPIE(N)=(EI(19)*QTAN(N,16)+EDT1(N)*PPIE(N))/EDT2(N)                TRA 1280
      PPIC(N)=(EI(19)*QTAN(N,15)+CDT1(N)*PPIC(N))/CDT2(N)                TRA 1290
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE CONTRIBUTIONS TO H AND T ARRAYS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 3050 HACH22=TCR+CC3(N)+BCC(N)                                           TRA 1300
      HACH33=TER+EC3(N)+BEC(N)                                           TRA 1310
      HACH23= -AF(N)*BER(N)                                              TRA 1320
      HACH32 = -AR(N)*BCR(N)                                             TRA 1330
      TACH2=  BCV(N)*TCR-BCI(N)-AF(N)*(BEV(N)*BER(N)-BEI(N))             TRA 1340
     1 -DBCI(N) +BCC(N)*BC    -PPIC(N)                                    TRA 1350
      TACH3=  BEV(N)*TER -BEI(N) -AR(N)*(BCV(N)*BCR(N)-BCI(N))           TRA 1360
     1 - DBEI(N) +BEC(N)*BE    -PPIE(N)                                  TRA 1370
      IF (MONT) WRITE (6,9757) PPIE(N),PPIC(N),HACH22,HACH33,HACH23,     TRA 1380
     2     HACH32,TACH2,TACH3                                            TRA 1390
      GO TO (401,402,403,404),KO                                         TRA 1400
  401 H(KO1,KO1)=H(KO1,KO1)+HACH22+HACH33+HACH23+HACH32                  TRA 1410
      H(KO1,KO2)=H(KO1,KO2)-HACH22-HACH32                                TRA 1420
      H(KO1,KO3)=H(KO1,KO3)-HACH23-HACH33                                TRA 1430
      T(KO1)=T(KO1)+TACH2+TACH3                                          TRA 1440
```

```
      H(KO2,KO1)=H(KO2,KO1)-HACH22-HACH23                                    TRA 1450
      H(KO2,KO2)=H(KO2,KO2)+HACH22                                           TRA 1460
      H(KO2,KO3)=H(KO2,KO3)+HACH23                                           TRA 1470
      T(KO2)=T(KO2)-TACH2                                                    TRA 1480
      H(KO3,KO1)=H(KO3,KO1)-HACH32-HACH33                                    TRA 1490
      H(KO3,KO2)=H(KO3,KO2)+HACH32                                           TRA 1500
      H(KO3,KO3)=H(KO3,KO3)+HACH33                                           TRA 1510
      T(KO3)=T(KO3)-TACH3                                                    TRA 1520
      GO TO 200                                                              TRA 1530
  402 H(KO2,KO2)=H(KO2,KO2)+HACH22                                           TRA 1540
      H(KO2,KO3)=H(KO2,KO3)+HACH23                                           TRA 1550
      T(KO2)=T(KO2)-TACH2                 +(HACH22+HACH23)*EI(KO1)           TRA 1560
      H(KO3,KO2)=H(KO3,KO2)+HACH32                                           TRA 1570
      H(KO3,KO3)=H(KO3,KO3)+HACH33                                           TRA 1580
      T(KO3)=T(KO3)-TACH3                 +(HACH32+HACH33)*EI(KO1)           TRA 1590
      GO TO 200                                                              TRA 1600
  403 H(KO1,KO1)=H(KO1,KO1)+HACH22+HACH33+HACH23+HACH32                      TRA 1610
      H(KO1,KO3)=H(KO1,KO3)-HACH23-HACH33                                    TRA 1620
      T(KO1)=T(KO1)+TACH2+TACH3           +(HACH22+HACH32)*EI(KO2)           TRA 1630
      H(KO3,KO1)=H(KO3,KO1)-HACH32-HACH33                                    TRA 1640
      H(KO3,KO3)=H(KO3,KO3)+HACH33                                           TRA 1650
      T(KO3)=T(KO3)-TACH3                 -HACH32*EI(KO2)                    TRA 1660
      GO TO 200                                                              TRA 1670
  404 H(KO1,KO1)=H(KO1,KO1)+HACH22+HACH33+HACH23+HACH32                      TRA 1680
      H(KO1,KO2)=H(KO1,KO2)-HACH22-HACH32                                    TRA 1690
      T(KO1)=T(KO1)+TACH2+TACH3           +(HACH23+HACH33)*EI(KO3)           TRA 1700
      H(KO2,KO1)=H(KO2,KO1)-HACH22-HACH23                                    TRA 1710
      H(KO2,KO2)=H(KO2,KO2)+HACH22                                           TRA 1720
      T(KO2)=T(KO2)-TACH2                 -HACH23*EI(KO3)                    TRA 1730
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE TOTAL COLLECTOR, EMITTER, AND BASE CURRENTS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  200 CICOLC(N) = TACH2 - HACH22*BCV(N) - HACH23*BEV(N)                       TRA 1740
      CIEMTR(N) = HACH32*BCV(N) + HACH33*BEV(N) - TACH3                      TRA 1750
      CIBASE(N) = CIEMTR(N) - CICOLC(N)                                      TRA 1760
      IF (MONT) WRITE(6,9753) T(KO1),T(KO2),T(KO3),H(KO1,KO1),               TRA 1770
     2    H(KO2,KO2),H(KO3,KO3),H(KO1,KO2),H(KO2,KO1),H(KO1,KO3),            TRA 1780
     3    H(KO3,KO1),H(KO2,KO3),H(KO3,KO2),CICOLC(N),CIEMTR(N),CIBASE(N)     TRA 1790
  909 CONTINUE                                                               TRA 1800
      RETURN                                                                 TRA 1810
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     * * * P A R T (2)- COMPUTE DC JUNCTION CURRENTS AND TEST FOR
C     CONVERGENCE * * *
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 9200 DO 9997 N=1,NTRANS                                                     TRA 1820
      KO=K (N)                                                               TRA 1830
      KO1=K2(N)                                                              TRA 1840
      KO2=K3(N)                                                              TRA 1850
      KO3=K4(N)                                                              TRA 1860
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE BASE-COLLECTOR VOLTAGE BC AND BASE-EMITTER VOLTAGE BE AT
C     PRESENT DELT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      GO TO (9001,9002,9003,9004),KO                                         TRA 1870
 9001 BC=V (KO1)-V (KO2)                                                     TRA 1880
      BE=V (KO1)-V (KO3)                                                     TRA 1890
      GO TO 9100                                                             TRA 1900
 9002 BC=EI (KO1)-V (KO2)                                                    TRA 1910
      BE=EI (KO1)-V (KO3)                                                    TRA 1920
      GO TO 9100                                                             TRA 1930
 9003 BC=V (KO1)-EI (KO2)                                                    TRA 1940
      BE=V (KO1)-V (KO3)                                                     TRA 1950
      GO TO 9100                                                             TRA 1960
 9004 BC=V (KO1)-V (KO2)                                                     TRA 1970
      BE=V (KO1)-EI (KO3)                                                    TRA 1980
 9100 CO=CC2(N)                                                              TRA 1990
```

```
      EO=EC2(N)                                                          TRA 2000
      CIO1=QTAN(N,5)                                                     TRA 2010
      EIO1=QTAN(N,10)                                                    TRA 2020
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE NEW DC JUNCTION CURRENTS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      CI=(BC-BCV(N))*BCR(N)+BCI(N)                                       TRA 2030
      EZ=(BE-BEV(N))*BFR(N)+BEI(N)                                       TRA 2040
      IF(K1(N).LE. 0) GO TO 9101                                         TRA 2050
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PNP TRANSISTOR ONLY (UNTIL 9101)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      BC=-BC                                                             TRA 2060
      CI=-CI                                                             TRA 2070
      BE=-BF                                                             TRA 2080
      EZ=-EZ                                                             TRA 2090
 9101 IF (MONT) WRITE (6,9754) BC,BE,CC,EC,CIO1,EIO1,CI,EZ               TRA 2100
      IF(BC*CI .GE. -1.0E-20) GO TO 9104                                 TRA 2110
      ABC=1.0                                                            TRA 2120
      IF (MONT  ) WRITE (6,9998)                                         TRA 2130
      BCR(N)=BCI(N)/BCV(N)                                               TRA 2140
      GO TO 9108                                                         TRA 2150
 9104 IF(CI .GT.(-CIO1-CIO1)) GO TO 9107                                 TRA 2160
      ABC=1.0                                                            TRA 2170
      IF (MONT  ) WRITE (6,9998)                                         TRA 2180
      BCI(N)=-CIO1                                                       TRA 2190
      BCV(N)=BC                                                          TRA 2200
      GO TO 9302                                                         TRA 2210
 9107 IF(BC*CO  .LT. 1.0) GO TO 9108                                     TRA 2220
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     TEST FOR CONVERGENCE OF BASE-COLLECTOR DC CURRENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DEXP=BC*CO                                                         TRA 2230
      IF(DEXP .GT. 60.0) GO TO 9106                                      TRA 2240
      DEXP=CIO1*(EXP(DEXP)-1.0)                                          TRA 2250
      IF(ABS(CI-DEXP) .LT. 0.1*DEXP) GO TO 9108                          TRA 2260
 9106 ABC=1.0                                                            TRA 2270
      IF (MONT  ) WRITE (6,9998)                                         TRA 2280
 9105 BCI(N)=CI                                                          TRA 2290
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE NEW BASE-COLLECTOR VOLTAGE AND I/V SLOPE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      BCV(N)=ALOG(CI/CIO1+1.0)/CO                                        TRA 2300
 9302 BCR(N)=(BCI(N)+CIO1)*CO                                            TRA 2310
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE EFFECT OF BASE-COLLECTOR JUNCTION CAPACITANCE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      BCV2=AMIN1(BCV(N),CC1(N))                                          TRA 2320
      BCC(N) = CDT3(N) / SQRT(1.0 - BCV2 * CC4(N))                       TRA 2330
      IF(K1(N).LE. 0) GO TO 9108                                         TRA 2340
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PNP TRANSISTOR ONLY (UNTIL 9108)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      BCI(N)=-BCI(N)                                                     TRA 2350
      BCV(N)=-BCV(N)                                                     TRA 2360
 9108 IF (MONT) WRITE (6,9755) ABC,BCV2,DEXP,BCI(N),BCV(N),BCR(N)        TRA 2370
      IF(BE*EZ .GE. -1.0E-20) GO TO 9115                                 TRA 2380
      ABC=1.0                                                            TRA 2390
      IF (MONT  ) WRITE (6,9998)                                         TRA 2400
      BER(N)=BEI(N)/BEV(N)                                               TRA 2410
      GO TO 9000                                                         TRA 2420
 9115 IF(EZ .GT.(-EIO1-EIO1))GO TO 9118                                  TRA 2430
      ABC=1.0                                                            TRA 2440
      IF (MONT  ) WRITE (6,9998)                                         TRA 2450
      BEI(N)=-EIO1                                                       TRA 2460
      BEV(N)=BE                                                          TRA 2470
```

```
      GO TO 9306                                                        TRA 2480
 9118 IF(BE*EO .LT. 1.0) GO TO 9000                                     TRA 2490
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     TEST FOR CONVERGENCE OF BASE-EMITTER CURRENT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DEXP=BE*EO                                                        TRA 2500
      IF(DEXP .GT. 60.0) GO TO 9117                                     TRA 2510
      DEXP=EIO1*(EXP(DEXP)-1.0)                                         TRA 2520
      IF(ABS(EZ-DEXP) .LT. 0.1*DEXP) GO TO 9000                         TRA 2530
 9117 ABC = 1.0                                                         TRA 2540
      IF (MONT  ) WRITE (6,9998)                                        TRA 2550
 9119 BEI(N)=EZ                                                         TRA 2560
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE NEW BASE-EMITTER VOLTAGE AND I/V SLOPE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      BEV(N)=ALOG(EZ/EIO1  +1.0)/EO                                     TRA 2570
 9306 BER(N)=(BEI(N)+EIO1)*EO                                           TRA 2580
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE EFFECT OF BASE-EMITTER JUNCTION CAPACITANCE
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      BEV2=AMIN1(BEV(N),EC1(N))                                         TRA 2590
      BEC(N) = EDT3(N) / SQRT(1.0 - BEV2 * EC4(N))                      TRA 2600
      IF(K1(N).LE. 0) GO TO 9000                                        TRA 2610
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     PNP TRANSISTOR ONLY (UNTIL 9000)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      BEI(N)=-BEI(N)                                                    TRA 2620
      BEV(N)=-BEV(N)                                                    TRA 2630
 9000 PWTR(N)=BEV(N)*(BEI(N)-BCI(N)*AR(N))+BCV(N)*(BCI(N)               TRA 2640
     1 - BEI(N)*AF(N)) +BC**2*CC3(N)+BE**2*EC3(N)                       TRA 2650
      IF (MONT) WRITE (6,9756) BER(N),BEV2,BEC(N),BEI(N),BEV(N),DEXP,   TRA 2660
     2    PWTR(N)                                                       TRA 2670
 9997 CONTINUE                                                          TRA 2680
      RETURN                                                            TRA 2690
C     . . . . . .
 2272 FORMAT (56H TTOOLD,CDT1(),CDT2(),CDT3(),EDT1(),EDT2(),EDT3()      TRA 2700
     2     ,2(/,5E24.15))                                               TRA 2710
 9501 FORMAT (38H LOOPING IN COLLECTOR OF TRANSISTOR    ,I5)            TRA 2720
 9502 FORMAT (36H LOOPING IN  EMITTER OF TRANSISTOR     ,I5)            TRA 2730
 9751 FORMAT (58H BC,BE,AF,AR,CO,EO,CI61,EI61,BCI(),BCV2,BCR(),BCV(),DEXTRA 2740
     2P     4(/,5E24.15))                                               TRA 2750
 9752 FORMAT (68H BCV2,BCC(),BEI(),BEV(),BER(),BEC(),DBCI(),DBEI(),TCR,TTRA 2760
     2ER,DEXP,BEV2      ,4(/,5E24.15))                                 TRA 2770
     1     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )                TRA 2780
 9753 FORMAT (36H T(1),T(2),T(3),H(1,1),H(2,2),H(3,3)                   TRA 2790
     2     ,   43H,H(1,2),H(2,1),H(1,3),H(3,1),H(2,3),H(3,2)  ,         TRA 2800
     3      30H,CICOLC(),CIEMTR(),CIBASE()                    ,        TRA 2810
     4     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )                TRA 2820
 9754 FORMAT (29H BC,BE,CO,EO,CIO1,EIC1,CI,EZ ,                         TRA 2830
     1     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )                TRA 2840
 9755 FORMAT (33H ABC,BCV2,DEXP,BCI(),BCV(),BCR()  ,                    TRA 2850
     1     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )                TRA 2860
 9756 FORMAT (42H BER(),BEV2,BEC(),BEI(),BEV(),DEXP,PWTR() ,/,          TRA 2870
     1     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )                TRA 2880
 9757 FORMAT (55H PPIE(),PPIC(),HACH22,HACH33,HACH23,HACH32,TACH2,TACH3 TRA 2890
     2     ,2(/,5E24.15))                                               TRA 2900
 9998 FORMAT (18H DID NOT CONVERGE    )                                 TRA 2910
 9999 FORMAT (13H TRANSISTORS  )                                        TRA 2920
      END                                                               TRA 2930
```

```
      SUBROUTINE CORE                                                    COR 0010
C     CORE COMPUTES CONTRIBUTIONS TO H AND T ARRAYS AND CURRENTS
C               OF MAGNETIC CORES

                    COMMON VARIABLES

      LOGICAL MONCOR , AITKEN                                            COR 0020
      REAL NS,NP                                                         COR 0030
      INTEGER STSWEX                                                     COR 0040
      EQUIVALENCE (KCORE,KC),(K1CORE,K1C),(K2CORE,K2C)                   COR 0050
      DIMENSION CIJM1(20,10), CIJM2(20,10), GC(20,10),                   COR 0060
     2 KC(20,10), K1C(20,10), K2C(20,10), PHDTEP(20), VC(20,10)          COR 0070
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
      IF(NCORE .LE. 0) RETURN                                            COR 0080
      IF (MONCOR) WRITE (6,9999)                                         COR 0090
      IF(NG  .NE. -1) GO TO 65                                           COR 0100
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C               * * * INITIAL CONDITIONS * * *
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF (IBC .GT. 0) GO TO 35                                           COR 0110
      DO 44 N = 1,NCORE                                                  COR 0120
      NWSUBN = NW(N)                                                     COR 0130
      DO 43 M = 1,NWSUBN                                                 COR 0140
      KO = KC(N,M)                                                       COR 0150
      KO1 = K1C(N,M)                                                     COR 0160
      KO2 = K2C(N,M)                                                     COR 0170
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE CONTRIBUTIONS TO H AND T ARRAYS (INITIAL CONDITIONS, EVERY
C     ITERATION)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      GW = 1.0/RW(N,M)                                                   COR 0180
      IF (NS(N,M).GT.1.0) GW = GW / NS(N,M)                              COR 0190
      IF (NP(N,M).GT.1.0) GW = GW * NP(N,M)                              COR 0200
      GO TO (40,41,42), KO                                               COR 0210
   40 H(KO1,KO1) = H(KO1,KO1) + GW                                       COR 0220
      H(KO1,KO2) = H(KO1,KO2) - GW                                       COR 0230
      H(KO2,KO1) = H(KO2,KO1) - GW                                       COR 0240
      H(KO2,KO2) = H(KO2,KO2) + GW                                       COR 0250
      GO TO 743                                                          COR 0260
   41 H(KO1,KO1) = H(KO1,KO1) + GW                                       COR 0270
      T(KO1) = T(KO1) + EI(KO2) * GW                                     COR 0280
      GO TO 743                                                          COR 0290
   42 H(KO2,KO2) = H(KO2,KO2) + GW                                       COR 0300
      T(KO2) = T(KO2) + EI(KO1) * GW                                     COR 0310
  743 IF (MONCOR) WRITE (6,776) H(KO1,KO1),H(KO1,KO2),H(KO2,KO1),        COR 0320
     2      H(KO2,KO2),T(KO1),T(KO2)                                    COR 0330
   43 CONTINUE                                                           COR 0340
   44 CONTINUE                                                           COR 0350
      RETURN                                                             COR 0360
   35 IF (ABC  .GT. 0.0) RETURN                                          COR 0370
      IF (JKKJ .GT. 0) RETURN                                            COR 0380
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE WINDING CURRENTS (INITIAL CONDITIONS, AFTER ACHIEVING
C     CONVERGENCE)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DO 55 N = 1,NCORE                                                  COR 0390
      NWSUBN = NW(N)                                                     COR 0400
```

```
      DO 54 M = 1,NWSUBN                                                    COR 0410
      KO = KC(N,M)                                                          COR 0420
      KO1 = K1C(N,M)                                                        COR 0430
      KO2 = K2C(N,M)                                                        COR 0440
      GO TO (46,47,48) ,KO                                                  COR 0450
   46 CICORE(N,M)=(V(KO1) - V(KO2))/RW(N,M)                                 COR 0460
      GO TO 49                                                              COR 0470
   47 CICORE(N,M)=(V(KO1)-EI(KO2)) / RW(N,M)                                COR 0480
      GO TO 49                                                              COR 0490
   48 CICORE(N,M) = (EI(KO1) - V(KO2)) / RW(N,M)                            COR 0500
   49 IF (NP(N,M).GT.1.0) CICORE(N,M) = CICORE(N,M) / NP(N,M)               COR 0510
      IF (MONCOR) WRITE (6,777) CICORE(N,M)                                 COR 0520
   54 CONTINUE                                                              COR 0530
   55 CONTINUE                                                              COR 0540
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE AND SET INITIAL VALUES OF F, PHIDOT, AND PHI AT PRESENT
C     AND PREVIOUS DELTS. THIS PORTION (26 TO 30) IS ALSO ENTERED
C     INITIALLY FROM THE MAIN PROGRAM IF NG = 0 OR NG = 1
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   26 DO 30 N =1,NCORE                                                      COR 0550
      FMM = 0.0                                                             COR 0560
      NWSUBN = NW(N)                                                        COR 0570
      DO 28 M = 1,NWSUBN                                                    COR 0580
      CIJM1(N,M) = CICORE(N,M)                                              COR 0590
      FMM = FMM + XNT(N,M) * CICORE(N,M)                                    COR 0600
      IF (MONCOR) WRITE (6,774) CIJM1(N,M),FMM                              COR 0610
   28 CONTINUE                                                              COR 0620
      IF (NWSUBN .GT. 9) GO TO 29                                           COR 0630
      J = NWSUBN + 1                                                        COR 0640
      DO 407 M = J, 10                                                      COR 0650
      CIJM1(N,M) = 0.0                                                      COR 0660
      CICORE(N,M) = 0.0                                                     COR 0670
  407 CONTINUE                                                              COR 0680
   29 F(N) = FMM                                                            COR 0690
      FM1(N) = FMM                                                          COR 0700
      FM2(N) = FMM                                                          COR 0710
      PHIDOT(N) = 0.0                                                       COR 0720
      PHDTM1(N) = 0.0                                                       COR 0730
      IF (NSVPHI .EQ. 0) PHI(N) = PHIO(N)                                   COR 0740
      PHI(N) = PHI(N)+P1(N)*FMM*ALOG((ABS(FMM)+    P2(N)    )             COR 0750
     2                               /(ABS(FMM)+    P3(N)    ))            COR 0760
      PHIM1(N) = PHI(N)                                                     COR 0770
      PHIM2(N) = PHI(N)                                                     COR 0780
      IF (MONCOR) WRITE (6,775) NWSUBN,PHI(N)                               COR 0790
   30 CONTINUE                                                              COR 0800
      RETURN                                                                COR 0810
   65 IF(IRC .GT. 0) GO TO 85                                               COR 0820
      IF (JSJ .EQ. 0) GO TO 26                                              COR 0830
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     * * PART (1) - COMPUTE CONTRIBUTIONS TO H AND T ARRAYS * *
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      FLXSW = 0.0                                                           COR 0840
      IF (ABC .GT.0.0)GO TO 8009                                            COR 0850
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE DELT-DEPENDENT TERMS AT THE FIRST ITERATION
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      AITKEN = .TRUE.                                                       COR 0860
      TM1 = -DELT                                                           COR 0870
      TM2 = TM1 - DTM1                                                      COR 0880
      TM3 = TM2 - DTM2                                                      COR 0890
      SUMT = TM1 + TM2 + TM3                                                COR 0900
      DELTAV = SUMT / 3.0                                                   COR 0910
      IF (DELT .LT. 1.0E-10) DELTAV = -TE/NDELT                             COR 0920
      SUMTT = TM1**2 + TM2**2 + TM3**2                                      COR 0930
      SMTSMT = SUMT**2                                                      COR 0940
      SUMTSQ = -3.0*SUMTT + SMTSMT                                          COR 0950
```

```
      IF (MONCOR) WRITE (6,8765) TM1,TM2,TM3,SUMT,SUMTT,SMTSMT,SUMTSQ   COR 0960
 8009 DO 80 N=1,NCORE                                                    COR 0970
      IF (MONCOR) WRITE (6,1001) N                                       COR 0980
      NWSUBN = NW(N)                                                     COR 0990
      IF(ABC .GT. 0.0) GO TO 68                                          COR 1000
      IF (MONCOR) WRITE (6,800) FM3(N),FM2(N),FM1(N),PHDTM2(N),PHDTM1(N)COR 1010
     2                  ,PHIM3(N),PHIM2(N),PHIM1(N),PDTEM1(N),          COR 1020
     3                  (CICOR1(N,J),J=1,NWSUBN)                        COR 1030
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE FDOTAV AND INITIAL GUESSES FOR F AND PHI
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      SUMF = FM1(N) + FM2(N) + FM3(N)                                    COR 1040
      SUMFT = FM1(N)*TM1 + FM2(N)*TM2 + FM3(N)*TM3                       COR 1050
      SMFTN = -3.0*SUMFT + SUMF*SUMT                                     COR 1060
      FDOTAV    = SMFTN / SUMTSQ                                         COR 1070
      F(N) = (SUMF - SUMT*FDOTAV    ) / 3.0                              COR 1080
      PHI(N)=PHIM1(N)+(DELT/DTM1)*(PHIM2(N)-PHIM1(N))+2.0*DELT*PHDTM1(N)COR 1090
      IF (MONCOR) WRITE (6,801) SUMF,SUMFT,SMFTN,FDOTAV    ,F(N),PHI(N)  COR 1100
      DO 4450 M=1,NWSUBN                                                 COR 1110
      CIJM2(N,M) = CICOR1(N,M)                                           COR 1120
      CIJM1(N,M) = CICOR1(N,M)                                           COR 1130
 4450 CONTINUE                                                           COR 1140
      GO TO 70                                                           COR 1150
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     UPDATE ITERATIVE WINDING CURRENTS IF NITER IS GREATER THAN 2
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   68 DO 69 M = 1,NWSUBN                                                 COR 1160
      CIJM2(N,M) = CIJM1(N,M)                                            COR 1170
      CIJM1(N,M) = CICORE(N,M)                                           COR 1180
   69 CONTINUE                                                           COR 1190
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     IF F IS NEGATIVE, CHANGE SIGNS OF F AND PHI
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   70 SIGNF = 1.0                                                        COR 1200
      IF (F(N) .LT. 0.0) SIGNF = -1.0                                    COR 1210
      SF = SIGNF*F(N)                                                    COR 1220
      SPHI = SIGNF*PHI(N)                                                COR 1230
      IF (MONCOR) WRITE (6,792) NITER,SIGNF,SF,SPHI                      COR 1240
      IF (LASTIC .LE. 0) GO TO 311                                       COR 1250
      IF (NAW .LT. 4) GO TO 8000                                         COR 1260
          LASTIC = 0                                                     COR 1270
          WRITE (6,8001)                                                 COR 1280
          GO TO 311                                                      COR 1290
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE PHDTE(ELASTIC PHIDOT) AND PHDTEP(ELASTIC PHIDOT PRIME)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 8000 IF (NITER .EQ. 1) GO TO 8002                                       COR 1300
      IF (NDELT  .GE. 3) GO TO 311                                       COR 1310
 8002 S3 = 1.0 / (SF + P3(N))                                            COR 1320
      S2 = 1.0 / (SF + P2(N))                                            COR 1330
      Q6 = ALOG(S3/S2) * P1(N)                                           COR 1340
      EPS = P1(N) *  SF * (S2-S3) + Q6                                   COR 1350
      EPSP=SIGNF*P1(N)*(S2-S3)* (2.0 - SF * (S2+S3))                     COR 1360
      IF (NDELT .GE. 3) GO TO 8003                                       COR 1370
      IF (NSVVAR .EQ. 1) GO TO 8003                                      COR 1380
      IF (KONT .EQ. 1) GO TO 8003                                        COR 1390
      PHDTE(N)=(F(N)-FM1(N))* EPS/DELT                                   COR 1400
      PHDTEP(N) = (EPS + EPSP*(F(N)-FM1(N) ) )/DELT                      COR 1410
      GO TO 8010                                                         COR 1420
 8003 PHDTEP(N) = 0.0                                                    COR 1430
      DLPDTE    = ABS(PDTEM1(N)) + ZV(N)                                 COR 1440
      PHDTE(N) = EPS*FDOTAV                                              COR 1450
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK AND LIMIT PHDTE WITHIN BOUNDS
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      Q = PDTEM1(N) + DLPDTE                                             COR 1460
```

```
      IF (PHDTE(N)-Q) 8005,8008,8007                                    COR 1470
 8005 Q = PDTEM1(N) - DLPDTE                                            COR 1480
      IF (PHDTE(N)-Q) 8007,8008,8008                                    COR 1490
 8007 PHDTE(N) = Q                                                      COR 1500
      GO TO 8010                                                        COR 1510
 8008 PHDTEP(N)=EPSP*FDDTAV   - EPS/DELTAV                              COR 1520
 8010 IF (MONCOR) WRITE (6,8006) EPS,EPSP,FDDTAV   ,PHDTEP(N),PHDTE(N)  COR 1530
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE STATIC PHI(F) AND PHIDPRIME(F)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  311 IF (SF .GT. FD1(N)) GO TO 312                                     COR 1540
      Q1 = P1(N)*ALOG((SF - P2(N))/(SF - P3(N)))                        COR 1550
      PHID(N) = Q1*SF - PHIR(N)                                         COR 1560
      PHIDP = Q1 + P4 (N) * SF / ((SF-P2(N))*(SF-P3(N)))                COR 1570
      GO TO 316                                                         COR 1580
  312 IF (SF .GT. FD2(N)) GO TO 313                                     COR 1590
      Q2 = P8(N) * (SF-FD1(N)) ** P9(N)                                 COR 1600
      PHID(N) = (P6(N)+Q2) * (SF-FD1(N)) + P5(N)                        COR 1610
      PHIDP = P6(N) + P7(N) * Q2                                        COR 1620
      GO TO 316                                                         COR 1630
  313 IF (SF .GT. FD3(N)) GO TO 314                                     COR 1640
      PHID(N) = P10(N)*SF - P11(N)                                      COR 1650
      PHIDP =P10(N)                                                     COR 1660
      GO TO 316                                                         COR 1670
  314 IF (SF .GT. FD4(N)) GO TO 315                                     COR 1680
      Q4 =P13(N) * ALOG(P14(N)* SF / (SF-P15(N)))                       COR 1690
      PHID(N) = (P12(N)+Q4) * SF - P16(N)                               COR 1700
      PHIDP = P12(N)+ Q4 - P19(N) / (SF-P15(N))                         COR 1710
      GO TO 316                                                         COR 1720
  315 Q5 =P13(N)*ALOG((SF -P17(N))/(SF - P15(N)))                       COR 1730
      PHID(N) = Q5*SF - P18(N)                                          COR 1740
      PHIDP = Q5 + P20(N) * SF / ((SF-P15(N))*(SF-P17(N)))              COR 1750
  316 IF (MONCOR)WRITE(6,791) Q1,Q2,Q4,Q5,Q6,PHID(N),PHIDP              COR 1760
      IF(SPHI .GE. PHID(N)) GO TO 324                                   COR 1770
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE PHDTP (PEAK INELASTIC PHIDOT) AND PHDTPP (PHDTP PRIME)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  401 IF(SF .GT. FD1(N)) GO TO 402                                      COR 1780
      PHDTP = 0.0                                                       COR 1790
      PHDTPP = 0.0                                                      COR 1800
      GO TO 322                                                         COR 1810
  402 IF (SF .GT. FDB(N)) GO TO 403                                     COR 1820
      PHDTP = XLMDAD(N) * (SF-FD1(N)) **XNUD(N)                         COR 1830
      PHDTPP = PHDTP * XNUD(N) / (SF-FD1(N))                            COR 1840
      GO TO 322                                                         COR 1850
  403 IF(SF .GT. FB(N)) GO TO 404                                       COR 1860
      PHDTP = XLAMDA(N)*(SF-FOPP(N))**XNU(N)                            COR 1870
      PHDTPP = PHDTP * XNU(N) / (SF - FOPP(N))                          COR 1880
      GO TO 322                                                         COR 1890
  404 IF (SF .GT. FB1(N)) GO TO 405                                     COR 1900
      PHDTP = ROP(N)*(SF - FO(N))                                       COR 1910
      PHDTPP = ROP(N)                                                   COR 1920
      GO TO 322                                                         COR 1930
  405 PHDTP = ROP1(N) * (SF-FO1(N))                                     COR 1940
      PHDTPP = ROP1(N)                                                  COR 1950
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE PHDTI(INELASTIC PHIDOT) AND PHDTIP(INELASTIC PHIDOTPRIME)
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  322 ETA = 1.0 - ((SPHI+SPHI+PHIS(N)-PHID(N)) / (PHIS(N)+PHID(N))) ** 2COR 1960
      PHDTI = SIGNF * PHDTP * ETA                                       COR 1970
      PHDTIP = ETA*PHDTPP+4.0*PHDTP*(SPHI+SPHI+PHIS(N)-PHID(N))*(SPHI+  COR 1980
     2     PHIS(N))*PHIDP/(PHIS(N)+PHID(N))**3                          COR 1990
      GO TO 325                                                         COR 2000
  324 PHDTI  = 0.0                                                      COR 2010
      PHDTIP = 0.0                                                      COR 2020
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
C     COMPUTE PHIDOT, PHDTPR, AND PHI
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  325 PHIDOT(N)=PHDTE(N)+ PHDTI                                        COR 2030
      PHDTPR = PHDTEP(N) + PHDTIP                                      COR 2040
      SPHI=SIGNF*(PHIM1(N)+0.5*DELT*(PHDTM1(N)+PHIDOT(N)))             COR 2050
      IF(SPHI .LE. PHID(N)) GO TO 415                                  COR 2060
      IF (LASTIC .LE. 0) Q6 = P1(N) * ALOG((SF+P2(N))/(SF+P3(N)))      COR 2070
      PHIDS=PHIR(N) + SF*Q6                                            COR 2080
      IF (PHIDS .LT. PHID(N)) PHIDS = PHID(N)                          COR 2090
      IF (SPHI .GT. PHIDS) SPHI = PHIDS                                COR 2100
  415 PHI(N) = SIGNF*SPHI                                              COR 2110
      IF (MONCOR) WRITE (6,790) PHDTP,PHDTPP,ETA,PHDTI,PHDTIP,PHIDOT(N),COR 2120
     2    PHDTPR,SPHI,PHIDS,PHI(N)                                     COR 2130
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE CONTRIBUTIONS TO H AND T ARRAYS OF EACH WINDING
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      DO 76 M = 1,NWSUBN                                               COR 2140
      GCNM    = 1.0/(XNT(N,M)**2 *PHDTPR + RW(N,M))                    COR 2150
      VCNM    = XNT(N,M)*PHIDOT(N) - XNT(N,M)**2 * PHDTPR * CIJM1(N,M) COR 2160
      IF (NS(N,M) .LE. 1.0) GO TO 7750                                 COR 2170
      GCNM = GCNM / NS(N,M)                                            COR 2180
      VCNM = VCNM * NS(N,M)                                            COR 2190
 7750 IF (NP(N,M) .GT. 1.0) GCNM = GCNM * NP(N,M)                      COR 2200
      KO = KC(N,M)                                                     COR 2210
      KO1 = K1C(N,M)                                                   COR 2220
      KO2 = K2C(N,M)                                                   COR 2230
      IF (MONCOR) WRITE (6,779) GCNM,VCNM,KO,KO1,KO2                   COR 2240
      GC(N,M) = GCNM                                                   COR 2250
      VC(N,M) = VCNM                                                   COR 2260
      GO TO (72,73,74) ,KO                                             COR 2270
   72 H(KO1,KO1) = H(KO1,KO1) + GCNM                                   COR 2280
      H(KO1,KO2) = H(KO1,KO2) - GCNM                                   COR 2290
      GCVC = GCNM * VCNM                                               COR 2300
      T(KO1) = T(KO1) + GCVC                                           COR 2310
      H(KO2,KO1) = H(KO2,KO1) - GCNM                                   COR 2320
      H(KO2,KO2) = H(KO2,KO2) + GCNM                                   COR 2330
      T(KO2) = T(KO2) - GCVC                                           COR 2340
      GO TO 769                                                        COR 2350
   73 H(KO1,KO1) = H(KO1,KO1) + GCNM                                   COR 2360
      T(KO1) = T(KO1) + GCNM    *(VCNM     + EI(KO2))                  COR 2370
      GO TO 769                                                        COR 2380
   74 H(KO2,KO2) = H(KO2,KO2) + GCNM                                   COR 2390
      T(KO2) = T(KO2) + GCNM    *(-VCNM    + EI(KO1))                  COR 2400
  769 IF (MONCOR) WRITE (6,776) H(KO1,KO1),H(KO1,KO2),H(KO2,KO1),      COR 2410
     2     H(KO2,KO2),T(KO1),T(KO2)                                    COR 2420
   76 CONTINUE                                                         COR 2430
      IF(ABS (PHDTI) .GT. ZV(N)) FLXSW = 1.0                           COR 2440
   80 CONTINUE                                                         COR 2450
      RETURN                                                           COR 2460
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     * * PART (2) - COMPUTE WINDING CURRENTS AND TEST FOR CONVERGENCE *
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   85 AITKEN = .NOT. AITKEN                                            COR 2470
      DO 100 N=1,NCORE                                                 COR 2480
      IF (MONCOR) WRITE (6,1001) N                                     COR 2490
      NWSUBN = NW(N)                                                   COR 2500
      F(N) = 0.0                                                       COR 2510
      DO 98 M = 1,NWSUBN                                               COR 2520
      KO = KC(N,M)                                                     COR 2530
      KO1 = K1C(N,M)                                                   COR 2540
      KO2 = K2C(N,M)                                                   COR 2550
      GO TO (87,88,89),KO                                              COR 2560
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     COMPUTE WINDING VOLTAGES
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
   87  BV = V(KO1) -V(KO2)                                                  COR 2570
       GO TO 90                                                             COR 2580
   88  BV = V(KO1) - EI(KO2)                                                COR 2590
       GO TO 90                                                             COR 2600
   89  BV = EI(KO1) - V(KO2)                                                COR 2610
C      - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C      COMPUTE WINDING CURRENT CIW AND MODIFY ACCORDING TO AITKIN FORMULA
C          EVERY OTHER ITERATION (OR HALF-FACTOR FORMULA EVERY ITERATION)
C      IF CIW IS OSCILLATORY
C      - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   90  CIW     = GC(N,M) * (BV - VC(N,M))                                   COR 2620
       IF (NP(N,M).GT.1.0)  CIW     = CIW     / NP(N,M)                     COR 2630
       IF (MONCOR) WRITE(6,795) BV,CIW,CIJM1(N,M),CIJM2(N,M)                COR 2640
       IF (CIJM1(N,M) .EQ. CIJM2(N,M) )  GO TO 95                           COR 2650
       IF (PSTEP .LT. 0.0) GO TO 93                                         COR 2660
       IF (AITKEN) GO TO 95                                                 COR 2670
       IF ((CIW-CIJM1(N,M))*(CIJM1(N,M)-CIJM2(N,M)) .LT. 0.0) CIW = CIW     COR 2680
      2   - (CIW-CIJM1(N,M))**2 / (CIW-2.0*CIJM1(N,M)+CIJM2(N,M))           COR 2690
       GO TO 94                                                             COR 2700
   93  IF ((CIW-CIJM1(N,M))*(CIJM1(N,M)-CIJM2(N,M)) .LT. 0.0) CIW =         COR 2710
     10.5 * (CIJM1(N,M) + CIW)                                              COR 2720
   94  IF (MONCOR) WRITE(6,796) CIW                                         COR 2730
C      - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C      TEST FOR CONVERGENCE OF EACH WINDING CURRENT
C      - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
   95  IF (CIW .EQ. 0.0) GO TO 110                                          COR 2740
       IF (ABS(1.0-CIJM1(N,M) /CIW).GT. (RELERR+ABSERR/ABS(CIW)))GO TO 97   COR 2750
       CICORE(N,M) = ABS(PSTEP)*CICORE(N,M) + (1.0-ABS(PSTEP))*CIW          COR 2760
       GO TO 114                                                            COR 2770
  110  IF ((CIJM1(N,M).EQ.0.0).AND.(CIJM2(N,M).EQ.0.0)) GO TO 112           COR 2780
   97  ABC = 1.0                                                            COR 2790
       IF (MONCOR) WRITE (6,9998)                                           COR 2800
  112  CICORE(N,M) = CIW                                                    COR 2810
C      - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C      COMPUTE MMF FOR THE NEXT ITERATION
C      - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  114  F(N) = F(N) + XNT(N,M) * CICORE(N,M)                                 COR 2820
       IF (MONCOR) WRITE (6,797)N,M,KO1,KO2,CIW,CICORE(N,M),F(N),ABC        COR 2830
   98  CONTINUE                                                             COR 2840
C      - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C      PREVENT STOP- SWITCHING EXIT BEFORE INELASTIC SWITCHING STARTS
C      - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
       IF (ABS (F(N)) .GT. FDI(N)) NOEXIT = 0                               COR 2850
  100  CONTINUE                                                             COR 2860
       IF (ABC         .GT. 0.0) RETURN                                     COR 2870
C      - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C      AFTER CONVERGENCE IS ACHIEVED, UPDATE THE VALUES OF F,PHIDOT, PHI,
C      AND WINDING CURRENTS OF PREVIOUS DELTS
C      - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
       DO 772 N = 1,NCORE                                                   COR 2880
       FM3(N) = FM2(N)                                                      COR 2890
       FM2(N) = FM1(N)                                                      COR 2900
       FM1(N) = F(N)                                                        COR 2910
       PHDTM2(N) = PHDTM1(N)                                                COR 2920
       PHDTM1(N) = PHIDOT(N)                                                COR 2930
       PDTFM1(N) = PHDTE(N)                                                 COR 2940
       PHIM3(N) = PHIM2(N)                                                  COR 2950
       PHIM2(N) = PHIM1(N)                                                  COR 2960
       PHIM1(N) = PHI(N)                                                    COR 2970
       NWSUBN = NW(N)                                                       COR 2980
       DO 2326 M=1,NWSUBN                                                   COR 2990
  2326 CICOR1(N,M) = CICORE(N,M)                                            COR 3000
   772 CONTINUE                                                             COR 3010
```

```
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
C     CHECK IF THE RUN SHOULD TERMINATE DUE TO STOP-SWITCHING EXIT
C     - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      IF(STSWEX       .LE. 0.0) RETURN                                  COR 3020
      IF(FLXSW        .NE. 0.0) RETURN                                  COR 3030
      IF (TE .LE. (TIMEIN+RNUDT*DELT)) RETURN                           COR 3040
      IF (NOFXIT      .EQ.   1) RETURN                                  COR 3050
      WRITE             (6,530)                                         COR 3060
      JCAT = 10                                                         COR 3070
      RETURN                                                            COR 3080
C  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .  .
  530 FORMAT(1HO,10X,19HSTOP-SWITCHING EXIT)                            COR 3090
  774 FORMAT (16H CIJM1(N,M),FMM   ,2E22.15)                            COR 3100
  775 FORMAT (16H NWSUBN,PHI(N)       ,I12,2E22.15)                     COR 3110
  776 FORMAT (39H H(1,1),H(1,2),H(2,1),H(2,2),T(1),T(2)                 COR 3120
     1     5(/,1H ,E22.15,E22.15,E22.15,E22.15,F22.15) )                COR 3130
  777 FORMAT (13H CICORE(N,M)  ,E22.15)                                 COR 3140
  779 FORMAT (22H GCNM,VCNM,KO,KO1,KO2    ,2E24.15,3I12)                COR 3150
  790 FORMAT (75H PHDTP,PHDTPP,ETA,PHDTI,PHDTIP,PHIDOT(N),PHDTPR,SPHI,PHCOR 3160
     2IDS,PHI(N)                                                        COR 3170
     3     ,/,1H ,5E22.15,/,1H ,5E22.15,/,1H ,I12)                      COR 3180
  791 FORMAT (30H Q1,Q2,Q4,Q5,Q6,PHID,PHIDP     ,                      COR 3190
     1     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )                COR 3200
  792 FORMAT (40H         NITER,SIGNF,SF,SPHI                 , I14,    COR 3210
     1     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )                COR 3220
  795 FORMAT(32H BV,CIW,CIJM1(N,M),CIJM2(N,M)      /5E24.15)            COR 3230
  796 FORMAT (13H CIW(AITKEN) , E24.15)                                 COR 3240
  797 FORMAT (38H N,M,KO1,KO2,CIW,CICORE(N,M),F(N),ABC    ,4I6,         COR 3250
     1     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )                COR 3260
  800 FORMAT (              42H FM3(N),FM2(N),FM1(N),PHDTM2(N),PHDTM1(N)COR 3270
     2           39H,PHIM3(N),PHIM2(N),PHIM1(N),PDTEM1(N),              COR 3280
     3           25H(CICOR1(N,J),J=1,NSUBN)  ,4(/5E24.15)  )            COR 3290
  801 FORMAT (              41H SUMF,SUMFT,SMFTN,FDOTAV(N),F(N),PHI(N)  COR 3300
     2       ,2(/,5E24.15))                                             COR 3310
 1001 FORMAT (35(3H .  ),4HCORE ,I3)                                    COR 3320
 8001 FORMAT (53H ELASTIC PHIDOT SET TO ZERO AFTER THREE CONVERGENCE     COR 3330
     2     ,10HFAILURES       )                                        COR 3340
 8006 FORMAT (36H EPS,EPSP,FDOTAV(N),PHDTEP,PHDTE(N)    ,/,5E24.15)     COR 3350
 8765 FORMAT (39H TM1,TM2,TM3,SUMT,SUMTT,SMTSMT,SUMTSQ                  COR 3360
     1     5(/,1H ,E22.15,E22.15,E22.15,E22.15,E22.15) )                COR 3370
 9998 FORMAT (18H DID NOT CONVERGE    )                                 COR 3380
 9999 FORMAT (13H CORES          )                                      COR 3390
      END                                                               COR 3400
```

INDEX*

---

* Unlabeled page numbers correspond to Volume I. Page numbers preceded by the label "II." correspond to Volume II.