SAND77-0939
Unlimited Release

3151

MASTER

AITRAC
Augmented Interactive Transient Radiation Analysis by Computer

User's Information Manual

Berne Electronics, Inc.
and
Sandia Laboratories, Albuquerque

Prepared by Sandia Laboratories, Albuquerque, New Mexico 87115 and Livermore, California 94550 for the United States Department of Energy under Contract AT(29-1)-789

Printed October 1977

Sandia Laboratories

SF 2900 Q(7-73)

DISCLAIMER

Issued by Sandia Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE

Printed in the United States of America

Available from
National Technical Information Service
U. S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161

Price: Printed Copy $9.50; Microfiche $3.00

SAND77-0939
Unlimited Release
Printed October 1977

AITRAC:
AUGMENTED INTERACTIVE TRANSIENT RADIATION ANALYSIS BY COMPUTER
USER'S INFORMATION MANUAL

A CAD Program by
Berne Electronics, Inc.
and
Systems Division 1132
Sandia Laboratories
Albuquerque, NM 87115

Sandia NOS Update

FOREWORD

Adapted from the TRAC and SPICE programs by BERNE ELECTRONICS, Inc., AITRAC: Augmented Interactive Transient Radiation Analysis by Computer; performs DC, Transient, and radiation analyses of electronic circuits.

The program operates in conversational time-sharing mode on all large-scale computers which can provide this interactive environment.

This manual instructs users on how to perform all AITRAC analyses. Specifically, it describes program features, how to access the time-sharing system, how to prepare data, how to operate the program, and how to interpret the output results.

Berne Electronics maintains and updates this program and is always available to users for help, technical questions, and consulting. Comments on how to improve the documentation and the program are solicited, and will be welcomed.

CONTENTS

CONTENTS
(continued)

CONTENTS
(continued)

CONTENTS
(continued)

1. Introduction

AITRAC is a program designed for on-line, interactive, DC, and transient analysis of electronic circuits. The program solves linear and nonlinear simultaneous equations which characterize the mathematical models used to predict circuit response. The program features:

* 100 external node - 200 branch capability
* Conversational, free-format input language
* Built-in junction, FET, MOS and switch models
* Sparse matrix algorithm with extended-precision H matrix and T vector calculations, for fast and accurate execution
* Linear transconductances: beta, GM, MU, ZM
* Accurate and fast radiation effects analysis
* Special interface for user-defined equations
* Selective control of multiple outputs
* Graphical outputs in wide and narrow formats
* On-line parameter modification capability

The user describes the problem by entering the circuit topology and part parameters as they are requested by the program. The program then automatically generates and solves the circuit equations, providing the user with printed or plotted output. The circuit topology and/or part values may then be changed by the user, and a new analysis requested. Circuit descriptions may be saved on disk files for storage and later use.

The program contains built-in standard models for: resistors, voltage and current sources, capacitors, inductors including mutual couplings, switches, junction diodes and transistors, FETS, and MOS devices. Nonstandard models may be constructed from standard models or by using the special equations interface. Time functions may be described by straight-line segments or by sine, damped sine, and exponential functions. Initial conditions may be input by the user, calculated by the program, or input from data files stored on disk.

Some of the internal models and algorithms used by AITRAC have been adapted from the TRAC and SPICE programs, References 1 and 2.

2. Program Description and Features

2.1 Standard Part Models

The program contains standard mathematical models which may be used to construct an analog of the circuit under investigation. These models are recognized by the program, the corresponding equations are automatically written, and the solution is computed. The various models are discussed and illustrated in the following sections.

2.1.1 Resistor

The resistor model is shown in Figure II-1 with current and voltage conventions. The F and T node designations are arbitrary and determine only the positive sign of the calculated branch current; they do not affect the solution process. Positive current flow is from the F node to the T node.

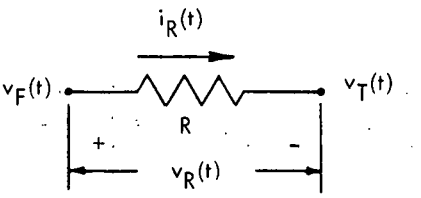


Figure II-1. Resistor Model and Conventions

The resistor model is defined by the current equation:

$$i_R(t) = \frac{v_R(t)}{R} \quad ; \quad p_R(t) = v_R(t) * i_R(t) \quad .$$

The instantaneous model branch current and branch power are calculated by the program and are available for output.

2.1.2 Capacitor

The capacitor is shown in Figure II-2 with current and voltage conventions. Again, the F and T node designations are arbitrary and determine only the positive flow of the calculated branch current; they do not affect the solution process. Positive current flow is from the F node to the T node.

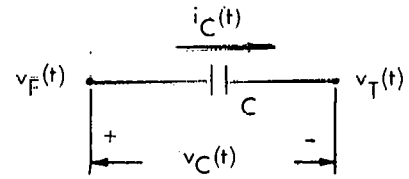
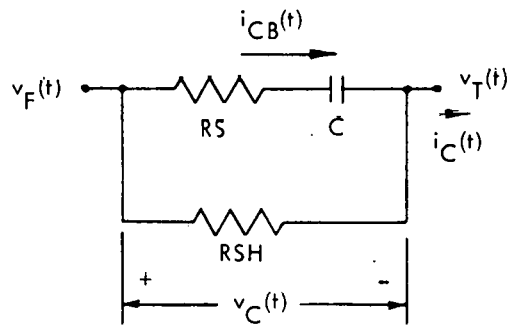


Figure II-2. Capacitor with Program Conventions

The capacitor model is shown in Figure II-3, and is defined by the following current equations:

$$i_C(t) = i_{CB}(t) + \frac{v_C(t)}{RSH} ; \qquad i_{CB}(t) = C\left[\frac{d[v_C(t) - i_{CB}(t)\,RS]}{dt}\right] ,$$

where RS is the equivalent series resistance and RSH is the leakage resistance. The instantaneous model branch current and branch power are calculated and available for output. Note that the branch current does NOT include the current in RSH.



$$p_C(t) = i_C(t) * v_C(t)$$

Figure II-3. Capacitor Model

2.1.3 Inductor

The inductor is shown in Figure II-4 with the current and voltage conventions. The F and T node designations are arbitrary and determine only the positive flow of the calculated branch current; they do not affect the solution process. Positive current flow is from the F node to the T node.

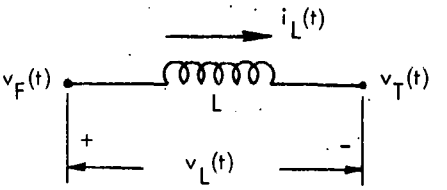


Figure II-4. Inductor with Program Conventions

The inductor model is shown in Figure II-5, and is defined by the following current equations:

$$i_L(t) = i_{LB}(t) + \frac{v_L(t)}{RSH} \quad ; \quad p_L(t) = i_L(t) * v_L(t)$$

and

$$i_{LB}(t) = \frac{1}{L} \int_o^t \left[v_L(t) - i_{LB}(t)\ RS \right] dt + i_{LB}(0) \quad ,$$

where RS is the series (DC) resistance and RSH is the leakage resistance. The instantaneous model branch current and branch power are calculated and available for output. Note that the branch current does NOT include the current in RSH.

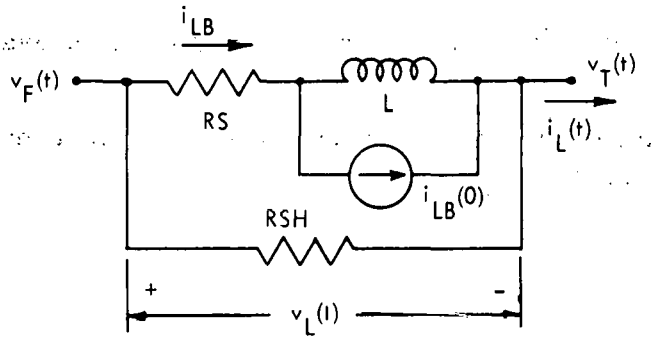


Figure II-5. Inductor Model

2.1.4 Grounded Voltage Source

The grounded voltage source model is shown in Figure II-6 with program conventions.

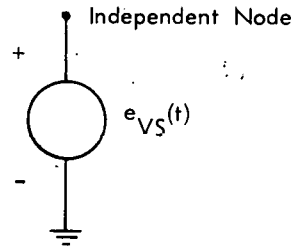


Figure II-6. Grounded Voltage Source Model with Conventions

An independent node is a node which is either ground, $v(t) = 0$, or connected to a grounded voltage source, $v(t) = e_{vs}(t)$, whose value is explicitly defined. A dependent node is a node which is not connected to ground or a grounded voltage source.

2.1.5 Floating Voltage Source

The floating voltage source model is shown in Figure II-7 with current and voltage conventions. The model is defined by the current equation:

$$i_{FVS}(t) = \left[v_{FVS}(t) - e_{FVS}(t)\right] / RS$$

branch power: $p_{FVS}(t) = i_{FVS}(t) * v_{FVS}(t)$.

The instantaneous model branch current and branch power are available for output.

Due to the voltage source in the branch, care should be exercised when interpreting branch power outputs.

2.1.6 Floating Current Source

The current source model is shown in Figure II-8 with current and voltage conventions.

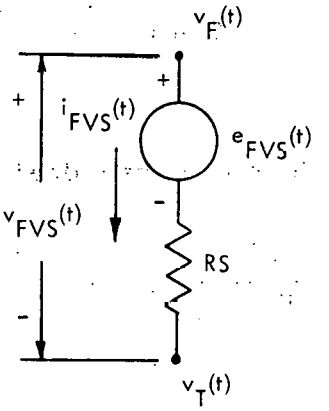


Figure II-7. Floating Voltage Source, with Conventions

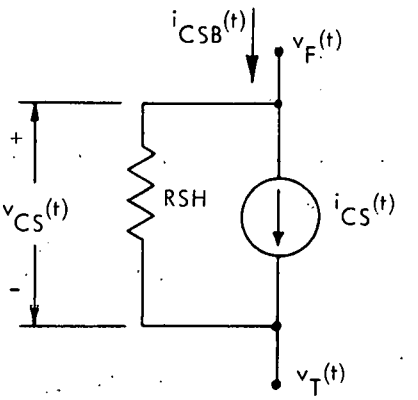


Figure II-8. Floating Current Source, with Conventions

The model is defined by the current equation:

$$i_{CSB}(t) = i_{CS}(t) + v_{CS}(t)/RSH$$

branch power: $p_{CS}(t) = i_{CS}(t) * v_{CS}(t)$.

The instantaneous model branch current and branch power are available for output. The branch current "BC" which is printed out does not include the independent current term "i_{CS}". It does however include any dependent currents which arise due to transconductance terms: GM, Beta, etc.

Caution is to be used when interpreting power output printouts, due to the current source in the model and the polarity conventions.

NOTE: The sign conventions of floating sources are such that the voltage drop from F to T is taken as positive.

2.1.7 Junction Diode

The junction diode is shown in Figure II-9 with current and voltage conventions.

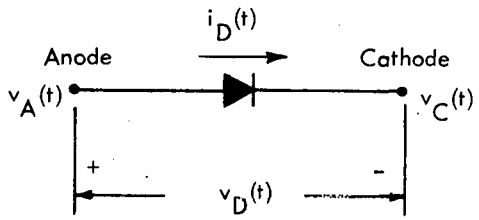


Figure II-9. Diode with Program Conventions

The diode model is shown in Figure II-10 and is defined by the following current equations:

$$i_D(t) = i_{DI}(t) + TD\,\frac{d[i_{DI}(t)]}{dt} + c_D\,\frac{d[_D(t)]}{dt} + \frac{v_D(t)}{RDL} - i_{PPD}(t)$$

$$i_{DI}(t) = IS\,(\exp[v_D(t)/MD \cdot \theta)] - 1)$$

$$c_D = \frac{CDO}{[1 - v_D(t)/VDBI]^{1/2}} \quad ; \quad v_D(t) \leq 0.9*VDBI \quad .$$

Voltages $v_A(t)$ and $v_C(t)$ are the voltages at the anode and cathode terminals, respectively, and

θ = k*TEMP/q ($\approx$.026 volts at 300 Kelvin)

k = Boltzmann's constant, 1.38054 E-23 erg/deg Kelvin

q = charge on electron, 1.6021 E-19 coulomb

TEMP = temperature in Kelvin, $T_{ambient}$ + 273.15

IS = diode reverse saturation current

MD = diode emission constant

RDL = diode leakage resistance

CDO = diode junction capacitance at $V_D = 0$

VDBI = diode intrinsic built-in voltage

TD = diode time constant

c_D = diode junction capacitance

IPPD = diode steady-state primary photocurrent

IPPD and $I_{PPD}(t)$ are parameters provided only for compatibility with versions of the program intended for radiation analysis and are omitted from the diode model when radiation effects are not to be studied.

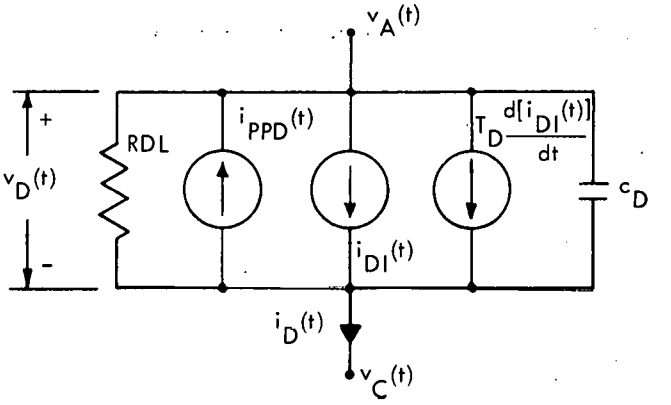


Figure II-10. Junction Diode Model

The instantaneous model branch current and branch power are calculated and are available for output. The real diode power can be calculated by the expression:

$$\text{Power} = v_D(t)\left[i_{DI}(t) - i_{IPPD}(t) + \frac{v_D(t)}{RDL}\right]$$

The model simulates the complete temperature effects, but does not model avalanche or bulk resistances, which can be added externally when required.

2.1.8 Bipolar Junction Transistor (BJT)

Bipolar transistors are shown in Figure II-11 with current and voltage conventions.

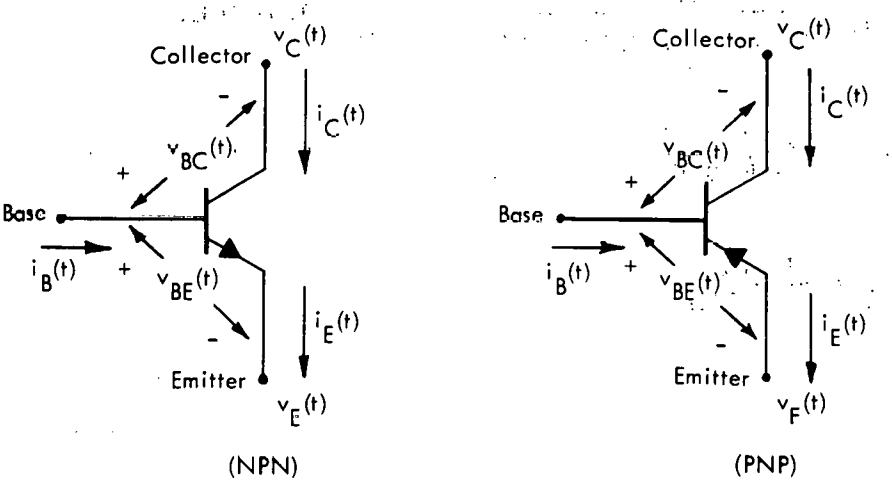


Figure II-11. Transistors with Program Conventions

The transistor models are shown in Figures II-12 and II-13. The NPN transistor model is defined by the following current equations:

$$i_C(t) = \alpha_N i_{EI}(t) - i_{CD}(t)$$

$$i_E(t) = i_{ED}(t) - \alpha_I i_{CI}(t)$$

$$i_B(t) = i_E(t) - i_C(t) \quad ,$$

where

$$i_{CD}(t) = i_{CI}(t) + TI \frac{d[i_{CI}(t)]}{dt} + c_C \frac{d[v_{BC}(t)]}{dt} + \frac{v_{BC}(t)}{RCL} - i_{PPC}(t)$$

$$i_{ED}(t) = i_{EI}(t) + TN \frac{d[i_{EI}(t)]}{dt} + c_E \frac{d[v_{BE}(t)]}{dt} + \frac{v_{BE}(t)}{REL} - i_{PPE}(t)$$

$$i_{CI}(t) = ICS(\exp [v_{BC}(t)/(MC \cdot \theta)] - 1)$$

$$i_{EI}(t) = IES(\exp [v_{BE}(t)/(ME \cdot \theta)] - 1)$$

$$c_C = \frac{CCO}{[1 - v_{BC}(t)/VCBI]^{1/2}} \quad ; \quad v_{BC} \leq 0.9*VCBI$$

$$c_E = \frac{CEO}{[1 - v_{BE}(t)/VEBI]^{1/2}} \quad ; \quad v_{BE} \leq 0.9*VEBI$$

$$\alpha_N = \frac{HFEN}{1 + HFEN}$$

$$\alpha_I = \frac{HFEI}{1 + HFEI}$$

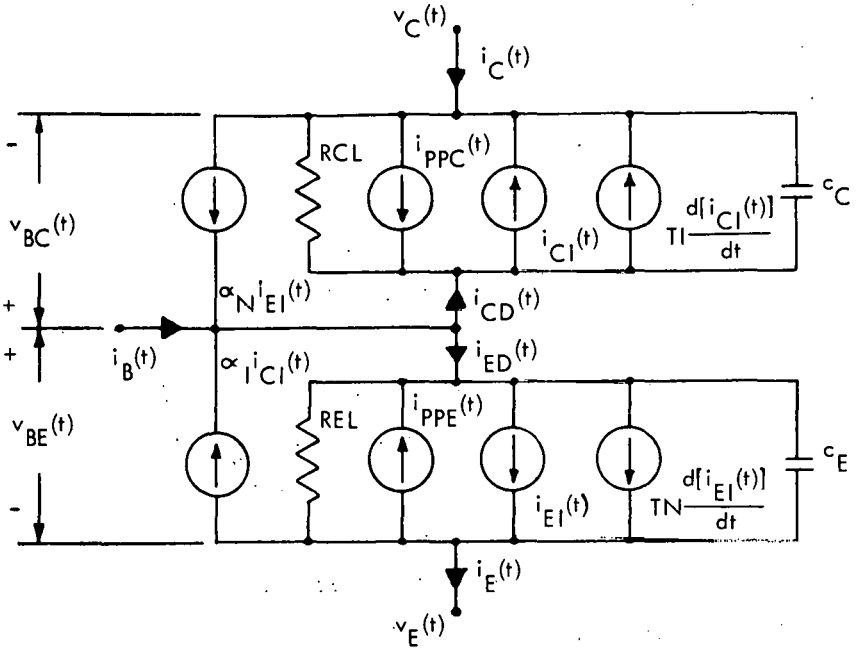


Figure II-12. NPN Bipolar Junction Transistor Model

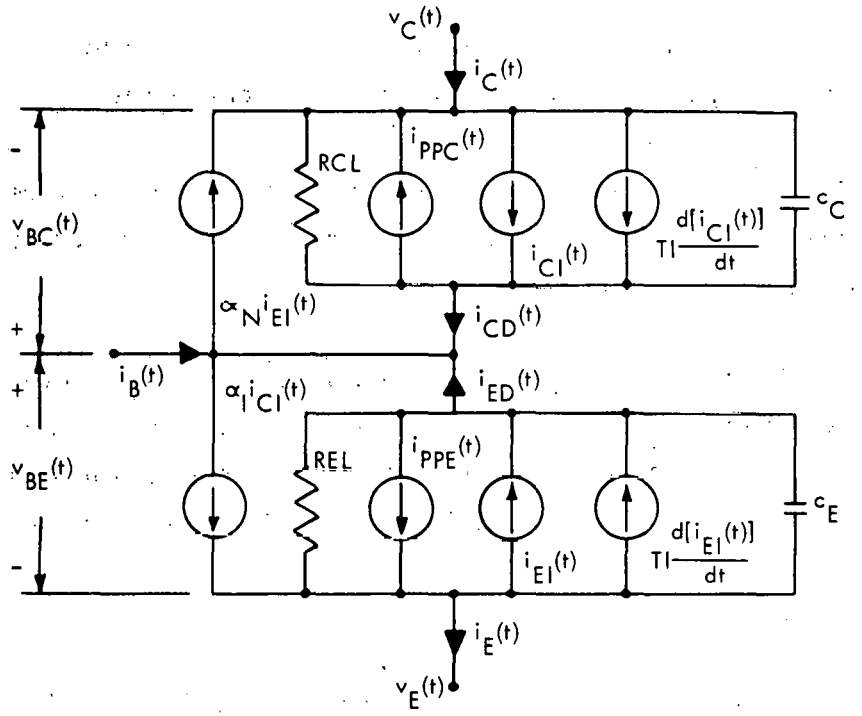


Figure II-13. PNP Bipolar Junction Transistor Model

The transistor parameters are defined as follows:

HFEN = normal transistor beta

HFEI = inverted transistor beta

α_N = normal transistor alpha

α_I = inverted transistor alpha

TN = emitter junction time constant

TI = collector junction time constant

ICS = collector junction diode saturation current

MC = collector junction diode emission constant

c_C = collector junction diode capacitance (transition)

CCO = collector junction diode capacitance at $V_{BC} = 0$

VCBI = collector junction intrinsic built-in voltage

RCL = collector junction leakage resistance

IES = emitter junction diode saturation current

ME = emitter junction diode emission constant

c_E = emitter junction diode capacitance

CEO = emitter junction diode capacitance at $V_{BE} = 0$

VEBI = emitter junction intrinsic built-in voltage

REL = emitter junction leakage resistance

IPPC = collector junction steady-state primary photocurrent

IPPE = emitter junction steady-state primary photocurrent

IPPC and IPPE are parameters provided only for compatibility with versions of the program intended for radiation analysis and are omitted from the transistor models when radiation effects are not to be studied.

Voltages $v_B(t)$, $v_C(t)$, and $v_E(t)$ are the voltages at the base, collector, and emitter terminals of the transistor. Model branch current numbers are assigned to the base and collector currents only. The instantaneous model branch currents $i_B(t)$ and $i_C(t)$ are calculated and are available for output. The power may be calculated by the expression:

$$\text{Power} = v_{BE}(t)\left[i_{EI}(t) - i_{PPE}(t) + \frac{v_{BE}(t)}{REL} - \alpha_I i_{CI}(t)\right]$$

$$+ v_{BC}(t)\left[i_{CI}(t) - i_{PPC}(t) + \frac{v_{BC}(t)}{RCL} - \alpha_N i_{EI}(t)\right] .$$

The total power dissipated in each transistor is available for printout or plotting by requesting the power in EITHER of the two branches assigned to any transistor.

For convenience, either branch contains the TOTAL device dissipation.

2.1.9 MOSFET Devices

Two different models of MOSFET devices are available, as well as a JFET model. Each can be specified either as an N-channel of P-channel device.

The "S" model MOSFET is the well-known Schichman-Hodges Model (Reference 10) used in most circuit analysis programs (see Figure II-14).

The "A" model MOSFET is useful for device manufacturers, as it allows the fabrication parameters and geometrical dimensions of the devices to be entered directly as data. The electrical performance characteristics are similar to those of the "S" model. Geometrical details are shown in Figure II-15. The actual implementation is based on the FET diode described in Reference 17. The FET diode concept allows a model to be developed which closely parallels the operation of the Ebers-Moll BJT model for all regions of operation. It also provides a convenient method for performing nonlinear iterations and for convergence checks during analysis.

The JFET model is also based on the Schichman-Hodges Model (see Figure II-16).

All models include the effect of substrate bias on threshold voltage and terms to model channel length modulation.

NOTES

1. Models for N-channel devices, with voltage polarities corresponding to operation in normal mode are shown. The user may specify both N and P channel devices, operating in either normal or inverted modes. AITRAC will automatically adjust the signs of all voltages and currents to account for either channel type and either mode of operation. The user need not be concerned about the necessary algebraic manipulations.

2. Some users may have developed models which include additional refinements and require more parameters to be entered. AITRAC has additional provisions to take care of such requirements. Please consult Berne for special modeling requirements.

2.1.10 S Model MOS

The S Type MOS device is characterized by the following 12 parameters:

NO.	NAME	PARAMETER	DEFAULT	UNITS
1	VTO	Threshold voltage	2.0	V
2	BETA	Transconductance parameter	1.0E-4	A/V^2
3	GAMM	Source-Substrate Threshold param.	0.7	$V^{1/2}$
4	PHI	Source-Substrate Offset potential	0.7	V
5	LAMB	Channel Length modulation param.	0.01	V^{-1}
6	CGS	Gate-source capacitance	1.0	pf
7	CGD	Gate-drain capacitance	1.0	pf
8	CGB	Gate-body capacitance	1.0	pf
9	CBD	Body-drain capacitance	2.5	pf
10	CBS	Body-source capacitance	2.5	pf
11	PHIB	Substrate junction potential	0.7	V
12	IS	Substrate junction saturtn. currnt.	1.0E-14	A

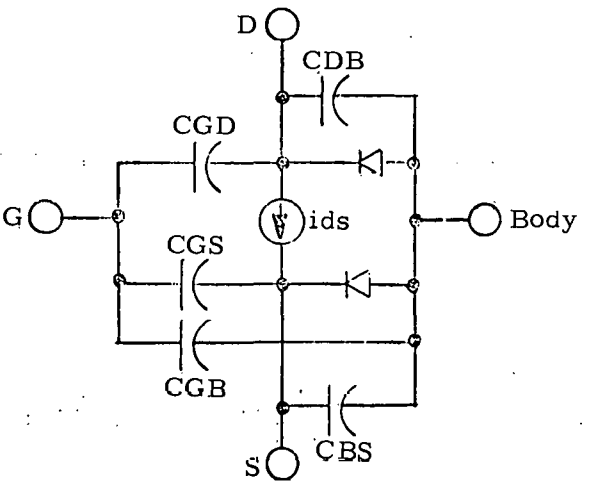


Figure II-14. S Model MOSFET

The governing equations for an N-channel device are as follows:

Forward Region: VDS > 0

$$VTE = VTO + GAMM\left[(VSB + PHI)^{0.5} + (PHI)^{0.5}\right]$$

Operating Conditions:

Cutoff:	VGST ≤ 0	or	VGS ≤ VTE
Saturated:	VGST ≤ VDS	and	VGD ≤ VTE
Triode:	VGST > VDS	or	VGS > VDS + VTE

Device Current:

$$ID = \begin{cases} 0 & VGS \le VTE \\ BETA[VGS - VTE]^2\,(1 + LAMB*VDS) & VGS \le VDS + VTE \\ BETA*VDS[2(VGS - VTE) - VDS](1 + LAMB*VDS) & VGS > VDS + VTE \end{cases}$$

In the reverse region: VDS < 0. VGD (Gate to Drain) is substituted for VGS (Gate to Source), VDB (Drain to Body) is substituted for VSB (Source to Body), and the sign of VDS (Drain to Source) is reversed in the above equations.

NOTES

1. Depending on the method of derivation or reference used, the above equations can be written in equivalent form as follows:

$$ID = \begin{cases} 0 \\ 2*BETA\left[\dfrac{VGS^2}{2} - VT*VGS + \dfrac{VT^2}{2}\right](1 + LAMB*VDS) \\ 2*BETA\left[\left(\dfrac{VGS^2}{2} - VT*VGS\right) - \left(\dfrac{VGD^2}{2} - VT*VGD\right)\right](1 + LAMB*VDS) \end{cases}.$$

2. The parameter LAMB(DA) is used to model the finite output conductance of the device in the saturated region:

$$g_d sat = BETA * LAMB\ (VGS - VTE)^2 \approx LAM\,(I_D) \quad .$$

The output conductance is assumed to vary linearly with drain current. The parameter LAMB(DΛ) is analogous to the Early Voltage (VA) used in the Gummel-Poon model of the bipolar junction transistor.

The two substrate (Body) junctions are normally reverse biased and are modeled by ideal diodes. (Note: VSB and VDB are normally positive quantities.)

$$ISB = Is\left[\exp\left(\frac{-qVSB}{KT}\right) - 1\right]$$

$$IDB = Is\left[\exp\left(\frac{-qVDB}{KT}\right) - 1\right]$$

The voltage-dependent depletion capacitances are defined by:

$$CSB = CSBO\left[1 + \frac{VSB}{PHIB}\right]^{-1/2} \quad ; \quad |VSB| < PHIB$$

$$CDB = CDBO\left[1 + \frac{VDB}{PHIB}\right]^{-1/2} \quad ; \quad |VDB| < PHIB \quad .$$

2.1.11 A Model MOS

The A Type MOS device is characterized by the following sixteen (16) parameters:

NO.	NAME	PARAMETER	DEFAULT	UNITS
1	KP	(= K') Device gain factor	1.0E-5	A/V^2
2	VTH	Threshold voltage at zero volts	0.5	V
3	SBC	Source-body coefficient	0.5	$V^{1/2}$
4	SBE	Source-body exponent	0.5	--
5	DW	Process induced increase of drawn device width	+0.1	mils†
6	DL	Process induced increase of drawn device length	-0.1	mils
7	OL	Actual gate-drain and gate-source overlap (after spreading)	0.05	mils
8	SEP	Average separation of two†† connected devices	1.0	mils
9	CGOX	Capacitance of gate oxide	0.25	pf/mil^2
10	CDIF	Capacitance of diffused region	0.1	pf/mil^2
11	DCC	Diffused capacitance coefficient	0.2	$V^{1/2}$
12	DCE	Diffused capacitance exponent	1.0	--
13	WDRN	Drawn device width	3.0	mils
14	LDRN	Drawn device length	0.3	mils
15	RO	Crimping resistor	100	ohms
16	ZED(PHI)	Source-body offset	0.7	V

†Units: Units of "mils" and "pf/mil^2" are shown. However, any self-consistent set of units may be used, such as microns, and pf/micron squared, or mm and pf/sq mm.

††SEP is the average separation of two connected devices. However, any length may be inserted for this parameter to provide the correct "pad" area.

The governing equations for an N-channel device are as follows:

$$VTE(\text{effective}) = VTH + SBC(CVS - VB + \underset{(PHI)}{ZED}^{SBE} - \underset{(PHI)}{ZED}^{SBE})$$

$$VDS = VGS - VGD \quad \text{and} \quad VGST = VGS - VT \ .$$

Operating Conditions:

Cutoff:	$VGST \le 0$	or	$VGS \le VTE$
Saturated:	$VGST \le VDS$	and	$VGD \le VTE$
Triode:	$VGST > VDS$	or	$VGS > VDS + VTE$

Device Characteristic Parameters:

$$KS = KP \frac{(WDRN + DW)}{(LDRN + DL)} \quad \text{and} \quad KU = \frac{KS}{1 + KS \cdot RO(VGD - VT)}$$

Device Current:

$$iDS = \begin{cases} 0 & \text{Cutoff} \\ \frac{KS}{2}(VGS - VTE)^2 & \text{Saturated} \\ KU\left[\left(\frac{VGS^2}{2} - VTE \cdot VGS\right) - \left(\frac{VGD^2}{2} - VTE \cdot VGD\right)\right] & \text{Triode} \end{cases}$$

The device capacitances are voltage dependent, depending on the region of operation, and are defined as follows:

$$CSB = \frac{SEP \cdot (WDRN + DW) \cdot CDIF}{1 + DCC(VSB)^{DCE}}$$

$$CDB = \frac{SEP \cdot (WDRN + DW) \cdot CDIF}{1 + DCC(VDB)^{DCE}}$$

The following capacitances are defined for notational and computational convenience:

$$CK1 = OL \cdot (WDRN + DW) \cdot CGOX$$

$$CK2 = (LDRN + OL) \cdot (WDRN + DW) \cdot CGOX$$

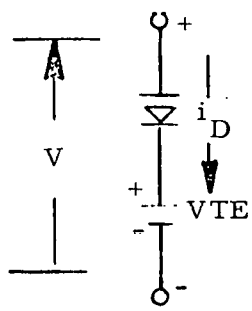
Capacitances CGB, CGS, and CGD depend on the state of the MOSFET, as follows:

MOSFET STATE	CGB	CGS	CGD
Cutoff-Normal	CK2	CK1	CK1
Cutoff-Inverted	-VGST(CK2)	CK1 + 0.7(1 - VGST)CK2	CK1 + 0.3(1 - VGST)CK2
Saturated	0	CK1 + 0.7CK2	CK1 + 0.3CK2
Triode	0	CK1 + 0.5CK2	CK1 + 0.5CK2

This model is implemented by using an FET diode, described in Reference II-11.

The use of this FET diode allows the MOS device to be modeled for all regions of operation with a single set of equations, in a manner similar to the Ebers-Moll equations applied to the bipolar junction transistor (BJT).

FET Diode Voltage-Current Relationships:



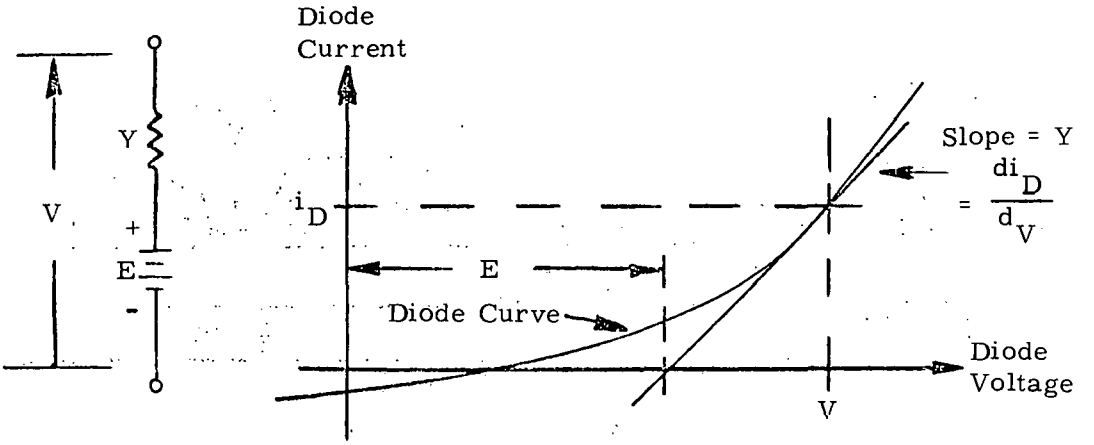
Cutoff: $i_D = V*10^{-12}$

Saturated: $i_D = KS\left(\frac{V^2}{2} - VTE*V\right)$ for: $V > VTE$

$i_D = 0$ for: $V \leq VTE$

Triode: $i_D = KU\left(\frac{V^2}{2} - VTE*V\right)$

The implementation for Newton-Raphson iteration is similar to that used for junction diodes:



The following derivation is used for the FET diode model, for use in the nonlinear iterations:

$$i_D = K\left(\frac{V^2}{2} - VTE*V\right)$$

$$\frac{di_D}{d_V} = K(V - VTE)$$

$$E = V - i_D/Y$$

Refer to the sketch above.

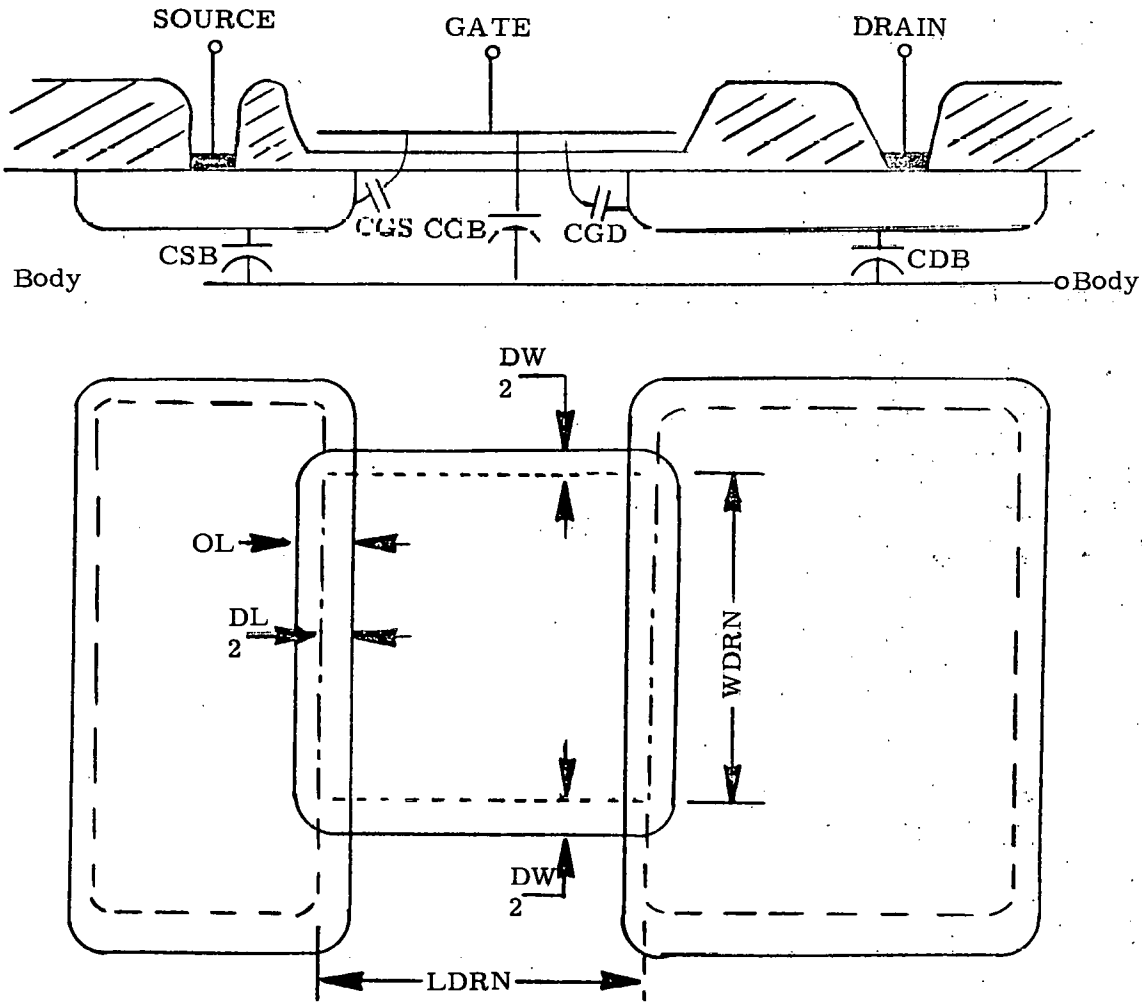


Figure II-15A. A Model - Geometry

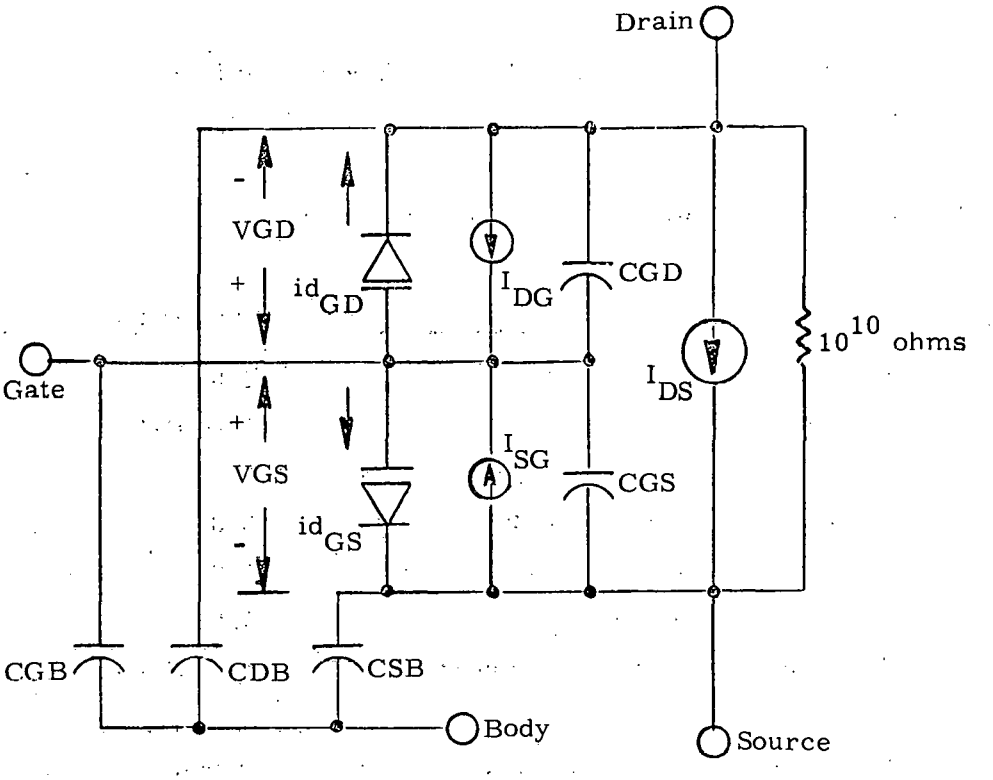


Figure II-15B. A Model MOSFET Implementation

$$I_{DS} = \begin{cases} 0 & \text{Cutoff} \\ \left(\frac{VGS^2}{2} - VT*VGS\right)KS + VDS*10^{-10} & \text{Saturated} \\ \left(\frac{VGS^2}{2} - VT*VGS\right)KU - \left(\frac{VGD^2}{2} - VT*VGD\right)KU + VDS*10^{-10} & \text{Triode} \end{cases}$$

2.1.12 JFET Model

The JFET Model is similar to the MOSFET Model, and is also implemented using the FET Diode of Reference II-11.

The following parameters define the N-Channel Model:

NAME	PARAMETER	DEFAULT	UNITS
VTO	Threshold Voltage	2.0	V
BETA	Transconductance	1.0 E-4	A/V^2
LAMB	Channel Length Modulation Parameter	0.01	V^{-1}
CGS (0)	Gate-Source Capacitance (Zero bias)	5.0	pf
CGD (0)	Gate-Drain Capacitance (Zero bias)	1.0	pf
PHIB	Gate-Junction Potential	0.6	V
IS	Gate-Junction Saturation Current	1.0 E-14	A

The governing equations for an N-Channel device, are as follows:

Forward Region: VDS > 0

Operating Conditions:

Cutoff: VGS ≤ VTO

Saturated: VGS ≤ VDS + VTO and VGD ≤ VTO

Triode: VGS > VDS + VTO

Device Current:

$$ID = \begin{cases} 0 & \text{Cutoff} \\ BETA \cdot [VGS - VTO]^2 (1 + LAMB * VDS) & \text{Saturated} \\ BETA * VDS \cdot [2(VGS - VTO) - VDS] \cdot (1 + LAMB * VDS) & \text{Triode} \end{cases}$$

The notes in Section 2.1.10 regarding P-Channel reverse operations, as well as provisions for more complex modeling, apply to the JFET device as well.

The gate junctions G-S and G-D are modeled by ideal diodes:

$$IGS = Is\left[\exp\left(\frac{-qVGS}{KT}\right) - 1\right]$$

$$IGD = Is\left[\exp\left(\frac{-qVGD}{KT}\right) - 1\right] \quad .$$

The polarites of VGS and VGD are such that these diodes are normally nonconducting.

The depletion capacitances are defined by:

$$CGS = CGSO \cdot \left[1 + \frac{VGS}{PHIB}\right]^{-1/2} \qquad |VGS| < PHIB$$

$$CDB = CDBO \cdot \left[1 + \frac{VDB}{PHIB}\right]^{-1/2} \qquad |VDB| < PHIB$$

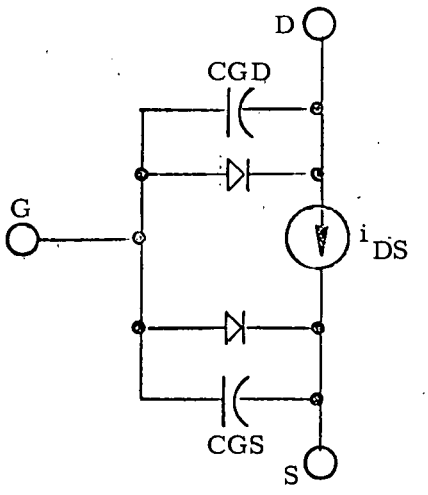


Figure II-16. JFET Model

2.1.13 Switch and Latch

The switch, or switched element, allows the user to vary element values in a circuit based on some changing conditions which occur during the analysis.

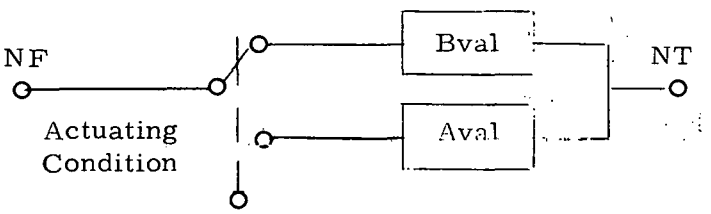
The elements which are allowed to be switched from one value to another upon switch actuation are:

S - floating sources, V or CU
R - resistors
C - capacitors } switching of RS or
L - inductors } RSH is NOT allowed
T - transconductances

An element is switched from one value to another upon switch actuation and returns to its initial value if the switch is deactuated due to circuit conditions. The user must be careful in his specifications so that switches are not actuated and then immediately deactuated, simulating a buzzer!

For such cases, the latch option is used. Latch operation is identical to the switch except that, once actuated, the latch stays in its actuated position for the rest of the simulation, regardless of circuit conditions.

In general, a switched element is of the form:



Bval = value before switching

Aval = value after switching

The actuating condition can be any of the following:

TFnn	Time function (voltage source)		
NVnn	Node voltage nn	[GT	
BCnn	Branch current nn		] Value
RXnn	Variable in SET command	[LT	]
TIME	Time point in simulation		

NOTE

Switches (and latches) are NOT activated in DC analysis. They are only activated in the TR analysis.

2.2 Temperature

One temperature value is entered in degrees Celsius (centigrade) to simulate a change in ambient for all semiconductor devices. The models use this value of temperature to calculate $\theta = kT/qM$ and for calculating new values of saturation currents. Thus, COMPLETE temperature effects are simulated by the program.

NOTE

Each diode and transistor may have an individual temperature specification, in order to vary its individual temperature, above or below the ambient temperature setting (see Section 3.13.1).

The original versions of this program used a value of θ (Boltzmann/Electron Charge) slightly different from the corrected value in this version. Thus, the results for circuits comprising diodes and transistors will give slightly different results. In order to compare calculations, the temperature, in older versions should be set to: 23.4652°C. This temperature will give results identical to those now calculated in AITRAC.

2.2.1 Reverse Saturation Currents

The reverse saturation currents for diodes (IS) and for transistors (ICS and IES) are strong functions of temperature, and must be accounted for if accurate temperature effects are to be simulated. In general:

$$IS_{(T)} = IS_o\left[T_o + \Delta T\right]^{3/M} * \exp\left[-Eg/(k(T_o + \Delta T))\right] ,$$

where

$IS_{(T)}$ is the saturation current at temperature = $T_o + \Delta T$

IS_o is the room temperature value of saturation current (the value in the semiconductor Library)

EG is the "Energy Gap" voltage for the material being used

M is the diode emission constant (MD, MC, ME) .

EG is defaulted to 1.11 volts for silicon. The user may enter any other value (such as 0.67 for germanium) by using the MODIFY command (see Section 3.13.1).

2.3 Time Functions

Functions of time may be entered and used for transient analysis. These functions may be entered in three ways:

a. As straight-line segment functions. An initial value at time t = 0 is entered, followed by up to 14 pairs of time and amplitude values. This defines a waveform of up to 15 segments.

b. As SINE waves. Three values are entered for peak amplitude, frequency, and initial phase.

c. As DAmped sine waves. Four values are entered for peak amplitude, frequency, phase, and envelope time constant (see Section 3.2.2).

d. As Double Exponential function. Three values are entered for magnitude and two time constants (see Section 3.2.2).

e. By SPECIAL EQUATIONS (see Section 5).

Some typical examples of time functions are shown in Figure II-17.

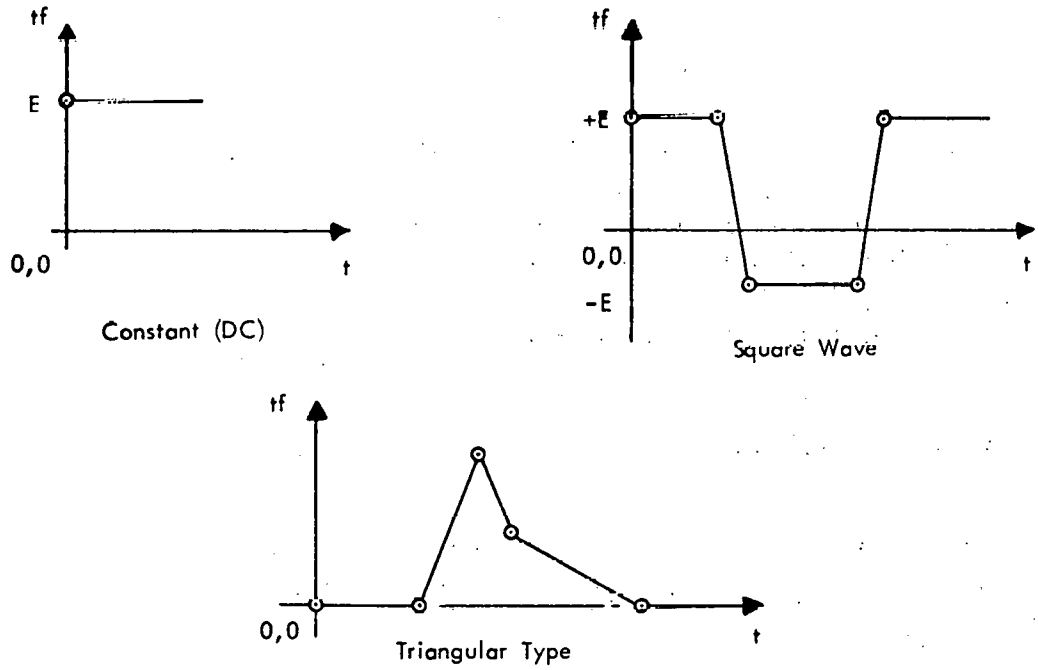


Figure II-17. Examples of Straight-Line Time Functions

The function is described to the program by its coordinates. The time functions may be cycled, or the last value may be held. Multivalued time functions are not allowed.

2.4 Initial Conditions

For a transient analysis, the user has the following options available for describing the initial conditions.

a. The initial conditions are zero (the default if no other option is specified).

b. The program calculates the DC solution (assuming one exists), and the DC solution will determine the initial conditions.

c. The user enters his own initial conditions. The data required for proper initialization are:

   (1) Voltages of nodes which have capacitors, inductors, diodes, and transistors connected to them, as well as: Capacitor and inductor branch currents.

   (2) Source values consistent with the above.

   (3) Use of the TRY option, in the EXECUTE command.

Inconsistent initial conditions cause the initial transient behavior to be abnormal. The requirement for the initial conditions to be consistent is that the currents at the nodes, which have capacitors, inductors, diodes, and transistors connected, satisfy Kirchhoff's current law.

When the zero initial conditions mode is used and when the source values at time zero are not equal to zero, the initial conditions may be inconsistent when the sources are connected to models which are dependent on the past terminal voltage values (e.g., capacitor, inductor, diode, and transistor). All source values must then be equal to zero at time zero (e.g., DC sources must be ramped from zero to their DC level).

2.5 Output

2.5.1 Printed Output

The user may specify any combination of the following as printed output:

a. Time functions
b. Node voltages
c. Branch currents
d. Branch powers

These may be printed in full or selectively. During a transient analysis, the time and current delta time are automatically printed out at each time solution. The frequency of printout (i.e., the ratio of printed to calculated points) may also be specified. The specified items are automatically printed out at the following solutions:

a. The initial conditions
b. The first time solution
c. All time function coordinates
d. Δt range end times
e. The last time solution

The printed output may be directed to the user's terminal; the line printer, if available; or a disk file.

2.5.2 Plotted Output

The user may specify any combination of the following as plotted output

a. Time functions
b. Node voltages
c. Branch currents
d. Branch powers

All results are plotted as a function of time. Four plots are allowed per grid, and multiple plots may be obtained without recalculating the solution. The plots are available in two formats for the Teletype terminal or in a format suitable for wide (120-character-width) terminals. The frequency of storing points for plotting may be specified. Plotted output may be directed to the user's terminal; the line printer, if available; or a disk file.

2.6 Delta Time (Time Step)

2.6.1 Delta Time Input and Automatic Program Termination

Maximum values of delta time may be entered for ten different time ranges. Usually, not all ten time regions need be defined. The last time region defined determines the time for normal program termination (i.e., simulation complete).

2.6.2 Program Delta Time Control

Delta time is controlled first by user input. Second, Δt is constrained to less than one-tenth the distance between the defined time coordinates for all straight-line time functions. Third, Δt will be reduced, if necessary, so that a solution may be made on the time function coordinate (within a tolerance). Fourth, if convergence, based on the nonlinearities, is not achieved in 10 iterations, Δt will be reduced by a factor of two. Fifth, the Δt for the next solution will not be increased if the number of iterations for solution is greater than three. The projected Δt will otherwise be allowed to double (up to the maximum Δt). Should Δt ever fall below 1. E-20, the program will terminate, assuming that there is a gross error in the simulation.

2.6.3 User Delta Time Selection

The primary objective is to have Δt as large as possible to keep the computation time (i.e., the number of time solutions) to a minimum, and at the same time achieve good engineering accuracy. When the delta time is made indiscriminately too small, poor accuracy can actually result due to numerical errors (e.g., round-off, etc.). The program does not have any delta time control based on internal problem time constants. The maximum delta time input can be based on the smallest significant time constant in the problem. Usually, the diode and transistor time constants can be ignored in the delta time consideration, except for high-frequency circuits where the period of the dominant harmonic approaches these time constants. From experience, it has been observed that a maximum delta time equal to one-tenth the smallest significant time constant yields good engineering accuracy. Should there be any doubt regarding the size of delta time, the problem can be rerun with another delta time value, and a check can be made for result disparities. In many instances, the maximum delta time size can be determined by the minimum number of solutions desired. For many semiconductor circuits, the delta time control discussed in the preceding section reduces the complications inherent in the delta time selection.

2.7 Topology Restrictions

No special topology restrictions exist in AITRAC. However, as a guide, each standard model should have only one terminal connected to ground or to an independent node (grounded voltage source). This is a practical limitation. For example, a diode connected between two voltage sources will produce a circuit which will not converge to a solution. Nodes must be numbered consecutively.

2.8 Problem Size

The problem size, subject to machine restrictions and core available, is as follows:

a.	Node unknowns	100 (plus ground)
b.	Branches (i.e., sum of resistors, capacitors, inductors, floating sources, and linear transconductances)	200
c.	Switched elements	200
d.	Junction diodes and/or JFETS	30
e.	Junction transistors and/or MOS devices	30
f.	Time functions and grounded volt. souces	18
g.	Radiation ionizing function	1
h.	Ratio of max to min resistance at any node	10^{10}

3. Program Operation

3.0 Access to the Time Sharing System

The following procedures are used to access the computer system and AITRAC from an interactive terminal.

3.0.1 Dialup

From the terminal location, the user turns on terminal power and dials the appropriate telephone number. Upon connection as evidenced by the tone signal, the user places the telephone handset into the audio coupler, observing the proper position of the telephone cord.

The user strikes the RETURN key, to which the computer responds:

```
6000 system up
```

After the user again strikes the RETURN key, the computer prompts:

```
77/07/11.  21.36.57.
Sandia NOS SN53 ECS          NOS 1.1-430
User number, password
```

The user then enters an authorized user number and password followed by a RETURN.

Following authentication by the computer, the following confirmation is displayed on the terminal.

```
Terminal:  35, TTY
RECOVER/SYSTEM:
```

If the user cannot reach this point, the user should check that the user number and password are current and correct.

The user is also referred to the Sandia NOS news notes.

3.0.2 Accessing AITRAC

AITRAC is most effectively accessed from the batch subsystem using the user-entered command:

```
BATCH
```

The user then enters the commands:

ATTACH, AITRAC/UN=LIBRARY

and

AITRAC

to which the computer responds with:

SANDIA AITRAC-100 Version 1.0 UPDATES=1

The user is now ready to use AITRAC by following the instructions given in Section 3.

3.0.3 Terminating AITRAC

AITRAC may be terminated by typing "Q" and RETURN when in command mode. If AITRAC is in circuit input mode, a "Q" and RETURN in response to a parameter prompt takes the program to command code. If AITRAC is requesting a plot title, a "QUIT" and RETURN takes the program to command mode.

NOTE: Before logging off, the user should SAVE any local files that might be needed in future work.

3.0.4 Logging Off

After returning from AITRAC to systems level, the command:

BYE

effects the logging off function.

After the computer prints the termination greeting and billing information, the telephone should be cradled, and terminal power turned off.

3.0.5 Errors in Typing

The backspace key (CTRL H) is used to back the printhead to the point of error so that corrected data may be typed over the bad data. The entire line can be deleted by simply depressing the ESC key before striking a RETURN key.

3.1 Using the Program

This section describes the steps required to use the program on-line from a remote terminal. The program is controlled by simple commands, listed below:

a. NEW - OLD: to read in a new circuit description from the user's terminal or retrieve a circuit description previously saved on disk file

b. PRINT - PLOT: to specify the items to be printed or plotted at the terminal, system printer, or on a disk file

c. TIME: to specify the different time steps over different time regions during the analysis

d. LIST: to list all or only some circuit elements currently stored in memory

e. EXECUTE: to execute the circuit currently in memory

f. TEMPERATURE: to set the value of ambient temperature

g. MODIFY: to alter only the values, not the connections, of the circuit elements

h. CHANGE: to alter the connections (as well as the values) of the circuit elements

i. DELETE - ADD: to delete unwanted circuit elements or to add new circuit elements

j. DO: to read and execute commands from a disk file or to store initial conditions to be used

k. SAVE: to save the circuit description on a disk file

l. MAKE: to create private semiconductor data files and to convert parameters from model in other programs

m. BC-NV-BP: to print out all currents, voltages, or branch powers after a DC execution

n. SUMMARY: to print a table of elements and connections

o. SCALE: to scale amplitudes of voltage and HP sources

p. STEP: to inhibit time step increase during execution

q. CRITERIA: to alter algorithm and convergence criteria

r. INFORM: to print out internal matrices for debugging

Complete descriptions of these commands are given in the following sections. A user's pocket size "AITRAC Programming Aid" is also available.

Every line read in by the program is divided into "fields." These fields are groups of characters, separated by at least one blank or by one comma. For example, the line:

```
PLOT, NV23 -10. 20.
```

consists of four fields: (1) PLOT (2) NV23 (3) -10. (4) 20.

A field may be INTEGER, REAL, or ALPHAMERIC.

INTEGER fields consist of an integer number, written without a decimal point, as follows:

```
1        0        32        -5        +6
```

REAL fields contain a real number, with either a decimal point or one of the exponential multipliers: E, U, K, M, P, etc.: for example,

```
-1.        0.        3.28        -38.473E-3        1.E+6
```

whose values are, respectively:

```
-1.        0.        3.28        -0.038473        1000000.
```

The letter E (exponent) is used to multiply the number by the power of ten which follows. The engineering multipliers may also be used as follows, with their equivalents:

A	F	P	N	U	L	K	M	G	T
E - 18	E - 15	E - 12	E - 9	E - 6	E - 3	E + 3	E + 6	E + 9	E + 12

Examples:

2.4K -5U 22.M 33P 1.234N 3.1416L

are equivalent to:

2.3E3 -5.E-6 2.2E7 3.3E-11 1.234E-9 3.1416E-3

NOTE: A decimal point is optional when specifying REAL numbers† in most instances. Thus, -15 is equivalent to -15., and 0 is the same as 0. This gives the user extra freedom on input, but he should always be careful to check his data entries by means of the LIST command before executing his problem. In particular, ambiguous entries such as EX VO 30 could mean that node 1 is initialized at 30. volts, or that node 30 is initialized at 0. volts (see Section 3.7). To avoid ambiguity, enter real numbers with decimal point during input, which is always the correct format.

ALPHAMERIC fields are neither integer nor real, and contain any sequence of characters; for example:

MYFILE NV12 2N654 BP3 SAVE

A NULL LINE is a line which contains no fields, i.e., a blank line. It is entered by hitting the carriage return key only.

Any line may contain up to 80 characters, but any line may be continued by a comma following the last field. Thus:

```
PRINT NV1  BC2, TF3  BC34  RATIO 5
```

is equivalent to the three lines:

```
PRINT  NV1,
BC2  TF3  BC34  RATIO,
3
```

If a semicolon character starts any field, then the rest of that line is ignored. Thus in the line:

```
TIME    10N    1U    ;THIS IS A COMMENT
```

the characters ;THIS IS A COMMENT will be disregarded.

---

†See Section 3.7.1 for other restrictions.

3.1.1 Input from Disk Files

Files may be prepared using the System's Text Editor, and saved on disk, for access by AITRAC. The following features will be of help to the user:

A dollar sign ($) in column one will be treated as if the user entered a blank line; i.e., all information on that line is ignored. This feature is useful when entering data from paper tape, as it helps keep track of how many blank lines are entered when skipping over elements not used in the circuit.

A double semicolon (;;) encountered on input causes all following information to be treated as comments, as for a single semicolon. However, the information after a double semicolon will be typed out at the terminal when the data are read in from a disk file. This allows additional comments to be stored on the file for special information and reminders. See Appendix B.

3.1.2 Input from Terminal

The sections which follow describe each of the program commands in detail. The following rules apply to the syntax description:

a. Words in CAPITAL letters are to be typed in EXACTLY as shown. The underlined portion of the command indicates the minimum abbreviation which is allowed.

b. Words in lower case are used to represent something indefinite. For example, the words:

filename value1 node2

represent, respectively, the name of a disk file, the value to be assigned to a circuit component, and a node connection for a component. Each would be replaced by the actual name, actual value, and actual node number.

c. Quantities in brackets represent options, which the user may, but need not, use.

3.2 NEW Command

The NEW Command is used to read in a new circuit description. Any circuit currently residing in memory is destroyed. The format for the NEW command is:

NEW

Upon entering the NEW command, the program will respond with

NAME:

The user may then type in a name to be given to the circuit. The name may be up to 40 characters long, including blanks. If no name is desired, the user should hit carriage return.

This NAME will be used as the default TITLE for the plots, if no other title is entered (see Section 3.5.1).

The circuit description for each element is then read in from the terminal. The program will request each parameter from the user, in turn. The formats for entering each set of data are described below.

3.2.1 OLD Command

The OLD Command is used to read in a circuit description which was previously saved on a disk file. The format for the OLD Command is:

OLD [[filename] [LIST]]

The circuit description is read in from the file with the given "filename". If the LIST option is also specified, the file is listed on the terminal, as it is read in. The reader is referred to Appendix B for information regarding the formats of the data stored on disk files.

3.2.2 Grounded Voltage Sources and Time Functions

After entering the circuit name, the program will respond with:

V1,

The letter V is used to represent grounded voltage sources and straight-line time functions, and the number 1 is used to indicate that information is being requested for the first time function or source. The user may now respond with one of the following:

a. A real number, to indicate a constant (DC) value for the source or time function. Thus,

V1, 5. E1

indicates a source or function with a constant value of 50.

b. A sequence of real numbers, to designate a straight-line segment time function. Example:

V1, 50. 1U 75. 2U .1K

indicates a time function of 50. volts at time = 0, 75 volts at t = 1 μs, and 100 volts at t = 2 μs.

If the last value entered is not to be held, the option CYCLE may be added, at the end of the list, to make the waveform repeat, e.g.,

V1, 50. 1U 75. 2U 100. CYCLE

Multivalued functions are not permitted, i.e., each value of time entered must be greater than the previous value. Up to 14 time-amplitude pairs may be entered, after the initial specification of amplitude at time t = 0. Thus, up to 15 segments may define a waveform.

c. The designations SINE, DASINE, DEXP, followed by three or four real numbers, to describe the waveform as follows:

Name	Input	Value1	Value2	Value3	Value4
Sine wave	SINE	Amplitude	Frequency	Phase	---
Damped Sine	DASINE	Amplitude	Frequency	Phase	Time Const.
Double Exp.	DEXP	Magnitude	T. Const. 1	T. Const. 2	---

The defining equations are:

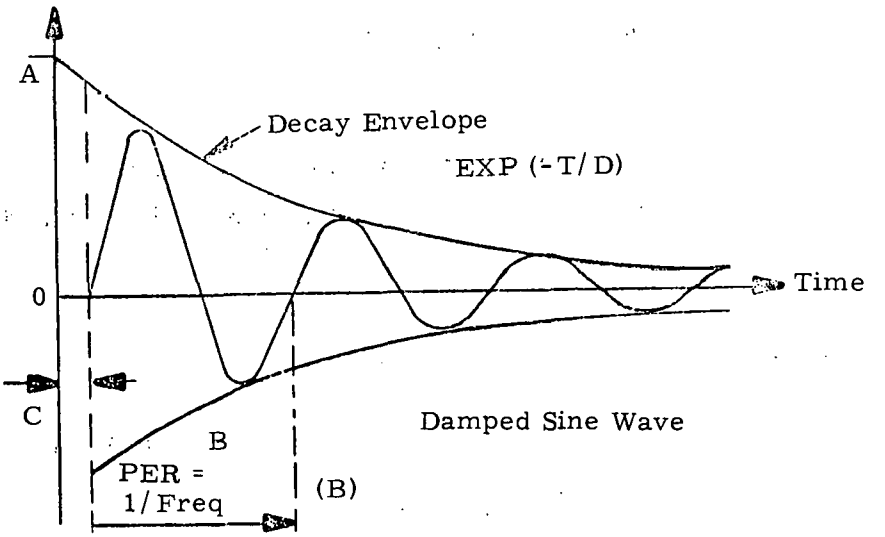
SINE: $A*SIN(2\pi(B*T + C/360))$

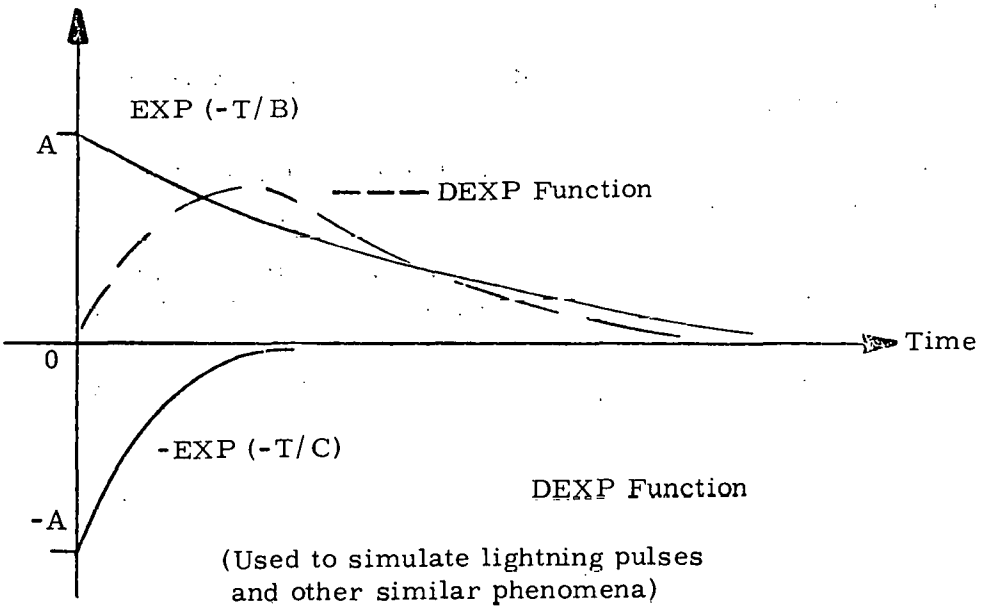
DASI: $A*SIN(2\pi(B*T + C/360))\ (EXP(-T/D))$

DEXP: $A[EXP(-T/B) - EXP(-T/C)]$

Example: V5, SINE 100. 500K 45.

is the input required to specify a sinusoidal source of 100 volts peak of a frequency of 500 KHz and a phase of 45 degrees.





DEXP Function

(Used to simulate lightning pulses and other similar phenomena)

After the value(s) for the first source, V1, is entered, the program will respond with:

V2,

and the user may enter the data for the second time function or source. When input for all sources is complete, the user enters a null line (carriage return) when information for the next source is requested by the program. The program will then advance to the next item, i.e., floating sources (see below).

3.2.3 Floating Voltage and Current Sources

Floating voltage or current sources are indicated by the letter S. The program requests information for the first source by typing:

(1) S1,

Numbers in parentheses always denote the branch numbers assigned to each component by the program. In the example above, floating source 1 is assigned branch number 1 in the circuit.

The format for describing floating sources is:

Node-f Node-t value [series or shunt res.] [CUrrent] ,

where

a. Node-f designates the connection of the F terminal of the source, and is either:

   (1) An unsigned integer to indicate the node number to which the terminal is connected.

   (2) The letter G or integer 0 to indicate that the terminal is connected to ground.

   (3) The letter V followed by an integer, value n, to indicate that the terminal is connected to grounded voltage source n.

b. Node-t designates the connection of the T terminal of the source.

c. Value indicates the value of the source, and is either:

(1) A real number to denote a constant DC value

(2) The words USE Vn to denote that the source is to use the values associated with time function n.

d. Series-shunt-resistance is optional, and, if present, is a real number indicating the value of a series resistance for voltage sources or shunt resistance for current sources. If omitted (or entered as zero), a value of 1E9 will be used for shunt resistances and 0.001 for series resistances.

e. CURRENT is optional, and, if present, indicates that the source is a floating current source. Otherwise, the source is assumed to be a floating voltage source.

Examples:

(1)S1, 2 G .1 1K CURRENT

indicates that floating source 1 is a current source connected from node 2 to ground, with a value of 0.1 ampere and a shunt resistance of 1 kilohm.

(3)S3, 1 V2 USE V3

indicates that floating source 3 is assigned branch number 3, is a floating voltage source connected from node 1 to grounded voltage source 2, and it is to assume the values associated with time function 3. A series resistance of 0.1 ohm is assumed.

(2)S2, 0 4 USE V1 CURRENT

indicates that floating source 2 is a current source connected from ground to node 4, and it is to assume the values associated with time function 1. A shunt resistance of 1E9 ohms is assumed.

The user terminates information on floating sources by entering a null line after describing the last source.

3.2.4 Resistors

Resistors are denoted by the letter R. The program will request information for the first resistor by typing:

(n)R1,

where n is the branch number being assigned to resistor 1. The format for describing resistors is:

node-f node-t value

where

a. Node-f designates the connection of the F terminal of the resistor.

b. Node-t designates the connection of the T terminal of the resistor.

c. Value is a real number indicating the resistance in ohms.

Examples:

(4)R1, 3 G 5K

indicates resistor 1 is assigned branch number 4, is connected from node 3 to ground, and has a resistance of 5 kilohms.

(6)R3, V1 2 6.3K

indicates resistor 3 is assigned branch number 6, is connected from grounded voltage source 1 to node 2, and has a resistance of 6.3 kilohms.

3.2.5 Capacitors

Capacitors are denoted by the letter C. The program will request information for the first capacitor by typing:

(n)C1,

where n is the branch number being assigned to capacitor 1. The format for describing capacitors is:

node-f node-t value [series-rest.] [shunt rest.]

where

a. Node-f designates the connection of the F terminal of the capacitor.

b. Node-t designates the connection of the T terminal of the capacitor.

c. Value is a real number indicating the capacitance in farads.

d. Series-resistance is optional, and, if present, is a real number indicating the series resistance in ohms. If omitted, a value of 0.001 ohm is assumed.

e. Shunt-resistance is optional, and, if present, is a real number indicating the shunt resistance in ohms. If shunt resistance is desired, then series resistance must also be entered (although the default value of 0.001 can be used). If omitted, a value of 1E9 ohms is assumed.

Example:

(8)C1, 2 V3 1U 1E4

indicates that capacitor 1 is assigned branch number 8, is connected from node 2 to grounded voltage source 3, has a capacitance of 1E-6 farads, and a series resistance of 10 kilohms. The shunt resistance is assumed to be 1E9 ohms.

3.2.6 Inductors

Inductors are denoted by the letter L. The program will request information for the first inductor by typing:

(n)L1,

where n is the branch number being assigned to inductor 1. The format for describing inductors is:

node-f node-t value [series-R] [shunt-R]

The parameters are analogous to those in the capacitor model.

Example:

(10)L1, 3 5 1U

indicates that inductor 1 is assigned branch number 10, is connected between nodes 3 and 5, and has inductance of 1E-6 henrys. The shunt resistance is assumed to be 1E9 ohms. The series resistance is assumed to be 0.001 ohm.

3.2.7 Junction Diodes

Diodes are denoted by the letter D. The program will request information on the first diode by typing:

(n)D1,

where n is the branch number being assigned by the program to the first diode. The format for describing diodes is:

node-a node-c [COPY Dn] [USE filename] [SEARCH diode-name] [LIBRARY diode-name]

where

a. Node-a and node-c designate the anode terminal and the cathode terminal connections of the diode. This convention must be observed to obtain proper polarity.

b. COPY, USE, SEARCH, and LIBRARY are optional, and are described below.

In the absence of the COPY, USE, SEARCH, or LIBRARY options, the program will type:

PARAMS:

indicating that it is requesting the seven diode parameters from the user. The diode parameters needed are:

IS, MD, RDL, CDO, VDBI, TD, IPPD

The meaning of these parameters is discussed in Section 2.1.8. (The parameter IPPD is included only for compatibility with versions of the program equipped to handle radiation analysis, and if radiation effects are not to be studied, a value of 0. should be entered for this parameter.)

The user should type in seven real numbers to enter a value for each of the parameters. The program will continue to accept input data until all seven parameters are entered. If a null line is entered, the program will type out the names of the parameters. The user should then enter a value for each parameter, in the proper order, as before.

Instead of typing a value for each parameter, the user may utilize the COPY, USE, or SEARCH options:

a. COPY Dn -- tells the program to copy the parameters associated with diode n and use them for the present diode.

b. USE filename -- tells the program to read the diode parameters from a file with the specified filename, rather than from the terminal. The file contains the parameter values just as they would be typed in at the terminal.

c. SEARCH diode-name -- tells the program to search for the diode parameters in the system parameter bank. If the diode type is found, the parameters retrieved will be used. If the specified diode type is not in the library, the user will be so notified, and must try another option.

d. LIBRARY diode-name -- tells the program to search for the diode parameters in the user's private data bank library. This private library may contain data for any semiconductor device, and is stored on a disk file in the user's own private area.

   The instructions for creating this private library are given in Section 3.18.1.

Examples:

```
(12)D1, 2 3 . . . .          (typed by program)
PARAMS:                      (user hits car. return)
IS, MD, RDL, CDO, VDBI, TD, IPPD .   (typed by program)
1N, 2., 1G, 10P, 0.70, 10N, 0.  .    (typed in by user)
```

In the example above, diode 1 is assigned branch number 12 (by the program) and is connected between nodes 2 and 3. When the parameters were requested, the user entered a null line and the program typed out the diode parameter headings. The seven diode parameters were entered, using engineering abbreviations. Refer to Section 2.1.7 for diode parameter details.

```
(14)D3, 3 V2 COPY D1
```

In this case, diode 3 is connected between node 3 and grounded voltage source 2. The parameters to be used for this diode are to be copied from those used for diode 1.

```
(15)D4, 1 2 USE MYDIOD
```

Diode 4 is connected between nodes 1 and 2. Its parameter values are to be retrieved from the user's file named "MYDIOD".

```
(13)D2, 55 77 SEARCH 1N914B
```

The user requests that the data be searched for in the system parameter library. If data for the diode type requested are found, the program will automatically use these data. If the diode type is not found in the system library, a message will be printed out informing the user of this fact. The user must then enter data using another option.

```
(173)D27, 43 86 LIBRARY 1N914B
```

The user has requested that the data be searched for in his private library. The data must have been previously entered for this diode name and saved on the user's disk area. If the diode name is not found in the user's library, a message will be printed out, and the data must be entered using another option.

3.2.8 Bipolar Junction Transistors (BJT)

Transistors are denoted by the letter Q. The program will request information for the first transistor by typing:

(n)Q1, (E-B-C),

where n is the branch number being assigned to the base of the transistor, and n + 1 is automatically assigned to the collector of the transistor. The format for describing transistors is:

```
                                  [NPN]   [COPY Qn]
node-e   node-b   node-c [PNP]   [USE filename]
                                  [SEARCH transistor-name]
                                  [LIBRARY transistor-name]
```

where

a. Node-e node-b node-c designate the emitter, base, and collector terminals of the transistor. This connection sequence must be observed.

b. NPN and PNP -- are optional; if omitted NPN is assumed.

c. COPY, USE, SEARCH, and LIBRARY are optional, and are described below.

In the absence of a COPY, USE, SEARCH, or LIBRARY field, the program will type:

PARAMS:

indicating that it is requesting the 16 bipolar junction transistor parameters for the Ebers-Moll model. If a carriage return is hit, the parameter headings will be printed as follows:

HFEN, HFEI, TN, TI, ICS, MC, CCO, VCBI,
RCL, IES, ME, CEO, VEBI, REL, IPPC, IPPE

The meaning of these parameters is discussed in Section 2.1.8. (The parameters IPPC and IPPE are included only for compatibility with versions of the program equipped to handle radiation analysis, and if radiation effects are not to be studied, values of 0. should be entered for these parameters.)

The user should type in 16 real numbers to enter a value for each of the parameters. The program will continue to accept input until all parameters have been entered. The user must enter the parameters in the correct order.

If an error in parameter data is made during input, the incorrect parameter may be modified to the correct value, using the MODIFY command (see Section 3.13).

Instead of typing in a value for each of the required parameters the user has the options: COPY, SEARCH, USE, and LIBRARY at his disposal, which facilitate and speed up data entry. The usage is analogous to that for diodes.

Example:

```
(16)Q1, (E-B-C) 12 33 G PNP . . .        (typed by program)
PARAMS:                                  (user hits car. ret.)
HFEN, HFEI, TN, TI, ICS, MC, CCO, VCBI
RCL, IES, ME, CEO, VEBI, REL, IPPC, IPPE  (typed by program)
100., 1. 1N 1U 10U 1. 10P                (entered
.75 10M 1P 1. 5E-11                        by
.80 100M 0. 0.                             user)
```

In the example above, transistor 1 is assigned branch numbers 16 and 17, and is connected between nodes 12, 33, and ground. The device is of type PNP. The 16 parameters are entered by the user.

The connection sequence: E-B-C is printed out for the first transistor only, to prompt the user for the correct connection sequence.

NOTES

It is not necessary to use a comma (,) at the end of the line in order to enter the diode or transistor parameters on several lines. The program continues to expect, and accepts, real numbers from the input terminal, until the entire list of 7 diode or 16 transistor parameters has been entered.

When using a PNP transistor, the option "PNP" must be typed in before the COPY or SEARCH option; otherwise, an NPN transistor is assumed, having the same numerical parameter values.

The program remembers diode and transistor USe and SEarch requests, and prints out this information when the LIst command is invoked. This allows the user to identify the source of his parameter data.

The COPY option, when used on diode and transistor input, is remembered. For example, if on Q5 input, the option COPY Q1 is used, then any subsequent changes to parameters in Q1 will also change the appropriate parameters in Q5. In fact, the parameters in Q5 cannot be altered (via a MODIFY command), until the original "COPY Q1" is released; this can be done by typing:

*CHANGE Q5

(n)Q5, (same connections as before) COPY Q5

Q5 parameters can now be changed by using the MODIFY command.

During the input phase, when entering Q5, the option "COPY Q1" may be used, but not: "COPY Q8," since Q8 does not yet exist in memory. After input has been completed, however, the option "Q5 (connections, PNP) COPY Q8" is valid, as is: "Q8.....COPY Q1."

3.2.9 Transconductances

Transconductances are denoted by the letter T. After requesting information on the last transistor, AITRAC will respond:

(n)T1,

If there are no transconductances in the circuit, the user should hit carriage return to terminate circuit input. Otherwise, the user should type:

from-branch to-branch type value

where

a. From-branch is the branch number of the controlling branch ("from-branch") of the transconductance. Note that the branch number is obtained from AITRAC (it is the number typed in parentheses for each element). For example,

(8)C1,

means capacitor 1 is assigned branch number 8 by AITRAC.

b. To-branch is the branch number of the controlled branch ("to-branch") of the transconductance.

c. Type is one of the following:

(1) BETA for current controlled current sources
(2) GM for voltage controlled current sources
(3) MU for voltage controlled voltage sources
(4) ZM for current controlled voltage sources

d. Value is a real number, indicating the value of the transconductance.

(10)T1, 6 8 BETA 50.

Transconductance 1: "from" branch is 6; "to" branch is 8; the value of BETA is 50.

The values of all T's in the circuit may be obtained by the command:

LIST T

or to obtain information on specific T's:

LIST T2 T4 etc.

RESTRICTIONS

The use of Zm or Mu when the "to" branch is either an inductance or capacitance is NOT recommended. The user is advised to study the equivalent circuits of Figure III-1 and use alternate methods of modeling his circuit.

Furthermore, the user is warned that different circuit analysis programs may define "branch current" in different ways, and comparisons of results from different programs may require careful examination.

Refer to Figure III-2, which shows a typical circuit branch, sometimes referred to as a "Kron" Branch, after Gabriel Kron who did basic work on the properties and methods of solution for complex networks.

The PRint BC command prints out the value of the current J' for all circuit elements (R, C, L, and floating voltage sources), and the current J'-I for all floating current sources.

Thus, the printed current "BC" does NOT include the current I_S in the shunt branch of capacitors and inductors, NOR does it include the independent current I in floating current sources.

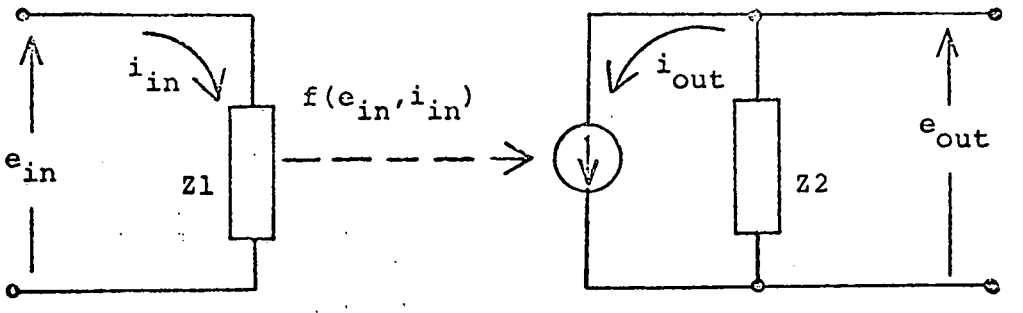


Figure III-1

The user wishing to use the linear transconductances, should first acquaint himself with the relationships given below, which are used by AITRAC for network calculations.

The basic definitions of the various transconductances are:

$$BETA = i_{out}/i_{in}$$

$$Gm = i_{out}/e_{in}$$

$$Mu = e_{out}/e_{in}$$

$$Zm = e_{out}/i_{in}$$

From the above, the following relationships can be derived:

Beta = Gm*Z1	Gm = Beta/Z1	Mu = Beta*Z2/Z1	Zm = Beta*Z2
Beta = Mu*Z1/Z2	Gm = Mu/Z2	Mu = Zm/Z1	Zm = Mu*Z1
Beta = Zm/Z2	Gm = Zm/Z1*Z2	Mu = Gm*Z2	Zm = Gm*Z1*Z2

The following relationships for voltage gain also hold:

$$A_v = Mu$$

$$A_v = Gm*Z2$$

$$A_v = Zm/Z1$$

$$A_v = Beta*Z2/Z1$$

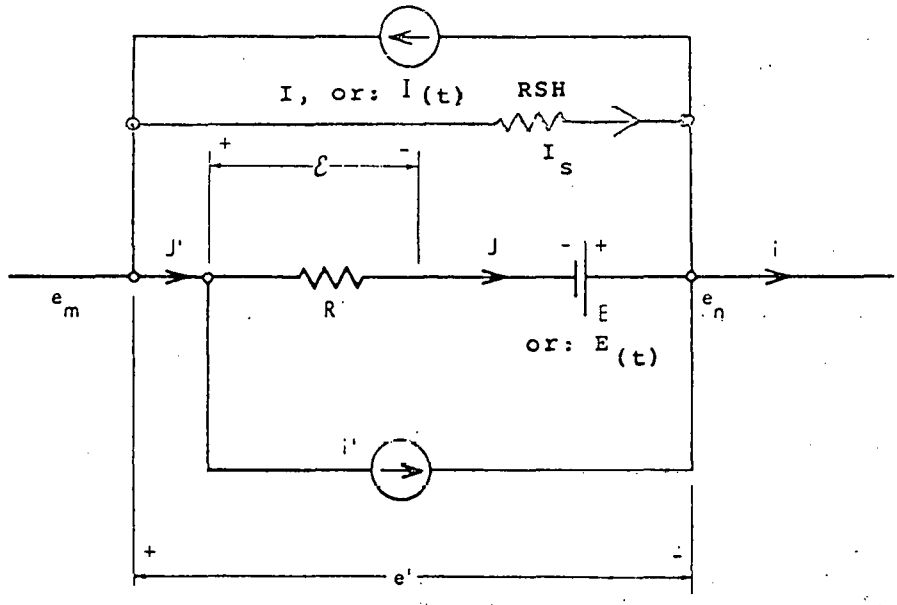


Figure III-2. Generalized AITRAC Circuit Branch, with Definitions

Independent Voltage: E, or $E_{(t)}$

Branch Voltage (e') = $e_m - e_n$

Element Voltage ($\mathcal{E}$) = $(e_m - e_n) + E$

Independent Current: I, or $I_{(t)}$

Branch Current (i) = J' - I

Dependent Current = i'
= $GM * \mathcal{E}_{(from)}$

Element Current including all transconductance terms (J') = J + i'

Shunt Current (for C's and L's) = I_s†

†The shunt current in the shunt resistance branches of capacitors and inductors is NOT included in the printed output of the current obtained with either the PRint BC, or PLot BC commands.

3.2.10 Mutual Inductances

Mutual inductances and coefficient of coupling may be specified in a linear transconductance element: T.

Consider that a mutual inductance is a bilateral transconductance as opposed to the other transconductances: Beta, Mu, Gm, and Zm, which are only unilateral.

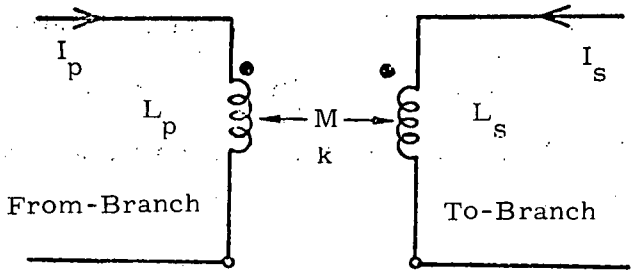
Coupled inductances are entered as follows:

(nn) Tm, from-branch to-branch $\begin{bmatrix} LM \\ LK \end{bmatrix}$ value

The entry LM is used to specify the mutual inductance in henrys. The entry LK is used to specify the coupling coefficient, where

$$LM = LK\sqrt{L_p L_s} \quad ; \quad 0.0 \le |LK| \le 1.0 \quad .$$

The polarity convention is as shown below:



Multiwinding transformers may be modeled by using the appropriate coupling values between each winding to every other winding.

3.2.11 JFETs

JFETs are denoted by the letter J. The program will request information for the first JFET by typing:

(n)J1, (G-S-D) ,

where n is the branch number being assigned by the program to the first JFET. The branch n+1 is assigned automatically to the JFET also.

The format for describing JFETs is:

node-g node-s node-d [NJ] [PJ] [COPY Jn] [USE filename] [SEARCH JFET-name] [LIBRARY JFET-name]

where:

a. Node-g node-s node-d designate the gate, source, and drain terminals of the JFET. This sequence must be observed.

b. NJ and PJ -- are optional; if omitted, N-type is assumed.

c. COPY, USE, SEARCH, or LIBRARY are optional, and are described below.

When COPY, USE, SEARCH, or LIBRARY option is not used, the program will type:

PARAMS:

indicating that it is requesting seven JFET parameters which must be entered by the user. The parameters needed are:

VTO, BETA, LAMB, CGS, CGD, PHIB, IS

The description of these parameters is given in section 2.1.10. The user should type in seven real numbers. The program will continue to accept input until the parameter list is satisfied. If the user types in a null line, the program will

type out the list of names of the parameters required. The user can then enter the seven required parameter values.

Instead of typing in a value for each parameter, the user may enter the COPY, USE, SEARCH, or LIBRARY options:

a. COPY Jn -- tells the program to copy the parameters associated with JFET n and use them for the present JFET being entered.

b. USE filename -- tells the program to read the JFET parameters from a file with the specified filename in the user's private disk area. The parameters are listed in the file just as they are entered on the terminal.

c. SEARCH JFET-name -- tells the program to search for the JFET parameters in the system parameter bank. If the JFET type is not in the system library, the user will be so notified, and he must try another option.

d. LIBRARY JFET-name -- the program will search the user's private library of devices, and search for the particular JFET name. The private library of devices, prepared by the user, must have the filename: FELIBRY

Examples:

```
(15)J1, (G-S-D), 5  9  3  PJ

VTO, BETA, LAMB, CGS, CGD, PHIB, IS
2.5, 3.3E-4, .1, 4P, 2.3P, 0.75, 2.5E-14
```

In the above example, JFET 1 is assigned branches 15 and 16, with the device connected to nodes 5, 9, and 3. The device is of P-Channel construction. When the parameters were requested the user typed a blank line, and the program printed out the JFET parameter headings. The seven required values were then typed in by the user.

(23)J3, 5 7 V2 COPY J1

JFET 3 is assigned branches 23 and 24, and is connected to nodes 5 and 7, and to voltage source V2. The parameters will be copied from those entered for JFET 1.

(7)J8, 7 8 2 USE MYFET

JFET 8 will be assigned the parameters found in the user's private disk file called 'MYFET'. An 'N-type' device is assumed.

(13)J2, G 4 6 PJ SEARCH 3N160

JFET 2 will be assigned values from the system parameter library, if the device is listed. A 'P-type' device will be modeled.

(76)J27, 56 25 76 LIBRARY XFETZ

The private library (file: FELIBRY) will be searched, and the parameters listed under the device 'XFETZ' will be used as data.

3.2.12 MOSFETs

MOSFETs are denoted by the letter M. The program will request information for the first MOSFET by typing:

(n)M1, (G-S-D-B) ,

where n is the branch number being assigned to the device. The branch n+1 is automatically assigned to the device also.

The format for describing either the 'S' or 'A' model is:

node-g node-s node-d node-b [NA / PA / NS / PS] [COPY Mn / USE filename / SEARCH MOSFET-name / LIBRARY MOSFET-name]

where:

a. node-g, node-s, node-d, node-b designate the connections to the gate, source, drain, and body, of the device. This sequence must be observed.

b. NA or PA indicate either N-type or "P-type" for the A Model.

c. NS or PS indicate either N-type or P-type for the S Model.

   NOTE: The default is NA, if this field is blank

d. COPY, USE, SEARCH, or LIBRARY are optional, and are described below.

In the absence of a COPY, USE, SEARCH, or LIBRARY field, the program will type:

PARAMS:

indicating that it is requesting the appropriate parameters for the model.

In the absence of an NA, PA, NS, or PS field, the default: NA is assumed by the program.

For the A model, the following 16 parameters are needed:

KP, VTH, SBC, SBE, DW, DL, OL, SEP,
CGOX, CDIF, DCC, DCE, WDRN, LDRN, RO, PHI(ZED)

The description of these parameters are given in section 2.1.11.

The user should type in 16 real numbers, one each for the parameters in the model. The program will continue to accept input data until all 16 parameters have been entered.

The user may enter a null line, in which case the program will print out the names of all the required parameters. The user must enter all parameters in their proper order.

If one of the parameters is entered in error, the MODIFY command may be used later to modify any single parameter which needs a correction.

Instead of entering 16 parameters for each MOSFET device, the user may take advantage of the COPY, USE, SEARCH, or LIBRARY options. The function of these options is identical to those described for the JFET Model, section 3.2.11.

For the S model, the field NS or PS must be specified, and the 12 parameters needed are:

VTO, BETA, GAMM, PHI, LAMB, CGS,
CGD, CGB, CDB, CSB, PHIB, IS

The description of these parameters is given in section 2.1.12.

The user may enter the required 12 parameters directly, or enter a COPY, use search, or LIBRARY command.

NOTE

When using the COPY command, and an A type device is being used, an S type may NOT be copied, and vice-versa. The program will check for this, and will print a diagnostic message.

An N type A model may copy a P type A model, and an N type S model may copy a P type S model.

Examples:

```
(12)M1, (G-S-D-B), 4 7 11 8                          (user hits
PARAMS:                                               carriage return)

KP, VTH, SBC, SBE, DW, DL, OL, SEP,                  (heading typed
CGOX, CDIF, DCC, DCE, WDRN, LDRN, RO, PHI             by program)

.2E-5, 0.75, .5, .6, .1, -.1, .05, 0.9, 0.3, 0.22,   (user enters
0.3, 1.1, 3., 0.5,                                    16 parameters)
125, 0.8
```

In the example above, MOSFET 1 is assigned branches 12 and 13. The default A model is assumed, and connected to nodes 4, 7, 11, and 8. The required 16 parameters are entered by the user, on as many lines as convenient. A comma at the end of the line is an automatic continuation, until the list is satisfied.

```
(14)M2, 66 22 0 5 PS COPY M1
ERROR, FIELD 7: SUPERFLUOUS OR INVALID FIELD
```

The user tried to use an S type device, and copy data, previously entered for an A type device.

```
(14)M2, 66 22 0 5 PS LIBRARY PMOSW
```

The private library (file: FELIBRY) will be searched, and the parameters listed under the device 'PMOSW' will be used as data.

3.2.13 Switches

Circuit elements may be switched from one value to another depending on the state of a switch (see section 3.2.14). A switch element is denoted by the letter: W. (The letter S is used for floating sources).

Three types of switches are allowed:

SWITCH, LATCH, and DOUBLE SWITCH.

The switch will activate, if and when, a user determined parameter value is greater than (or less than) a preset value. The switch will deactivate (return to its original state) if the user determined parameter value now becomes less than (or greater than) the preset value during the course of the simulation.

The latch will remain activated for the entire simulation once it has been activated.

The double switch will activate if and when the first preset condition is reached, and will deactivate if and when the second preset condition is reached.

The program will request information for the first switch by typing:

(n) W1 ,

where n* is the branch number automatically assigned by the program, the format for directing switches is:

SW
LA { NVn, TFn, TIME, RXn, BCn } [GT, LT] Value

DS { NVn, TFn, TIME, RXn, BCn } [GT, LT] Value1 { NVn, TFn, TIME, RXn, BCn } [GT, LT] Value2

where:

SW is used to specify a switch.

LA is used to specify a latch.

DS is used to specify a double-switch.

NVn, TFn, TIME, RXn, BCn, are respectively:

> A node voltage, a time-function (voltage source), the time (program independent variable), a special parameter used in a SET command, or a branch current. The subscript n indicates which specific parameter the user wishes to reference.

The MODIFY command may be used to alter the switch type from LA to SW or SW to LA, and to modify the value specification for the switching action.

All other fields must be altered by using the CHANGE command.

---

*Switches do not actually use a branch location in the same sense as a resistor does, for example. The branch number is only used for reference by the program.

3.2.14 Switched Element Specifications

The S, R, C, L, or T elements to be altered in value by switch actuation are specified during the language input phase, or by using the CHANGE command.

The formats of the switched elements are:

S: node-f node-t (Bval Aval Wn) [series R]
[Shunt R CU]

R: node-f node t

L, C: node-f node-t [series R Shunt R]

T: branch-f branch-t {BE, MU, GM, ZM, LM, LV} (Bval Aval Wn)

where:

node-f, node-t, branch-f, branch-t are the node terminals, or branches of the pertinent circuit elements.

Bval is the element value before switch actuation.
Aval is the element value after switch actuation.
Wn is the pertinent switch which is controlling the switched element.

The CHANGE command must be used to alter any switched element values.

A particular switch may control more than one branch at the same time.

Series or shunt resistances may NOT be controlled by switches.

3.3 TIME Command

The TIME command specifies the time regions for a transient analysis. The format is:

TIME max-delta1 finish-time1 max-delta2 finish-time2...

where

a. Max-delta -- is the maximum value of delta time for the given time region.

b. Finish-time -- is the finish time for the given time region. Each finish time must be greater than the previous finish time.

Up to ten time regions can be specified. The reader is referred to Section 2.6 for a discussion of delta time selection and control.

Examples:

TIME 1E-6 1E-4

specifies that time is to go from 0 to 1E-4 seconds, with a maximum Δt of 1E-6 seconds.

TIME 1E-6 1E-4 1E-7 2E-4

specifies time is to go from 0 to 2E-4 seconds, in two regions:

a. 0 to 1E-4, with a maximum Δt of 1E-6 seconds.

b. 1E-4 to 2E-4 seconds, with a maximum Δt of 1E-7 seconds.

NOTES

A new TIme entry overrides a previous TIme specification.

The command LIst TIme will produce the following printed message:

APPROX nnnn TIME POINTS REQUESTED

This will inform the user of approximately how many calculations are to be performed. The number "nnnn" does NOT include any time step reductions during time function slope changes, or any time step reductions encountered during convergence attempts which also require a time-step reduction.

The message is meant to be informative, in case the user has specified inadvertently an excessive number of points to be calculated.

3.4 PRINT Command

The PRINT command is used to specify the items to be printed. The format is:

$$\text{PRINT output1 output2...output4 } \left[\underline{\text{RATIO}}\ n\right]\left[\underline{\text{ON}}\ \begin{matrix}\underline{\text{FILE}}\ \text{filename}\\ \underline{\text{PRINTER}}\end{matrix}\right],$$

where

a. Output -- specifies what is to be printed and may be:

(1) NV for node voltages

(2) BC for branch currents

(3) TF for time functions (grounded voltage sources)

(4) BP for branch powers

A selected output may be specified by including its number after the output indicator, e.g., NV3 indicates node voltage 3, BC9 branch current 9, etc. Up to four selected outputs may be specified.

†b. RATIO n -- is optional and, if present, specifies the ratio of points stored for plotting to printed points (transient analysis only). If no plotting is specified, this will instead indicate the ratio of calculated points to printed points. If omitted, the ratio is assumed to be one.

c. ON FILE filename -- is optional and indicates that the printed output is to go to the specified file rather than the user's terminal.

d. ON PRINTER -- is optional and indicates that the printed output is to go to the line printer, if available.

PRINT NV BC RATIO 20†

†With the addition of the special "sieve" algorithm in the plotting routines, the option 'RATIO' may no longer be necessary (see Section 3.5, page 3-34).

specifies all node voltages and branch currents are to be printed at the user's terminal. The ratio of plotted (or calculated, if no plotting) to printed points is 20.

```
PRINT NV2 NV3 TF ON FILE ANS
```

specifies node voltages 2 and 3, and all time functions are to be printed on the file ANS. Every point stored for plotting (or calculated, if no plotting) is printed.

```
PRINT BC4 NV BC2 ON PRINTER
```

specifies branch currents 2 and 4, and all node voltages, are to be printed on the line printer, if available.

3.5 PLOT Command

The PLOT command specifies the items to be plotted. The format is:

```
PLOT  item1  [item2 ... item4] [option1 option2 ...]
```

where

a. Item1 -- specifies the item to be plotted and may be:

   (1) any Node Voltage (NV)
   (2) any Branch Current (BC)
   (3) any Branch Power (BP)
   (4) any Time Function (TF)
   (5) the HP function (TF19)

b. Item2 ... item4 are optional, and specify up to three other items to be plotted. A maximum of four plots per grid is allowed.

c. The options allowed are:

(1) None -- outputs the plot in default format. The grid of 51 x 101 points, with the time axis running horizontally, is split up into two sections, suitable for plotting on terminals having only 80-column carriages. The plot is output with enough overlap between the two sections to make it easy to join them and thus obtain a full page plot in wide format.

(2) ON FILE filename -- outputs the plot on the named file, rather than on the user's terminal.
NOTE: Printed and plotted outputs must go to two different files and may not go to a single file.

(3) ON PRINTER -- outputs that plot to the system printer, if available, in wide format: 51 x 101 points, with horizontal time axis.

(4) LPT -- outputs that plot on the terminal, in wide format, suitable for wide carriage (120-132 characters) terminals, with horizontal time axis.

(5) XTTY -- outputs the plot to the terminal, in narrow format (51 x 51 points) with the time axis running vertically down the page.

(6) RATIO n -- specifies the ratio of stored points for plotting, to the points calculated. If omitted, the ratio is assumed to be one.

(7) NOW -- specifies that the plot is to be output "now," i.e., without recomputing the solution, but by using data stored on disk from the previous calculation. This option can be used only after a plot has been printed. The items to be plotted must be similar to any items plotted previously; e.g., node voltages cannot be plotted if only branch currents were requested previously. This restriction saves disk space and reduces disk searches.

(8) TIME -- used to specify the limits on the time axis. See examples of usage which follow.

(9) OSUM -- prints only a summary table of computed values, and not the plot itself. It can be useful when used with the NOW option, to print a table of computed values versus time, without having to recompute the solution. (Note, however, that a PLOT command must have been given on the original run.)

The program will scale the X and Y axes automatically, by using the min and max values of the data as the limits. The program will then attempt to find "round" numbers, such as multiples of 1, 2, 5, etc., for the axes scales, based on the range of the data.

The user may override the default scaling option in two ways.

a. by specifying min and max values after any item,

b. by entering an asterisk ("*") after any item. This will cause the data to be plotted without any axis scaling.

A new PLot specification overrides a previous entry.

Example:

PLOT NV2

plots node voltage 2 at the user's terminal, in default format with time axis running horizontally. Every point calculated is stored for plotting. The axes are scaled automatically.

PLOT BC5 XTTY ON FILE TRP OSUM

plots branch current 5 in narrow format with the time axis running vertically down the paper. The plot is stored on the file TRP, and a table of plotted points is also included. Every point calculated is stored for plotting, and the axes are scaled automatically.

PLOT NV1 NV3 1. 5. RATIO 8

plots node voltages 1 and 3 at user's terminal in default format. Node voltage 1 is scaled automatically, but the node voltage 3 axis runs from 1. to 5. Every eighth point calculated is stored for plotting.

PLOT NV6 TF2 TIME 1E-6 5E-6 LPT ON PRINTER

plots node voltage 6 and time function 2 on the line printer, if available. The time axis runs from 1E-6 to 5E-6 seconds (data points calculated outside this range are disregarded). The other axes are scaled automatically, with "round" numbers.

PLOT NV3 NV4 NOW

plots node voltages 3 and 4 without recalculating the solution. A plot of node voltages must have been done previously.

PLOT NV1 1. 6. NV4 1. 9. TIME 0. 2E-3

plots node voltages 1 and 4 with the limits for each item and the time axis, as indicated.

PLOT NV1 * RATIO 2

plots node voltage 1, with no scaling used for the limits. Every other point calculated is plotted.

PLOT NV3 BC5 OSUM NOW

prints a table of calculated points for node voltage 3 and branch current 5 without recomputing.

NOTE

> Whenever the NOW option is used, any voltage may be printed or plotted without recalculation, similarly for BC, TF, or BP. However, if only node voltages were requested in the original Print or Plot request and the user then requests branch currents with the NOW option, the program will, of necessity, calculate the branch currents.

3.5.1 Special Sorting of Data Points

If more than 150 points are encountered when attempting to plot the data calculated and stored on disk, the plotting routine will automatically enter a "sieve" routine, and a message will be printed out at the user's terminal. This proprietary routine will sort up to 1000 points and plot ONLY those which are within the plot grid tolerance, i.e., ±1% vertically and ±1/2% horizontally, for a 51 x 101 point grid.

All peaks, valleys, and small perturbations (even smaller than these tolerances) will be retained by using another routine designed to cull these points from amid all others. Only if the routines fail to reduce the data points to 150 points or less will the user be required to select 1/2, 1/3, etc., points, from all those saved on the disk.

The user is always assured of obtaining all the pertinent information, without loss of detail.

3.5.2 TITLE for Plots

When the plotting routine is entered and the points to be plotted have been selected by the "sieve" routine, the message:

ENTER PLOT TITLE:

will be printed out. The user may type in a title, up to 40 characters in length, which will be printed as the first line of the plot. A new title may be entered for each plot.

If a blank line (carriage return) is entered, the circuit NAME (Section 3.2) will be used as the default title for the plot.

The program will then print out:

---- SET PAPER 3 HOLES ABOVE FOLD, THEN HIT CARRIAGE RETURN ----

This will allow the user to line up his paper properly, before the plot is printed.

If any nonblank character is entered, the message "ENTER PLOT TITLE" will be printed again, and the user may change his title.

3.6 LIST Command

The LIST command is used to obtain information about the circuit currently being analyzed. The format for the LIST command is:

LIST [option1] item 1 [option2] item2...

where

a. The items to be listed may be:

(1) The letter: V, S, R, C, L, D, Q, T, J, M, or W to obtain a listing, respectively, of grounded voltage sources, and of time functions, floating sources, resistors, capacitors, inductors, diodes, transistors, transconductances, FETS, MOS devices, and switches. Any of the above may be followed by an integer number to obtain a listing of a single item (see examples).

(2) TEMPERATURE -- lists the present value of temperature, only if other than 25° Celsius.

(3) PRINT -- lists the current print request, if any.

(4) PLOT -- lists the current plot request, if any.

b. The options which may be used are:

(1) CONNECTIONS -- lists only the nodal connections of the items which follow.

(2) FULL -- lists all information on the items which follow (the default).

(3) ON PRINTER -- lists the items which follow on the system printer, if available.

(4) ON FILE filename -- lists and saves the items which follow on the specified file. This is the standard way to save a circuit description -- after a circuit is LISTed on a file, it can be read in later using the OLD command (see Section 3.2.1).

If no items are specified, the entire circuit is listed. The format and order in which items are listed is the same as that in which they are entered. A new LIST specification overrides a previous specification.

Examples:

```
LIST
```

will list the entire circuit description at the user's terminal.

```
LIST R1
```

will list only resistor 1 at the terminal.

```
LIST CONNECTIONS R1 Q2
```

will list the nodal connections of resistor 1 and transistor 2.

LIST C R2 L D PRINT

will list all capacitors, inductors, and diodes, as well as resistor 2. The current PRINT request will also be listed.

LIST ON FILE SAMP

will list the entire circuit description on the file SAMP. The circuit will then be saved on that file, and may be restored at any later time by typing the command: OLD SAMP (see OLD Command, Section 3.2.1).

LIST CONN R1 Q FULL C

will list the nodal connections of resistor 1 and all transistors, and all information on all capacitors.

LIST TEMP ON PRINTER

will list the current temperature, if other than 25° Celsius, on the line printer (if the system has a line printer available).

LIST ON PRINTER

will list the entire circuit data on the system printer.

NOTE

To delete a previous PRint, PLot, or TIme command, the PR, PL, or TI command is entered without any arguments following it. Thus:

PR (car. ret.) deletes all previous PRint requests

PL (car. ret.) deletes all previous PLot requests

TI (car. ret.) deletes all previous TIme requests

The current values in the CRITeria Command are listed by:

LIst CRITeria

Refer to Section 3.20.

3.7 EXECUTE Command

The EXECUTE command is used to execute the circuit currently in memory. The allowable formats are:

	[EXECUTION OPTIONS]	(INITIAL CONDITIONS)
	None	VOLTAGES node, value, nod2, val2
EXECUTE	DC	CURRENTS bnch, value, brn2, val2,...
	IDC	TRY VOLTAGES....CURRENTS....

3.7.1 The EXECUTION OPTIONS specify the type of analysis to be performed. The valid execution options are:

a. None -- a TRansient analysis is performed, assuming that:

(1) All node voltages (not connected to grounded voltage sources) are zero.

(2) All branch currents (not including floating current sources) are zero.

(3) All initial conditions (capacitor voltages and inductor currents) are zero.

NOTE

A DC analysis is performed if no TIME command has been specified. The same conditions, as above, are assumed.

b. IDC -- calculate a TRansient analysis, using program calculated DC initial conditions. This option is used on a transient analysis when it is desired to have the program calculate the DC solution, and these values are used for the initial conditions. Otherwise, all initial conditions are assumed to be zero.

c. DC -- to do a DC analysis only. This corresponds to the physical situation of the circuit having operated for an infinitely long period of time with DC sources, i.e., all initial turn-on transients have decayed out so that all voltages and current derivatives are zero. Capacitors are opened and inductors are shorted, i.e., the capacitor model is reduced to its shunt resistance, and the inductor model is reduced to its series and shunt resistances. Diodes are reduced to diffusion current and leakage resistance.

A DC analysis will be assumed if no TIME command has been entered.

3.7.2 The INITIAL CONDITIONS OPTIONS specify the initial conditions for a DC solution iteration or for the initial conditions of a TRansient solution.

User-specified initial conditions options are useful for:

... providing starting values for more rapid convergence of solutions; sometimes helpful in circuits with feedback loops.

... providing control of solution of symmetrical networks such as bistable multivibrators; initial conditions can set one or the other, side on or off.

... providing initial conditions of oscillator networks with long buildup times and short oscillation periods.

... providing initial conditions for transient analysis which differ from the steady-state (IDC option) conditions.

The valid initial-conditions options are:

a. None -- DC or TRansient analysis is performed, depending on the EXECUTION OPTIONS specified, with all voltages and initial conditions at zero (see Section 3.7.1, paragraph 1).

b(1). VOLTAGES node numbers, voltage values -- Used to specify voltages for the nodes, in TRansient analysis. This option inserts the user-given voltages, instead of the voltages which would be determined by a dc solution prior to the TRansient analysis.

b(2). CURRENTS branch numbers, current values -- Used to specify currents for the branches, in TRansient analysis. This option uses the given currents instead of the currents which would be determined by a dc solution prior to the TRansient analysis.

Branch currents are a derived parameter in AITRAC, determined by finding the differences between node voltage and dividing by the branch resistance. Therefore, specifying initial values for branch currents is meaningful only for floating sources and inductor currents.

NOTE

The VOLTAGE and/or CURRENT specification effectively override a DC solution or an initial condition solution prior to a TRansient solution.

Thus, the following two commands are equivalent:

```
EXECUTE   VO ...
EXECUTE   IDC  VO ...
```

In both cases, a TRansient solution will be performed using the values specified in the VO ... entry for the initial values of node voltages (and/or CURRENTS).

Furthermore, the entry:

```
EXECUTE   DC   VO..
```

will cause the voltages specified in the VO ... entry to be printed out. In effect, the specification above states: "Use the given values as the DC solution." It can be used as a convenient way to print out a long list of initial conditions, prior to TRansient executions. Actually, no computations are performed.

c. TRY.... -- This option may be used prior to, and in conjunction with, the VOLTAGE or CURRENT options, when used to specify initial conditions. The TRY option tells AITRAC to use the indicated node voltages as a starting point for the iterative process to determine a DC solution or the DC initial conditions.

TRY may help in cases where convergence is otherwise difficult or ambiguous. For example, by using the TRY option, bistable circuits can easily be forced to start from a predefined state. Ordinarily, flip-flops will show both sides conducting equally, either both on or both off.

A DC analysis will be assumed if no TIME command has been entered.

NOTE

The use of the DO command (Section 3.12) is especially useful when many initial conditions have to be entered.

The reader is advised to refer to Section 2.4 for advice on user-specified initial conditions.

3.7.3 Examples of EXECUTE Command Usage Without User-Specified Initial Conditions

EXECUTE

The circuit currently stored in memory is executed. A transient analysis is done, with the initial conditions zero. However, if no TIME command has been entered, a DC analysis is performed.

EXECUTE IDC

The circuit is executed, and the DC solution is used as the initial conditions of the transient solution.

EXECUTE DC

A DC solution is performed with initial conditions zero.

3.7.4 Examples of EXECUTE Command Usage with User-Specified Initial Conditions

General form: EXECUTE VOLTAGES .... CURRENTS ..

Form 1: EXECUTE VOLTAGES 5.3 6.2 3.4 ....
CURRENTS 3. 4.5 8.9 10.4 ....

A TRansient execution is performed with the following initial conditions:

Node Voltage 1:	5.3 volts	Branch Current 1:	3.0 amperes
2:	6.2	2:	4.5
3:	3.4	3:	8.9
		4:	10.4

All other voltages and currents are assumed zero.

NOTE

If no node numbers specified explicitly, sequential node numbers are assumed, beginning with "1."

Form 2: EXECUTE VOLTAGES 3 5.6 2.8 7.4 8 1.4 11 12.2 13.4
CURRENTS 1 4.9 2.8 2.4 7 8.9 10 12.6

A TRansient execution is performed with the following initial conditions:

Node Voltage	3:	5.6 volts	Branch Current	1:	4.9 amperes
	4:	2.8		2:	2.8
	5:	7.4		3:	2.4
	8:	1.4		7:	8.9
	11:	12.2		10:	12.6
	12:	13.4			

All other voltages and currents are assumed zero.

NOTE

Integer numbers must be used to denote node and branch numbers. Real numbers must be used to enter voltage and current values for the initial conditions.

3.7.5 Examples of EXECUTE Commands with Initial Conditions for Starting Iterations of dc Solutions

a. EXECUTE IDC TRY VOLTAGES ... CURRENTS ...
EXECUTE --- TRY VOLTAGES ... CURRENTS ...

The starting values, specified after VOLTAGES and CURRENTS, are used as a starting point for the iterations of a dc solution, to determine the initial conditions for a TRansient analysis. These two commands will give identical results.

b. EXECUTE DC TRY VOLTAGES ..... CURRENTS

The initial values, specified after VO and CU, are used as a starting point for the iterations of a DC solution.

NOTE

The command: EXECUTE TRY VO ... CU (without the DC spec.) will NOT produce the same results as the command above. Instead a TRansient analysis will be performed. Refer to Section 3.7.5, paragraph (a) above, and to paragraph c of Section 3.7.2.

3.7.6 Diagnostic Messages for Hanging and Unconnected Nodes

If the circuit contains hanging or unconnected nodes, a diagnostic message will be printed out, before execution, to warn the user of possible circuit specification errors.

A hanging node is one which has only one component connected to it.

An unconnected node is one which has no components connected to it. Usually, an unconnected node results either from deletion of components from an original circuit, or when nodes are not numbered sequentially when inputting a new circuit.

The circuit will execute with HANGING nodes, but will NOT execute if UNCONNECTED nodes are detected.

A simple remedy is to connect a dummy resistor from any unconnected node to ground and try to execute. Renumbering of the nodes can then be done if the circuit is otherwise correct.

3.8 TEMPERATURE Command

The TEMPERATURE command is used to change the value of the ambient temperature for ALL semiconductor devices, as a global command. The format is:

TEMPERATURE value

where

value -- is the new value (expressed as a real number) of temperature. The temperature need not be entered unless it is different from 25. °C (see Section 2.2).

Example: TEMP -55.

The ambient for all devices is now set at -55° Celsius.

Individual device temperatures above or below this ambient temperature may be specified via the MODIFY Command (see Section 3.13.1 for details).

NOTE

COMPLETE temperature effects are simulated by the program, i.e., the value of θ, as well as the value of saturation currents, is altered by the TEmperature Command. Refer to Sections 2.2 and 2.2.1.

3.9 CHANGE Command

The CHANGE command is used to change the node connections and value of any element in the circuit. The format is:

CHANGE element-id

where

element-id -- identifies the element to be changed. The element may be V, S, R, C, L, D, Q, T, J, M, or W. The program will request new information for the element, and it is to be typed in with exactly the same format as when entering a new circuit description.

Example:

CHANGE R2

The program responds:

(n)R2,

where n is the branch number which is being assigned to the second resistor. The user then enters the new information for R2.

NOTE

Most element values may be altered using the MODIFY command, if the node connections are not to be changed. If the node connections are changed, however, the entire set of data for that element must be entered.

3.10 ADD Command

The ADD command is used to add a new element to the circuit. The format is:

ADD element-id

where

element-id -- identifies the type of element to be added (i.e., either V, S, R, C, L, D, Q, T, J, M, or W). Information will be requested by the program on the new element. After entering a new element, the branches in the circuit will be renumbered. The user should do a LIST command to obtain the new branch assignments.

Example:

ADD L

The program responds:

(m)Ln,

where m is the branch number being assigned to the new inductor and n is the number of the new inductor. The user then enters the inductor information.

3.11 DELETE Command

The DELETE command is used to delete components from a circuit. The format is:

DElete component-id

where

component-id -- identifies the component to be deleted: i.e., S, R, C, L, D, Q, T, J, M, or W.

NOTE

Grounded voltage sources and time functions cannot be deleted. They may be set to 0.0 if not needed.

The components specified in the DElete command are removed from the circuit. Caution must be used when deleting components, as some nodes may be left "hanging" (only one component connected to a node) or "unconnected" (nodes numbered out of sequence).

After a branch or component is deleted, the program automatically renumbers all the subsequent branches. The user should LIst his circuit to obtain the new numbering.

Transconductances which use branches which were NOT deleted, will have NEW branch numbers assigned to them automatically.

Example:

```
DELETE  Q3
```

Transistor Q3 is removed from the circuit, and all subsequent branch numbers will be renumbered. Note that subsequent branches will differ by "2," since two branches were actually removed when a transistor was deleted. Only one branch for all other components.

NOTE

If a branch which references a transconductance is deleted, a warning message will be printed out to inform the user For example:

(part of a circuit description)

```
...
...
(1)  R1,   0  1  1K
(2)  R2,   6  9  330.
...
(16) T1,   1  2  BETA 100.
...
```

The command: DELETE R2 will produce the following message:

"WARNING REFERENCED BY T1 CONNECTION DATA NOT ADJUSTED FOR T1"

R2 will be deleted, but if T1 is listed it will show that branch 2 is still referenced by T1, and the user must take appropriate action.

3.12 DO Command

The DO command executes a series of commands stored on a file. The format is:

DO filename [LIST]

where

a. Filename -- identifies the file which contains the commands to be executed. Any command described in Section 3 may be stored on the file, except the NEW and OLD commands

b. If the LIST option is present, the commands will be listed as they are executed.

Example:

Assume the file COM contains:

```
PRINT NV1
EXECUTE VOLTAGES 1. 4. 6. CURRENTS 3. 9. 2. 4.
```

The command:

DO COM

will execute those commands.

This command is very useful when executing problems which have many initial conditions which must be set prior to execution.

3.13 MODIFY Command

The MODIFY command allows users to alter element values selectively. It is used instead of the CHANGE Command whenever the nodal connections are not changed or if only a single semiconductor parameter needs to be altered. The format is:

MOdify element-id [parameter] value

where

a. Element-id -- designates the element whose value is to be modified and can be: S, R, C, L, D, Q, T, J, M, or W, followed by an integer; e.g., R5, C3, W125, etc. If the element to be modified is a semiconductor, then the PARAM field must be included to indicate the particular semiconductor parameter to be modified; e.g., HFEN, HFEI, IS, etc. Refer to Sections 3.2.7, 3.2.8, 3.2.11, and 3.2.12 for descriptions of the semiconductor parameters.

b. Value -- is a real number, specifying the new value to be given to the element.

Examples:

```
MOdify R12  5.1K
```

changes the value of resistor 12 to 5.1 kilohms.

```
MO  Q3  ME  1.22
```

changes the value of ME in Q3 to 1.22.

The modified values remain in effect until they are again changed by another MOdify or CHange command.

The MOdify command also accepts: P1, P2, ... P16 instead of the actual semiconductor parameter names. The numbers correspond to the order in which the parameters are input normally. Thus, the following two commands are equivalent:

```
MO Q3 ME 1.22   or   MO Q3 P14 1.22
```

NOTES

1. Node connections and series and shunt resistors in capacitors and inductors and in floating sources can NOT be altered with the MODIFY command. To change these, the CHANGE command must be used.

2. Grounded voltage sources and time functions can only be changed by using the CHANGE command.

3.13.1 MODIFY Command (Extensions)

Additional parameter modifications may be made using the MODIFY Command. The format is:

MODIFY device-id parameter [value]

The devices may be: diodes (D), BJT's (Q), JFET's (J), or MOS (M). The parameters which may be modified are as follows:

	BJT, JFET, MOS	Alternate Spec.	Diodes
Initial states	NORMAL	ON	NORMAL
	INVERTED	OFF, REVERSED	INVERTED
	ZERO	(default state)	ZERO
	SATURATED		
	BREAKDOWN		
Temp. diff. from ambient	TEMP value	†	TEMP value
Energy gap voltage	EG value	††	EG value
Type	NPN , PNP		
Examples:	MODIFY D3 EG .67		(Ge diode specification)
	MODIFY J4 IN		(J4 initially inverted mode)
	MO Q13 TE -38.5		(Q13 is -38.5°C below ambient)
	MO M7 SA		(M7 initially saturated)
	MO Q6 PNP		(Q6 changed from NPN to PNP)

NOTE

The EG, TE, and initial states (NOrmal, INverted, etc.) must all be specified before the NPN or PNP and the SEARCH or USE options are entered on the same line.

†Default = 0.0, i.e., Device temp = ambient temp (-273 ≤ TE ≤ 1000.)

††Default = 1.11, i.e., Silicon Eg voltage. EG > 0.0

When a COPY instruction is used, only the EG specification is copied for the device copying another device. Thus, if

```
(27) Q1  3 4 7  SAT  TE 125. EG  .75 PNP SE 2N3906
(29) Q2  5 8 2  COPY Q1
```

ONLY the EG value and the parameters entered from the device library will be used for Q2. Thus, Q2 will be at ZERO (the default), its temperature will be equal to that set in the TEmperature Command (Section 3.8), and an NPN type will be assumed.

The LIST command will print out all the semiconductor device parameters which are set by the MODIFY command.

3.13.2 Modification of D, Q, J, and M Parameters

The use of the MODIFY command described in Section 3.13 may be ambiguous when a semiconductor device is copying another device. Consider the following portion of a circuit.

```
  ----
  ----
(27) Q3 3 4 5  COPY  Q1
(29) Q4 7 6 2  COPY  Q2
  -----

  * MODIFY Q3 HFEN 75.
```

Does the user intend to MODIFY HFEN of Q3 ONLY? Or does he want to modify ALL the devices which COPY each other? This ambiguity must be resolved, and the MODIFY command works as follows:

a. When a device is copying another device, the message:
   "IGNORED-USE: CHANGE, INSERT CONNECTIONS
   THEN COPY Qn" (or Dn)
   The user must issue a CHANGE command, re-enter the node connections, and use the COPY command to make the device copy itself, i.e., Q nn N1 N2 N3 COPY Q nn. Refer to the last four paragraphs of Section 3.2.8.

b. When the first device used by subsequent devices in a COPY construction is modified, then ALL the devices which copy this first device are modified.

3.14 SAVE Command

The SAVE command has the format:

SAVE filename

It saves the circuit description on a disk file, with the "filename" given. This command is identical in function to the alternate command:

LIST ON FILE filename

3.15 SUMMARY Command

When the user types SUMMARY (abbreviation SU), AITRAC will type the number of nodes in the circuit, as well as the number of each of the different element types in the circuit (i.e., resistors, capacitors, transistors, etc.).

3.16 BC, NV, and BP Commands

The commands, BC, NV, BP are used to obtain any branch currents, voltages, or powers after a solution has been obtained. The format is:

```
BC
NV [n]
BP
```

The commands type out the requested node voltages, branch currents, or powers as of the last circuit execution. If the n is omitted, all of the items are typed out; otherwise, just the n'th item will be typed out. The command is quite useful during a DC solution in typing out circuit values other than those requested during execution (via selective PRINT) without having to recompute the solution.

3.17 SCALE Command

The SCALE command is used to scale voltage or HP sources. The format is:

SCALE [Vn / V / HP] value

In the first form, source Vn is scaled by the factor given. In the second form, all voltage sources are scaled by the value given. This includes HP sources if there are any in the circuit. In the third form, only the HP source is scaled. Only the AMPLITUDES are scaled; the time coordinates are NOT altered.

Examples:

SCALE V3 3.	Multiplies all amplitudes in V3 by the factor 3.
SCALE V .25	All V sources and HP sources are multiplied by 1/4
SCALE HP 5E6	The HP function is multiplied by the factor: 5E6

NOTE

The S sources are NOT scaled by the SCALE command unless they were defined with a "USE Vn" option in the input. The SCALE command actually changes the values of the sources so that the scaled values will be saved and listed.

3.18 MAKE Command

The command, MAKE, is used to create private diode and transistor parameter files. The format is:

MAKE [D] filename [CIRCUS / SCEPTRE / SPICE / Qn / Dn]

If the second field is D, the user is creating a diode parameter file; otherwise, transistor is assumed. If the fourth field is omitted, the program responds by asking for the appropriate AITRAC parameters. After entering all the parameters, they are written on file with "filename" given.

If the fourth field is CIRCUS, SCEPTRE, or SPICE, the program responds by asking for the appropriate diode or transistor CIRCUS, SCEPTRE, or SPICE parameters. The user should type in the parameters; they will then be automatically converted to AITRAC parameters and written on the file with the given "filename".

If the fourth field is Qn, then the parameters from the n'th transistor of the circuit currently in memory are written on file. If the fourth field is Dn, then the n'th diode is used.

In all of the above cases, the generated file is intended to be used with the USE option on diode and transistor input. The MAKE command provides a convenient way to generate private diode and transistor libraries without the use of external editing programs.

NOTE

> The MAKE command can be aborted at any time by typing in the letter Q for any requested parameter.

3.19 TOLERANCE Command

The TOLERANCE command is used to change the value of the DC convergence tolerance (TOL1), which is normally set at 1. E-5. The format is:

TOLERANCE value

The Command: LIst TOlerance will type out the current value of TOlerance, if other than the default value of 1. E-5.

If the DEFAult option is used in the CRITeria Command, the value of TOlerance is automatically reset to 1. E-5. Refer to Section 3. 20 below.

3.20 CRITERIA Command

The CRITERIA command is used to set various program control variables, in order to modify some modes of operation in certain calculation algorithms.

NOTE

> Indiscriminate use of these parameter settings is to be avoided, especially by the inexperienced user. The program defaults these parameters to values which give most accurate results, with the minimum computer time, for most circuits.
>
> Sometimes a particular circuit requires particular algorithm adjustments, but this is usually a rare occurrence. Extreme care should always be exercised

Control Variable	Default Value	Description and Functions
TOL1	1. E-05	DC node voltage convergence tolerance.
TOLD	0. 1	Diffusion curve current tolerance for DC analysis.
TOLT	0. 1	Diffusion curve current tolerance for TR analysis.
TOMI	1. E-20	Minimum value of Δt. Program will terminate if $\Delta t \leq$ TOMI.
ITOT	3	Δt is not increased if number of iterations at any time point is $\leq$ ITOT.
ITTA	10	Δt is reduced by a factor of 2 if number of iterations at any time is $\leq$ ITTA.
IAW	10	Max solution attempts at any time before solution is accepted and program continues.
IDC	200	Max number of iterations for DC solution.
ALGORITHM CONTROLS:		
MU	0. 5	Trapezoidal-rectangular integration control: $0.5 \leq MU \leq 1.0$. MU = 1.0 corresponds to rectangular.
TRAC	TRAC	Original TRAC (current iteration) algorithm for diode curve solution.
SPC1	TRAC	SPICE-I Program (voltage iteration) algorithm for diode curve solution.
SPC2	TRAC	SPICE-II Program (voltage and current iteration) algorithm for diode curve solution.
DEFAult	[TRAC, MU = 0. 5]	All control variables are reset to their DEFAULT values.

NOTE

All specifications set in the CRITeria Command remain in effect for all subsequent calculations, until they are reset, explicitly, by the user. The NEW and OLD commands do NOT reset any of the control variables; therefore, the DEFAult option should be used at the beginning of a new problem, if the default variables are desired.

Most CRITERIA command control-variable subspecifications use FOUR LETTER abbreviations, except as noted below. The form of the command is:

CRITERIA

and the program will respond with: CRITERIA:

The user may now enter any option listed, after which the program will ask:

VALUE:

if a numerical value is necessary.

To exit from the CRITeria Command, a blank line (carriage return) is typed in.

3.20.1 CRITERIA Parameter Setting

The current values of the various CRITERIA Control-Variable Parameters may be obtained with the LIst Command. The format is:

LIst CRITeria

CAUTION

The SPC1 and SPC2 options were added to allow the user to have several convergence algorithms available, within the same program. To date, no single convergence algorithm has been found which always converges or which gives the minimum number of iterations on all circuits.

The original TRAC algorithm (current iteration on diode diffusion curve) performs best, in general, when all diodes and transistors are initialized at ZERO; i.e., all junctions are started off with zero volts across them.

In some cases, such as symmetrical flip-flops, one of the stages should be initialized ON, to obtain the correct solution.

The capability of specifying the states of any, or all, semiconductor devices via the MODIFY Command gives the user full control over the method of solution for any particular circuit he wishes to analyze.

Both the SPC1 and the SPC2 options ALWAYS start the semiconductor devices in their NORMAL (ON) states. The TOL1 criterion is also automatically set at 1. E-3, whereas TRAC uses 1. E-5.

For both SPC1 and SPC2 specifications, the semiconductor states: NORM, SAT, INVE, OFF, etc., are IGNORED, as these algorithms are designed for starting in the normal, on, condition.

It is advisable for the user to experiment with a single circuit with which he is already familiar, in order to gage the effects of each option on the speed (number of iterations) and accuracy of the solutions obtained (see Section 3.21, Inform Command).

In general, each algorithm will produce slightly different results, especially for devices which are operating in an OFF state in the circuit; e.g., diodes with reverse voltages. This is true, even if the TOL1 criterion is adjusted to be 1.E-5 for each method of convergence.

3.21 STEP Command

The STEP command is used to prevent a time-step reduction from occurring during a TRansient analysis. The format is:

STEP

A warning message:

"NO PROGRAM REDUCTION OF TIME STEP"

is printed out, when the calculation phase is entered.

Automatic time-step reduction under program control is reset by again entering the command STEP.

NOTE

The STEP command remains in effect for all subsequent calculations, unless reset, explicitly, by the user.

The STEP command is NOT reset by the DEFAult option in the CRITeria command (refer to Section 3.20).

The STEP command does not eliminate the time-step reductions which occur during a straight-line segment of each Time Function or of the HP Function.

Preventing the program from reducing the time step will usually lead to NON CONVERGENCE problems.

The use of the STEP option is provided primarily as a tool to simulate other programs, for which smaller time steps must be used to obtain satisfactory results.

The user should refer to Section 3.20 and study the effect of the MU option under CRITERIA in order to obtain a different control over the integration algorithm used in this program.

3.22 INFORM Command

The INFORM command is used mainly as a program debugging tool during program development. It is NOT meant to be a standard user feature, nor is this command function maintained or guaranteed. The format is:

INFORM

The program will print out: WHICH?

The user has the following options:

COUNTS	which will print out, after an execution: KONDCS = nn, KONTRS = mm, KONSTP = jj showing the number of DC and TR iterations, as well as the number of time-step reductions during a DC and transient solution.

The remaining commands are strictly for debugging purposes.

SPARsity	will print out the sparsity of the original circuit matrix, and the sparsity after optimal re-ordering. SPAR(after) SPAR(before) indicate a reduction of zero terms.
INTErnal	Internal circuit node numbering.
EXTErior	External (user input) node numbering.
NODE	Nodal conversion vector, from user to internal node numbering.
DUMP	To obtain a dump of the H matrix and T vector. The program will respond: PLACE, TIMES? The user enters a number between 1 and 13 for PLACE, and a number 1-N for the number of times he wishes to obtain a printout.

The following information is for reference only. It is not maintained or documented.

The user should not use these options unless requested to do so specifically for tests and debugging purposes by Berne.

Dump No.	What	Where
1	HT (matrix and T vector)	SEQSOL
2	PT (pointers)	SEQSOL
3	HT	SEQSOL
4	PT	SEQSOL
5	HT	SEQSOL
6	PT	SEQSOL
7	HT	TRAC
8	Initial PT	SPARSE
9	PT	SPARSE
10	PT	RENUMB
11	PT	INSERT
12	PT	BUILD
13	PT	FIND

3.23 On-Line Interrupt

The NOS Time Sharing System allows an external interrupt character to be recognized during execution.

The execution of any circuit may be interrupted at any time by typing in the character "S" or "I", at the TTY terminal, during a transient simulation.

Execution will be interrupted as soon as the contents of the output buffers are cleared.

The program will respond with:

INT(E)...

indicating an External interrupt.

The user may then type in any of the following commands:

a. A -- ABORT current execution, and return to the AITRAC input language. The circuit can then be modified, a new circuit can be run, etc.

b. S -- SUPPRESS future typeout on the terminal until printout is restored by an R command.

c. R -- RESTORE output at the terminal.

d. X, option -- EXAMINE current status of the circuit, where the "option" may be:

   (1) NV, BC, or TF to examine any node voltages, branch currents, or time functions. Example: X, BC will print out the current value of ALL branch currents.

   (2) NVn, BCn, TFn, to examine a particular node voltage, branch current, or time function. Example: X, NV5 will type out the current value of voltage at node 5.

(3) T -- TIME, types out the current value of time in the simulation.
The user can thus check the progress of the simulation if there seem to be convergence problems.

(4) D -- DELTA time, types out the current value of the time step, again valuable if convergence problems are encountered.

(5) N -- NUMBER of times the simultaneous equations for this circuit have been solved; e.g., X, N.

e. C, option -- CONTINUE execution of the circuit, and the "option" may be:

(1) None, i.e., just the command "C," to continue.

(2) T = value -- continues execution of the circuit and automatically interrupts execution when the simulation time reaches the specified value. Example: C, T = 5E-6 continues the execution until time = 5 microseconds, or surpasses it.

(3) P -- continues execution, until the next regular printout of answers is obtained.

(4) N + i -- continue execution, until the simulataneous equations have been solved "i" more times; e.g., C, N + 10 will perform 10 more iterations.

(5) I -- continues execution until equations have been solved one more time. Equivalent to: N + 1; e.g., C, I (or C, N + 1).

Of course, after execution is resumed, AITRAC may be interrupted again at any time, by typing "S" or "I".

Whenever the interrupt routine is entered, the program types:

INT(letter)

where "letter" indicates the cause of the interrupt, as follows:

a. E -- Eternal interrupt caused by

b. P -- Previous C,P request

c. N -- Previous C,N request

d. T -- Previous C,T - value request

e. I -- Previous C,I request

3.24 QUIT Command

The QUIT command is used to exit from AITRAC and return the user to the System level. The format is:

QUIT

The user will be returned to System level, where he can log out or use the time-sharing system for whatever purpose he wishes.

NOTE

For convenience, three temporary files are stored on the user's local area when a plot is performed; these files are named:

TRACTMP - stores the data points for plotting

TRA1TMP - stores the COMMON area while plotting

TRA2TMP - stores the number of points on disk

These files may be retained on disk by using the "REPLACE, filename" command on each of the above three files. The user may come back later and make more plots, by using the "NOW" option, without recomputing, even after logging out.

If the user is sure that no further plots are needed at a later time, these files should be deleted from his disk area.

The system command to delete these files is:

PURGE, TRACTMP, TRA1TMP, TRA2TMP

4. Radiation Effects

4.1 Description

AITRAC is equipped to perform radiation analysis accurately. To study the effects of radiation, one must define the hole-pair generating function, as well as the appropriate semiconductor parameters.

a. i_{PPD} = Diode primary photocurrent (diodes)

b. i_{PPC} = Collector junction diode primary photocurrent (transistors)

c. i_{PPE} = Emitter junction diode primary photocurrent (transistors)

For a diode, the radiation induced photocurrent $i_{PPD}(t)$ can be calculated by one of the following expressions:

Type 0 - First-order differential equation solution to arbitrary shape gamma-dot(t):

$$i_{PPD}(t) + TD^* \frac{d|i_{PPD}(t)|}{dt} = IPPD^*[\text{gamma-dot}(t)]$$

Type 1 - Error function (ERF) solution of $i_{PPD}(t)$ to unit-step changes of gamma-dot(t)

$$i_{PPD}(t) = IPPD \sum_{j=1}^{NU} \left| \text{gamma-dot}(t)_j U(t - T_j) ERF \left| \frac{t - T_j}{2*TD} \right|^{1/2} \right|$$

where: NU = number of unit steps in gamma-dot(t) occurring at times T_j.

For an ideal rectangular pulse NU = 2.

For a transistor, the radiation-induced photocurrents $i_{PPE}(t)$ and $i_{PPC}(t)$ can be calculated by one of the following:

Type 0 - First-order differential equation solution to arbitrary shape gamma-dot(t):

$$i_{PPC}(t) + TI^* \frac{d|i_{PPC}(t)|}{dt} = IPPC^*[\text{gamma-dot}(t)]$$

$$i_{PPE}(t) + TN^* \frac{d|i_{PPE}(t)|}{dt} = IPPE^*[\text{gamma-dot}(t)]$$

Type 1 - Error function solution of $i_{PP}(t)$, to unit step changes of gamma-dot(t):

$$i_{PPC}(t) = IPPC \sum_{j=1}^{NU} \left| \text{gamma-dot}(t)_j U(t - T_j) ERF \left| \frac{t - T_j}{2*TI} \right|^{1/2} \right|$$

$$i_{PPE}(t) = IPPE \sum_{j-1}^{NU} \left| \text{gamma-dot}(t)_j U(t - T_j) ERF \left| \frac{t - T_j}{2*TN} \right|^{1/2} \right|$$

NU = number of unit steps in gamma-dot(t) occurring at times T_j.

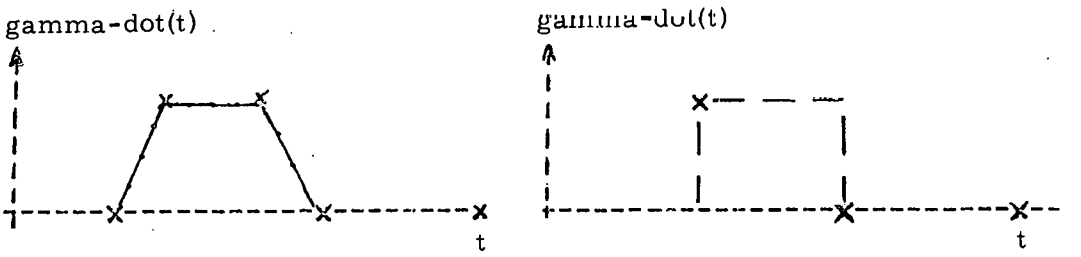
The hole-pair generating function is a time function which represents the ionizing dose rate gamma-dot(t). Two types of semiconductor responses are possible with the ATTRAC program.

Type 0 - The gamma-dot(t) function is treated exactly as the usual time functions. Each induced primary photocurrent is the solution of a first-order differential equation having gamma-dot(t) function as a driving function. It is described by a set of ramps. The coordinates of the straight line intersections of each ramp define the Type 0 function.

Type 1 - The gamma-dot(t) function is described by discrete plateau changes. Each induced primary photocurrent will be the sum of error functions. It is described by a set of "plateaus."

The coordinates at each discrete plateau level change define the Type 1 function.

Figure IV-1 illustrates how a rectangular pulse would be input for each type.



A. Type 0 B. Type 1

Figure IV-1. Input Description of Type 0 and Type 1 gamma-dot(t)

Gamma-dot ($\dot{\gamma}$) = Transient Induced Dose Rate (rads/sec)

The coordinates at the straight-line intersections define the type zero function. The coordinates at each discrete plateau level change define the Type 1 function.

4.2 Data Input

The diode and transistor radiation parameters are entered with the normal diode and transistor parameter input. IPPD is the seventh diode parameter entered, while IPPC and IPPE are the 15th and 16th transistor parameters necessary. See Sections 2.1.7 and 2.1.8 for a complete description of how to input semiconductor

parameters. (NOTE: Radiation parameters are not currently stored in the public semiconductor parameter library; values of 0. will be automatically retrieved, so the user should issue a MODIFY command to change the radiation parameters to the values desired.)

The hole-pair generating function is input by use of the AITRAC command: HP. The format for the HP command is:

$$\mathrm{HP}\ \begin{pmatrix}0\\1\end{pmatrix}\ \mathrm{Value0,\ time1\ value1,\ time2\ value2,\ .\ .\ ,timen\ valuen}$$

where

a. The second field is an integer 0 or 1 to denotc either the Type 0 or Type 1 function.

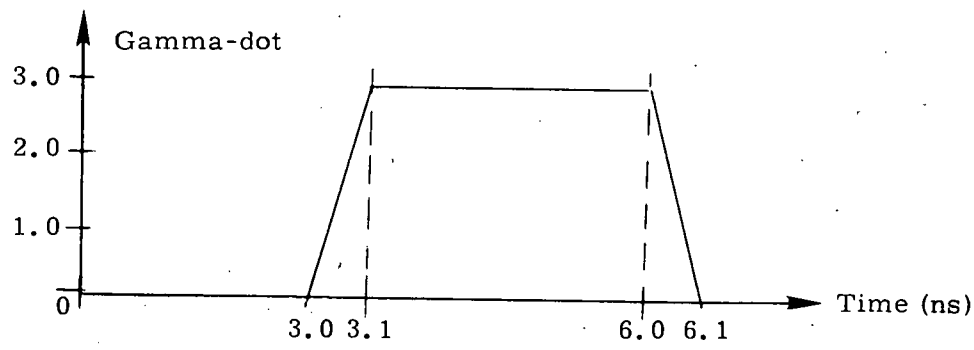
b. The third field is a real number denoting the value of the function at t = 0.

c. The fourth and subsequent fields are pairs of real numbers denoting the coordinates of the hole-pair generating function (time and value) as described above. The hole-pair generating function may be described by up to 15 straight-line segments.

Example:

HP 0 0. 3E-9 0. 3.1E-9 3. 6E-9 3. 6.1E-9 0.

describes the following Type 0 function.



To list or verify the values of the HP functions, the LIST command may be used. The format is:

LIST HP

This will list the type and the values of the HP function.

NOTE

The hole-pair generating function is stored internally as the 19th Time Function.

The following commands must be used to print or plot the HP function.

PRINT TF19

or

PLOT TF19

4.3 Improved Type 0 Response

The Type 0 response was approximated originally by using a single time constant exponential function to simulate the Error Function response† characteristic of radiation phenomena.

This approximation is very coarse and produces large errors, especially when large time steps are used to save CPU time.

A new, proprietary algorithm has been developed and incorporated which uses two exponential terms to approximate the theoretical error-function response.

A 40-fold error reduction is obtained with minimal additional CPU time. Moreover, larger time steps may be used without incurring accuracy problems.

The results obtained with the Type 0 response are now practically identical to those obtained with Type 1, with much less CPU time.

---

†See: VOL-II, TRAC Manual, pages 27, ff.
Harry Diamond Labs DAAG39-68-C-0041

Moreover, the advantage of allowing a time function to be specified by a series of ramps, in the Type 0 description, instead of the plateaus, in the Type 1 description, makes the modeling much more flexible.

Technical Note RAD-5 "Improved Radiation Response Calculations" is available from Berne Electronics for those users who wish more detailed information.

5. Special Equations

AITRAC allows the experienced user to write FORTRAN equations and control statements in order to modify standard model parameters, create new models, and define auxiliary quantities (e.g., peak power, energy, etc.) to be printed or plotted.

When standard models are not sufficient to analyze the problem at hand, additional interaction with AITRAC is provided via the FORTRAN subroutine SPEQ.

Some knowledge of FORTRAN and familiarity with the System's Text Editor are needed to make efficient use of this extremely versatile and powerful feature. However, many of the system operations required have been made "transparent" to the user in order to keep the overall operation as simple and "foolproof" as possible. Thus, the mechanics of compiling, loading, etc., are done by a single procedure file. The guiding philosophy is to help the engineer analyze his problem and not force him to spend time mastering computer system intricacies.

Once the user has modified his SPEQ subroutine to include his model modifications and additions, the command:

MAKEQ (FN = fname)

is required to compile, load, and modify the program's overlay structure to incorporate the user's special equations.

5.1 Description

Users proficient in FORTRAN may include special equations to supplement the capabilities and further extend the flexibility of the AITRAC program. There are three main applications for special equations:

a. Definition of special time functions.

b. Addition of auxiliary equations.

c. Addition of nonstandard models.

Special equations are added to a subroutine in the AITRAC program called SPEQ (see Figure V-1). The subroutine is divided into four main parts.

```
        SUBROUTINE SPEQ(KK4)
      INTEGER     KK4
C
C$ALPHA
C---     DOUBLE PRECISION H,T,V,V1
         COMMON/ALPHA/ISETUP,IPOINT(618),H(901),TOLJD,TOLJT,
     *   IIPAT(30),RPAT(30),A(1),E(19,16,2),O,T(109),V(169),
     1   BV,CT(11,2),DI(30),DM,DR(30),DV(30),EI(20),E2(19),
     2   ST(19,2),TE,TI,TO,V1(169),
     3   V2(169),ABC,BCI(30),BCR(30),BCV(30),BEI(30),
     4   BER(30),BEV(30),DIP(30),E04,EP1(20),EP2(20),ERR,QCR(30,16),
     5   SI1,TEX,TE1,TOO(19),TOP,TO1,BCDT,BCIP(30),
     6   BCUR(290),BEIP(30),COND(30),CONP(200),CONT(30),DTIM,GRSP,
     7   PPIC(30),PPID(30),PPIE(30),PWCR(30),PWTR(30),QTAN(30,28),
     8   RMAG,TE11,TOL1,TOL2,TOT2,TO11,VAL1(200),VAL2(200),
     9   VAL3(200),BCDT1,BCUR1(290),PPIC1(30),PPID1(30),PPIE1(30),
     A   SYMB1,SYMB2,TOLPL,TOMIN,IAW,JSJ,IDC,NCUT,
     B   KTOT,IBC,NAW,ITOT,KTOT1,NV,NG,JJ,N3B,NBRAN,NBLOCK,
     C   ITOT1,JUJ,N2B,ISTRK,ISYM4,NZ,ND,NBB,NVM,NNNN1,ICP,
     D   IPRINT,NTPLP,NOPOUT(11),NA,NZ1,JJJ,KE,
     E   IFGDTC,JV,IPX,NA1,NT,NB,N1B,JCAT,NCORE,JVJ,NP,IPLCON,
     F   ISYM3,JG(19)
         DIMENSION NOP(4)
         EQUIVALENCE (ND1,NOPOUT(1)),(NOP(1),NOPOUT(2)),
     1   (NCR,NOPOUT(6)),(NTR,NOPOUT(7)),(NTS,NOPOUT(8)),
     2   (NJFET,NOPOUT(9)),(NMOSFT,NOPOUT(10)),(NPSWI,NOPOUT(11))
         COMMON/ALPHAB/NOCUST,ISPICE,ISENSE,IWORST,ITRAC,TWOPI,ROOT2,
     1   ABSZER,BASEDG,BOLTZQ,BANDEG,VCRIT,UM,MENU,KKLPAR,VAL4(200),
     2   NOSWI(200),XMU(11,2)
CALPHA$
         DIMENSION RX(25),IX(25)
         EQUIVALENCE (NRX,V(101)),(NIX,V(102)),
     1   (RX(1),V(111)),(IX(1),V(136))
C
       GO TO (9000,9003,9002,9001), KK4
9000   IF(ISETUP .EQ. 1) GOTO 9004
C   CALLS TO SUBROUTINE * BUILD * INSERTED HERE
       RETURN
9004   CONTINUE
C        PART 1 - MATRIX AND TIME FUNCTIONS HERE
       RETURN
 9003  CONTINUE
C      PART 2
       RETURN
 9002  CONTINUE
C      PART 3
       RETURN
 9001  CONTINUE
C      PART 4 - AUXILIARY EQUATIONS AFTER THIS CARD.
       RETURN
       END
```

Figure V-1. User's Special Equations Interface: SPEQ

Part 1 is used for the definition of special, forcing, and time functions and nonstandard model equations. Modifications of H matrix and T vector terms are also inserted in this first section.

Part 2 is used for special convergence equations. The iteration procedure and convergence criteria for any nonlinear models generated by the user must be placed in this second section.

Part 3 is used for special solution acceptance equations. Equations which do not affect the iteration directly, but which can affect the acceptance of an otherwise converged solution, are placed in this third section.

In addition, Part 0 is entered only once, the first time that SPEQ is called by AITRAC. In this section the user must define any nodes which are not defined in the standard language input.

Since sparse matrix techniques are used to conserve core size, there is no simple relationship between an H matrix and T vector element and its node number(s). Thus, the pointer system used to keep track of each of these elements must be correctly initialized and proper core locations set up before the user can access the correct H and T terms.

Entering nonstandard time functions:

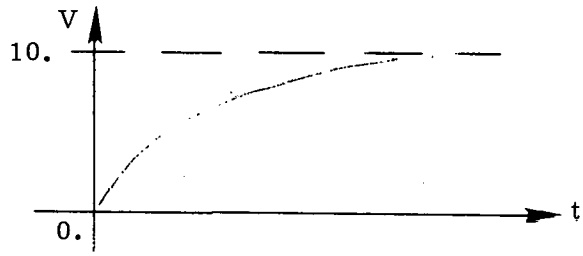
Nonstandard time functions are entered in Part 1 of the SPEQ subroutine. Time functions in the AITRAC program are stored in the FORTRAN array EI. See† Table I, User's Table, for a complete identification of AITRAC program variables; these variables are kept in COMMON and are accessible from the SPEQ subroutine. Thus, for example, to code time function 3 as a nonstandard function, one might enter the FORTRAN statement

```
EI(3) = 10.*[1. - EXP(-TE/7.1128)] ,
```

where TE is the AITRAC variable for time.

†Table I follows Section 5.7, and will be expanded and updated periodically.

This represents an exponential voltage source, with a time constant of 7.1128 seconds which reaches an asymptotic value of 10.0 volts.



This statement is to be placed in Part 1 of the SPEQ subroutine prior to the RETURN statement. When nonstandard time functions are used, the time function must first be defined in the AITRAC input as existing; that is, dummy values of 0. may be entered. For example, during AITRAC input, the statement

V3. 0.

defines time function 3 as existent, and the corresponding equation for EI(3) may then be coded in SPEQ. Of course, any time function may be printed or plotted by use of the TF option, in the PRINT or PLOT statements.

5.2 Coding Auxiliary Equations

Parameter values may be modified based on a dependent variable. The value generated will be used for the next time solution. Such auxiliary equations used to modify a parameter value are inserted in Part 4 of the SPEQ subroutine.
Example: the value of resistor 2 (assigned as branch 4) is to be changed from its nominal value of 1 kilohm to 10 kilohms if node voltage 5 is greater than 20 volts. Then (by reference to Table I), the appropriate FORTRAN statement to be inserted in Part 4 would be:

IF(V(5).GT.20.) VAL2(4) = 10E3

The reader should study the examples in Appendix E and run the problems shown to familiarize himself with the operation of AITRAC using auxiliary equations.

5.3 Coding Nonstandard Models

When the standard models are not adequate to represent the problem at hand, nonstandard models may be included by writing the appropriate equations in Part 1 of the SPEQ subroutine. The basic AITRAC program is designed to solve simultaneous linear node equations of the following form:

$$w_1 = h_{1,1}\, v_1 + h_{1,2}\, v_2 + \cdots + h_{1,NV}\, v_{NV}$$

$$\vdots \qquad\qquad \vdots$$

$$w_{NV} = h_{NV,1}\, v_1 + h_{NV,2}\, v_2 + \cdots + h_{NV,NV}\, v_{NV}$$

or in matrix form,

$$\begin{vmatrix} w_1 \\ \vdots \\ w_{NV} \end{vmatrix} = \begin{vmatrix} h_{1,1} & h_{1,2} & \cdots & h_{1,NV} \\ \vdots & & & \vdots \\ h_{NV,1} & h_{NV,2} & \cdots & h_{NV,NV} \end{vmatrix} * \begin{vmatrix} v_1 \\ \vdots \\ v_{NV} \end{vmatrix}$$

where

$$h_{i,j} = f_{i,j}(b_1, b_2, \ldots b_k)$$

$$w_i = f_i\ (b_1, b_2, \ldots b_k)$$

b_m = a known parameter value, such as a resistance, source voltage, transistor current gain, time, etc.

v_i = dependent node voltage unknown

NV = number of dependent nodes ($1 \geq NV \geq 100$)

Since the equations are linearly indedendent, they are solved by

$$|v| = |h|^{-1}\,|w| \ .$$

The $h_{i,j}$ terms are admittances and have units of mho. The w terms are currents and have units of amperes.

In the program, linear differential/integral equations are solved by a difference equation mechanization (e.g., capacitor and inductor standard model equations). Nonlinear equations are solved by iterative algorithms (e.g., diode and transistor standard model equations).

The matrix equations are set up in the form that follows using the AITRAC notation as listed in Table I. If auxiliary matrix equations are to be written in conjunction with the standard models equation writer, see also Appendix F.

$$\begin{vmatrix} T(1) \\ \cdot \\ \cdot \\ \cdot \\ T(NV) \end{vmatrix} = \begin{vmatrix} H(1,1) & H(1,2) & \cdots & H(1,NV) \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ H(NV,1) & H(NV,2) & \cdots & H(NV,NV) \end{vmatrix} * \begin{vmatrix} V(1) \\ \cdot \\ \cdot \\ \cdot \\ V(NV) \end{vmatrix}$$

NOTE

The elements of the matrix are NOT stored in a square array as shown above, since sparse matrix techniques are used. Pointers are set up so as to be able to store nonzero H matrix elements in a linear array. The user has available special routines, BUILD, SHTADD, and BRAVOL, for use when he wishes to access these elements.

Thus, by entering his own circuit's node numbers, the correct elements are accessed directly and transparently, regardless of the program's internal node renumbering.

5.3.1 AITRAC Program Variables

The SPEQ subroutine allows the user to interface directly with all the program variables which are in COMMON. The program variables and their description are given in Table I.

A brief look at these tables of variables will show, for example, that the program's "time" variable is stored in the variable "TE," that the beta (forward) of a transistor is the variable QTAN(N, 1), where N is the transistor's number, and that the variable V(I) is the node voltage at node I.

5.3.2 Transmitting Parameters to the SPEQ Subroutine

It is frequently useful to be able to read in auxiliary parameter values by using the AITRAC input language for use in the SPEQ routine. This can be accomplished by using the SET RX and SET IX commands. The format is:

```
SET RX value1  value 2  ... value v
SET IX numbr1  numbr 2 ... numbr n
```

The SET RX command loads real numbers: value1, value2, etc., into an array RX for use in the SPEQ routine. Likewise, the SET IX command loads the integer values: numbr1, numbr2, etc., into an array IX for use in the SPEQ routine. [The arrays RX(25) and IX(25) are stored in COMMON.] Example:

```
SET RX  1.   3.14   2.718
```

The values 1., 3.14 and 2.718 are read into the array RX, so that

RX(1) = 1. RX(2) = 3.14 RX(3) = 2.718

An extended form of the SET command is:

```
SET RX2  5.
```

which will change only RX(2) to the value 5.0. Up to 25 real and 25 integer auxiliary parameters may be stored. The values of the auxiliary variables can be checked at any time by typing the command:

```
LIST SET
```

5.4 Use of the SPEQ Subroutine with Sparse Matrix Techniques

The use of sparse matrix techniques to save core space necessitates the use of pointers to keep track of the internal reordering of the node numbers during program setup and execution.

If, in the input language section, nodes 1 through 3 are interconnected in a circuit, then the pointers are all set up during the scanning of the input data.

For example, consider the following circuit.

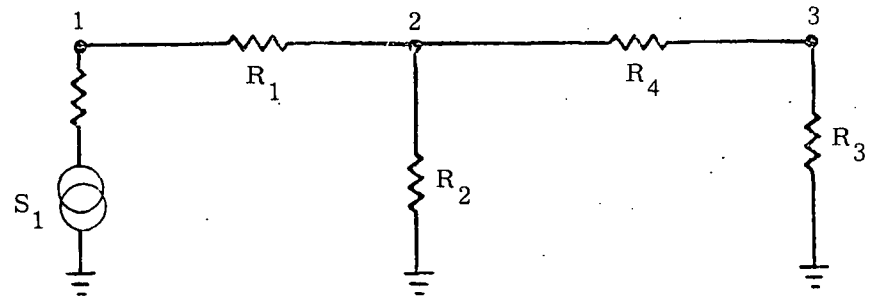


Figure V-2

Data for this circuit may be entered in the standard manner in the AITRAC input language section. Suppose now that R4 is not a linear component, but has a value which is a function of the branch voltage across it. Thus

$$R_4 = 1000./(V_2 - V_3)^4$$

or, in SPEQ program variables:

```
VAL2(4) = 1000./(V(2) - V(3))**4 .
```

Several simple steps must be followed in order to implement this model in SPEQ:

a. The pointers for nodes 2 and 3 must be set up in Part "0."

b. A model suitable for iteration of nonlinearities† and for convergence checks must be derived.

†A typical method is detailed in Section 7, pp. 39-42, of Reference 1, Volume II.

c. H matrix and T vector components must be added to incorporate the contribution of R4 to the network's operation in Part 1.

d. Convergence criteria must be defined to check if the program may increase the value of time for its next iteration, in Part 2.

e. The value of the branch current in R4 must be printed out to check if the current-voltage relationship is implemented correctly. This can be done in Part 4.

5.4.1 Pointer System Setup

If an element is to be connected between nodes N_a and N_b, the following calls MUST be included in Part 0 of SPEQ:

CALL BUILD (N_a, N_b)
CALL BUILD (N_b, N_a)

A similar pair of calls must be made for any other elements which the user may wish to connect between any other pairs of nodes. Subroutine BUILD in the AITRAC program sets up pointers and core locations for each OFF-DIAGONAL matrix term to be accessed.

NOTE

The node numbers N_a and N_b are the positive integer values of the actual node numbers between which the new element is connected. When one terminal of an element is connected to an INDEPENDENT node, i.e., a fixed voltage source (time function) or to ground, then the "effective node number" is the NEGATIVE integer number corresponding to the fixed source. For GROUND (Node 0), the proper value to be used is: -20.

For example:

a. R4 connected between V3 and Node 7, the calls would be:

```
CALL BUILD(3, 7)
CALL BUILD(7, 3)
```

b. R4 connected between Node 5 and ground:

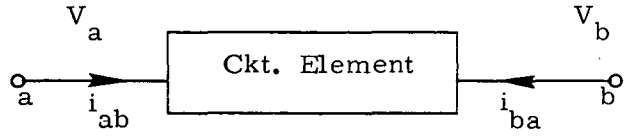
```
CALL BUILD (5, -20)
CALL BUILD (-20, 5)
```

It is good practice to insert these calls in Part 0 of SPEQ, even if a dummy element is inserted across the desired node pairs in the input language section. Then, if the dummy resistor is DELETED (in the language), the pointer system will still be set up correctly. Also, it is useful to insert a dummy element (a 100-megohm resistor, for instance) across the nodes of a special element, so that the BRANCH CURRENT may be printed or plotted conveniently, via the PRINT and PLOT commands. See Section 5.4.5.

These calls need to be made only once, on the very first pass through SPEQ. The program does this automatically by changing the variable ISETUP. The user must not alter the value of ISETUP in his own FORTRAN equations.

5.4.2 Modification of H Matrix and T Vector Terms

Any passive element between two dependent nodes obeys the following equations:



$$i_{ab} = \Delta H_{aa} V_a + \Delta H_{ab} V_b - \Delta T_a$$

$$i_{ba} = \Delta H_{bb} V_b + \Delta H_{ba} V_a - \Delta T_b$$

$$\Delta H_{aa} = \Delta H \qquad \Delta H_{bb} = \Delta H \qquad \Delta T_a = T_a$$

$$\Delta H_{ab} = -\Delta H \qquad \Delta H_{ba} = -\Delta H \qquad \Delta T_b = -\Delta T$$

If one of the nodes is an independent node, i.e., a voltage source or ground, the H matrix and T vector contributions will be different. The user is referred to Section 6 for the detailed mathematical model information.

The most general case of a complete branch is shown below. The branch contains: a series element, Y_T, a shunt element, Y_{TS}, an independent voltage source, E_T, an independent current source, I_I, and a dependent current, I_D, introduced by a transconductance term.

The general H matrix contributions are given by:

Series and Shunt Element Contributions	Transconductance Term Contributions
$\Delta H_{aa} = +(Y_T + Y_{TS})$	$\Delta H_{av} = +GM$
$\Delta H_{bb} = +(Y_T + Y_{TS})$	$\Delta H_{aw} = -GM$
$\Delta H_{ab} = -(Y_T + Y_{TS})$	$\Delta H_{bv} = -GM$
$\Delta H_{ba} = -(Y_T + Y_{TS})$	$\Delta H_{bw} = +GM$

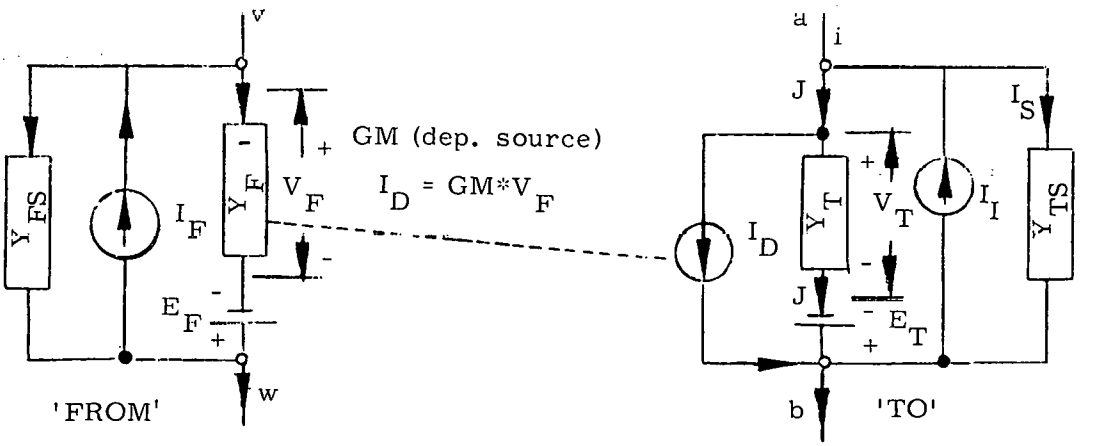
The generalized T vector contributions are given by the sum of all the currents in the "to" branch. The internal independent voltage source E_T (if present) is replaced by an equivalent current generator, using Norton's theorem:

$$I_T = -E_T * Y_T$$

Then

$$\Delta T_a = +I_I - E_T * Y_T - GM * V_F$$

$$\Delta T_b = -I_I + E_T * Y_T + GM * V_F$$



General Circuit Branches
Including Transconductances

The basic equations which also apply to the generalized "to" branch are:

Kirchhoff's Current Law: $i = J' - (I_I - I_s)$

$J' = J + I_D$

Kirchhoff's Voltage Law: $V = e + E_T$

$e = e_a - e_b$

If the dependent current I_D is not caused by a transconductance GM between the "from" and "to" branches, the equivalent GM term can always be obtained by using the relationships given in Figure III.2.

Once the correct ΔH and ΔT terms have been determined, they must be added to the existing H and T terms; the procedure for doing this has been simplified as much as possible for the user, as described in the following sections.

5.4.3 Determination of Branch Voltages

To determine the voltage across an element, such as R4 in our example, we must evaluate the following expression:

Branch Voltage - XBV = V(2) - V(3) .

However, when using sparse matrix techniques, the program's internal node numbering differs from the external (user input) numbering. Therefore, a special function, BRAVOL, is available to the user for obtaining the branch voltage across any pair of nodes, simply and directly. To obtain the voltage across any two nodes, the function is used as follows:

XBV = BRAVOL (N_a, N_b, V, EI)

where

XBV is any variable name assigned by the user

N_a is the "from" node of the new element

N_b is the "to" node of the new element

V is the array (in COMMON) which contains the present value of node voltages

EI is the array (in COMMON) which contains the present value of time functions or grounded voltage sources.

When capacitors and other energy storing elements are to be modeled, the PAST values of node voltages are required also. In this case, function BRAVOL is used as follows:

$$VPAST = BRAVOL\ (N_a, N_b, V1, EP1)$$

where

V1 is the array (in COMMON) which contains the first past value of node voltages

EP1 is the array (in COMMON) which contains the first past value of the time functions or grounded voltage sources.

Again, the positive integer values of the node numbers are entered for N_a and N_b, when independent nodes are referenced. The negative integer value is entered when referring to dependent nodes, i.e., voltage sources. The value -20 is entered for the ground node.

The voltage across two independent nodes (voltage sources and/or ground) may NOT be obtained, since an element must NOT be connected between two independent nodes. Therefore, check to see if either N_a or N_b are entered as negative values, but not both.

5.4.4 H Matrix and T Vector Terms

The updating of the H matrix and T vector terms, in Part 1, is performed automatically for the user by means of subroutine SHTADD (Simplified H and T Additions) in this manner:

$$CALL\ SHTADD\ (N_a, N_b, HTERM, TTERM)$$

where

N_a, N_b are the nodes to which the element is connected. Negative values are entered for voltage sources, and -20 for ground.

HTERM is the absolute value of the ΔH component to be added to the H matrix terms.

TTERM is the absolute value of the ΔT term to be added to the T vector.

If either the ΔH or ΔT terms are zero, an argument of 0.0 is entered for the appropriate term.

Subroutine SHTADD performs the updating of the H_{aa}, H_{ab}, H_{ba}, H_{bb}, T_a, and T_b terms automatically with only one CALL, in Part 1 of SPEQ.

5.4.5 Special Model Convergence Equations

The iteration procedure convergence criteria and tests for any special nonlinear models introduced by the user are placed in Part 2. Such tests are required when special models are nonlinear and changing sufficiently fast to require iteration within a time step; for example, in the case of diodes which obey exponential relationships. A practical example is given in the example at the end of this section.

5.4.6 Special Solution Acceptance Equations

Equations which do not affect the iteration directly, but which could affect the acceptance of an otherwise converged solution, are entered in Part 3.

5.4.7 Auxiliary Quantities and Functions of Dependent Node Voltages and of Converged Solutions

After each converged solution has been obtained, auxiliary quantities (such as branch current, branch powers, etc.) can be obtained in Part 4. Such quantities cannot be obtained until convergence has been achieved, since they are all derived from the dependent node voltages, and until convergence has been achieved, any values computed would be meaningless.

As an example, if, in Figure V-2, the ratio of the voltage across Nodes 1 and 3 and the voltage across R_3 is desired, the following equations could be used.

```
XBV13 = BRAVOL (1, 3, V, EI)
EI(4) = XBV13/V(3)  .
```

These calculations would store the desired voltage ratio into Time Function 4. Then, the user could use either PRINT TF4 or PLOT TF4, with the standard print and plot commands in the AITRAC language. Note that, first, an entry

V4, 0.0

must be included in the language section as detailed in Section 5.1.1.

5.5 Accessing the Special Equations

To access the specially created AITRAC routine, the user must issue the command:

SET SPECIAL ON

This AITRAC command should be issued after the NEW command and before the EXECUTE command.

AITRAC will use the user's special SPEQ subroutine, and perform the calculations desired.

The user may stop using the special equations feature at any time by issuing the command:

SET SPECIAL OFF .

5.6 Example of SPEQ Use

Refer to the circuit of Figure V-2. Assume that R4 has a resistance of:

$$R = 1000./e^3$$

where "e" is the branch voltage ($V_2 - V_3$) across resistor R4. The current-voltage relationship is, mathematically

$$I = e^4/1000 \quad .$$

Figure V-3 shows this characteristic.

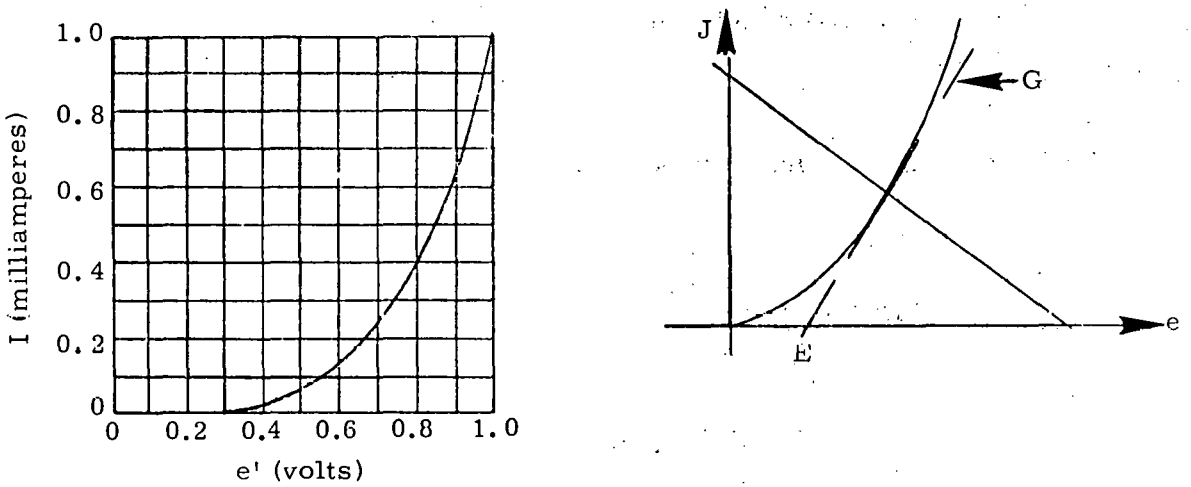


Figure V-3. Diode Characteristic and Tangent Model

A "tangent model" can now be used, in a manner similar to the diode model used in ATRAC. The following steps must be followed.

a. A nonlinear resistor is replaced by a linearized model at each iteration. Thus,

where GEE is the conductance at the tangent point; thus:

$$GEE = \frac{di}{de} = 4.*e^3/1000. = e^3/250 \quad .$$

b. The voltage EEE is the difference between the branch voltage, XBV = $V_a - V_b$, and the voltage drop across EEE. Thus,

$$EEE = XBV - J_{(XBV)} * \frac{1.}{GEE} \quad .$$

The T vector component T = DELT, is given by

$$DELT = GEE*EEE = XBV*GEE - J_{(XBV)} \quad .$$

The current, $J_{(XBV)}$, is assumed to be equal to the fourth power ideal mathematical relationship which is desired at any branch voltage XBV. Thus,

$$J_{(XBV)} = RCUR(XBV) = (XBV)^4/1000.$$

and

$$DELT = XBV*GEE - RCUR(XBV) \quad .$$

c. The ΔH and ΔT term contributions from R_4 must be added to the proper H and T terms. This is done by calling subroutine SHTADD:

```
CALL SHTADD(1, 2, GEE, DELT)
```

in Part 1.

d. The actual branch current, XCUR, is then computed in Part 2:

```
XCUR = (XBV-EEE)*GEE .
```

e. A check must be made at each iteration to see if the actual current, XCUR, obtained from the tangent model matches (within a given tolerance set up by the user) the desired current obtained from the mathematical, fourth power relationship. If the currents are within this set tolerance, thc program will automatically stop iterating and continue with the next time step, because the nonlinear CONVERGENCE FLAG, ABC, will be set equal to -1 in the subroutines for nonlinear models (diodes, transistors, etc.).

The main section of the program thinks that thcrc are no nonlinear models, since no diodes, transistors, etc., have been included in the circuit.

Therefore, if the currents do not match within a set tolerance, the nonconvergence flat, ABC, must be set to a value of +1. Thus:

```
IF(ABS((XCUR-DCUR)/XCUR).GT.TOL) ABC=1.
```

When ABC is set to +1, it indicates that convergence is NOT achieved; therefore, the program will perform another iteration, without increasing the time step.

f. If convergence is not achieved, then the tangent model must be re-linearized, with the new value of branch voltage, so that a better approximation can be achieved with the next iteration.

g. All of these steps are followed and coded as shown in the listing in the Appendix, page AE-42. For added flexibility some parameters are entered with RX(1) for the resistance scale factor, and RX(2) for the current tolerance, so they may be changed by the user at will.

Note that in Part 4, the converged value of branch current in the nonlinear resistor XCUR, is added to the branch current of the "dummy" 100 megohm resistor R4, so that the total current in Branch 5 can be printed or plotted with the PR or PL commands.

5.6.1 Input Data

The input data, printed results, plotted output and plot data tabulation for the nonlinear resistor problem are included in Appendix E.

5.7 Sandia NOS AITRAC Special Equations

The following steps are required to utilize the special equations feature with Sandia NOS AITRAC.

a. Obtain a copy of the special equations subroutine SPEQ.

(1) If equations are being inserted for the first time, the SPEQ routine may be obtained from the system library by typing:

```
GET,ASPEQ/UN=LIBRARY
```

(2) If the equations have already been inserted into subroutine SPEQ and the routine has been previously saved (say, as permanent file TRAC), retrieve it by typing:

```
GET,TRAC
```

b. Insert the desired special equations into the SPEQ routine by using XEDIT. The user may desire to save these special equations as a permanent file for later use, as indicated in step a.(2) above.

c. Run the procedure file by typing:

```
GET,MAKEQ/UN=LIBRARY
```

d. Run the procedure file by typing:

```
-MAKEQ(FN=fname)
```

Where fname is a local or permanent file prepared according to steps a and b above. This procedure file compiles the subroutine SPEQ and any other subprograms in fname, and loads them along with AITRAC into memory leading the user directly into the input phase of AITRAC. Compiler errors are directed to local files and terminal diagnostic messages are issued.

Error conditions: Should the user receive the message

FORTRAN ERRORS IN FN - PLEASE CORRECT

after typing -MAKEQ (FN=fname), this indicates that there was an error from the FORTRAN compiler in compiling the SPEQ routine. The user should re-examine his equations, correct his errors, and re-execute steps a through d.

TABLE I

User's Table of FORTRAN
Variables Used in AITRAC

This table provides a reference for the user who needs to access the AITRAC Program Variables, when using the SPEQ (Special Equations) option.

Periodic expansion and updatings to this table will be provided to allow maximum flexibility of SPEQ use by the user.

TABLE I

User's Table
(Notation and Symbols - AITRAC)

AITRAC FORTRAN Notation	Description	Text Symbols
NV	Number of dependent nodes $0 \leq NV \leq 100$	NV
H(I, J)	Element of the coefficient matrix (H matrix). I, J represents the row and column numbers of the matrix. $1 \leq I \leq NV$ (maximum matrix size 100 x 100) $1 \leq J \leq NV$	$H_{i,\ j}$
T(I)	Element of the vector matrix (T matrix). I represents the row number of the matrix. $1 \leq I \leq NV$	w_i
V(I)	Element of the unknown matrix (node voltage). I represents the row number (node number) of the V matrix. $1 \leq I \leq NV$	$v_i(t_n)$
V1(I)	First past value of the unknown matrix (node voltage) $1 \leq 1 \leq NV$	$v_1(t_{n-1})$
TE	Present value of program time	t, t_n
TO	Present valuc of program delta time	dt. Δt_n
TE11	Past value of time (TE-TO)	Δt_{n-1}
TO11	Past value of delta time	t_{n-1}
TI	Present value of entered maximum delta time	

TABLE I
(continued)

AITRAC FORTRAN Notation	Description	Text Symbols
ND1	Number of time functions and grounded voltage sources. $0 \leq ND1 \leq 18$.	
ND	Total number of time functions. ND1 + hole-pair generating function. $1 \leq ND \leq 18 + 1$	
EI(K)	Present value of a time function or grounded voltage source K	$tf_k(t)$, $tf_k(t_n)$
		$e_{vs}(t)$, $e_{vs}(t_n)$
EP1(K)	First past value of EI(K) The value of EI(K) at TE11	$tf_k(t_{n-1})$
		$e_{vs}(t_{n-1})$
EP2(K)	Second past value of EI(K) The value of EP1(K) at TELL, (or EI(K), at TE11 - TO11)	$tf_k(t_{n-2})$
	where $1 \leq K \leq 20$	$e_{vs}(t_{n-2})$
	K = 19 for hole-pair generating function gamma-dot(t)	
	K = 20 for ground (EI(20) = 0. volts)	
NOP(1)	Number of floating voltage and current sources	
NOP(2)	Number of resistors	
NOP(3)	Number of capacitors	
NOP(4)	Number of inductors	
NTS	Number of transconductances	
NCR	Number of diodes	
NTR	Number of transistors	
NJFET	Number of JFETS	
NMOSFT	Number of MOSFETS	

TABLE I
(continued)

AITRAC FORTRAN Notation	Description	Text Symbols
NPSWI	Number of Parameter Switches	
	$0 \leq \sum_{I=1}^{4} NOP(I) \leq 200 - NTS - NPSWI$	
	$0 \leq NCR + NJFET \leq 30$	
	$0 \leq NTR + NMOSFT \leq 30$	
NBRAN	Number of assigned branches	
	$NBRAN = \sum_{I-1}^{4} \left[NOP(I)\right] + NCR + 2*NTR$	
	$+ NTS + NJFET + 2*NMOSFT + NPSWI$	

Parameters of Branch K

	Floating Voltage Source	Current Source	Resistor	Capacitor	Inductor
VAL1 (K)	Voltage	Current	Not used	Capacitance	Inductance
VAL2 (K)	Series Resistance	Shunt Resistance	Resistance	Series Resistance	Series Resistance
VAL3 (K)	Power	Power	Power	Shunt Resistance	Shunt Resistance
VAL4 (K)	Switched value of branch K, controlled by a switch				

TABLE I
(continued)

AITRAC FORTRAN Notation	Description	Text Symbols
	Parameters of Transconductance Branch K	
VAL1 (K)	Original Transconductance Value	
VAL2 (K)	SWITCH { 1st value for switching condition (Packed information)	
VAL3 (K)	2nd value for switching condition (Packed information) }	
VAL4 (K)	Switched Transconductance Value	

BCUR (K)	Present value of the current in the assigned model branch K. (For the capacitor and inductor models, this value does not include the current through the shunt resistors.)	
BCUR1 (K)	First past value of the current in the assigned model branch K. The value of BCUR (K) at TE11.	
	Where the branch numbers are assigned as follows:	
	1. Floating voltage and current sources. K = Number of the source	
	2. Resistor K = NOP (1) + Number of the resistor	
	3. Capacitor K = NOP (1) + NOP (2) + Number of capacitor	
	4. Inductor K = NOP (1) + NOP (2) + NOP (3) + Number of the inductor	

TABLE I
(continued)

AITRAC FORTRAN Notation	Description	Text Symbols

5. Diode

$$K = \sum_{I=1}^{4} (NOP(I)) + \text{Number of the diode}$$

6. Transistor

J = Number of the transistor

a. Base Current

$$K = \sum_{I=1}^{4} (NOP(I)) + NCR + 2J-1$$

b. Collector Current

$$K = \sum_{I=1}^{4} (NOP(I)) + NCR + 2J$$

7. Transconductance

$$K = \sum_{I=1}^{4} (NOP(I)) + NCR + 2*NTR + \text{Number of Transconductance}$$

NOTE: Even though Transconductances carry a branch number designation, no branch current or power can either be calculated or printed.

8. JFET (Channel Current)

$$K = \sum_{I=1}^{4} (NOP(I)) + NCR + 2*NTR + NTS + \text{Number of JFET}$$

TABLE I
(continued)

AITRAC FORTRAN Notation	Description	Text Symbols
	9. MOSFET	
	J = Number of MOSFET in question	
	a. Channel Current:	
	$K = \sum_{I=1}^{4} (NOP(I)) \mid NCR \mid 2*NTR + NTS + NJFET + 2J - 1$	
	b. Gate-Source FET Diode Current:	
	$K = \sum_{I=1}^{4} (NOP(I)) + NCR + 2*NTR + NTS + NJFET + 2J$	
	10. Parameter Switch	
	$K = \sum_{I=1}^{4} (NOP(I)) + NCR + 2*NTR + NTS + NJFET + NMOSFT +$ Number of parameter switch	
	NOTE: Even though Parameter Switches carry a branch number designation, no branch current or power can either be calculated or printed for switches.	

O	k*TEMP/q (0.026 at 300 deg. K)	θ
	Boltzmann's constant (1.38062E-23 J/K	k
	Charge on electron (1.60219E-19 Coul)	q
BOLTZQ	k/q (8.61708E-5)	

TABLE I
(continued)

AITRAC FORTRAN Notation	Description	Text Symbols
A(1)	Ambient temperature (deg. K) (temperature in deg. C + 273.15)	
	Device Temperature:	
	= Ambient + difference of element temperature from ambient specified	TEMP
ABSZER	Value of zero degree Celsius on the absolute temperature scale, i.e: -273.15 deg. K.	
BASEDG	Programs's reference temperature for device parameters	
BANDEG	Energy band Gap (volts) of device being analyzed	
VCRIT	Critical voltage of diode diffusion curve, often point of min. radius of curvature, sometimes set to: 10*O (θ)	
TWOPI	$2*\pi$ (6.28318....)	
ROOT2	SQRT(2.) (1.414....)	
UM	Variable order of integration coefficient set in CRITERIA Command	MU
XMU	Time varying value of UM (reserved for use in the future)	

TABLE I
(continued)

AITRAC FORTRAN Notation	Description	Text Symbols
	JUNCTION DIODES	
QCR (N, 1)	Diode reverse saturation current	IS
QCR (N, 2)	Diode proportionality constant (usually $1 \leq MD \leq 2$)	MD
QCR (N, 3)	Diode leakage resistance	RDL
	Diode junction capacitance	CD
QCR (N, 4)	Diode junction capacitance at $V_D = 0.0$	CDO
QCR (N, 5)	Diode intrinsic built-in voltage	VDBI
QCR (N, 6)	Diode time constant	TD
QCR (N, 7)	Diode steady state primary photocurrent	IPPD
QCR (N, MDJUNC)	Initial diode state (on, off, etc.)	
QCR (N, MDDIFT)	Diode working temperature Difference from ambient	
QCR (N, MDEG)	Energy bandgap for the diode	EG
QCR (N, MDISTE)	Temperature corrected value of IS	
QCR (N, MDCRIT)	Critical voltage on diffusion curve	VCRIT
QCR (N, MDMOS)	Set to 0.0, if device is a diode	
PWCR (N)	Instantaneous diode real power	
PPID (N)	Diode primary photocurrent	$i_{PPD}(t)$
PPID1 (N)	Past value of diode primary photocurrent	
DV (N)	Diode voltage	$v_D(t)$

TABLE I
(continued)

AITRAC FORTRAN Notation	Description	Text Symbols
DI (N)	Ideal diode current	$i_{DI}(t)$
DIP (N)	Past value of ideal diode current	
DR (N)	Ideal diode tangent conductance where N = number of the diode	

TABLE I
(continued)

AITRAC FORTRAN Notation	Description	Text Symbols
	JUNCTION TRANSISTORS	
QTAN (N, 1)	Normal transistor beta	HFEN
QTAN (N, 2)	Inverted transistor beta	HFEI
	Normal transistor alpha	a_N
	Inverted transistor alpha	a_I
QTAN (N, 3)	Emitter junction time constant	TN
QTAN (N, 4)	Collector junction time constant	TI
QTAN (N, 5)	Collector junction diode saturation current	ICS
QTAN (N, 6)	Collector junction diode proportionality constant (usually $1 \leq MC \leq 2$)	MC
	Collector junction diode capacitance	CC
QTAN (N, 7)	Collector junction diode capacitance at $V_{BC} = 0$	CCO
QTAN (N, 8)	Collector junction intrinsic built-in voltage	VCBI
QTAN (N, 9)	Collector junction leakage resistance	RCI
QTAN (N, 10)	Emitter junction diode saturation current	IES
QTAN (N, 11)	Emitter junction diode proportionality constant (usually $1 \leq ME \leq 2$)	ME
	Emitter junction diode capacitance	CE
QTAN (N, 12)	Emitter junction diode capacitance at $V_{BE} = 0$	CEO
QTAN (N, 13)	Emitter junction intrinsic built-in voltage	VEBI

TABLE I
(continued)

AITRAC FORTRAN Notation	Description	Text Symbols
QTAN (N, 14)	Emitter junction leakage resistance	REL
QTAN (N, 15)	Collector junction steady state primary photocurrent	IPPC
QTAN (N, 16)	Emitter junction steady state primary photocurrent	IPPE
QTAN (N, MQJUNC)	Initial transistor state (on, off, etc.)	
QTAN (N, MQDIFT)	Transistor working temperature, difference from ambient	
QTAN (N, MQEG)	Energy bandgap for transistor	EG
QTAN (N, MQICST)	Temperature corrected value of ICS	
QTAN (N, MQIEST)	Temperature corrected value of IES	
QTAN (N, MQCCRI)	Critical voltage of collector diode diffusion curve	
QTAN (N, MQECRI)	Critical voltage of emitter diode diffusion curve	
QTAN (N, MQMOS)	Set to 0.0 if device is bipolar transistor	
PWTR (N)	Instantaneous transistor real power	
PPIC (N)	Collector junction diode primary photocurrent	$i_{PPC}(t)$
PPIC1 (N)	Past value of collector junction diode primary photocurrent	
BCV (N)	Collector junction diode voltage	$v_{BC}(t)$
BCI (N)	Collector junction ideal diode current	$i_{CI}(t)$
BCIP (N)	Past value of collector junction ideal diode current	
BCR (N)	Collector junction ideal diode tangent conductance	

TABLE I
(concluded)

AITRAC FORTRAN Notation	Description	Text Symbols
PPIE (N)	Emitter junction diode primary photocurrent	$i_{PPE}(t)$
PPIE1 (N)	Past value of emitter junction diode primary photocurrent	
BEV (N)	Emitter junction diode voltage	$v_{BE}(t)$
BEI (N)	Emitter junction ideal diode current	$i_{EI}(t)$
BEIP (N)	Past value of emitter junction ideal diode current	
BER (N)	Emitter junction ideal diode tangent conductance	
	where N = Number of transistor	

6. Mathematical Formulation

Reprinted from:

MTRAC
Computation, Program and Application

By: D. Nitzan and J. R. Herndon

Prepared for:

Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, California 91103

SRI Project 6408

Report 6, June 1969

I METHOD OF COMPUTATION IN MTRAC

The basic computation method used in the MTRAC computer program is described first. It will be shown that the unknown circuit time variables can be solved from the matrix equation $T] = [H] \cdot V]$, where $T]$ and $V]$ are column matrices of currents and nodal voltages, respectively, and $[H]$ is a square conductance matrix. On the basis of this method, the contributions to the $[H]$ and $T]$ matrices will then be derived for various circuit elements, such as a current source, a resistor, a voltage source with internal series resistance, a capacitor, an inductor, a zener diode, a diode, a transistor, and a square-loop magnetic core.

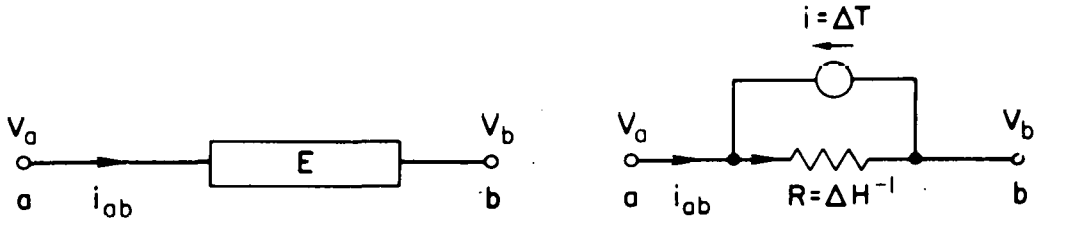
A. Basic Computation Method

1. Matrix Equation $T] = [H] \times V]$

a. Derivation of $T] = [H] \times V]$

Consider a linear or nonlinear two-terminal circuit element E, Fig. 1(a). The element is connected between two nodes, a and b, whose voltages (to ground) are V_a and V_b, respectively. As we shall show later, the current from Node a to Node b at a given time step can be approximated by the linear expression

$$i_{ab} = \Delta H(V_a - V_b) - \Delta T \quad , \qquad (1)$$



(a) SCHEMATIC REPRESENTATION (b) APPROXIMATE EQUIVALENT ELEMENT

TA-6408-1

Figure 1 A TWO-TERMINAL CIRCUIT ELEMENT

where ΔH and ΔT are constants representing conductance and current, respectively. Thus, the approximate equivalence of a two-terminal element is a conductance ΔH in parallel with a current source ΔT, as shown in Fig. 1(b). In the case of a nonlinear circuit element, Eq. (1) is based on "linearization" of the element's operational (or dynamic) characteristics.

Now suppose that n_p two-terminal circuit elements are connected in parallel between Nodes a and b, as shown in Fig. 2. These elements may be represented by an equivalent two-terminal circuit element, enclosed by the dashed line in Fig. 2, whose current is

$$i_{ab} = \sum_{j=1}^{n_p} i_j \qquad (2)$$

Applying Eq. (1) to each jth current in Eq. (2) gives

$$i_{ab} = \sum_{j=1}^{n_p} \Delta H_j (V_a - V_b) - \sum_{j=1}^{n_p} \Delta T_j$$

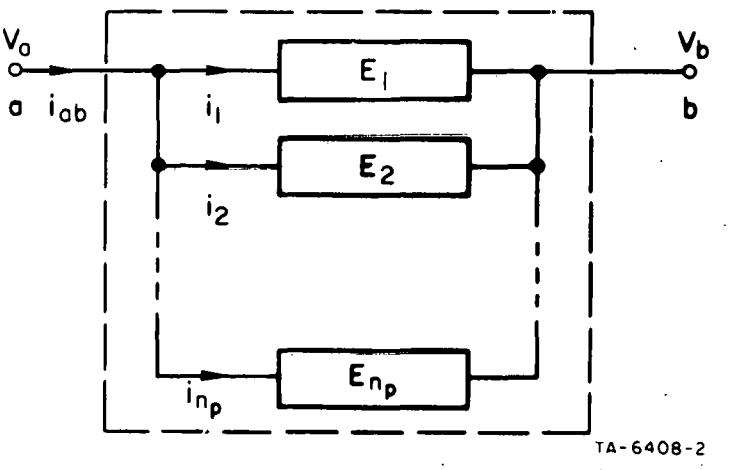


Figure 2 TWO-TERMINAL CIRCUIT ELEMENTS IN PARALLEL

Hence, ΔH and ΔT of the equivalent circuit element are

$$\Delta H_{(p)} = \sum_{j=1}^{n_p} \Delta H_j \qquad (3)$$

and

$$\Delta T_{(p)} = \sum_{j=1}^{n_p} \Delta T_j \qquad (4)$$

If the n_p elements are identical, then Eqs. (3) and (4) are reduced to

$$\Delta H_{(p,i)} = n_p \Delta H \qquad (5)$$

and

$$\Delta T_{(p,i)} = n_p \Delta T \qquad (6)$$

where the subscript i is added to designate *identical* elements and where ΔH and ΔT correspond to each individual element.

Consider now n_s identical two-terminal circuit elements which are connected in series between Nodes a and b, as shown in Fig. 3. The current through each element is i_{ab}. Applying Eq. (1) to the current of each element, we obtain

$$i_{ab} = \Delta H(V_a - V_1) - \Delta T$$

$$i_{ab} = \Delta H(V_1 - V_2) - \Delta T$$

$$\vdots$$

$$i_{ab} = \Delta H(V_{n_s-1} - V_b) - \Delta T$$

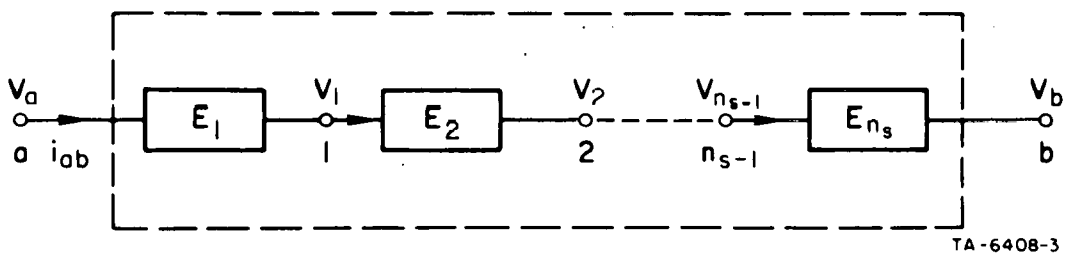


Figure 3 IDENTICAL TWO-TERMINAL CIRCUIT ELEMENTS IN SERIES

Adding these n_s equations and dividing by n_s gives

$$i_{ab} = \frac{\Delta H}{n_s}(V_a - V_b) - \Delta T \quad . \tag{7}$$

Identifying Eq. (7) with Eq. (1) gives ΔH and ΔT of an equivalent element of n_s identical two-terminal elements in series:

$$\Delta H_{(s,1)} = \Delta H / n_s \tag{8}$$

and

$$\Delta T_{(s,1)} = \Delta T \quad . \tag{9}$$

Comparing the equivalent circuits of elements connected in parallel and in series, we observe the following:

(1) Unlike the case of elements in parallel, the current i_{ab} through an equivalent element of n_s *different* elements in series cannot be brought into the form of Eq. (1). Consequently, we can represent n_s elements in series by an equivalent element only if they are identical.

(2) Physically, Eqs. (5) and (8) simply state that the conductance of n_p identical elements in parallel is n_p times larger than and the conductance of n_s identical elements in series is n_s times smaller than the conductance of an individual element.

(3) Equations (6) and (9) agree with the fact that currents add in a parallel connection, but are the same in a series connection.

Consider now an entire network of n nodes, any one of which may be Node a, *i.e.*, $1 \leq a \leq n$. Any jth element between Node a and the remaining $n - 1$ nodes is represented by Node b in Eq. (1). For each of these $n - 1$ elements, Eq. (1) is written as

$$i_{aj} = \Delta H_j (V_a - V_j) - \Delta T_j \quad . \tag{10}$$

If there is no element between Node a and, say, Node m, then $\Delta H_m = 0$ and $\Delta T_m = 0$; hence, $i_{am} = 0$, as expected. For reference, let us assume that all the $n - 1$ currents *leave* Node a. An example for the case of $a = 1$ is shown schematically in Fig. 4. Applying Eq. (10) to each of the

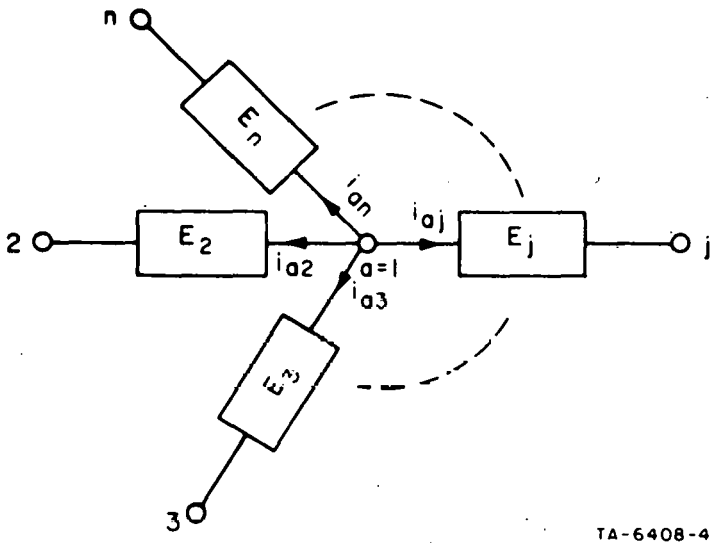


Figure 4 CIRCUIT ELEMENTS BETWEEN NODE a AND OTHER NETWORK NODES.

n - 1 currents and summing, we obtain

$$\sum_j i_{aj} = \left(\sum_j \Delta H_j\right) V_a - \sum_j \Delta H_j V_j - \sum_j \Delta T_j \quad .$$

But, following Kirchoff's current law, $\sum_j i_{aj} = 0$. Hence,

$$T_a = H_{aa} V_a + \sum_j H_{aj} V_j \quad . \tag{11}$$

where

$$H_{aa} = \sum_j \Delta H_j \quad , \tag{12}$$

$$H_{aj} = -\Delta H_j \quad , \tag{13}$$

and

$$T_a = \sum_j \Delta T_j \quad . \tag{14}$$

Applying Eq. (11) to each of the n nodes (*i.e.*, letting $a = 1, 2, \ldots n$), we obtain the following matrix equation

$$\begin{bmatrix} T_1 \\ T_2 \\ \cdot \\ \cdot \\ \cdot \\ T_i \\ \cdot \\ \cdot \\ \cdot \\ T_n \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & \cdots & H_{1j} & \cdots & H_{1n} \\ H_{21} & H_{22} & \cdots & H_{2j} & \cdots & H_{2n} \\ \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & & \cdot \\ H_{i1} & H_{i2} & \cdots & H_{ij} & \cdots & H_{in} \\ \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & & \cdot & & \cdot \\ H_{n1} & H_{n2} & \cdots & H_{nj} & \cdots & H_{nn} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ \cdot \\ \cdot \\ \cdot \\ V_i \\ \cdot \\ \cdot \\ \cdot \\ V_n \end{bmatrix} \qquad (15)$$

or, in brief,

$$[T] = [H] \cdot [V] \qquad (16)$$

b. Filling the $[H]$ and $[T]$ Matrices

In Fig. 4 and Eq. (11), the attention was focused on one node (Node a), and the effect of all the circuit elements on this node was examined. In filling the $[H]$ and $[T]$ matrices, however, each circuit element is examined separately by computing its contribution to the H and T elements that correspond to its terminals. Initially, the elements of the $[H]$ and $[T]$ matrices are set to zero. All the circuit elements of a given type (*e.g.*, sources) are scanned, and the contributions of each of these elements to the matrix elements corresponding to its two nodes (or three nodes in case of a transistor) are computed. This computation procedure is repeated for the remaining circuit-element types. If no circuit element exists between, say, Nodes a and m, then $\Delta H_m = 0$ and, following Eq. (13), $H_{am} = 0$. On the other hand, if Node a is common to ℓ circuit elements, then, following Eqs. (12) and (14), H_{aa} is the sum of ℓ ΔH terms and T_a is the sum of ℓ ΔT terms.

The ΔH and ΔT terms corresponding to each node whose voltage is to be solved will next be examined separately for a two-terminal circuit element and for an active three-terminal circuit element (transistor).

2. A Two-Terminal Circuit Element

Consider a two-terminal circuit element across Nodes a and b, Fig. 1. In filling the $[H]$ and $T]$ matrices, three cases are distinguished, as shown in Fig. 5:

(1) Both V_a and V_b are unknown

(2) V_a is unknown and $V_b - V_{sb}$ is known

(3) $V_a \quad V_{sa}$ is known and V_b is unknown.

In Case (1), both nodes are floating. In Case (2), Node b is tied to either a voltage source ($V_{sb} \neq 0$) or ground ($V_{sb} = 0$). In Case (3), Node a is tied to either a voltage source ($V_{sa} \neq 0$) or ground ($V_{sa} = 0$). Let us examine each case separately.

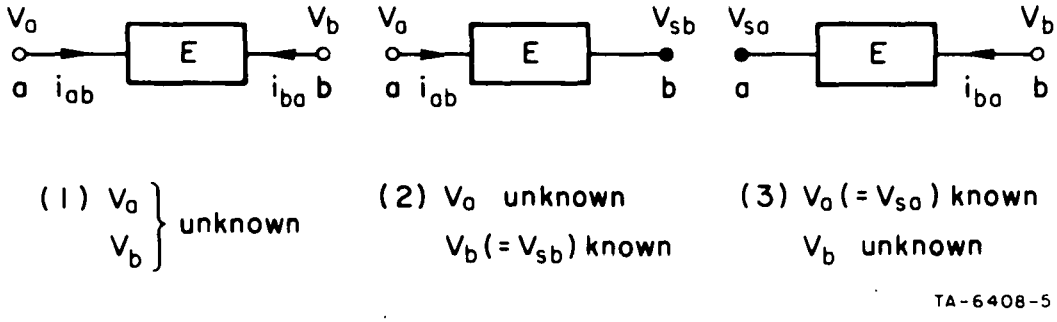


Figure 5 THREE CASES OF NODAL VOLTAGES OF A TWO-TERMINAL CIRCUIT ELEMENT

a. Case (1): V_a And V_b Are Unknown

Let us first consider a current flow from Node a to Node b. Following Eqs. (12), (13), and (14), the contributions of this circuit element to the matrix elements H_{aa}, H_{ab}, and T_a are

$$\Delta H_{aa} = \Delta H \quad , \tag{17}$$

$$\Delta H_{ab} = -\Delta H \quad , \tag{18}$$

and

$$\Delta T_a = \Delta T \quad . \tag{19}$$

Thus, Eq. (1) may be written in the form

$$i_{ab} = \Delta H_{aa} V_a + \Delta H_{ab} V_b - \Delta T_a \quad . \tag{20}$$

So far we have referred to a current flow from Node a to Node b. However, in order to completely fill the matrices, we also have to consider the current i_{ba} from Node b to Node a. Since $i_{ba} = -i_{ab}$, Eq. (1) yields

$$i_{ba} = \Delta H(V_b - V_a) + \Delta T \quad . \tag{21}$$

However, by interchanging a and b in Eq. (20) we also obtain

$$i_{ba} = \Delta H_{bb} V_b + \Delta H_{ba} V_a - \Delta T_b \quad . \tag{22}$$

Equating Eqs. (21) and (22), we find that

$$\Delta H_{bb} = \Delta H \quad , \tag{23}$$

$$\Delta H_{ba} = -\Delta H \quad , \tag{24}$$

and

$$\Delta T_b = -\Delta T \quad . \tag{25}$$

Note that $\Delta H_{aa} = \Delta H_{bb}$ and $\Delta H_{ab} = \Delta H_{ba}$, but $\Delta T_a = -\Delta T_b$.

b. Case (2): V_a Is Unknown; $V_b (= V_{sb})$ Is Known

Substituting $V_b = V_{sb}$ into Eq. (1) gives

$$i_{ab} = \Delta H(V_a - V_{sb}) - \Delta T \quad . \tag{26}$$

Since V_b is known, the matrix equation, Eq. (15), does not include V_b and its associated elements. The values of H_{ab}, H_{bb}, H_{ba}, and T_b are therefore not computed and Eq. (20) is modified to

$$i_{ab} = \Delta H_{aa} V_a - \Delta T_a \quad . \tag{27}$$

By equating Eqs. (26) and (27) we find that

$$\Delta H_{aa} = \Delta H \tag{28}$$

and

$$\Delta T_a = \Delta T + \Delta H \cdot V_{s\,b} \quad . \tag{29}$$

Case (3): $V_a (= V_{s\,a})$ Is Known; V_b Is Unknown

Substituting $V_a = V_{s\,a}$ into Eq. (21) gives

$$i_{b\,a} = \Delta H (V_b - V_{s\,a}) + \Delta T \quad . \tag{30}$$

The same arguments for Node b in Case (2) now hold for Node a. Thus, Eq. (22) is modified to

$$i_{b\,a} = \Delta H_{b\,b} V_b - \Delta T_b \quad . \tag{31}$$

By equating Eqs. (30) and (31) we find that

$$\Delta H_{b\,b} = \Delta H \tag{32}$$

and

$$\Delta T_b = -\Delta T + \Delta H \cdot V_{s\,a} \quad . \tag{33}$$

REFERENCES

1. Transient Radiation Analysis by Computer Program (TRAC), Vols. I & II Harry Diamond Labs DAAG39-68-C-0041.

2. Improved Radiation Response Calculations Berne Electronics Technical Note RAD-5.

3. NOS News Notes, Sandia Laboratories, Albuquerque, NM, Division 2614.

APPENDIX A

NONSTANDARD MODELING

APPENDIX A

NONSTANDARD MODELING

A1. Zener Diode

Zener diodes are shown in Figure A-1.

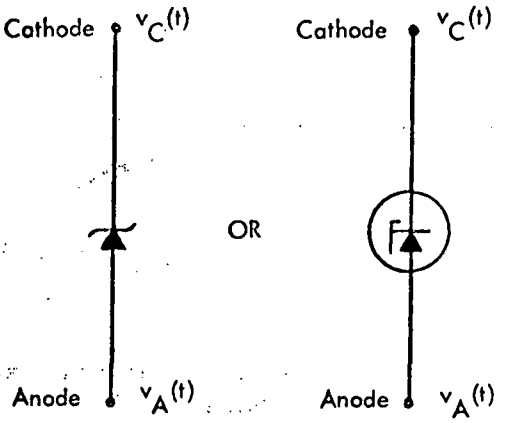


Figure A-1. Zener Diodes

The zener diode may be modeled by using the two diode-floating source combination in Figure A-2.

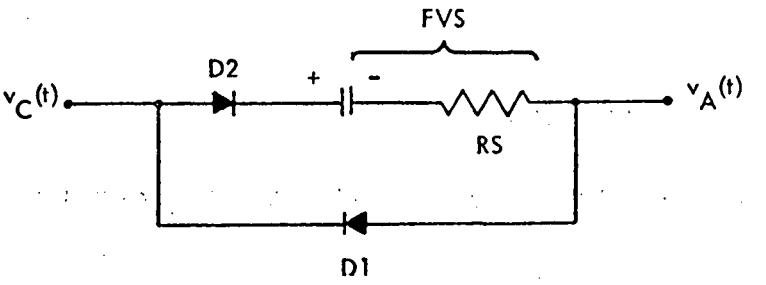


Figure A-2. Zener Diode Nonstandard Model

Diode D1 represents the forward characteristic of the zener diode. Diode D2 and FVS represent the zener characteristic of the zener diode. E_{FVS} is the voltage at the knee of the zener curve less V_{D2}, and R_S is the slope of the curve after breakdown.

A2. Integrated Circuit Transistor

An NPN integrated transistor with PN junction isolation may be modeled by the PNP-NPN transistor combination shown in Figure A-3.

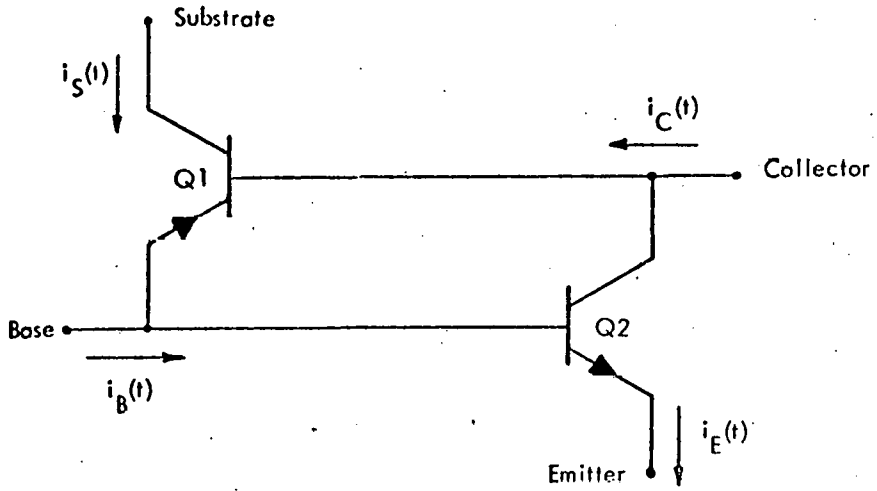


Figure A-3. Integrated Circuit Transistor Nonstandard Model

The following transistor parameter relationships are necessary.

ME1 = MC2 = actual base-collector value

IES1 = ICS2 = 1/2 actual value

α_{N1} = 2 x actual value (HFEN ≠ -1.)

α_{I2} = 2 x actual value (HFEI ≠ -1.)

VCBI1 = VCBI2 = actual value

The values below are recommended for the other parameters which must be modified.

TN1 = TI2 = actual value

CEO1 = CCO2 = 1/2 actual value

REL1 = RCL2 = 2 x actual value

IPPE1 = IPPC2 = 1/2 actual value

A3. Silicon Controlled Rectifier (SCR)

The SCR is shown symbolically in Figure A-4. The SCR may be modeled with the standard models shown in Figure A-5.

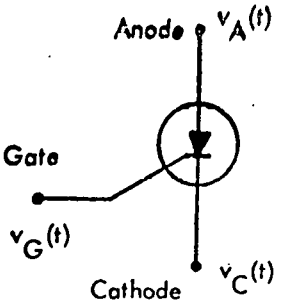


Figure A-4. SCR

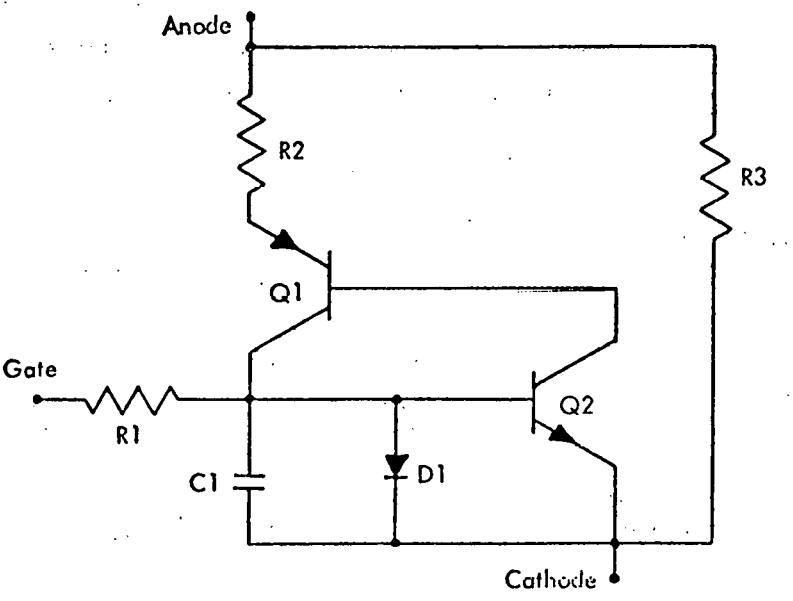


Figure A-5. SCR Nonstandard Model

The transistors provide for regenerative gain and switching action where their time constants define the turn on, turn off, and storage time of the SCR. The diode accounts for the variation of current gain with anode current. R1, C1 determine the gate impedance and the turn on delay time. R3 is the anode to cathode leakage resistance, and R2 is the small series resistance when the SCR is fully conducting.

A4. Four-Layer Diode (PNPN Switch)

The four-layer diode is shown symbolically in Figure A-6.

Figure A-6. Four-Layer Diode

The four-layer diode may be modeled by combining the floating voltage source and diode standard models with the nonstandard SCR model as shown in Figure A-7. E_{FVS} is the approximate forward breakover (switching) voltage.

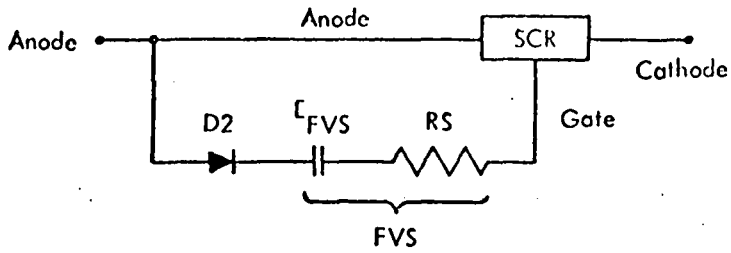


Figure A-7. Four-Layer Diode Nonstandard Model

A5. Unijunction Transistor (UJT)

The UJT is shown symbolically in Figure A-8.

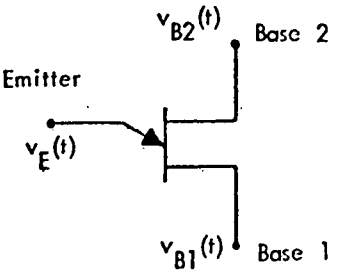


Figure A-8. UJT

The UJT may be modeled by the standard model combination shown in Figure A-9.

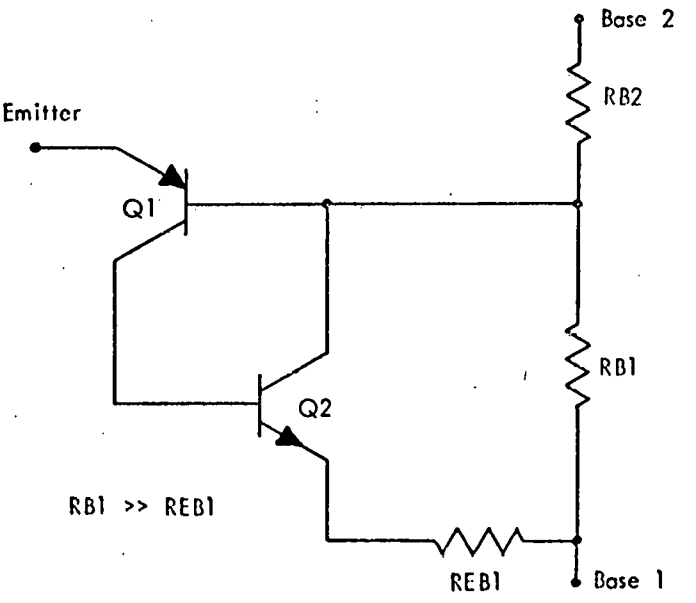


Figure A-9. UJT Nonstandard Model

Resistors RB1 + RB2 equals the interbase resistance and REB1 is the small emitter-base 1 resistance when the UJT is fully on. The intrinsic stand-off ratio is defined by RB1/RB1 + RB2). The transistors provide for the switching action of the UJT.

APPENDIX B

DATA INPUT FROM FILES

APPENDIX B

DATA INPUT FROM FILES

Users may prepare problem data for input from files, rather than by program interaction. To make such a data file, the system text editor is entered, and the user lists the circuit components in the same order as they would normally be requested by the program:

- V Grounded Voltage Sources and Time Functions
- S Floating Sources
- R Resistors
- C Capacitors
- L Inductors
- D Junction Diodes
- Q Transistors
- T Transconductances and Mutual Inductances
- J JFET's
- M MOS/FET's
- W Switches

A blank line, or a $ sign, must be used to terminate (or indicate none of) a particular component type. A $ sign is preferable as it makes the file easier to read. When input is taken from a disk file, the first field of each line is ignored by AITRAC; this is done so that the user may insert parameter identifications (such as: R1, C33, L12, etc.). With this one exception, each line contains the information exactly as it would have been typed into the program if the user were entering the data on-line. Refer to Section 3.1.1.

As an example, assume the disk file BRIDGE contains the following data:

```
BRIDGED TEE CIRCUIT          The circuit name must be included
$                            Blank line: no V sources
S1  1  G  -.02  1K  CURRENT  A floating current source
$                            Blank: no more floating sources
R1  3  0  1K                 A resistor
$                            Blank: no more resistors
C1  1  3  1U                 A capacitor
C2  2  0  1U                 A capacitor
C3  3  0  1U                 A capacitor
```

```
$                                   Blank: no more capacitors
L1 1  2  1.                         An inductor
L2 2  3  1.                         An inductor
$                                   Blank: no more inductors
$                                   Blank: no diodes
$                                   Blank: no transistors
$                                   Blank: no transconductances
$                                   Blank: no JFET's
$                                   Blank: no MOSFET's
$                                   Blank: no switches
```

This file contains data for the Bridged Tee circuit given in one of the examples in Appendix E.

These data may be read into the program by the command:

```
OLD BRIDGE
```

NOTE

When diode, transistor, JFET, or MOSFET parameters are entered directly as data, and not by means of the SEarch, USE, or LIbrary commands, the letters D, Q, J, or M must be entered in Column 1 of EACH line of data; for example, for a diode:

```
D12  17  22  ZERO  TE=55
D  1N  2.  .1E10 .11E-10
D  .75  10N  0.0
```

and for a transistor:

```
Q3  4  2  3  PNP  SATURATED  TEMP=+125
Q 100.  1.  1N  .1U  .1E-10
Q 1.1  .1E-10  0.75  10M  .1P
Q 1.33  .5E-10  0.84
Q  .1E9  0.0  0.0
```

The data may be entered on fewer lines, but EACH line must start with the letters D, Q, J, or M, as appropriate.

The user should familiarize himself with the proper methods for entering paper tape files, as well as for storing and editing disk files, by referring to the proper sections of the NOS news notes[†].

In particular refer to:

Section 2: "Logging In and Out"

Section 4: "XEDIT" Text Editor

Section 12: "Printing and Punching Files under NOS"

[†]NOS News Notes, Sandia Laboratories, Albuquerque, NM, Division 2614.

APPENDIX C

ERROR MESSAGES

APPENDIX C

ERROR MESSAGES

The following error messages may occur during input of data or commands. When applicable, the program will type:

ERROR, FIELD nn: "error message"

where nn is the number of the field in which the error occurred.

Error Message	Meaning
INVALID AITRAC COMMAND	A command is unrecognized, and should be re-entered. See Section 3 for valid commands.
REAL VALUE EXPECTED	A real number was expected in the indicated field. Check typing.
ODD NUMBER OF TIME VALUE PAIRS	A TIME command or time-function input; the last value entered was not paired with a value for "finish time" or amplitude.
MAX NO. OF DEVICES EXCEEDED	Too many of the device being entered are being specified, i.e., more than 30 diodes, etc. See Section 2.7.
ILLEGAL NODE OR SOURCE NUMBER	A referenced voltage source is not defined, or node number > 100.
NOT CONSTANT VALUE OR USE	A value for a floating source has been specified incorrectly.
SUPERFLUOUS OR INVALID FIELD	Too many fields are specified, or typing error.
SUPERFLUOUS	Self-explanatory.
TWO CONNECTIONS TO SOURCE OR GROUND	Only one terminal of an element may be connected to ground or to a fixed voltage source.
NONEXISTENT PARAMETER	The parameter being referenced does not exist in the circuit.

Error Message	Meaning
INTEGER EXPECTED	A real value or alpha field was entered instead of an integer.
FILE NOT FOUND	The file named does not exist in the user's directory. Check the correct spelling of the name.
NO CIRCUIT IN MEMORY	The command entered is meaningless as there is no circuit in memory at this time.
NO OUTPUT REQUESTED	EXECUTE command is ignored as no PRINT or PLOT command has been specified.
UNREALISTIC VALUE	The value specified is outside reasonable range for computation.
UNCONNECTED NODES	Execution is inhibited, and the list of unconnected nodes is printed. Can occur when components have been deleted from the circuit. Dummy elements may be connected between unconnected nodes and ground to force execution with "HANGING NODES" message.
WARNING HANGING NODES	A dependent node has only one element connected to it. Execution continues.
TIME VALUES MUST INCREASE	A time function is specified with time points which do not increase from the value specified in previous time fields.
ILLEGAL HP FUNCTION TYPE	HP Function must be type "0" or "1."
INCORRECT DEVICE TYPE	A transistor may not COPY a MOSFET, or other similar conflict.
INCORRECT MATERIAL TYPE	The material specified for a semiconductor device is incorrect.
REFERENCED BRANCH NOT AN INDUCTOR	Both branch numbers specified in a T (transconductance) specification using LK and LM must refer to branches containing only inductors.
SWITCH nn IS UNDEFINED	An element was specified with two values, and a reference to switch "nn," but a "W" (switch) specification for that switch number was not entered.

Error Message	Meaning
RIGHT PARENTHESIS ERROR	The right parenthesis was left out in a W (switch) specification.
DEVICE IS NOT IN DATA BANK	The device desired is not included in the standard AITRAC data bank. Use another option or enter data for each parameter.
DEVICE NOT IN LIBRARY	The device desired is not included in the user's private library. Use another option or enter data for each parameter.
USER LIBRARY NOT FOUND	The user's private library does not exist on his private directory (see Section 3.18).
NOT COPY, USE, SEARCH OR LIBRARY	A nonvalid option was specified for retrieving semiconductor data from disk storage.

Other self-explanatory messages may be printed occasionally to inform, warn, or guide the user during program operation.

The following error messages can occur during execution of a problem.

1. NONLINEAR MODELS WILL NOT CONVERGE

After the limit number of iterations (200) for the DC solution, the diode and/or transistor standard models have not converged. The convergence criterion is that the current through each junction calculated by using the ideal diode iteration parameters and the current calculated by using the model data and the ideal diode equation are within a 10% tolerance.

2. ****PROGRAM WILL NOT CONVERGE
PERCENT ERROR FOR NODE CALC. XX.XXXXX

After the limit number of iterations (200) for the DC solution, the program has not converged. The node values are changing by the percent indicated between the last two iterations (i.e., 199th and 200th). (Note: This diagnostic will always follow the above error.)

3. ****SINGULAR MATRIX X.
THE FOLLOWING ROWS ARE SINGULAR

The matrix for the determination of the unknown node values cannot be solved:

a. If the number X is positive, then there exists a diagonal matrix element whose value is zero.

b. If X is negative, this diagonal value of zero was encountered when the matrix was being reduced during the solution.

Cases (a) and (b) can occur when a parameter has an improper value whereby the numerical accuracy of the equation solver and/or machine accuracy are inadequate (roundoff error too large, mostly due to insufficient significance in the machine). In addition, case (a) can occur when there is incorrect parts connection (wrong number of nodes, i.e., unconnected nodes, duplicate node numbers, one terminal of a part not connected, etc.). Case (b) can occur when a parameter has an improper value or delta time is too small, which can result in numerical difficulties.

4. ****SYSTEM UNSTABLE

This indication occurs when the program has made 10 attempts to reach an acceptable solution. Based on the nonlinear models, convergence has not been achieved; the program gives up on this solution in time, the delta time is reduced by a factor of two, and another solution attempt is made. In many instances, the program pulls through, and the above message has little or no effect on the analysis results.

APPENDIX D

HINTS FOR PROGRAM USAGE

APPENDIX D

HINTS FOR PROGRAM USAGE

The following suggestions may be useful in obtaining more efficient program results. To reduce running time, the amount of calculations must be reduced. The running time is proportional to the number of iterations per time solution, the number of time solutions, the size of the unknown matrix (number of nodes) to solve, and the number of parts. Another consideration for improved usage is the ability to take full advantage of the program capabilities. The suggestions are general, and some will be directly applicable to many of the user's circuits.

1. For larger circuits, determine if any nodes may be eliminated and the parts count reduced (assuming the analysis will not suffer). Similiar parts in series or parallel may often be combined, and series and shunt resistances can be incorporated into the standard models.

2. Take advantage of the program's Δt control, when there are straight-line time functions involved (e.g., square wave inputs, etc.). Usually one defined Δt region is sufficient. Sometimes two defined regions are useful, i.e.,

$$\Delta t_1 \text{ small, } t_1 = \Delta t_1$$

$$\Delta t_2 \gg \Delta t_1, \; t_2 = \text{end time} \quad .$$

3. Transistors can be conveniently used as diodes, when necessary. They may be employed as two diodes back-to-back (HFEN = HFEI = 0) or as one diode. When used as one diode, it is best to connect the collector to the base and set ICS = IPPC = 0.

4. Ratio of calculated points to plotted points ≤ 5 unless the Δt is extremely small, in order that reasonable plots are made.

5. Ratio of plotted points to printed points > 10 since printing is time consuming and plots are generally more convenient.

6. Augment standard semiconductor models with bulk resistors only when essential to simulation results.

7. Make the data as realistic (and practical) as possible.

   a. Time function switching times (e.g., rise and fall times) reasonable. For example, do not attempt to switch in a very small (say 1E-15) or zero time, as the program is not mechanized to accept such an unrealistic change.

   b. Due to numerical limitations, resistor values must be significant in comparison with an associated resistor value. For example, a 1E-4 ohm resistor in series with a 1E-6 ohm resistance may cause the nodal matrix to approach singularity and, in some instances, the node equations may fail to solve (one of these resistances may be contained within a model).

   c. Semiconductor data should be reasonable. Examples of some parameter ranges and typical values for silicon semiconductors are given below.

Diode and Transistor	Transistor
$1 \leq M_D \leq 2$	$\alpha_N > \alpha_I$
$1E\text{-}15 \leq I_S \leq 1E\text{-}6$	RCL > REL
VDBI = .75	ICS > IES
IPPC = 1.E-9 - 1.E-13 amps/rad/sec	TI > TN CEO > CCO (for equal areas)
IPPE = (.1 - .01)*IPPC	ME ≤ MC

8. After any changes of node connections, parts additions, and deletions, it is good practice to issue the command: SAVE filename (the "filename" on which the problem is to be stored) and then the command: OLD, filename. In this way, a fresh copy of the problem is brought into memory, and all data will be properly initialized.

9. Always perform a DC analysis prior to attempting a TR analysis. If the circuit does not operate correctly at steady state, much computer time and user time can be saved by correcting those problems first!

One notable exception, however, is the analysis of oscillators which may not have a stable DC state. For these circuits, a transient analysis starting with all power supplies at zero (and ramping slowly to their required voltage) will probably produce a satisfactory simulation, whereas a DC analysis may not.

10. Numerical instability can occur if the value of a time step Δt is larger than the time constant in any loop containing an energy-storing element at the time that this loop is undergoing a significant perturbation (such as a multivibrator switching between states). This instability appears as a damped oscillation with alternate values describing the upper and lower bounds of the oscillation. If such numerical instability is suspected, the value of Δt should be reduced around the time that this oscillation occurs, and the simulation should be rerun.

Another method is to INCREASE the value of MU (see CRITeria command, Section 3.20) from 0.5 (the default for trapezoidal integration) to 0.75 or 1.0 (equivalent to rectangular integration). The time step should also be reduced by a factor of 2-5, and the simulation should be rerun.

APPENDIX E

SAMPLE RUNS

One-Stage Amplifier

The following pages demonstrate some sample runs of the program. The first run is of the one-stage amplifier shown in Figure E-1.

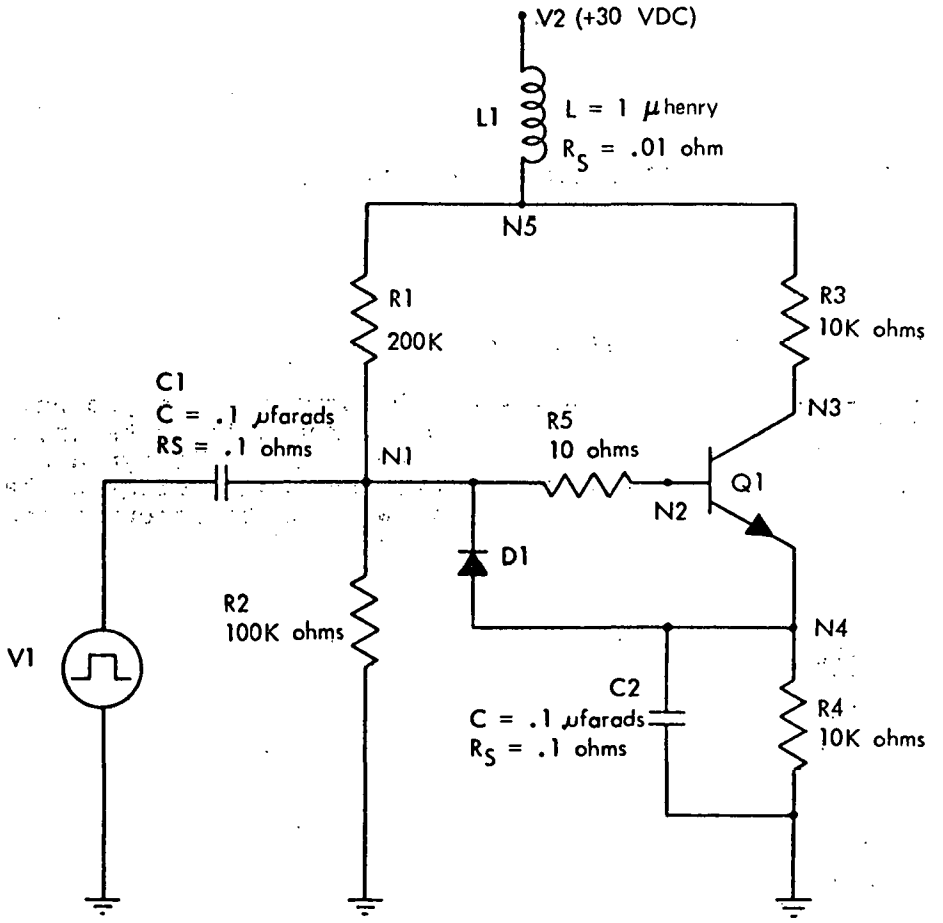


Figure E-1. One-Stage Amplifier

```
6000 system up


 77/05/27. 08.55.21.
SANDIA NOS SN53 ECS                          NOS 1.1-430
USER NUMBER, PASSWORD

TERMINAL:     37, TTY
RECOVER /SYSTEM: ATTACH,AITRAC/UN=LIBRARY

READY.
X,AITRAC



 SANDIA AITRAC-100      VERSION 1.0   UPDATES - 1



 AITRAC-100 READY  77/05/27  08.56.01

  * ? NEW <------------------------------ NEW command tells program that
  NAME: ? TEST                           new circuit is to be entered.
                                         (The asterisk is typed by the
  V1, ? 0 .1E-6 20 .9E-6 20 1E-6 0       program, and indicates that it
  V2, ? 30                               is waiting to accept a command.)
  V3, ?

 (1)S1, ?

 (1)R1, ? 5 1 .2E6
 (2)R2, ? 1 G .1E6
 (3)R3, ? 5 3 .1E5
 (4)R4, ? 4 G .1E5
 (5)R5, ? 1 2 10
 (6)R6, ?

 (6)C1, ? 1 V1 .1U
 (7)C2, ? 4 G .1U
 (8)C3, ?

 (8)L1, ? V2 5 1U .01
 (9)L2, ?

 (9)D1, ? 4 1
+PARAMS:
$$
? <------------------------------------ User hit carriage return here
+IS, MD, RDL, CDO, VDBI, TD, IPPD      to obtain diode headings.
$$
```

```
? 1E-9 2 .1E10 .11E-10 .75 1E-8 0
 (10)D2,?

 (10)Q1, (E-B-C)  ? 4 2 3
+PARAMS:
$$
?
+HFEN, HFEI, TN, TI, ICS, MC, CCO, VCBI
 RCL, IES, ME, CEO, VEBI, REL, IPPC, IPPE
$$
? 100 1 1E-9 .1E-6 .1E-10 1
? .1E-10 .75 .1E8 .1E-12 1 .5E-10 .75
? .1E9 0 0
 (12)Q2,?

 (12)T1,?

 (12)J1, (G-S-D)  ?

 (12)M1, (G-S-D-B)  ?

 (12)W1,?
  * ? LIST <------------------------ Circuit input complete; user
 TEST                                types LIST to get printout
      V1                0.0,         of circuit description.
      V      1.000E-07    2.000E+01,
      V      9.000E-07    2.000E+01,
      V      1.000E-06    0.0,

      V2                3.000E+01


  (1)R1     5     1     2.000E+05
  (2)R2     1     0     1.000E+05
  (3)R3     5     3     1.000E+04
  (4)R4     4     0     1.000E+04
  (5)R5     1     2     1.000E+01

  (6)C1     1    V1     1.000E-07   1.000E-03   1.000E+09
  (7)C2     4     0     1.000E-07   1.000E-03   1.000E+09

  (8)L1    V2     5     1.000E-06   1.000E-02   1.000E+09

  (9)D1     4     1         TE   0.        EG    1.11
      D       1.000E-09    2.000E+00    1.000E+09    1.100E-11
      D       7.500E-01    1.000E-08    0.

  (10)Q1     4     2     3         TE    0.        EG    1.11
      Q       1.000E+02    1.000E+00    1.000E-09    1.000E-07
      Q       1.000E-11    1.000E+00    1.000E-11    7.500E-01
      Q       1.000E+07    1.000E-13    1.000E+00    5.000E-11
      Q       7.500E-01    1.000E+08    0.           0.
```

```
 * ? PRINT NV1 NV4 BC1 BC3 RATIO 10 <----User specifies items to be
 * ? EXECUTE DC                          printed.

DC SOLUTION <----------Execute a dc analysis only.

    NV  1       NV  4        BC  1        BC  3
------------ ------------ ------------ ------------
 9.4893E+00   8.9006E+00   1.0255E-04   8.8242E-04


AITRAC-100 READY  77/05/27  09.02.23
                                  User specifies a time domain.
 * ? TIME 1E-6 5E-5 <-------------Execute a transient analysis IDC
 * ? EXECUTE IDC <                means program is to calculate dc
                                  initial conditions.
    TIME         DELTA        NV  1        NV  4        BC  1        BC  3
------------ ------------ ------------ ------------ ------------ ------------
 0.           0.           9.4893E+00   8.9006E+00   1.0255E-04   8.8242E-04

*START TIME= 0.           *END TIME=   5.0000E-05   *MAX DELTA= 1.0000E-06

 1.0000E-08   1.0000E-08   1.1480E+01   8.9101E+00   9.1427E-05   2.0671E-03
 1.0000E-07   1.0000E-08   2.8566E+01   9.8246E+00   7.9898E-06   2.0198E-03
 7.3000E-07   8.0000E-08   2.2128E+01   1.6263E+01   4.0090E-05   1.3769E-03
 9.0000E-07   1.0000E-08   2.1392E+01   1.6998E+01   4.2349E-05   1.2751E-03
 9.7000E-07   1.0000E-08   1.1608E+01   1.2782E+01   9.2432E-05   1.7231E-03
 1.0000E-06   1.0000E-08   8.6021E+00   9.7881E+00   1.0648E-04   2.0036E-03
 1.1150E-06   2.0000E-08   9.0113E+00   9.3801E+00   1.0531E-04   2.0642E-03
 1.1800E-06   5.0000E-09   9.0677E+00   9.3245E+00   1.0481E-04   2.0906E-03
 1.2500E-06   1.0000E-08   9.1010E+00   9.2920E+00   1.0457E-04   2.1156E-03
 3.5600E-06   1.0000E-06   9.1544E+00   9.2339E+00   1.0431E-04   2.9169E-06
 1.3560E-05   1.0000E-06   9.1558E+00   9.1421E+00   1.0430E-04   2.1440E-06
 2.3560E-05   1.0000E-06   9.1572E+00   9.0512E+00   1.0430E-04   2.1438E-06
 3.3560E-05   1.0000E-06   9.1587E+00   8.9612E+00   1.0429E-04   2.1439E-06
 4.3560E-05   1.0000E-06   9.1601E+00   8.8721E+00   1.0428E-04   2.1504E-06
 5.0000E-05   4.4000E-07   9.1610E+00   8.8152E+00   1.0411E-04   2.0144E-06


AITRAC-100 READY  77/05/27  09.03.07

 * ? LIST R1 <----------------------List information on resistor 1.
 (1)R1    5    1    2.000E+05

 * ? CHANGE R1 <--------------------Change the value of resistor 1.

(1)R1, ? 5 1 3E5
 * ? LIST R1 <----------------------Verify the change.
 (1)R1    5    1    3.000E+05
```

```
 * ? EXECUTE DC <-------Do a dc analysis with changed
                        parameter value.
DC SOLUTION

    NV  1        NV  4        BC  1        BC  3
------------ ------------ ------------ ------------
 7.1349E+00   6.5541E+00   7.6217E-05   6.5056E-04


AITRAC-100 READY  77/05/27  09.04.53

 * ? TEMPERATURE 30 <-------Now change temperature.
 * ? LIST TEMP
TEMPERATURE      3.000E+01
 * ? EXECUTE DC <---------- Do another dc analysis.

DC SOLUTION

    NV  1        NV  4        BC  1        BC  3
------------ ------------ ------------ ------------
 7.1341E+00   6.5635E+00   7.6220E-05   6.5149E-04


AITRAC-100 READY  77/05/27  09.05.42

 * ? CHANGE R1 <---------- Change the parameters back to their
                           original values.
(1)R1, ? 5 1 2E5
 * ? TEMP 25
 * ? LIST ON FILE DCAMP <---- Save the circuit on the file.
 * ? Q                        DCAMP.  A command exits from
READY.                        the program and returns to
BYE                           monitor level.


 09.06.50  77/05/27.
OFF RGMOSTE
TTY 037      6.409 SRU
```

Bridged Tee Circuit

The bridged-tee circuit, shown in Figure E-2, is analyzed in the following run. A plot of the transient response is obtained at the terminal.

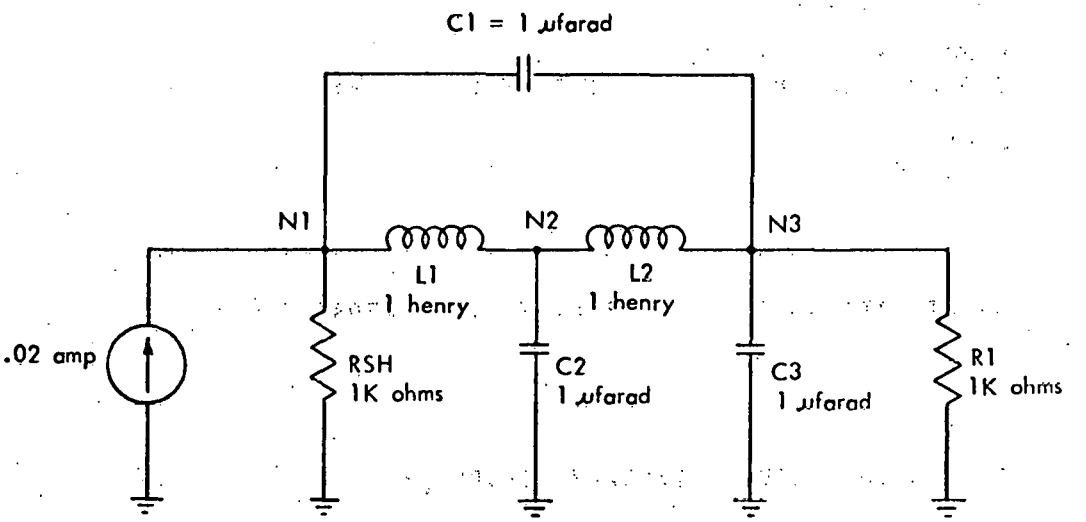


Figure E-2. Bridged-tee Circuit

```
6000 system up

 77/05/27. 09.08.23.
SANDIA NOS SN53 ECS                          NOS 1.1-430
USER NUMBER, PASSWORD

TERMINAL:      5, TTY
RECOVER /SYSTEM: ATTACH,AITRAC/UN=LIBRARY

READY.
X,AITRAC



 SANDIA AITRAC-100      VERSION 1.0   UPDATES = 1



 AITRAC-100 READY  77/05/27  09.09.03

  * ? NEW <------------------Start a new circuit.
  NAME: ? BRIDGED TEE CIRCUIT

  V1, ?

 (1)S1, ? 1 G -.02 1K CURRENT
 (2)S2, ?

 (2)R1, ? 3 0 1K
 (3)R2, ?

 (3)C1, ? 1 3 1U
 (4)C2, ? 2 0 1U
 (5)C3, ? 3 0 1U
 (6)C4, ?

 (6)L1, ? 1 2 1
 (7)L2, ? 2 3 1
 (8)L3, ?

 (8)D1, ?

 (8)Q1,  (E-B-C)  ?

 (8)T1, ?

 (8)J1,  (G-S-D)  ?

 (8)M1,  (G-S-D-B)  ?
```

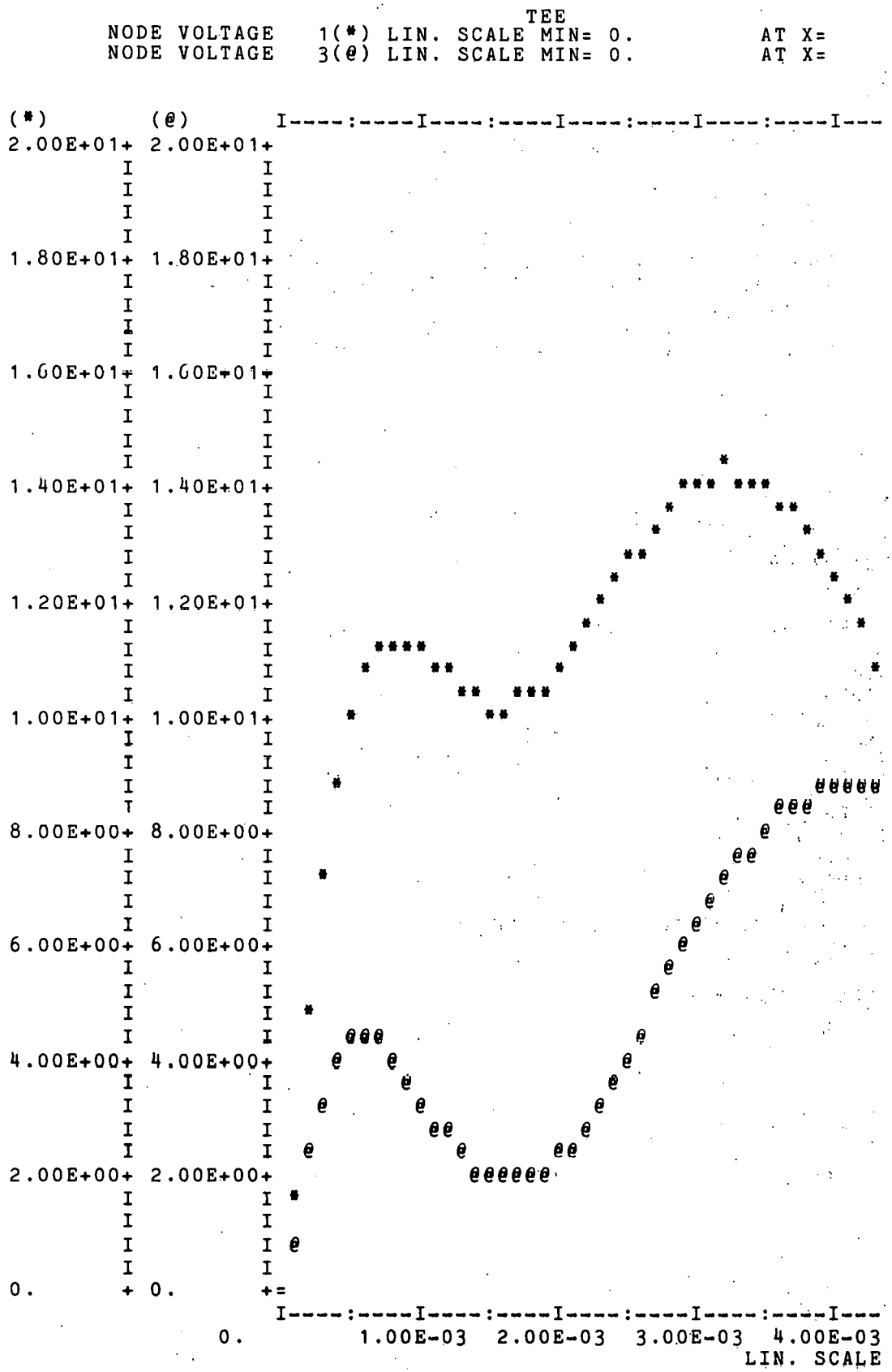
```
(8)W1, ?
 * ? LIST
BRIDGED TEE CIRCUIT

 (1)S1     1     0    -2.000E-02  1.000E+03 CURRENT

 (2)R1     3     0     1.000E+03

 (3)C1     1     3     1.000E-06  1.000E-03  1.000E+09
 (4)C2     2     0     1.000E-06  1.000E-03  1.000E+09
 (5)C3     3     0     1.000E-06  1.000E-03  1.000E+09

 (6)L1     1     2     1.000E+00  1.000E-03  1.000E+09
 (7)L2     2     3     1.000E+00  1.000E-03  1.000E+09
```

Specify time domain. User desires plot of node voltages 1 and 3 (auto. scaling)

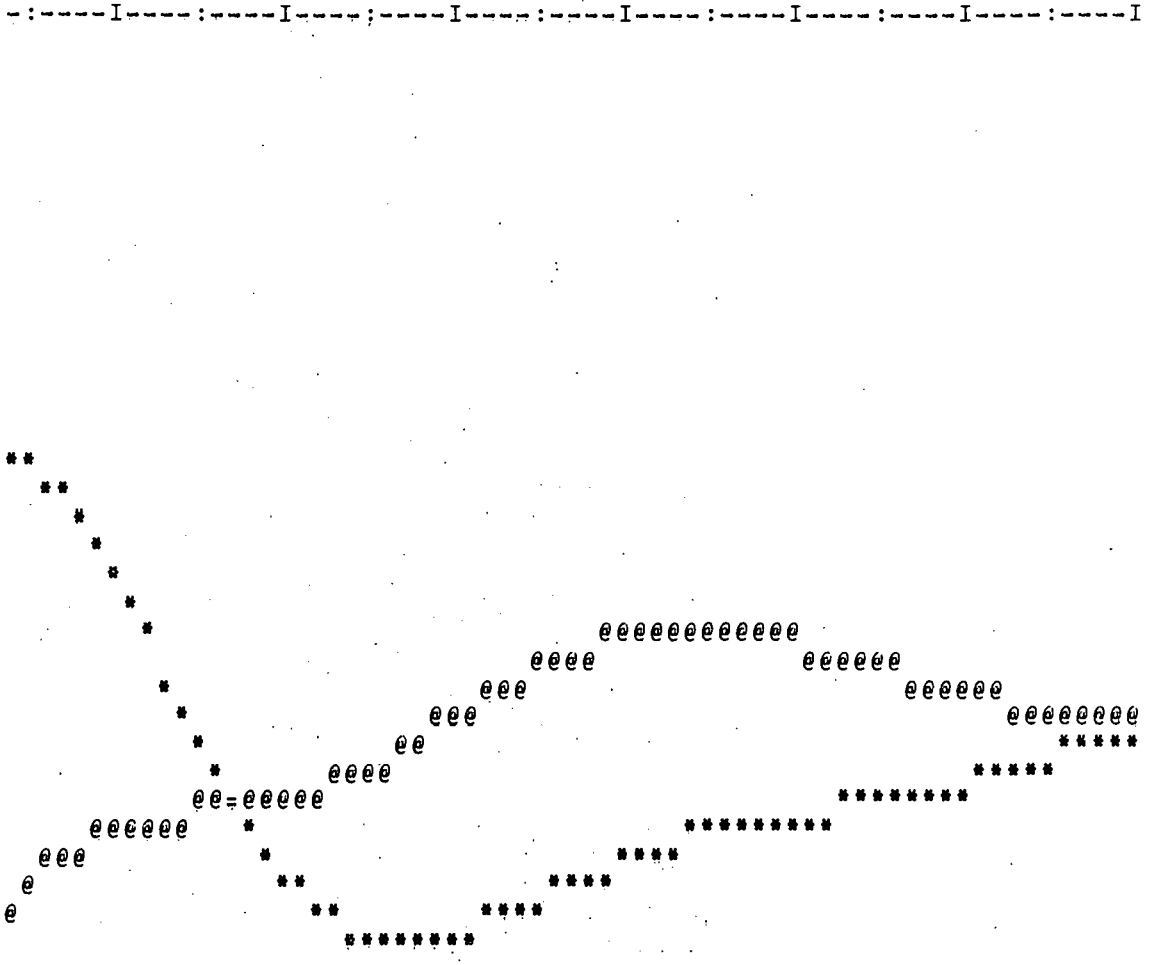
```
 * ? TIME 1E-4 1E-2
 * ? PLOT NV1 NV3
 * ? EXECUTE

*START TIME= 0.          *END TIME=  1.0000E-02  *MAX DELTA= 1.0000E-04

T= 9.0000E-04
T= 1.9000E-03
T= 2.9000E-03
T= 3.9000E-03
T= 4.9000E-03
T= 5.9000E-03
T= 6.9000E-03
T= 7.9000E-03
T= 8.9000E-03
T= 9.9000E-03
```

The value of time is printed out for every tenth point stored for plotting, so that user is informed of solution progress.

```
### NEW PLOT & SIEVE ROUTINE ENTERED ###

   101 POINTS TO BE PLOTTED...

 #

 ENTER PLOT TITLE: ? TEE

----- SET PAPER 3 HOLES ABOVE FOLD, THEN HIT CARRIAGE RETURN -----
?
```


TEE
NODE VOLTAGE 1(*) LIN. SCALE MIN= 0. AT X=
NODE VOLTAGE 3(@) LIN. SCALE MIN= 0. AT X=
(*)
(@)
2.00E+01
1.80E+01
1.60E+01
1.40E+01
1.20E+01
1.00E+01
8.00E+00
6.00E+00
4.00E+00
2.00E+00
0.
1.00E-03
2.00E-03
3.00E-03
4.00E-03
LIN. SCALE

```
 AT X= 0.          MAX= 1.4220E+01 AT X= 3.2000E-03
 AT X= 0.          MAX= 1.1631E+01 AT X= 7.4000E-03
```



```
  4.00E-03  5.00E-03  6.00E-03  7.00E-03  8.00E-03  9.00E-03  1.00E-02
LIN. SCALE  TIME (SECONDS)

AITRAC-100 READY  77/05/27  09.32.06

 * ? Q
```

Radiation Effects

The simple circuit shown in Figure E-3 is used to show the effects of nuclear radiation on semiconductors.

The IPPC and IPPE parameters for the transistor, as well as the specifications for the HP Functions, must be specified by the user.

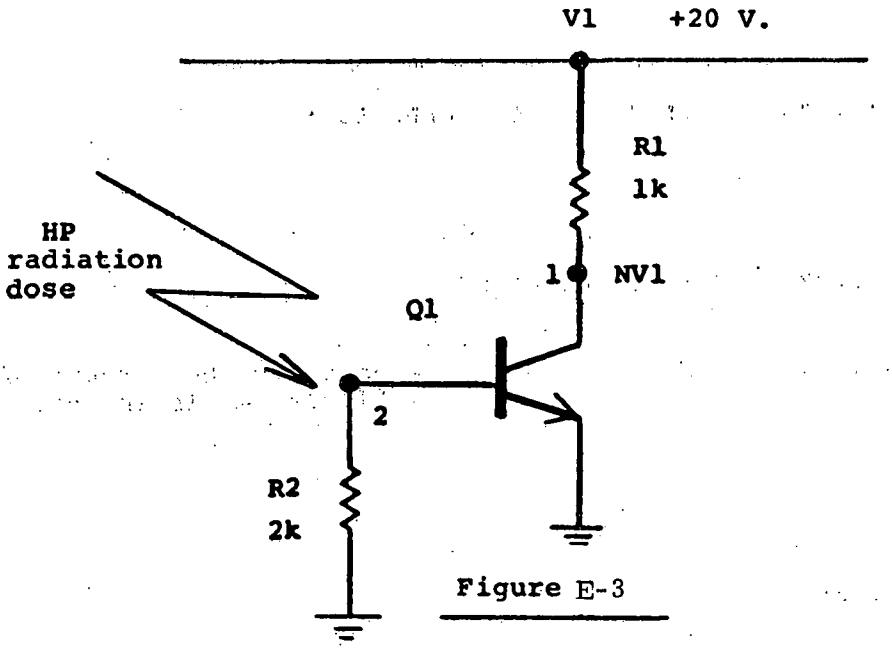


Figure E-3

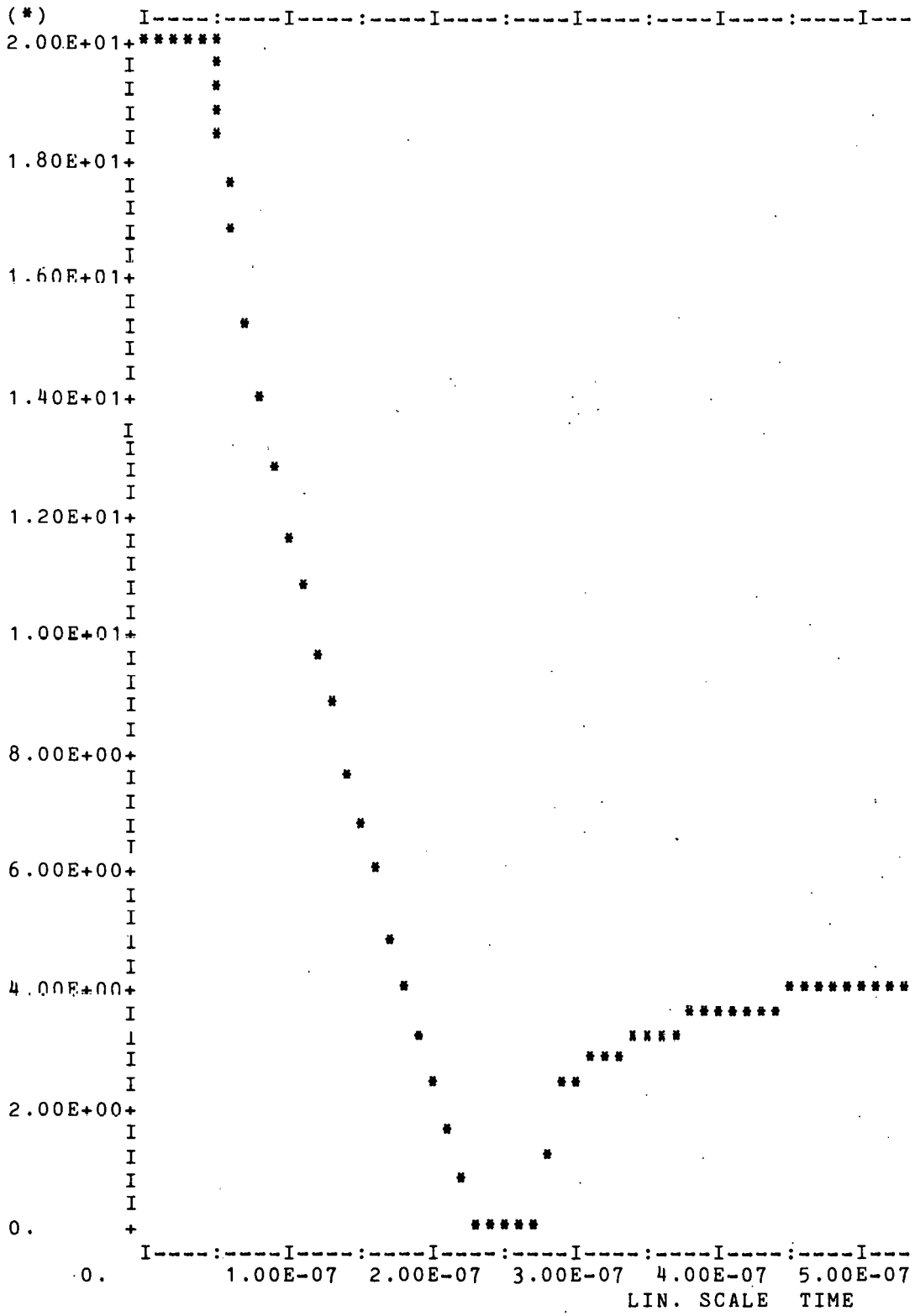
Reference: 'Improvements in Transistor Models and Circuit Hardening for TREE Applications', Technical Report Number: AFWL-TR-67-71, December 1967, Air Force Weapons Laboratory, Kirtland Air Force Base, Albuquerque, New Mexico.

```
6000 system up

 77/05/27. 09.43.12.
SANDIA NOS SN53 ECS                          NOS 1.1-430
USER NUMBER, PASSWORD
                                              AI TRAC is obtained
TERMINAL:     13, TTY                         from the library
RECOVER /SYSTEM: ATTACH,AITRAC/UN=LIBRARY

READY.
X.AITRAC



 SANDIA AITRAC-100     VERSION 1.0   UPDATES = 1




 AITRAC-100 READY  77/05/27  09.44.08

  * ? NEW
  NAME: ? RADIATION EFFECTS              - data for the circuit of
                                         - Figure V-3 is entered
  V1, ? 20
  V2, ?

 (1)S1, ?

 (1)R1, ? 1 V1 1K
 (2)R2, ? 2 G 2K
 (3)R3, ?

 (3)C1, ?

 (3)L1, ?

 (3)D1, ?

 (3)Q1,  (E-B-C)  ? 0 2 1
+PARAMS:
$$
? 6.4E1 7.5 1.227E-7 1.59E-7           - transistor parameters
? 3.65E-6 1.025 1.16E-11 1.2           - including IPPC and IPPE
? 1E7 3.11E-6 1.03 2.19E-11            - are entered
? 1.2 1E7 1E-10 1E-12
 (5)Q2, ?

 (5)T1, ?
```

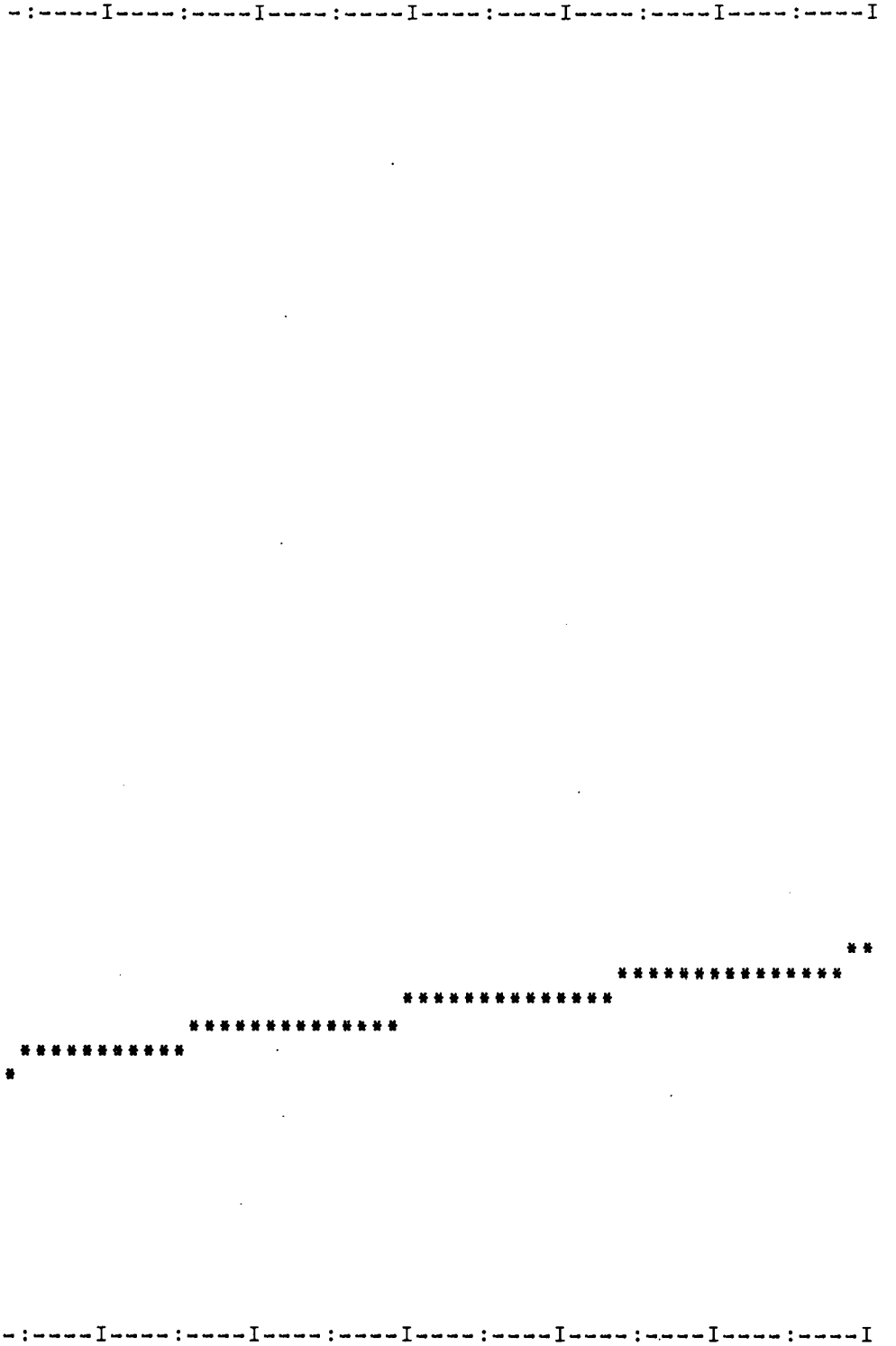
```
(5)J1,  (G-S-D)  ?

(5)M1,  (G-S-D-B)  ?

(5)W1, ?
 * ? HP 1 0 5E-8 1E8 2.5E-7 0          - Type 1 HP function entered
 * ? SAVE RADIATE                      - data is saved on file
 * ? LI                                  "RADIATE"
RADIATION EFFECTS
     V1                 2.000E+01      - entire file is listed
                                       - for verification of input

 (1)R1     1    V1     1.000E+03
 (2)R2     2     0     2.000E+03



 (3)Q1     0     2     1        TE    0.        EG    1.11
     Q        6.400E+01    7.500E+00    1.227E-07    1.590E-07
     Q        3.650E-06    1.025E+00    1.160E-11    1.200E+00
     Q        1.000E+07    3.110E-06    1.030E+00    2.190E-11
     Q        1.200E+00    1.000E+07    1.000E-10    1.000E-12



    HP              1     0.0,
           5.000E-08      1.000E+08,
           2.500E-07      0.0,

IMPROVED RADIATION RESPONSE CALCULATIONS USED

 * ? PLOT NV1 0 20 TIME 0 1U
 * ? TIME 1E-8 1U
 * ? EX IDC

*START TIME= 0.          *END TIME=  1.0000E-06  *MAX DELTA= 1.0000E-08

T= 4.5000E-08
T= 5.5110E-08
T= 1.5023E-07
T= 2.5000E-07                        - execution....
T= 2.6023E-07
T= 3.6023E-07                        - time is printed out
T= 4.6023E-07                        - for every 10 points
T= 5.6023E-07                        - which are calculated
T= 6.6023E-07
T= 7.6023E-07
T= 8.6023E-07
T= 9.6023E-07

### NEW PLOT & SIEVE ROUTINE ENTERED ###
 #

 ENTER PLOT TITLE: ? RADIATION

----- SET PAPER 3 HOLES ABOVE FOLD, THEN HIT CARRIAGE RETURN     -----
?
```

RADIATION

NODE VOLTAGE 1(*) LIN. SCALE MIN= 1.1841E-01 AT X=



AT X= 2.5127E-07 MAX= 1.9994E+01 AT X= 0.

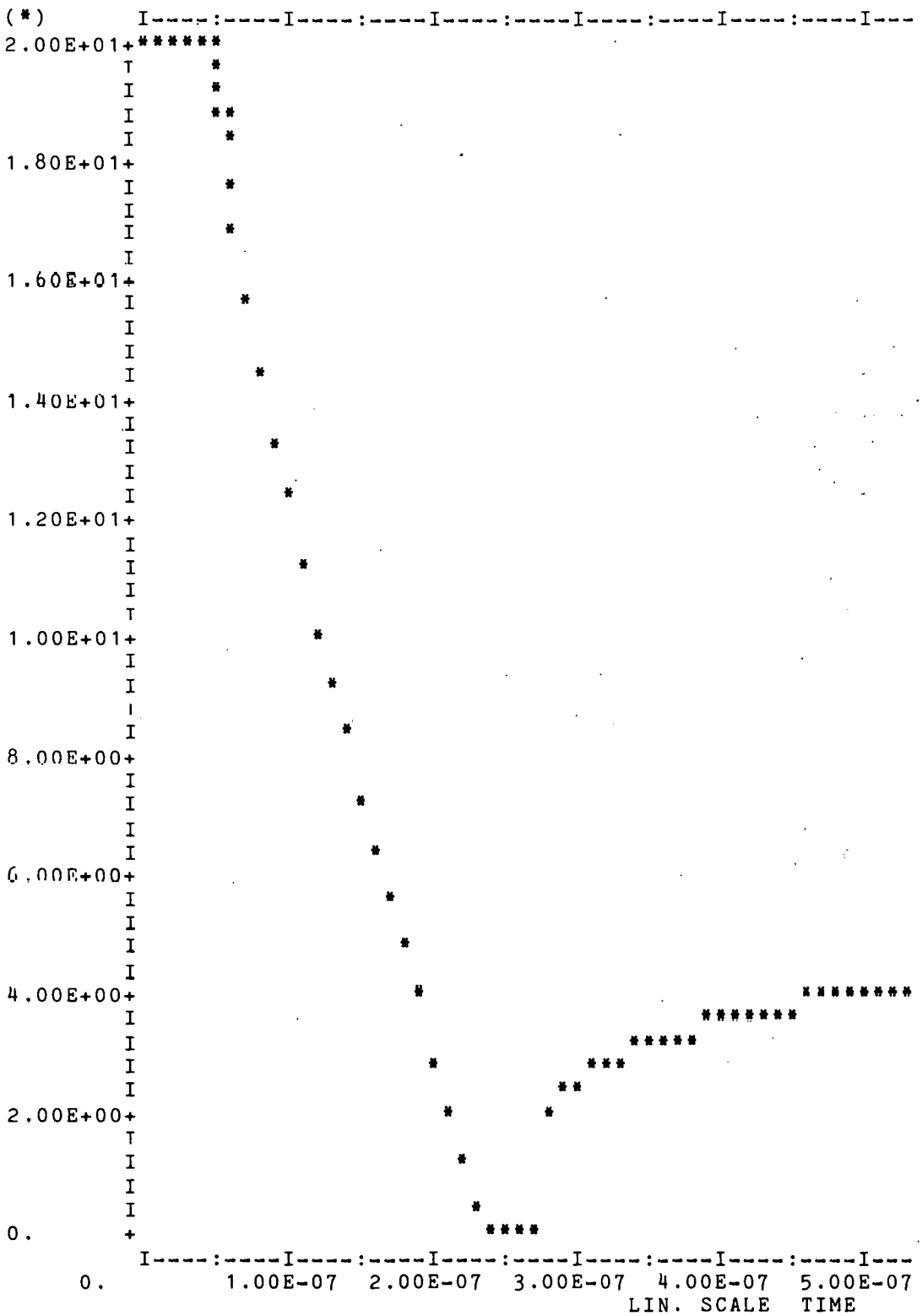


```
AITRAC-100 READY  77/05/27  10.03.13

 * ? HP 0 0 5E-8 0 5.5E-8 1E8 2.5E-7 1E8 2.55E-7 0 1U 0
   * ? LI HP
      HP              0     0.0,
              5.000E-08     0.0,
              5.500E-08     1.000E+08,
              2.500E-07     1.000E+08,
              2.550E-07     0.0,
              1.000E-06     0.0,

  IMPROVED RADIATION RESPONSE CALCULATIONS USED

   * ? EX IDC

  *START TIME= 0.          *END TIME=  1.0000E-06  *MAX DELTA= 1.0000E-08

  T= 4.5000E-08
  T= 5.4500E-08
  T= 1.1050E-07
  T= 2.1050E-07
  T= 2.5300E-07
  T= 2.8050E-07
  T= 3.8050E-07
  T= 4.8050E-07
  T= 5.8050E-07
  T= 6.8050E-07
  T= 7.8050E-07
  T= 8.8050E-07
  T= 9.8050E-07

  ### NEW PLOT & SIEVE ROUTINE ENTERED ###


   #

   ENTER PLOT TITLE: ? RADIATION

  ----- SET PAPER 3 HOLES ABOVE FOLD, THEN HIT CARRIAGE RETURN      -----
 ?
```

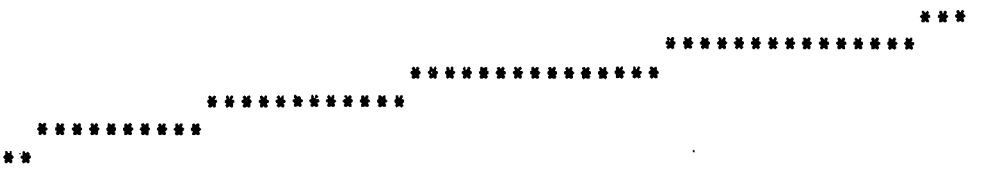
1

RADIATION

NODE VOLTAGE 1(*) LIN. SCALE MIN= 1.2665E-01 AT X=

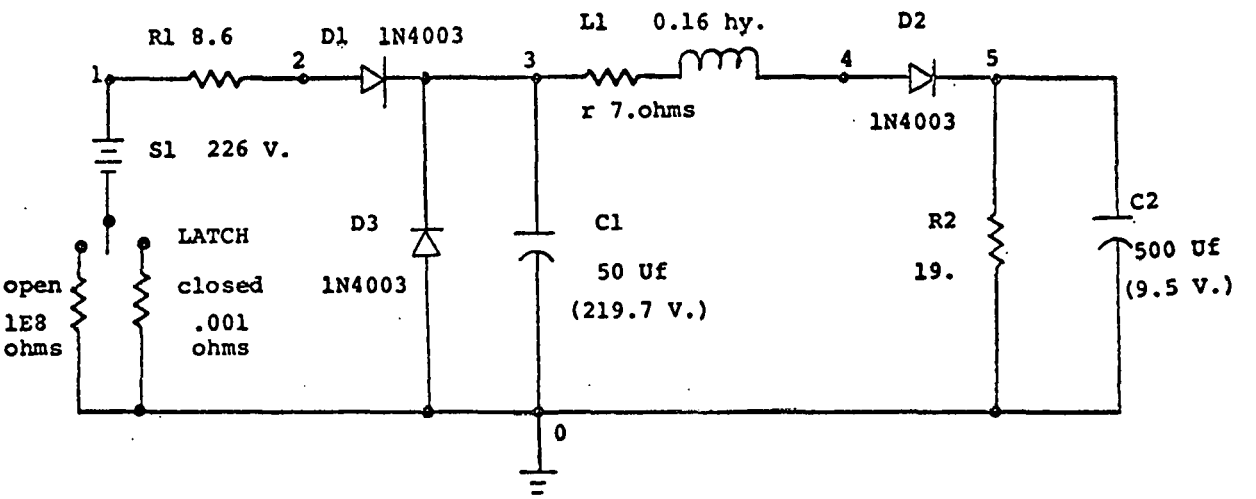


```
 AT X= 2.5550E-07    MAX= 1.9994E+01 AT X= 0.

-:----I----:----I----:----I----:----I----:----I----:----I----:----I
```



```
-:----I----:----I----:----I----:----I----:----I----:----I----:----I
  5.00E-07  6.00E-07  7.00E-07  8.00E-07  9.00E-07  1.00E-06
  TIME (SECONDS)
```

Switching Regulator



When NV5 exceeds 18.0 Volts, latch opens, after 15 msec.
When NV5 drops below 6.0 Volts, latch closes, after 15 msec.
Initially latch is closed. V3 = 219.7 V. ; V5 = 9.5 V.

Figure E-4

SPEQ FORTRAN Variables

V(5) Node Voltage 5

TE Present time

TDO Time to open latch

TDC Time to close latch

VAL2(1) Series resistance in floating voltage source(1)
Latch open = 1.E8, Latch closed = .001 Ohms.

SETOPN latch to open

SETCLO latch to close

OPEN open or close latch

"SPEQ" Implementation for Switching Regulator

The following four logical tests are used to open and close the latch, with a fifteen millisecond delay, at the correct times:

1. IF the latch is not set to open, and the controlling node voltage, V(5) is greater than 18.0 Volts:
   then set latch to open, and record the opening time, TDO, and proceed with test 3.
   otherwise, proceed with test 2.

2. IF the latch is not set to close, and the controlling node voltage, V(5) is less than 6.0 Volts:
   then, set latch to close, and record closing time, TDC, and proceed with test 3.
   otherwise, proceed with test 3.

3. IF the latch is set to open, and the present time, TE, is greater than the opening time TDO:
   then, open latch and set latch not to open; proceed with program
   otherwise, proceed with test 4.

4. IF latch is set to close, and present time TE, is greater than the closing time TDC:
   then, close latch, and set latch not to close; proceed with program
   otherwise, proceed with program

First, the logic and implementation of the FORTRAN equations must be thought out by the engineer. Then the actual coding of the SPEQ subroutine can be done. The steps outlined in Section 5.5 should be followed.

The SPEQ subroutine is obtained from the library.

The SPEQ subroutine is edited using XEDIT.

SPEQ is saved under the user's chosen name, e.g., TRHAIRY (the name was chosen since this could be a "hairy" problem!).

The MAKEQ procedure file is obtained from the library.

The MAKEQ procedure file is run, which enters the user directly into AITRAC after subroutine completion.

The user enters the topology data for the switching regulator circuit of Figure E-4 directly at the terminal or from a previously prepared file. In this case the circuit data file was prepared and saved with the name "HAIRY."

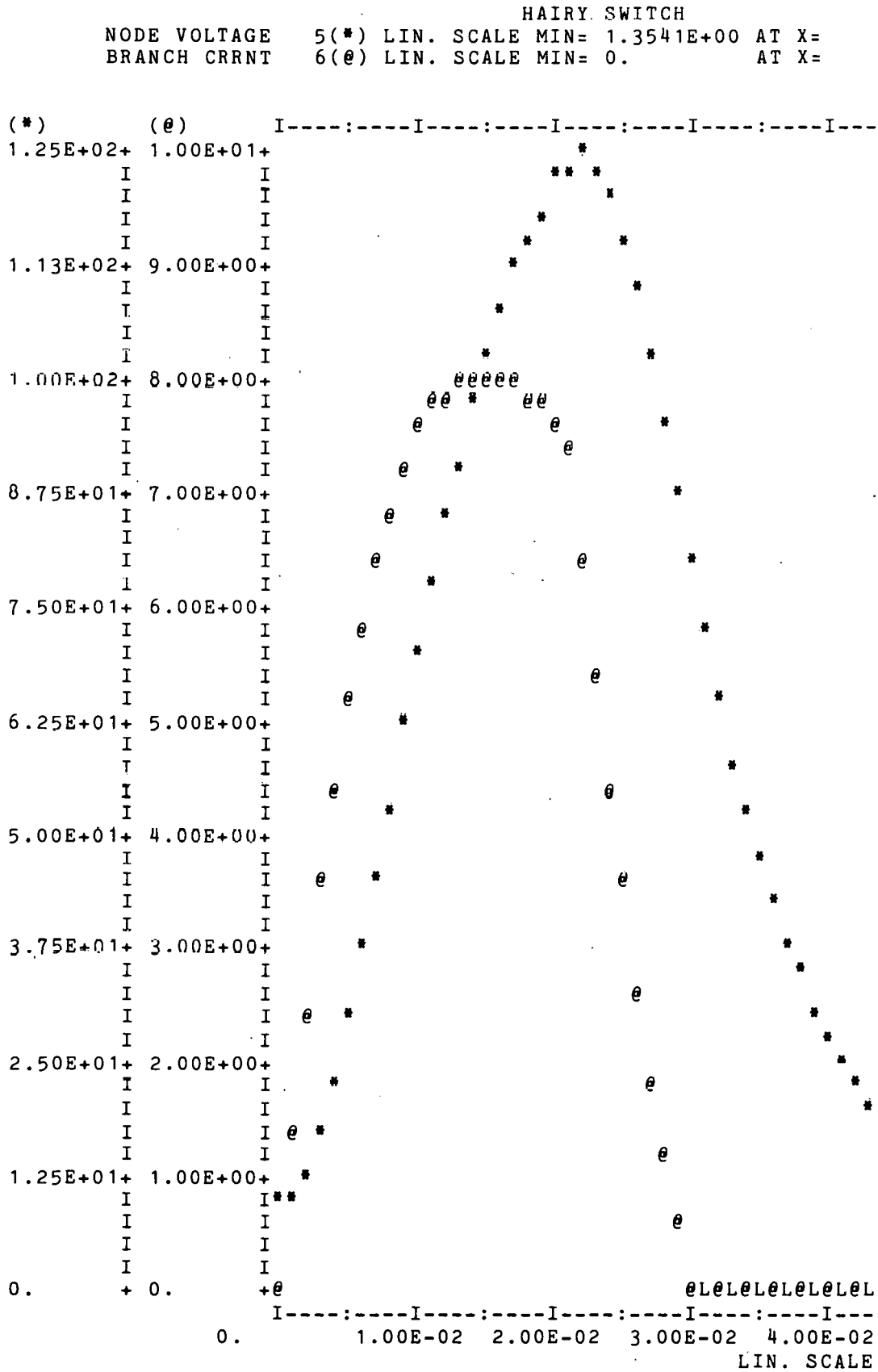
The user can now perform runs to check out his circuit. Listed first is only that portion of the file that contains the FORTRAN coding.

```
C
      LOGICAL OPEN, SETOPN, SETCLO
      DATA OPEN, SETOPN, SETCLO /.FALSE.,.FALSE., .FALSE. /
      DATA TOLCUR / .001 /
      GO TO (9000,9003,9002,9001), KK4
9000  IF(ISETUP .EQ. 1) GOTO 9004
C  CALLS TO SUBROUTINE * BUILD * INSERTED HERE
      RETURN
9004  CONTINUE
C       PART 1 - MATRIX AND TIME FUNCTIONS HERE
      RETURN
 9003 CONTINUE
C     PART 2
      RETURN
 9002 CONTINUE
C     PART 3
C  ZERO 'T' VECTOR FOR SUMMATION OF NODAL CURRENTS
      DO 1000 J=1,NV
      T(J)=0.0
1000  CONTINUE
      RETURN
 9001 CONTINUE
C     PART 4 - AUXILIARY EQUATIONS AFTER THIS CARD.
C  CHECK SUMMATION OF NODAL CURRENTS
C
      DO 1001 J=1,NV
      IF((ABS(T(J)) - TOLCUR ) .GT. 0.0) WRITE(8,5000)J,T(J)
5000  FORMAT(" NODE (",I3,") CURRENT= ",E12.5,"$$$$$$")
1001  CONTINUE
C
C  CODE FOR HAIRY SWITCH
C
C
      IF(.NOT.(.NOT. SETOPN .AND. V(5) .GT. 18.)) GO TO 10
      SETOPN= .TRUE.
      TDO=TE + 15E-3
      GOTO 20
10    CONTINUE
      IF(.NOT.(.NOT. SETCLO .AND. V(5) .LT. 6.)) GO TO 20
      SETCLO = .TRUE.
      TDC=TE + .015
20    CONTINUE
      IF(.NOT.(SETOPN .AND. TE .GT. TDO)) GO TO 30
      OPEN= .TRUE.
      SETOPN= .FALSE.
      GOTO 40
30    CONTINUE
      IF(.NOT.(SETCLO .AND. TE .GT. TDC)) GOTO 40
      OPEN= .FALSE.
      SETCLO= .FALSE.
40    CONTINUE
      VAL2(1)= .001
      IF(OPEN) VAL2(1)= 1E8
C  DIAGNOSTIC PRINTOUTS....PRINTS STATE OF SWITCHES
      WRITE(8,600)TE,SETOPN,SETCLO,OPEN,TDO,TDC
600   FORMAT(1X,E11.4,3L1,2(E12.4))
C
      RETURN
      END
END OF FILE
>?
```

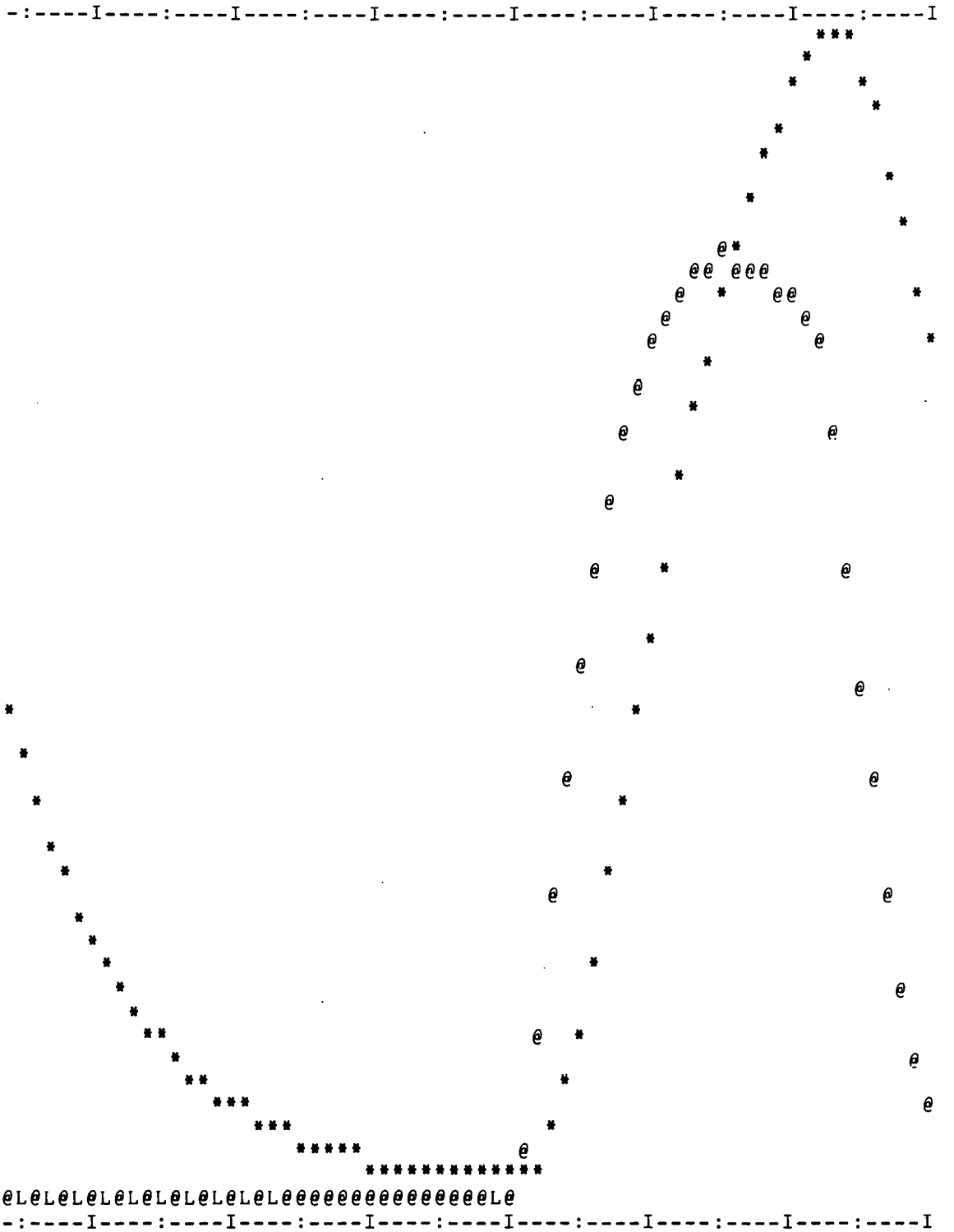
FORTRAN Coding of SPEQ for
Switching Regulator Simulation

```
 GET,MAKEQ/UN=LIBRARY
/-MAKEQ(FN=TRHAIRY)
 COMPILATION OF FN OK
GETTING LIBRARY FILES
 ADDING USER'S EQUATIONS
GENERATING SPECIAL AITRAC OVERLAYS
LOADMAP AVAILABLE ON FILE 'LOADMAP'
AITRAC OVERLAYS ON LOCAL FILE 'AITRAC'


 SANDIA AITRAC-100      VERSION 1.0    UPDATES = 1


AITRAC-100 READY  77/09/07  13.55.47

 * ? NEW HAIRY LIST
 HAIRY SWITCH
      V1                  0.0

  (1)S1     1     0     2.260E+02   1.000E-03

  (2)R1     1     2     8.600E+00
  (3)R2     5     0     1.900E+01

  (4)C1     3     0     5.000E-05   1.000E-03   1.000E+09
  (5)C2     5     0     5.000E-04   1.000E-03   1.000E+09

  (6)L1     3     4     1.600E-01   7.000E+00   1.000E+09

  (7)D1     2     3        TE   0.        EG    1.11      SE  1N4003
     ;D       4.200E-09    1.770E+00    2.000E+10    2.300E-11
     ;D       1.000E+00    6.830E-06    0.
  (8)D2     4     5        TE   0.        EG    1.11      COPY D1
  (9)D3     0     3        TE   0.        EG    1.11      COPY D1

 PL NV5 0. 125. BC6 0. 10.
 SET SPECIAL ON
 TIME,
    1.000E-03     1.000E-01,

  * ? EX VO 3 219.7 5 9.5
 USING SPECIAL EQUATIONS
```

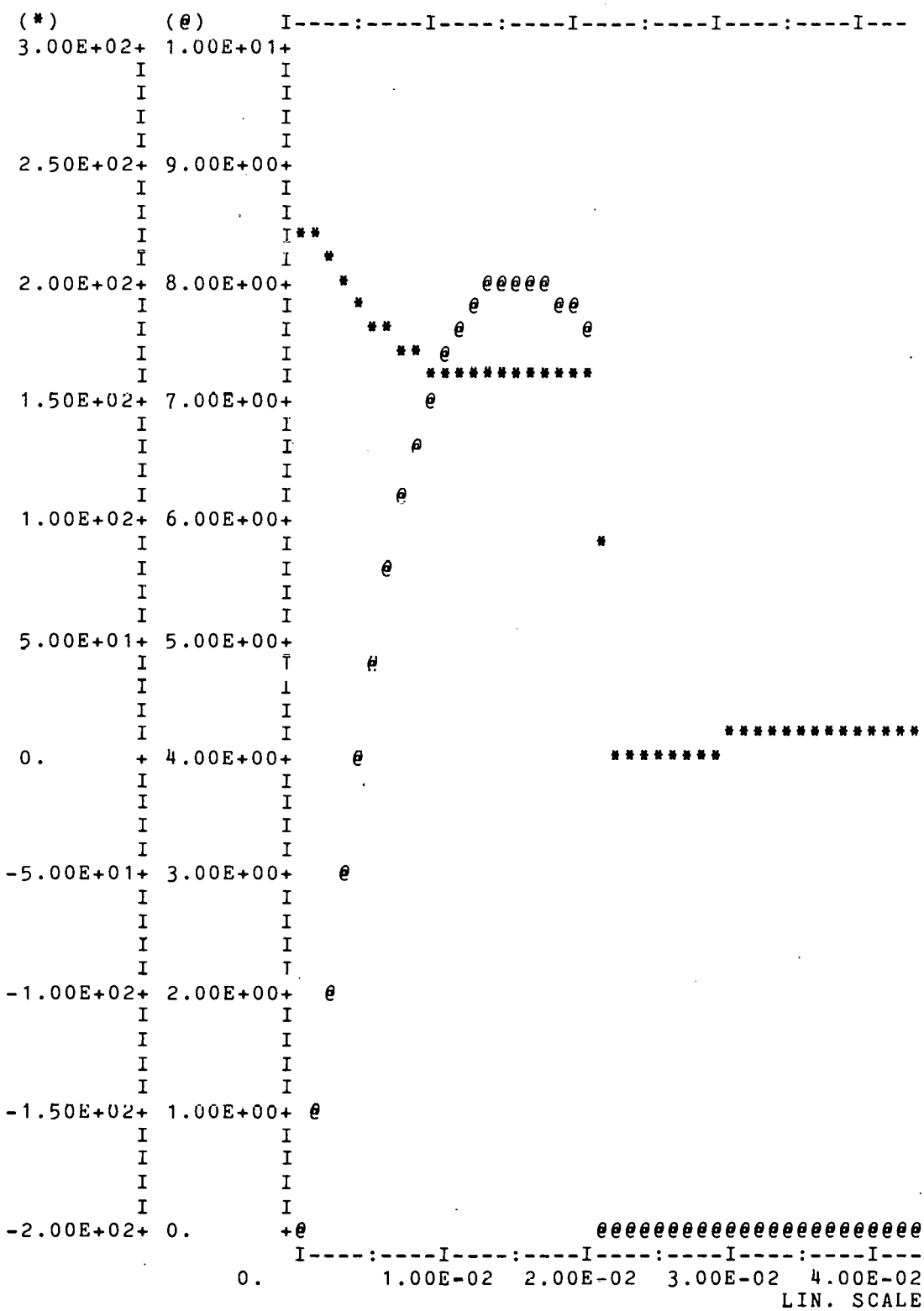
```
*START TIME= 0.          *END TIME=  1.0000E-01  *MAX DELTA= 1.0000E-03

T= 9.0000E-03
T= 1.9000E-02
T= 2.9000E-02
T= 3.9000E-02
T= 4.9000E-02
T= 5.9000E-02
T= 6.8500E-02
T= 7.8500E-02
T= 8.8500E-02
T= 9.8500E-02

### NEW PLOT & SIEVE ROUTINE ENTERED ###


   102 POINTS TO BE PLOTTED...
```


HAIRY SWITCH
NODE VOLTAGE 5(*) LIN. SCALE MIN= 1.3541E+00 AT X=
BRANCH CRRNT 6(@) LIN. SCALE MIN= 0. AT X=
(*)
(@)
1.25E+02
1.00E+01
1.13E+02
9.00E+00
1.00E+02
8.00E+00
8.75E+01
7.00E+00
7.50E+01
6.00E+00
6.25E+01
5.00E+00
5.00E+01
4.00E+00
3.75E+01
3.00E+00
2.50E+01
2.00E+00
1.25E+01
1.00E+00
0.
0.
0.
1.00E-02
2.00E-02
3.00E-02
4.00E-02
LIN. SCALE

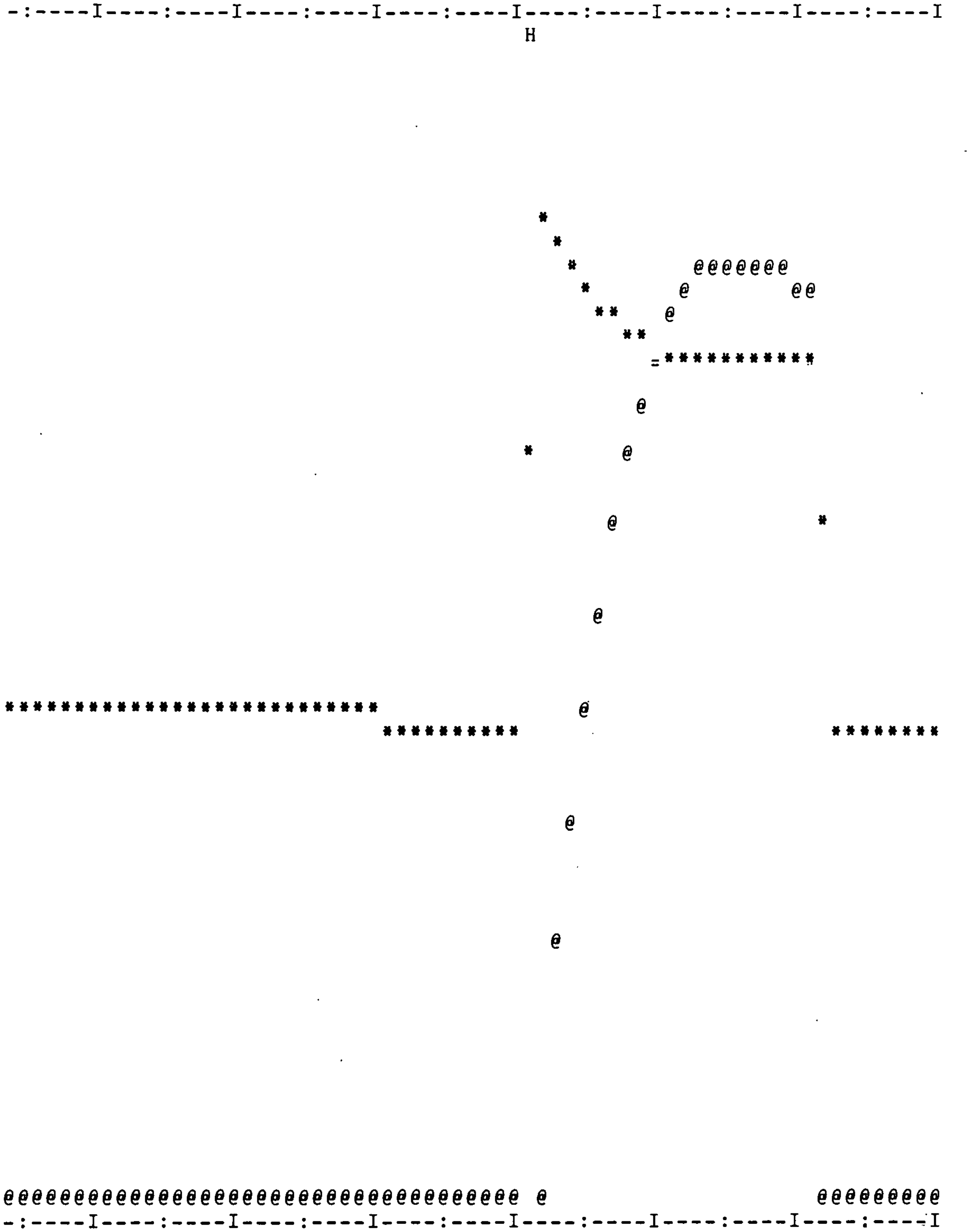
AT X= 7.0500E-02 MAX= 1.2491E+02 AT X= 9.3500E-02
AT X= 0. MAX= 8.1033E+00 AT X= 8.5500E-02



```
AITRAC-100 READY  77/07/15  13.41.11

 * ? PLOT NV3 -200. 300. BC2 0. 10. NOW
                                        HAIRY SWITCH
      NODE VOLTAGE   3(*) LIN. SCALE MIN=-9.6226E-01 AT X=
      BRANCH CRRN    2(@) LIN. SCALE MIN= 0.         AT X=
```



AT X= 2.3000E-02   MAX= 2.2499E+02 AT X= 7.2500E-02
AT X= 0.   MAX= 8.0966E+00 AT X= 8.6500E-02



4.00E-02 5.00E-02 6.00E-02 7.00E-02 8.00E-02 9.00E-02 1.00E-01
LIN. SCALE TIME (SECONDS)

AITRAC-100 READY 77/07/15 13.37.28

* ? PLOT BC6 NV3 OSUM NOW

TABLE OF PLOTTED VALUES

NO.	X-AXIS	* VALUE	@ VALUE
1	0.	0.	2.19700E+02
2	1.00000E-03	1.30017E+00	2.16604E+02
3	2.00000E-03	2.46788E+00	2.08324E+02
4	3.00000E-03	3.50946E+00	1.98704E+02
5	4.00000E-03	4.42027E+00	1.90381E+02
6	5.00000E-03	5.20576E+00	1.83152E+02
7	6.00000E-03	5.87260E+00	1.76972E+02
8	7.00000E-03	6.42873E+00	1.71776E+02
9	8.00000E-03	6.88292E+00	1.67491E+02
10	9.00000E-03	7.24449E+00	1.64038E+02
11	1.00000E-02	7.52297E+00	1.61336E+02
12	1.10000E-02	7.72788E+00	1.59302E+02
13	1.20000E-02	7.86851E+00	1.57856E+02
14	1.30000E-02	7.95380E+00	1.56919E+02
15	1.40000E-02	7.99220E+00	1.56418E+02
16	1.50000E-02	7.99158E+00	1.56281E+02
17	1.60000E-02	7.95916E+00	1.56444E+02
18	1.70000E-02	7.90152E+00	1.56848E+02
19	1.80000E-02	7.82454E+00	1.57441E+02
20	1.90000E-02	7.73342E+00	1.58174E+02
21	2.00000E-02	7.63268E+00	1.59006E+02
22	2.10000E-02	7.30216E+00	8.64139E+01
23	2.20000E-02	6.48803E+00	-9.50476E-01
24	2.30000E-02	5.44256E+00	-9.62264E-01
25	2.40000E-02	4.45417E+00	-9.27798E-01
26	2.50000E-02	3.53063E+00	-9.49800E-01
27	2.60000E-02	2.67773E+00	-8.88412E-01
28	2.70000E-02	1.89919E+00	-9.31430E-01
29	2.80000E-02	1.20277E+00	9.74489E-01
30	2.90000E-02	5.83186E-01	-9.02299E-01
31	3.00000E-02	6.34460E-02	8.71740E+00
32	3.10000E-02	-4.57930E-04	7.96881E+00
33	3.20000E-02	2.83255E-07	7.97343E+00
34	3.30000E-02	-9.71701E-07	7.97348E+00
35	3.40000E-02	1.95111E-02	7.77840E+00
36	3.50000E-02	-1.32580E-04	7.58468E+00
37	3.60000E-02	1.06490E-02	7.47954E+00
38	3.70000E-02	-7.24573E-05	7.37383E+00
39	3.80000E-02	1.35464E-02	7.23912E+00
40	3.90000E-02	-9.20696E-05	7.10464E+00
41	4.00000E-02	6.33651E-03	7.04223E+00
42	4.10000E-02	-4.33418E-05	6.97935E+00
43	4.20000E-02	9.50604E-03	6.88476E+00
44	4.30000E-02	-6.46414E-05	6.79040E+00
45	4.40000E-02	3.62855E-03	6.75480E+00
46	4.50000E-02	-2.62369E-05	6.71882E+00
47	4.60000E-02	6.72110E-03	6.65191E+00
48	4.70000E-02	-4.58170E-05	6.58521E+00

49	4.80000E-02	2.00460E-03	6.56566E+00
50	4.90000E-02	-1.36818E-05	6.54580E+00
51	5.00000E-02	4.71768E-03	6.49879E+00
52	5.10000E-02	-3.25645E-05	6.45199E+00
53	5.20000E-02	1.11131E-03	6.44125E+00
54	5.30000E-02	-7.63374E-06	6.43025E+00
55	5.40000E-02	3.22795E-03	6.39809E+00
56	5.50000E-02	1.13110E-03	6.35455E+00
57	5.60000E-02	2.61927E-03	6.31709E+00
58	5.70000E-02	6.72199E-03	6.22371E+00
59	5.80000E-02	1.25045E-02	6.03149E+00
60	5.90000E-02	1.89204E-02	5.71727E+00
61	6.00000E-02	2.49189E-02	5.27892E+00
62	6.10000E-02	2.95683E-02	4.73409E+00
63	6.20000E-02	3.21479E-02	4.11697E+00
64	6.30000E-02	3.22655E-02	3.47288E+00
65	6.40000E-02	2.98809E-02	2.85146E+00
66	6.50000E-02	2.52962E-02	2.29974E+00
67	6.60000E-02	1.90997E-02	1.85583E+00
68	6.70000E-02	1.20789E-02	1.54409E+00
69	6.80000E-02	5.16177E-03	1.37174E+00
70	6.85000E-02	2.03219E-03	1.33579E+00
71	6.95000E-02	-1.42011E-05	1.31566E+00
72	7.05000E-02	9.50898E-07	1.31584E+00
73	7.15000E-02	3.58555E-01	1.19922E+02
74	7.25000E-02	1.37860E+00	2.24988E+02
75	7.35000E-02	2.60500E+00	2.06749E+02
76	7.45000E-02	3.66768E+00	1.97451E+02
77	7.55000E-02	4.59370E+00	1.88938E+02
78	7.65000E-02	5.38787E+00	1.81615E+02
79	7.75000E-02	6.05797E+00	1.75387E+02
80	7.85000E-02	6.61284E+00	1.70185E+02
81	7.95000E-02	7.06213E+00	1.65929E+02
82	8.05000E-02	7.41594E+00	1.62533E+02
83	8.15000E-02	7.68449E+00	1.59908E+02
84	8.25000E-02	7.87790E+00	1.57968E+02
85	8.35000E-02	8.00601E+00	1.56626E+02
86	8.45000E-02	8.07820E+00	1.55800E+02
87	8.55000E-02	8.10330E+00	1.55412E+02
88	8.65000E-02	8.08950E+00	1.55389E+02
89	8.75000E-02	8.04429E+00	1.55664E+02
90	8.85000E-02	7.97443E+00	1.56176E+02
91	8.95000E-02	7.88595E+00	1.56870E+02
92	9.05000E-02	7.78417E+00	1.57698E+02
93	9.15000E-02	7.67368E+00	1.58618E+02
94	9.25000E-02	7.33328E+00	8.57556E+01
95	9.35000E-02	6.51100E+00	-9.51541E-01
96	9.45000E-02	5.45996E+00	-9.62062E-01
97	9.55000E-02	4.46649E+00	-9.29382E-01
98	9.65000E-02	3.53837E+00	-9.48991E-01
99	9.75000E-02	2.68139E+00	-8.91768E-01
100	9.85000E-02	1.89929E+00	-9.30118E-01
101	9.95000E-02	1.19709E+00	8.60631E-02

Application of Switches

This circuit demonstrates the use of switches and their operation. By using the switching elements many simulations can be performed without the use of special equations (SPEQ).

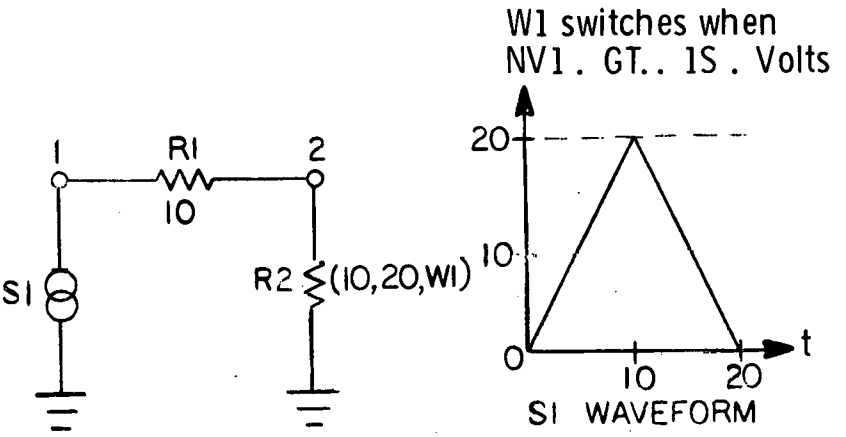


Figure E-5. Circuit with Switch Element

```
SANDIA AITRAC-100      VERSION 1.0 . UPDATES = 1



AITRAC-100 READY  77/07/18  14.28.03

 * ? NEW
 NAME: ? UP AND DOWN SWITCHING

 V1, ? 0,10 20,20 0
 V2, ?

(1)S1, ? 1 0 USE V1
(2)S2, ?

(2)R1, ? 1 2 10
(3)R2, ? 2 0 (10 20 W1)
(4)R3, ? Q
 * ? ADD W

(4)W1, ? SW NV1 GT 15
 * ? PL NV1 NV2 BC1 BC2 XTTY
 * ? TI .1 20
 * ? LIST
UP AND DOWN SWITCHING
     V1                  0.0,
     V       1.000E+01    2.000E+01,
     V       2.000E+01    0.0,


 (1)S1     1     0     USE V1     1.000E-03

 (2)R1     1     2     1.000E+01
 (3)R2     2     0    ( 1.000E+01   2.000E+01   W1)




 (4)W1    SW   NV1   GT    1.5000000E+01

PL NV1 NV2 BC1 BC2 XTTY
TIME,
   1.000E-01    2.000E+01,

APPROX.      200 TIME POINTS REQUESTED


 * ? EX
```

1 UP AND DOWN SWITCHING

TIME (SECONDS) X AXIS: LIN. SCALE

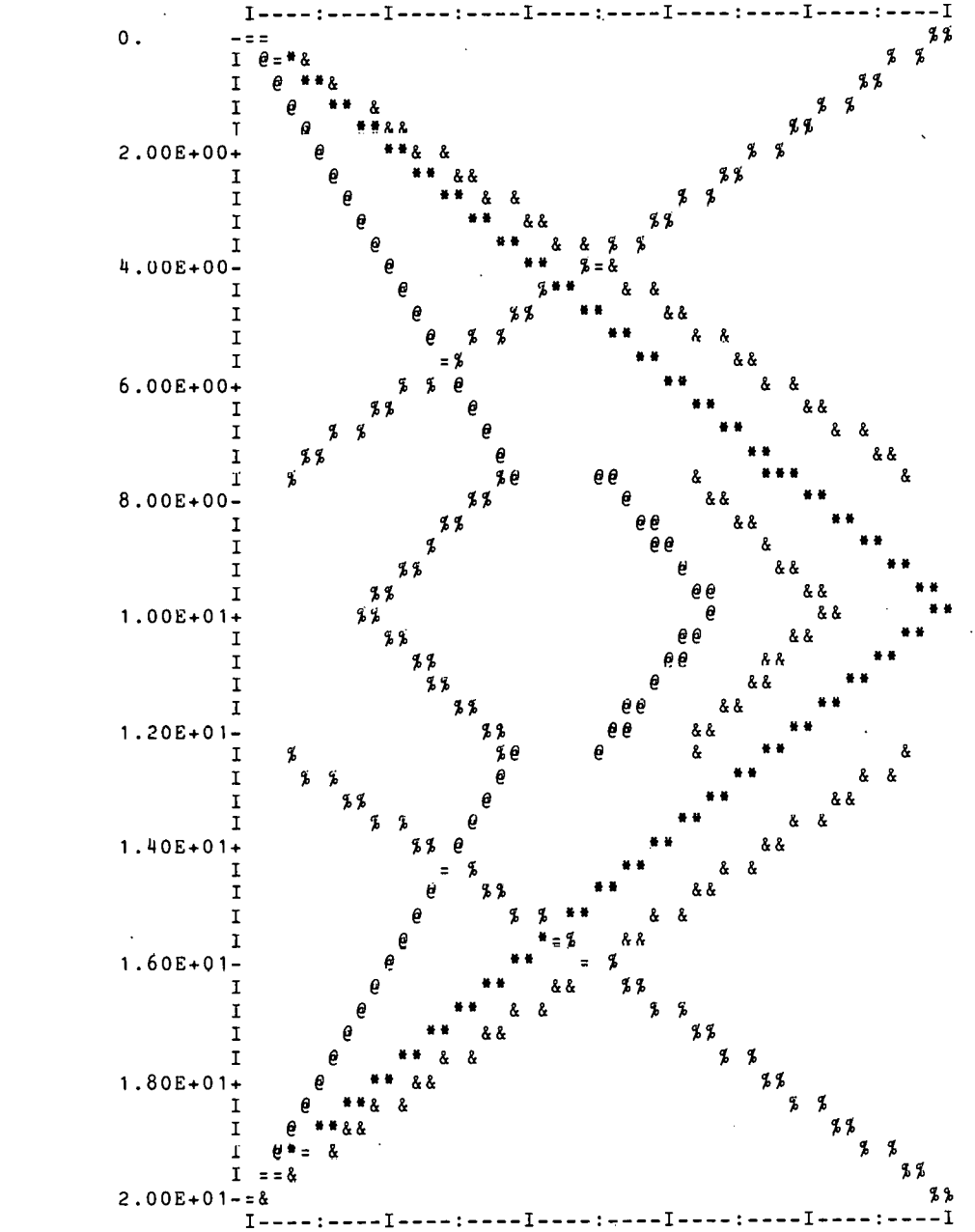
NODE VOLTAGE 1 (*) LIN. SCALE MIN= 0. AT X= 0.
MAX= 1.9999E+01 AT X= 1.0000E+01

NODE VOLTAGE 2 (@) LIN. SCALE MIN= 0. AT X= 0.
MAX= 1.3333E+01 AT X= 1.0000E+01

BRANCH CRRNT 1 (%) LIN. SCALE MIN=-7.4996E-01 AT X= 7.5000E+00
MAX= 0. AT X= 0.

BRANCH CRRNT 2 (&) LIN. SCALE MIN= 0. AT X= 0.
MAX= 7.4996E-01 AT X= 1.2500E+01

(*)	0.	4.00E+00	8.00E+00	1.20E+01	1.60E+01	2.00E+01
(@)	0.	4.00E+00	8.00E+00	1.20E+01	1.60E+01	2.00E+01
(%)	-8.00E-01	-6.40E-01	-4.80E-01	-3.20E-01	-1.60E-01	0.
(&)	0.	1.60E-01	3.20E-01	4.80E-01	6.40E-01	8.00E-01



AITRAC-100 READY 77/07/18 14.39.49

Nonlinear Resistor

Following are the terminal printouts of the SPEQ coding, the circuit topology, and the printouts and plots of the nonlinear resistor problem discussed in Section 5.7.

```
        EQUIVALENCE (NRX,V(101)),(NIX,V(102)),
     1  (RX(1),V(111)),(IX(1),V(136))
C
C FUNCTION DEFINITIONS; RX1  IS VALUE OF NON LIN RES. AT 1.0 V.
       RCUR(XX) = XX**4/RX(1)
       GCOND(XX)= XX**3/(RX(1)/4.)
C
      GO TO (9000,9003,9002,9001), KK4
9000  IF(ISETUP .EQ. 1) GOTO 9004
C  CALLS TO SUBROUTINE * BUILD * INSERTED HERE
       CALL BUILD(2,3)
       CALL BUILD(3,2)
      RETURN
9004  CONTINUE
C       PART 1 - MATRIX AND TIME FUNCTIONS HERE
       XBV=BRAVOL(2,3,V,EI)
C--   ADD SMALL CONDUCTANCE IN CASE GCOND = 0.
       GEE = GCOND(XBV) + 1E-12
       DELT = XBV * GEE -RCUR(XBV)
       EEE = XBV -RCUR(XBV)/GEE
C
       CALL SHTADD(2,3,GEE,DELT)
C
      RETURN
 9003 CONTINUE
C     PART 2
C--   CONVERGENCE TESTS
C MODEL CURRENT = XCUR
       XBV = BRAVOL(2,3,V,EI)
       XCUR = (XBV-EEE)*GEE
C DESIRED CURRENT = DCUR
       DCUR = RCUR(XBV)
C  RX2 IS % TOL. ON CURRENTS FOR CONVERGENCE; NOMINALLY = 0.1%
C  ABC IS NON-CONVERGENCE FLAG; +1 FOR NON CONVERGED SOLUTION
       IF(RX(2).EQ.0.)RX(2)=.001
       IF(ABS((XCUR-DCUR)/XCUR) .GT. RX(2)) ABC= 1.
C
C--  RELINEARIZE MODEL
       GEE = GCOND(XBV)
C
      RETURN
 9002 CONTINUE
C     PART 3
      RETURN
 9001 CONTINUE
C     PART 4 - AUXILIARY EQUATIONS AFTER THIS CARD.
C---  ADD MODEL CURRENT TO BRANCH 5 FOR PRINTOUT
C---  CAREFUL, BRANCH 1 IS S1, THEREFORE R4 IS BRANCH 5 (NOT 4)
       BCUR(5) = BCUR(5) + DCUR
      RETURN
      END
```

Nonlinear Resistor SPEQ Coding

```
 * ? LIST
 AITRAC SPECIAL EQUATIONS TEST
     V1                  0.0,
     V      1.000E+00    1.000E+05,


 (1)S1     1     0     USE V1     1.000E-05

 (2)R1     1     2     1.000E+00
 (3)R2     2     0     1.000E-05
 (4)R3     3     0     1.000E-03
 (5)R4     2     3     1.000E+08
```

```
 PR NV2 NV3 BC4 BC5
SET SPECIAL ON
SET RX    1.000E+03  1.000E-03
TIME,
   1.000E-01    1.000E+00,

APPROX.       10 TIME POINTS REQUESTED
```

```
 * ? EXECUTE
USING SPECIAL EQUATIONS
```

TIME	DELTA	NV 2	NV 3	BC 4	BC 5
0.	0.	0.	0.	0.	0.
*START TIME= 0.		*END TIME= 1.0000E+00		*MAX DELTA= 1.0000E-01	
1.0000E-01	1.0000E-01	9.9998E-02	1.0099E-10	1.0099E-07	1.0099E-07
2.0000E-01	1.0000E-01	2.0000E-01	1.6019E-09	1.6019E-06	1.6019E-06
3.0000E-01	1.0000E-01	2.9999E-01	8.1024E-09	8.1024E-06	8.1024E-06
4.0000E-01	1.0000E-01	3.9999E-01	2.5602E-08	2.5602E-05	2.5602E-05
5.0000E-01	1.0000E-01	4.9999E-01	6.2500E-08	6.2500E-05	6.2500E-05
6.0000E-01	1.0000E-01	5.9999E-01	1.2960E-07	1.2960E-04	1.2960E-04
7.0000E-01	1.0000E-01	6.9999E-01	2.4009E-07	2.4009E-04	2.4009E-04
8.0000E-01	1.0000E-01	7.9998E-01	4.0957E-07	4.0957E-04	4.0957E-04
9.0000E-01	1.0000E-01	8.9998E-01	6.5605E-07	6.5605E-04	6.5605E-04
1.0000E+00	1.0000E-01	9.9998E-01	9.9993E-07	9.9993E-04	9.9993E-04

```
AITRAC-100 READY  77/07/18  15.18.29

 * ?
```

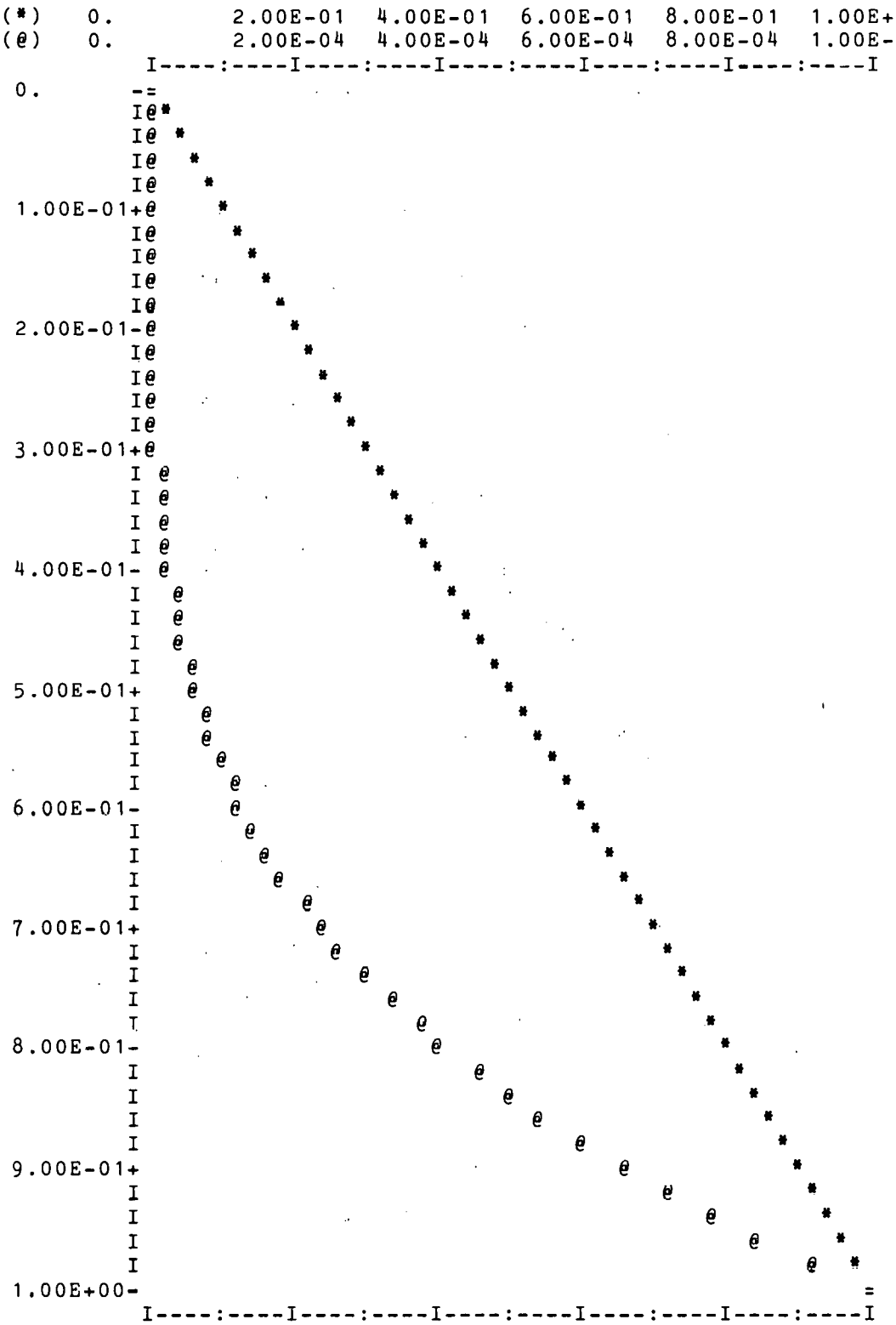
```
     PR
  * ? PL NV2 BC5 XTTY
  * ? TI 2E-2 1.
  * ? EX
 USING SPECIAL EQUATIONS


 *START TIME= 0.          *END TIME=  1.0000E+00  *MAX DELTA= 2.0000E-02

 T= 1.8000E-01
 T= 3.8000E-01
 T= 5.8000E-01
 T= 7.8000E-01
 T= 9.8000E-01

 ### NEW PLOT & SIEVE ROUTINE ENTERED ###


     51 POINTS TO BE PLOTTED...



  ENTER PLOT TITLE: ? NON LINEAR RESISTOR USING SPECIAL EQUATIONS

 ----- SET PAPER 3 HOLES ABOVE FOLD, THEN HIT CARRIAGE RETURN     -----
?
```

NON LINEAR RESISTOR USING SP. EQUATIONS

TIME (SECONDS) X AXIS: LIN. SCALE

NODE VOLTAGE 2 (*) LIN. SCALE MIN= 0. AT X= 0.
MAX= 9.9998E-01 AT X= 1.0000E+00

BRANCH CRRNT 5 (@) LIN. SCALE MIN= 0. AT X= 0.
MAX= 9.9993E-04 AT X= 1.0000E+00



NON LINEAR RESISTOR USING SP. EQUATIONS

TABLE OF PLOTTED VALUES

NO.	X-AXIS	* VALUE	@ VALUE
1	0.	0.	0.
2	2.00000E-02	1.99996E-02	3.59983E-10
3	4.00000E-02	3.99992E-02	2.95979E-09
4	6.00000E-02	5.99988E-02	1.35590E-08
5	8.00000E-02	7.99984E-02	4.17567E-08
6	1.00000E-01	9.99980E-02	1.00992E-07
7	1.20000E-01	1.19998E-01	2.08543E-07
8	1.40000E-01	1.39997E-01	3.85529E-07
9	1.60000E-01	1.59997E-01	6.56908E-07
10	1.80000E-01	1.79996E-01	1.05148E-06
11	2.00000E-01	1.99996E-01	1.60187E-06
12	2.20000E-01	2.19996E-01	2.34457E-06
13	2.40000E-01	2.39995E-01	3.31989E-06
14	2.60000E-01	2.59995E-01	4.57199E-06
15	2.80000E-01	2.79994E-01	6.14887E-06
16	3.00000E-01	2.99994E-01	8.10235E-06
17	3.20000E-01	3.19994E-01	1.04881E-05
18	3.40000E-01	3.39993E-01	1.33657E-05
19	3.60000E-01	3.59993E-01	1.67984E-05
20	3.80000E-01	3.79992E-01	2.08535E-05
21	4.00000E-01	3.99992E-01	2.56019E-05
22	4.20000E-01	4.19992E-01	3.11187E-05
23	4.40000E-01	4.39991E-01	3.74823E-05
24	4.60000E-01	4.59991E-01	4.47756E-05
25	4.80000E-01	4.79990E-01	5.30847E-05
26	5.00000E-01	4.99990E-01	6.25000E-05
27	5.20000E-01	5.19990E-01	7.31155E-05
28	5.40000E-01	5.39989E-01	8.50291E-05
29	5.60000E-01	5.59989E-01	9.83426E-05
30	5.80000E-01	5.79988E-01	1.13162E-04
31	6.00000E-01	5.99988E-01	1.29596E-04
32	6.20000E-01	6.19988E-01	1.47758E-04
33	6.40000E-01	6.39987E-01	1.67765E-04
34	6.60000E-01	6.59987E-01	1.89739E-04
35	6.80000E-01	6.79986E-01	2.13803E-04
36	7.00000E-01	6.99986E-01	2.40007E-04
37	7.20000E-01	7.19986E-01	2.68724E-04
38	7.40000E-01	7.39985E-01	2.99849E-04
39	7.60000E-01	7.59985E-01	3.33602E-04
40	7.80000E-01	7.79984E-01	3.70128E-04
41	8.00000E-01	7.99984E-01	4.09574E-04
42	8.20000E-01	8.19984E-01	4.52093E-04
43	8.40000E-01	8.39983E-01	4.97839E-04
44	8.60000E-01	8.59983E-01	5.46972E-04
45	8.80000E-01	8.79982E-01	5.99655E-04
46	9.00000E-01	8.99982E-01	6.56055E-04
47	9.20000E-01	9.19982E-01	7.16343E-04
48	9.40000E-01	9.39981E-01	7.80693E-04
49	9.60000E-01	9.59981E-01	8.49285E-04
50	9.80000E-01	9.79980E-01	9.22301E-04
51	1.00000E+00	9.99980E-01	9.99926E-04

APPENDIX F

SEMICONDUCTOR LIBRARY

****

S E M I C O N D U C T O R

L I B R A R Y

****

DATA OBTAINED FROM :

BERNE, CIRCUS, DICURT,
NET-1, SCEPTRE, SYSCAP,

and other sources.

(These Data Will Be Updated Periodically)

June 1972

Updated August 1974

Berne Electronics, Inc.
White Plains, N.Y., 10605

S E M I C O N D U C T O R

L I B R A R Y

****

This library is offered as a convenience to AITRAC users.

The required diode and transistor parameters have been abstracted from various sources, and are believed to be the best data available to date.

Berne Electronics does not guarantee the accuracy of these data, but stipulates that it has done considerable research into the availability of semiconductor data, and that it has checked the parameters, to the best of its ability.

If serious errors are found, the user is asked to report them to Berne Electronics, so that the data can be corrected as soon as possible. Also, if users have knowledge or access to other semiconductor data which can be made available to other AITRAC users, Berne will make all possible efforts to include such data in this library, thereby increasing its scope and usefulness.

N O T E :

All data can be taken as 'average', i.e: as applicable to the average parameters of any semiconductor.

For clariry, device names are listed with blanks, such as: 1N 459, 2N 174, or 2N 2907A. However, the user should disregard all blanks when issuing the 'SEarch' command in AITRAC.

Thus, the data for the above semiconductors are retrieved by the commands:

```
SEarch 1N459
SE     2N174
```

or:

```
SE 2N2907A
```

without any blanks in the device name.

D I O D E S

TYPE NO.	MAT.	USAGE	WORK.VLTS RATED VLTS	IF/VD MA/VLTS	IREV/VREV MA/VLTS	REMARKS
1N 63	GE	GP-R	VR 125	.4/1	50/125	N
1N 93	GE	GP-R	VR 300	150/.45	1.3M/300	N
1N 100	GE	GP-R	VR 100	20/1	50/50	N
1N 140	GE	GP-S	VR 70	40/1	300/50	N
1N 191	GE	GP-S	VR 90	5/1	125/50	N
1N 270	GE	GP-S	VR 80	200/1	100/50	N
1N 273	GE	GP-S	WV 30	100/1	20/30	C
1N 273A	GE	GP-S	WV 30	100/1	20/30	C
1N 273B	GE	GP-S	WV 30	100/1	20/30	C
1N 273C	GE	GP-S	WV 30	100/1	20/30	C
1N 273D	GE	GP-S	WV 30	100/1	20/30	C
1N 273E	GE	GP-S	WV 30	100/1	20/30	C
1N 273F	GE	GP-S	WV 30	100/1	20/30	C
1N 276L	GE	GP-S	VR 50	40/1	.1/50	S CURRENT < 200 AMPS.
1N 276	GE	GP-S	VR 50	40/1	.1/50	S CURRENT > 200 AMPS.
1N 279	GE	GP-S	VR 30	100/1	200/20	N
1N 456	SI	HCLL	WV 25	40/1	25M/25	B LOW LEAKAGE
1N 456FC	SI	HCLL	WV 25	40/1	25M/25	Y LOW LEAKAGE, FAIRCH.
1N 456TI	SI	HCLL	WV 25	40/1	25M/25	Y LOW LEAKAGE, TEX.INST.
1N 457	SI	GP-S	VR 60	20/1	.025/50	S
1N 459	SI	GP-S	VR 175	3/1	.025/50	S
1N 461	SI	GP-S	WV 25	15/1	.5M/25	Y
1N 482A	SI	GP-S	WV 36	100/1.1	250M/36	Y
1N 645	SI	GP-S	VR 225	400/1	.2/225	N SEE 1N4003
1N 645B	SI	GP-S	VR 225	400/1	.05/225	C SEE 1N4003
1N 646	SI	GP-S	VR 300	400/1	.2/300	S CURRENT > 280 UA
1N 646L	SI	GP-S	VR 300	400/1	.2/300	S CURRENT < 280 UA
1N 647	SI	GP-S	VR 400	400/1	.2/400	S SEE 1N4005
1N 648	SI	GP-S	VR 500	400/1	.2/500	S SEE 1N4005
1N 649	SI	GP-S	VR 600	400/1	.2/600	N SEE 1N4005
1N 658	SI	GP-S	VR 100	100/1	.05/100	S
1N 659	SI	GP-S	VR 50	6/1	5/50	C
1N 660	SI	GP-S	VR 100	6/1	5/100	S
1N 661	SI	GP-S	WV 200	6/1	10/200	N
1N 662	SI	HCMS	WV 80	10/1	200/50	Y HI SPEED

TYPE NO.	MAT.	USAGE	WORK. VLTS RATED VLTS	IF/VD MA/VLTS	IREV/VREV MA/VLTS		REMARKS
1N 695	GE	GP-S	VR 20	100/1	2/20	S	
1N 903	SI	GP-S	VR 40	10/1	.1/40	S	
1N 908	SI	GP-S	VR 40	10/1	.1/40	S	
1N 914	SI	GP-P	WV 20	10/1	5/20	N	
1N 914B	SI	GP-P	VR 75	10/1	5/75	S	
1N 917	SI	GP-S	VR 30	10/1	.05U/30		
1N 970B	SI	Z	400MW 24.	5%		N	ZENER
1N 971	SI	Z	400MW 27.	20%		C	ZENER
1N 995	GE	S	VR 15	10/.5	10/6	N	
1N 1313A	SI	Z	150MW 9.1	5%			ZENER SEE 1N5238B
1N 1315	SI	Z	150MW 12.	5%			ZENER SEE 1N5243A
1N 2199	SI	R	VR 400	6A/1.25	10M/400	N	
1N 3064	SI	GP-S	VR 75	10/1	.1U/75	B	
1N 3070	SI	HCHS	WV 175	100/1	.1U/175	Y	HI SPEED
1N 3071	SI	GP-S	VR 200	100/1	.1/200	S	
1N 3595	SI	HCLL	WV 125	200/1	1N/125	Y	HI SPEED LO LEAKAGE
1N 3600	SI	HCHS	WV 50	200/1	.1U/50	Y	ULT.FAST
1N 3605	SI	GP-S	VR 40	20/.55	.05/40	N	
1N 3611	SI	GP-S	VR 200	750/1	10/40	S	
1N 3669	SI	GP-S	VR 70	400/1.1	.25/70	S	
1N 4001	SI	R	WV 50	1A/1.6	30/50	S	
1N 4003	SI	R	VR 200	1A/1.6	30/200		REPL. FOR 1N645/A/B
1N 4005	SI	R	WV 600	1A/1.6	30/50		REPL. FOR 1N646-9
1N 4006	SI	R	WV 800	1A/1.6	30/50	S	
1N 4148	SI	HCHS	WV 75	10/1	250N/20	Y	ULT.FAST, MINIATURE
1N 4150	SI	HCHS	WV 50	200/1	100N/50	Y	ULT.FAST, MINIATURE
1N 4151	SI	HCHS	WV 50	50/1	50N/50	Y	ULT.FAST, MINIATURE
1N 4152	SI	HCHS	WV 30	20/.88	50N/30	Y	ULT.FAST
1N 4153	SI	HCHS	WV 50	20/.88	50N/50	Y	ULT.FAST
1N 4305	SI	GP-S	WV 75	1/.6	.1U/75	Y	
1N 4306	SI	MPHS	WV 50	50/1	50N/50	Y	MATCHED PAIR, HI SPEED
1N 4370	SI	Z	400MW 2.4	10%			ZENER
1N 4371	SI	Z	400MW 2.7	10%			ZENER
1N 4372	SI	Z	400MW 3.0	10%			ZENER
1N 4444	SI	GP-S	WV 50	100/1	50N/50	Y	
1N 4446	SI	HCHS	WV 70	20/1	25N/20	Y	ULT.FAST, MINIATURE
1N 4447	SI	HCHS	WV 70	20/1	25N/20	Y	ULT.FAST, MIN., LO CAP.
1N 4448	SI	HCHS	WV 70	100/1	25N/20	Y	ULT.FAST, MINIATURE

TYPE NO.	MAT.	USAGE	WORK.VLTS RATED	VLTS	IF/VD MA/VLTS	IREV/VRVV MA/VLTS	REMARKS
1N 4450	SI	HCHS	WV	30	200/1	50N/30	Y ULT.FAST, MINIATURE
1N 4454	SI	HCHS	WV	40	10/1	.10/50	Y ULT.FAST, MIN., LO CAP.
1N 4531	SI	HCHS	WV	75	10/1	25N/20	Y ULT.FAST, MINIATURE
1N 4532	SI	HCHS	WV	75	10/1	.10/50	Y ULT.FAST, MIN., LO CAP.
1N 4533	SI	HCHS	WV	40	20/.88	50N/40	Y
1N 4534	SI	HCHS	WV	50	20/.88	50N/50	Y
1N 4606FC	SI	HCHS	WV	70	200/1	250N/70	Y ULT.FAST, MIN., LO CAP.
1N 4606TI	SI	HCHS	WV	75	200/1	250N/70	Y ULT.FAST, MIN., LO CAP.
1N 4607FC	SI	HCHS	WV	70	350/1	250N/70	Y ULT.FAST, MINIATURE
1N 4607TI	SI	HCHS	WV	70	350/1	250N/70	Y ULT.FAST, MINIATURE
1N 4608	SI	HCHS	WV	70	350/1	250N/70	Y ULT.FAST, MINIATURE
1N 4610	SI	HCHS	WV	55	200/1	100N/55	Y ULT.FAST, MIN., LO CAP.
1N 4727	SI	HCHS	WV	20	10/.85	100N/20	Y ULT.FAST, MINIATURE
1N 4950	SI	HCHS	WV	25	300/1	100N/25	Y ULT.FAST, MINIATURE
1N 5282	SI	HCHS	WV	55	300/1	.10/55	Y ULT.FAST, MIN., LO CAP.
1N 5317	SI	HCHS	WV	55	300/1	.10/55	Y ULT.FAST, MIN., LO CAP.
1N 5318	SI	HCHS	WV	50	200/1	.10/50	Y ULT.FAST, MIN., LO CAP.
1N 5430	SI	HCHS	WV	50	200/1	.10/50	Y ULT.FAST, RAD.RESIS.
1N 5431	SI	HCHS	WV	55	500/1.15	.10/55	Y ULT.FAST, RAD.RESIS.
1N 5432	SI	HCHS	WV	10	50/1.1	50N/10	Y PICOSEC. SW., RAD.RESIS.
1N10000		GP	TO BE USED AS A GENERAL PURPOSE DEVICE				

A = AMPS U = MICROAMPS M = MILLIAMPS NA = NANOAMPS

GP = GEN.PURPOSE F = FAST K = RECTIFIER
S = SIGNAL Z = ZENER HCHS = HI CONDUCTANCE, HI SPEED

B = BERNE LIB. C = CIRCUS LIB. I = IMPACT LIB.
N = NET-1 LIB. S = SCEPTRE LIB. Y = SYSCAP LIB.

D = DICURT MANUAL, TM342-1-U, AUTONETICS, AUG., 1968
K = SCEPTRE BOOK (BOWERS & SEDORE, PRENTICE HALL, 1971)

MISCELLANEOUS DIODES

TYPE NO.	MAT.	USAGE	WORK. VLTS RATED VLTS	IF/VD MA/VLTS	IREV/VREV MA/VLTS	REMARKS
A 670						
FA 2008						
FA 2010						
FD 100	SI	SW	VR 50	10/1.	.1UA/50	S (FAIRCHILD)
FD 200	SI	SW	VR 150	100/1	.1UA/150	N
FD 300	SI	SW	WV 125	200/1	.001/125	N
FD 600	SI	SW	WV 50	200/1	.1/50	N
FD 700	SI	SW	WV 20	50/1.1	.05/20	N
FDA 630						
FDM1000						
FD6666	SI	SW	WV 50	300/1.1	.01/55	N
FSD220						
HPA2001						
ID3050T	GE	HS-SW	VR 6	200/1.	.1E/3.	N (INT. DIODE)
SD 500	SI	R	VR 400	500/1.2	10UA/100	N (INT.RECT.ENGLAND)
SG 5250	SI	GP-S	VR 50	100/.9	25NA/25	S (TRANSITRON)
SG 5270	SI	GP-S	VR 100	100/.9	.1UA/25	S (TRANSITRON)
PS 760						C
PS4750						
PS4902						
TIXD27						Y (TEXAS INSTR.)
TIXD28						Y (TEXAS INSTR.)
UT 262	SI	R	VR 200	900/1.	2UA/200	S (UNITRODE)
907821						

TRANSISTORS

TYPE NO.	MATL TYPE	USAGE	VCE VCB	RATED WATTS	HFE/MA	CUTOFF FR.MHZ	REMARKS
2N 174	GE-P	PW/A	80	100.	30/5A	.1	N
2N 315	GE-P	SW	20	.1	20/100	5.0	N
2N 329A	SI-P	GP	30	.30	60/3	.5	S
2N 336	SI-N	GP	45	.30	125/1	13.	S
2N 356	GE-N	SW	20	.1	30/100	3.0	N
2N 375	GE-P	PW/A	80	50.	50/1A	.007	N
2N 384	GE-P	PP	40	.12	90/1.5	100.	N
2N 385	GE-N	SW	25	.15	50/5	4.	N
2N 393	GE-P	SW	6	.025	20/50	25.	N
2N 396	GE-P	SW	105	.050	20/5	20.	N
2N 404	GE-P	SW	25	.150	30/12	4.	N
2N 414	GE-P	GP	30	.150	40/10	4.	N
2N 457	GE-P	PW/A	60	50.	50/5A	.004	N
2N 585	GE-P	SW	25	.12	20/20	4.	S
2N 597	GE-P	SW	45	.250	100/100	3.	N
2N 598	GE-P	SW	35	.250	100/100	5.6	N
2N 645	GE-P	SW	30	.12	20/10	20.	C
2N 695	GE-P	SW/HP	15	.075	25/10	40.	Y
2N 697	SI-N	GP	60	2.	75/150	50.	N
2N 705	GE-P	SW/HP	15	.300	25/10	300.	N
2N 706	SI-N	SW/HP	25	.30	20/10	200.	S
2N 706A	SI-N	SW/HP	25	1.	75/150	300.	N
2N 711A	GE-P	SW/HP	15	.150	80/10	150.	N
2N 718	SI-N	HP	60	.40	80/150	60.	S
2N 718A	SI-N	GP	75	.50	80/150	60.	C
2N 720A	SI-N	SW	120	.50	80/150	60.	S
2N 722	SI-P	HP	50	.40	60/150	70.	S
2N 743	SI-N	SW/HP	20	.30	40/10	300.	S
2N 760	SI-N	HP	45	.50	75/10	50.	S
2N 797	GE-N	SW/HP	20	.150	40/10	600.	N
2N 834	SI-N	SW/HP	40	.30	25/10	400.	S
2N 835	SI-N	SW/HP	25	.30	20/10	300.	S

TYPE NO.	MATL TYPE	USAGE	VCE VCE	RATED WATTS	HFE/MA	CUTOFF FR.-MHZ	REMARKS
2N 914	SI-N	SW/HP	40	.36	50/10	300.	S
2N 915	SI-N	HP	70	.36	100/10	300.	S
2N 916	SI-N	HP	45	.36	100/10	300.	S
2N 918	SI-N	HP	30	.20	20/3	750.	S
2N 955A	GE-N	SW/HP	12	.15	30/30	1000.	S
2N 964	GE-P	SW/HP	15	.150	40/10	300.	N
2N 976	GE-P	SW/HP	15	.100	30/20	250.	N
2N 995	SI-P	SW/HP	20	1.2	60/20	100.	N
2N 1016B	SI-N	SW/PW	100	150.	10/5A	.030	S
2N 1016E	SI-P	SW/PW	250	150.	10/5A	.030	S
2N 1037	SI-P	GP	50	.250	25/10	.150	N
2N 1039	GE-P	PW/A	60	20.	40/1A	.008	N
2N 1099	GE-P	PW/A	80	50.	50/5A	.010	S
2N 1131	SI-P	GP	50	2.0	26/150	70.	N
2N 1132	SI-P	GP	50	2.0	45/150	90.	N
2N 1154	GE-P	SW/PW	45	7.50	80/400	.50	S
2N 1225	GE-P	GP	40	.12	20/5	100.	S
2N 1228	SI-P	SW	15	.400	28/10	1.2	N
2N 1269	GE-N	SW/HP	20	.075	100/10	40.	C
2N 1301	GE-P	SW/PP	13	.150	30/10	35.	N
2N 1304	GE-N	SW	25	.150	100/10	5.	N
2N 1306	GE-N	SW	25	.150	120/10	10.	N
2N 1307	GE-P	SW	30	.150	120/10	10.	N
2N 1308	GE-N	SW	25	.150	80/10	15.	N
2N 1342	SI-N	HP/PV	150	.80	80/150	70.	S
2N 1483	SI-N	SW/PW	60	25.	30/750	1.25	N
2N 1486	SI-N	SW/PW	100	25.	75/750	1.25	S
2N 1490	SI-N	SW/PW	100	75.	50/1.5A	1.	N
2N 1499	GE-P	SW/HP	20	.025	20/10	100.	C
2N 1499A	GE-P	SW/HP	20	.200	30/10	5.	N
2N 1506	SI-N	GP	60	.80	50/100	140.	C
2N 1506A	SI-N	HP	80	.80	70/100	140.	S
2N 1613	SI-N	GP	75	3.0	80/150	80.	N
2N 1709	SI-N	HP/PW	75	15.	35/350	175.	S
2N 1711	SI-N	GP	75	3.0	130/150	100.	N
2N 1717	SI-N	A/PW	150	20.	70/200	16.	D
2N 1722	SI-N	A/PW	120	50.	50/2A	10.	C
2N 1724	SI-N	A/PW	120	50.	50/2A	10.	N

TYPE NO.	MATL TYPE	USAGE	VCE VCB	RATED WATTS	HFE/MA	CUTOFF FR.MHZ	REMARKS
2N 1893	SI-N	HP	120	.80	80/150	50.	S
2N 1900	SI-N	HP/PW	140	125.	10/10A	50.	S
2N 2048	GE-P	SW/HP	20	.15	50/10	150.	N
2N 2060	SI-N	GP	100	.50	80/10	60.	C
2N 2087	SI-N	SW/PP	120	.600	80/150	150.	N
2N 2102	SI-N	SW/HP	120	5.	40/10	60.	N
2N 2167	SI-P	SW/CH	30	.1 50	50/1	80.	S
2N 2188	GE-P	GP	40	.125	80/1.5	60.	N SEE 2N3323
2N 2192	SI-N	SW/PP	60	.80	200/150	50.	S
2N 2219	SI-N	SW/HP	60	.80	200/150	250.	C SEE 2N2218
2N 2222	SI-N	SW/HP	60	1.80	200/150	250.	N
2N 2222A	SI-N	SW/HP	75	1.80	200/150	300.	N
2N 2223	SI-N	GP	100	.50	100/10	50.	S
2N 2243A	SI-N	SW/HP	120	.80	80/150	50.	S
2N 2256	GE-P	SW/HP	7	.15	50/10	250.	K
2N 2368	SI-N	SW/HP	40	.36	40/10	400.	S
2N 2369	SI-N	SW	40	1.2	80/10	650.	K SEE 2N3227
2N 2369A	SI-N	SW/HP	40	.36	80/10	500.	S
2N 2411	SI-P	SW/HP	25	.30	40/10	140.	S SEE 2N3250
2N 2432	SI-N	SW/CH	30	.30	50/1	20.	D
2N 2453	SI-N	A/M	60	.50	300/1	60.	K
2N 2481	SI-N	SW/HP	40	1.2	75/10	300.	S SEE 2N2480
2N 2484	SI-N	LL/LN	60	1.2	430/10	75.	K
2N 2538	SI-P	SW/HP	60	.80	200/150	250.	Y SEE 2N2537
2N 2656	SI-N	GP	25	.36	75/.1	250.	S
2N 2695	SI-P	SW	25	.36	75/80	100.	S
2N 2708	SI-P	HP	35	.20	75/2	700.	S
2N 2784	SI-N	SW/HP	15	.200	40/30	1000.	N
2N 2800	SI-P	SW/HP	50	.80	60/150	120.	Y
2N 2801	SI-P	SW	50	.80	150/150	120.	S SEE 2N2800
2N 2802	SI-P	GP	25	.25	70/.1	60.	Y
2N 2804	SI-P	A/PW	25	.25	70/.1	60.	Y
2N 2808	SI-N	HP	30	.30	60/2	1000.	S

TYPE NO.	MATL TYPE	USAGE	VCY VCB	RATED WATTS	HFE/MA	CUTOFF FR.MHZ	REMARKS
2N 2845	SI-N	SW/HF	60	.36	75/150	250.	S
2N 2857	SI-N	HF/GP	30	.20	100/2	1000.	C
2N 2867	SI-N	HF/PW	100	25.	50/250	140.	S
2N 2894	SI-P	SW/HF	12	.36	100/30	400.	N
2N 2905	SI-P	SW	60	3.0	200/150	200.	C
2N 2906	SI-P	SW/HF	60	1.8	80/150	200.	Y
2N 2907	SI-P	SW/HF	60	1.8	200/150	200.	S SEE 2N2904
2N 2907A	SI-P	SW/HF	60	1.80	200/150	200.	S
2N 2918	SI-N	A/M	45	.36	300/10M	60.	D
2N 3013	SI-N	SW/HF	40	.36	75/30	350.	C
2N 3014	SI-N	SW/HF	40	.36	75/30	350.	I
2N 3017	SI-N	HF/PW	100	3.33	100/1A	200.	S
2N 3019	SI-N	HV	140	5.0	200/150	100.	R
2N 3026	SI-P	HF/PW	60	25.	100/1A	60.	K
2N 3039	SI-P	GP	50	.36	50/150	50.	Y
2N 3051	SI-P	GP	25	.25	70/.1	60.	Y
2N 3055	SI-N	PW/A	100	115.	40/4A	.020	N
2N 3108	SI-N	HV	100	5.0	70/150	65.	N
2N 3117	SI-N	LV/LN	60	1.2	400/1.	60.	N
2N 3119	SI-N	SW/HF	100	1.	100/100	250.	N
2N 3227	SI-N	SW/HF	40	.36	200/10	500.	B
2N 3244	SI-P	SW	40	1.	80/500	175.	S
2N 3251	SI-P	SW/HF	50	1.2	200/10	300.	N
2N 3252	SI-N	SW	60	1.	60/500	200.	S
2N 3283	GE-P	HF	25	.10	20/3	250.	S
2N 3287	SI-N	HF	40	.20	75/2	350.	S
2N 3309	SI-N	HF/PW	50	3.5	50/30	300.	Y
2N 3375	SI-N	HF/PW	65	11.60	60/250	400.	S SEE 2N3632
2N 3486	SI-P	SW/HF	60	2.	200/150	200.	C SEE 2N2904
2N 3498	SI-N	A/HF	100	1.	80/150	150.	S
2N 3499	SI-N	HF	100	1.	200/150	150.	S

TYPE NO.	MATL TYPE	USAGE	VCB VCE	RATED WATTS	HFE/MA	CUTOFF FE.MHZ	REMARKS
2N 3501	SI-N	HF	150	1.	200/150	150.	N
2N 3502	SI-P	SW	45	3.0	200/10	250.	N
2N 3503	SI-P	SW	60	3.0	200/10	250.	N
2N 3509	SI-N	SW/HF	40	.40	200/10	500.	C SEE 2N3506
2N 3553	SI-N	A/PW	65	7.	80/250	400.	C SEE 2N3375
2N 3632	SI-N	A/PW	65	23.	80/250	250.	C SEE 2N3375
2N 3633	SI-P	HF	30	.20	50/6	700.	S
2N 3635	SI-P	HF	140	1.	200/50	200.	N
2N 3723	SI-N	SW/HF	100	.80	80/100	300.	C SEE 2N3722
2N 3737	SI-N	SW/HF	75	.50	50/1A	250.	K
2N 3828	SI-N	HF	40	.30	100/12	300.	S
2N 3839	SI-N	A/PF	30	.20	30/3	1000.	C SEE 2N2857
2N 3866	SI-N	A/PW	55	5.	100/50	500.	C
2N 3904	SI-N	SW/HF	60	.31	200/10	300.	N
2N 3906	SI-P	SW/HF	40	.31	200/10	250.	N
2N 3913	SI-P	SW/HF	60	.40	90/1	4.	K
2N 3915	SI-P	SW/HF	60	.40	90/1	10.	K
2N 3959	SI-N	SW/HF	20	.40	120/10	1300.	N 2N4260 COMPL.
2N 3961	SI-N	HF/PW	65	10.	5/1A	400.	C See 2N3632
2N 4207	SI-P	SW/HF	6	.30	80/10	650.	C
2N 4208	SI-P	SW-HF	12	.30	70/10	700.	C
2N 4209	SI-P	SW/HF	15	.30	75/10	850.	C
2N 4251	SI-N	SW/HF	15	.25	100/10	1300.	C
2N 4260	SI-P	SW/HF	15	.20	25/10	1660.	N 2N3959 COMPL.
2N 4420	SI-N	SW	40	.25	75/30	250.	Y
2N 4923	SI-N	PW/A	80	30.	50/500	3.	N
2N10000	GP	TO BE USED AS A GENERAL PURPOSE NPN DEVICE					
2N10000	GP	TO BE USED AS A GENERAL PURPOSE PNP DEVICE					

MISCELLANEOUS TRANSISTORS

TYPE NO.	MATL TYPE	USAGE	VCE VCB	RATED WATTS	HFE/MA	CUTOFF FR.MHZ	REMARKS
A 210							N
MJE 521	SI-P	SW/PW	40	40.	40/500	2.5	N (MOTOROLA)
MM 2258	SI-N	SW	120	1.	35/100	150.	N (MOTOROLA)
MPS 404	SI-P	SW	25	.31	30/10	4.0	N (MOTOROLA)
MPS 404A	SI-P	SW	40	.31	30/10	4.0	N (MOTOROLA)
TC 229							N

A = AMPS U = MICROAMPS M = MILLIAMPS NA = NANOAMPS

GP = GE. PURPOSE PW = POWER A = AMPLIFIER
SW = SWITCH HF = HI.FREQUENCY RF = RADIO FREQUENCY
HV = HI.VOLTAGE CH = CHOPPER LL = LOW LEVEL
LN = LOW NOISE LV = LOW VOLTAGE M = MULTIPLE DEVICE

B = BERNE LIB. C = CIRCUS LIB. I = IMPACT LIB.
N = NET-1 LIB. S = SCEPTRE LIB. Y = SYSCAP LIB.

K = DICUBT MANUAL, TM342-1-8, AUTONETICS, AUG., 1968
K = SCEPTRE BOOK (BOWERS & SEDORE, PRENTICE HALL, 1971)

COMMENT SHEET

MANUAL TITLE: AITRAC USER INFORMATION MANUAL

PUBLICATION: SAND77-0939 REVISION: Sept 1977

FROM:

NAME: _____

BUSINESS ADDRESS: _____

_____

_____

COMMENTS:

This form is not intended to be used as an order blank. Your evaluation of this manual will be welcomed by R. G. Mosteller, SLA Division 1132.

Any errors, suggested additions or deletions, or general comments may be made below. Please include page number references.

DISTRIBUTION:

1116 C. W. Cook
1123 D. R. Breding
1126 G. J. Hansen
1126 R. E. Peppers
1126 L. T. Livingston
1132 A. B. Church
1132 J. P. Luette
1132 J. L. Lovell
1132 L. H. Skenandore
1132 J. L. Montoya
1132 H. E. Roberts
1132 R. G. Mosteller (75)
Attn: S. Bernstein (25)
Berne Electronics
1132 L. W. McEwen
1132 R. L. Peabody
1132 R. W. Durkee
1132 R. J. Burton
1242 H. H. Eyer
1242 S. B. Roeske
1243 J. P. Hamilton
1243 D. G. Beatson
1351 M. B. Sandoval
1732 V. W. Hansen
1735 F. G. Broell
2116 W. T. Corbett
2141 D. H. Loescher
2153 C. P. Stanton
2153 R. V. Baron
2153 R. P. Roberts
2155 J. McBrayer
2162 T. J. Williams
2162 M. F. Deveney
2165 J. L. Wilcoxen
2165 J. L. Johnson
2166 N. F. Siska
2167 R. A. White
2167 J. H. Stichman
2328 J. N. Shinkle
2331 J. K. Shane
2332 R. W. Stafford
2332 E. L. Lane
2332 R. G. Husa
2335 J. J. Anastasio
2335 A. P. Disch
2335 N. C. Widenhoefer
2336 A. E. Giddings
2336 D. N. Harstad
2345 F. E. Heard
2351 J. F. Mareda
2532 K. E. Nowotny
2552 P. J. Spellman
2552 J. O. Wheeler
4314 L. R. Dominguez
5131 T. J. Tucker
5132 V. R. Yarberry
5132 G. C. Steffen
5154 D. T. Sanders
5167 D. A. Benson
5233 W. K. Tucker
5233 E. C. Cnare
5241 K. D. Bergeron
5423 R. P. Toth
5736 T. D. Mc Connell
9344 F. Aragon
9354 J. E. Solberg
9355 P. B. Higgins
9421 J. P. Watterberg
9472 K. S. Hess
9482 C. M. Warthen
9521 J. C. Connell
9526 T. Starr
9624 G. R. Urish
8266 E. A. Aas
3141 C. A. Pepmueller (Actg) (5)
3151 W. L. Garner (3)
For DOE/TIC (Unlimited Release)
DOE/TIC (25)
(R.P. Campbell, 3172-3)