

Intusoft Newsletter

Personal Computer Circuit Design Tools

April 1991 Issue



Copyright © Intusoft, All Rights Reserved

(310) 833-0710 Fax (310) 833-9658

DELUXE UPGRADE OPTION FOR ICAP OWNERS

PC user's of Intusoft's ICAP circuit simulation systems can take advantage of a new upgrade option that adds several modeling features. The ICAPS deluxe option, which can be added to any ICAP package, includes SPICEMOD 1.4, the RF device library, the SPICE APPLICATIONS HANDBOOK, and all of the currently available vendor supplied op-amp models. SPICEMOD allow users to develop their own SPICE models, for a variety of semi-conductors, from data sheet parameters easily and quickly. The library additions include 40 RF BJT models, over 500 op-amp models from various manufacturers and all of the models associated with the SPICE APPLICATIONS HANDBOOK. The model libraries are Berkeley compatible and include SPICENET symbols and PRESPICE compatible ".LIB" files.

In This Issue

- 1 ICAP Deluxe Option
INTUSCOPE PC Update
- 2 **Modeling A Fuse**
- 8 Fuse Model Listings
- 9 Undocumented SPICE
Options You Can Use:
ITL6 Source Stepping

INTUSCOPE 3.1 Updates

The INTUSCOPE 3.1 update has been shipped. If you have not returned your INTUSCOPE free update card, you must do so by May 15, 1991 in order to qualify for the free update. INTUSCOPE requires a coprocessor, an EGA, VGA, or Super VGA (800x600) graphics adapter and a Microsoft or compatible mouse and driver. CGA and Hercules graphics adapters are not supported.

We would like to thank our beta sites for their assistance in developing the new INTUSCOPE program. We have also had enormous positive feedback from our users about the new look and features. Here are just some of their comments:

Fred Tourtellote, Omni Research, *"An impressive data analysis package."*

Ron Ward, Micro Design, *"The graph printouts are top quality."*

Jonathon Kramer, Drexelbrook Engineering, *"The new interface is great... great stuff."*

Sam Clark, Clark Labs, *"Awesome!"*

Glen Fasnot, Acron Standard, *"I'm really happy with its performance."*

John Barnes, Varian, Inc., *"I like it. A vast improvement. Good manual."*

Irv Weiner, Analogic, *"It gets me the results faster than any other data analysis package."*

Modeling A Fuse

Circuit protection is an important part of the circuit design process and its benefits are well known. In short circuit protection, the fuse plays a vital role. Fuses are the simplest means of automatically opening a circuit under short-circuit or overload conditions. Incorporating a fuse model into SPICE would allow a simulation to predict component and circuit stress under extreme conditions without actually having to subject the circuit to hazardous test conditions. In addition, the effect of the nonlinear fuse resistance, especially in low voltage circuits where fuse resistance can become significant, can be examined.

In this application note, we will develop a model for the normal blow 8AG glass tube fuse family. This work could, however, be extended to other fuse types.

Previously Developed Thermal Models

In the July and October 1988 newsletters (newsletters 10 and 11), Intusoft introduced thermal models for diodes, transistors, thermistors, and a tungsten lamp. The models are unique because they are able to use temperature as a state variable, a feature not inherently available in any SPICE program, where temperature is normally a constant throughout the simulation. This modeling feature allows an analysis to proceed with temperature playing a controlling force in the simulation process as opposed to a simple constant.

Some of the thermal modeling concepts, including those associated with the tungsten lamp, will be used in the fuse model development.

Fuse Characteristics

The fuse may seem like a simple element to simulate, but there are a number of important properties that prohibit the possible use of a simplified model such as an ideal current controlled switch. A fuse is a piece of metal wire, usually an alloy consisting of nickel, copper, brass, or zinc that has a low melting point. Fuse metals for electric fuses may also be made of such materials as bismuth cadmium, lead, silver, aluminum or tin. The fuse tube is sometimes filled with heat resisting powder which confines and extinguishes the arc quickly when the fuse material melts. Fuses will normally carry full-rated load current continuously, but will blow in a specified amount of time when the rated current is exceeded.

Fuse characteristics vary somewhat by the choice of the filament material, but the time and minimum current ratings are affected by the ambient temperature and the line current. The resistance of the fuse causes dissipation of energy, liberation of heat, and the rise of temperature. Sufficient current will cause the temperature to rise to the melting point of the material and open the circuit. The rating of a fuse therefore, depends upon its dimensions, mounting, enclosure, material, and several other factors which affect its heat dissipating capacity. Because of the dependences on heat, temperature, time, and a fuse's nonlinear resistance, a simple current monitoring switch model cannot be used for a realistic simulation.

Fuse Model Review

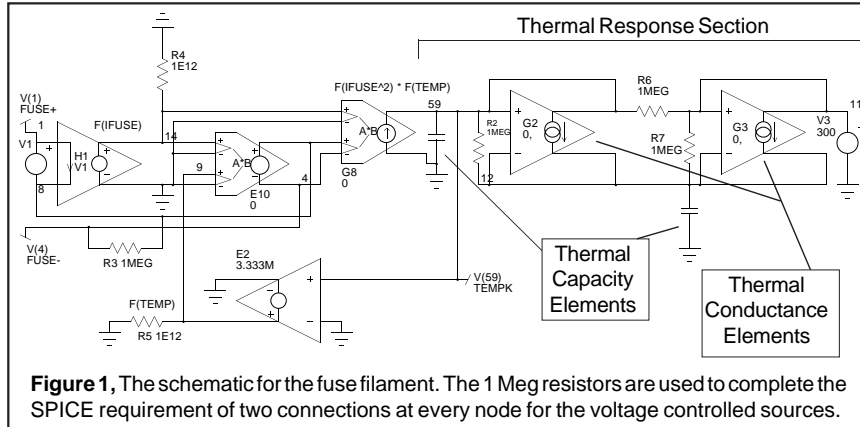
The complete Intusoft fuse model consists of two parts; the fuse filament, which encompasses the temperature response, and the fuse blowing circuit, which allows the fuse connection to be broken when the filament reaches its melting temperature.

The circuit temperature is normally a constant in SPICE 2, therefore, it is necessary to use an alternate simulation variable to represent the dynamic nature of temperature in the fuse. The flow of heat through a thermal resistance is analogous to the flow of direct current through an electrical resistance because both types of flow obey similar equations. The heat flow equation is $q = \Delta T / R$, where q is the heat flow, ΔT is the temperature potential, and R is the thermal resistance. If we replace heat flow by I , the current, the temperature potential by the electrical potential, i.e., the voltage difference, and the thermal resistance, by the electrical resistance, we obtain the equation for the flow rate of electricity; $I = E / R$.

Fuse Voltage-Current Relationship

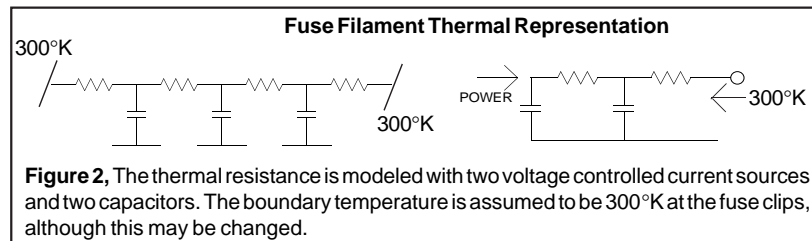
A schematic of the model for the fuse filament, similar to the tungsten lamp model in the Oct. 1988 newsletter, is shown in Figure 1. Fuse current is converted to a proportional voltage through H1. The voltage across the fuse, output of E10, is a function of the fuse current and temperature according to Ohm's law; $V_{\text{fuse}} = I_{\text{fuse}} * R(t)$. The temperature vs. electrical resistance response, $R(t)$, is a combination of E2, which represents the change in electrical resistance over temperature and a polynomial multiplier in E10, which represents the fuse's "cold" resistance. The resistance vs. temperature coefficient in E2 was chosen to be a constant. The voltage across the fuse and the fuse current are then multiplied by G8 to convert the fuse power to a current proportional to heat flow; which will be used to calculate the temperature.

Fuse Model Review (cont'd)



Thermal Response

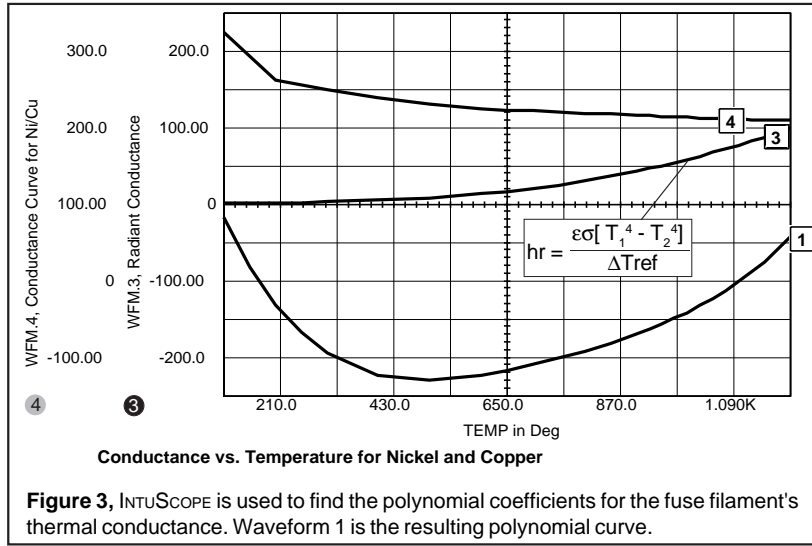
The right most portion of the circuit is where the power vs. temperature relationship is generated. In this section of the model we are using the following analogs; voltage = temperature in degrees K, current = power in watts, resistance = thermal resistance in degrees K per watt, and capacitance = thermal capacity in watt-sec per degrees K. Development of the values for this section requires knowledge of the filament material's thermal resistance and thermal capacity. The values used here were developed based on the assumption that the filament was made of copper and nickel. Other material combinations will require modifications to the values.



The thermal response of the filament is set up by the thermal resistance, the thermal capacity, and the electrical resistance vs. temperature. In setting up the thermal resistance, it was assumed that the fuse was made up a series of sections with the ends of the fuse bounded by the ambient temperature and the power inserted at the center. For the model, a Thevenin equivalent of a two section representation was constructed (Figure 2).

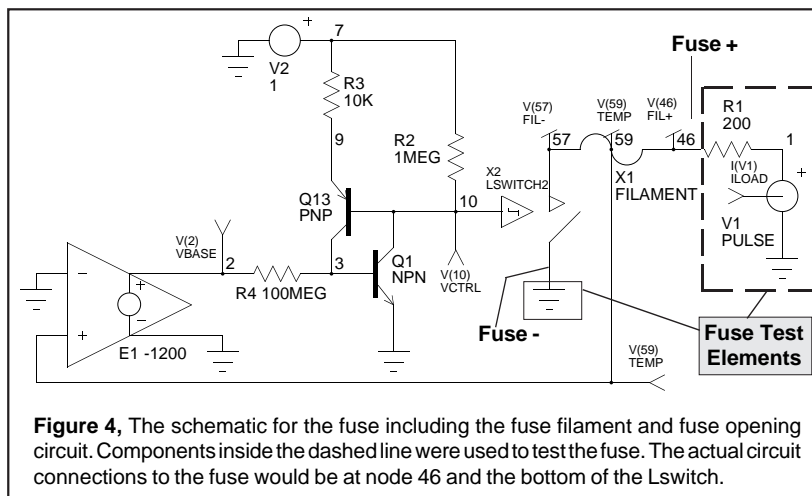
The components G2 and G3 represent the filament's thermal conductance, which was found more easily than the resistance. The conductance is made up of two components, conduction and radiation. The overall conductance, h , for a surface to or from

which heat flows by conduction and radiation is given by $h = hc + hr$. To find the proper relationship of conductance vs. temperature, the conductance curves for the material and radiation were added in INTUSCOPE and a polynomial regression was performed. The resulting polynomial coefficients were then inserted into the VCCS elements, G2 and G3. The 1 Meg resistors were added to satisfy SPICE's requirement for 2 connections at every node.



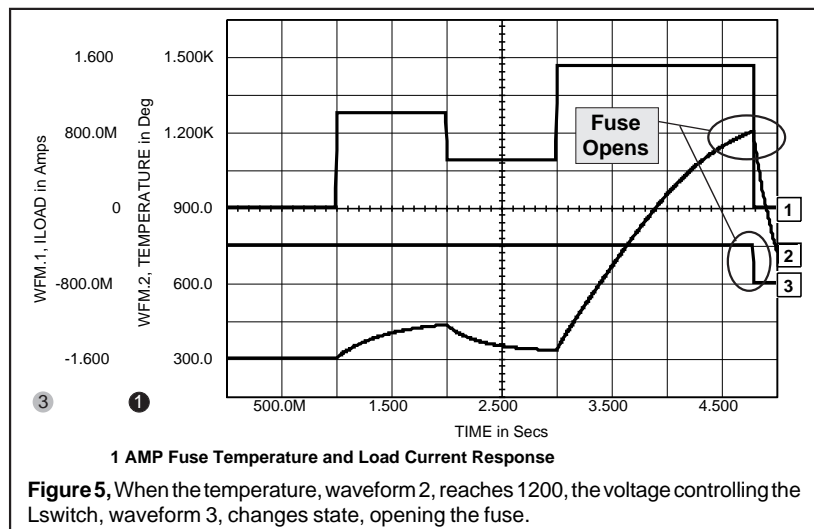
Opening The Fuse

The switching circuit, shown below, is used to stop the line current flowing through the fuse. It consists of an amplifier that controls a simple SCR switch. The amplifier, E1, has an offset equal to the temperature that the fuse is supposed to melt at, 1200°K.



Opening The Fuse (cont'd)

When the temperature rises to the offset, the SCR is triggered. This opens the Lswitch element and cuts off the line current. The Lswitch element is a nonlinear switch that is either open with 1E15 ohms, or closed with a small resistance. The SCR and its associated components act like a one-shot; once it is triggered, the Lswitch cannot be closed. This configuration allows the fuse temperature and resistance to change with the current (Figure 5). However, simulation results show the fuse will only accurately model the time to blow at currents exceeding 135% of rating.



Changing Fuse Parameters

Because of the wide variations in fuse construction it is not possible to create one “generic” fuse subcircuit. However, after setting a few fuse constants, the topology developed here does allow for a single subcircuit representation of an entire fuse family which can vary based on the current rating.

Referring to the model listings in Table 1, the thermal conductance, thermal capacity, and nominal electrical resistance were all made proportional to rated fuse current, I_B . This can be done because it is assumed that the fuse resistance will vary inversely with the fuse rating making the power to blow a fuse constant across a fuse family.

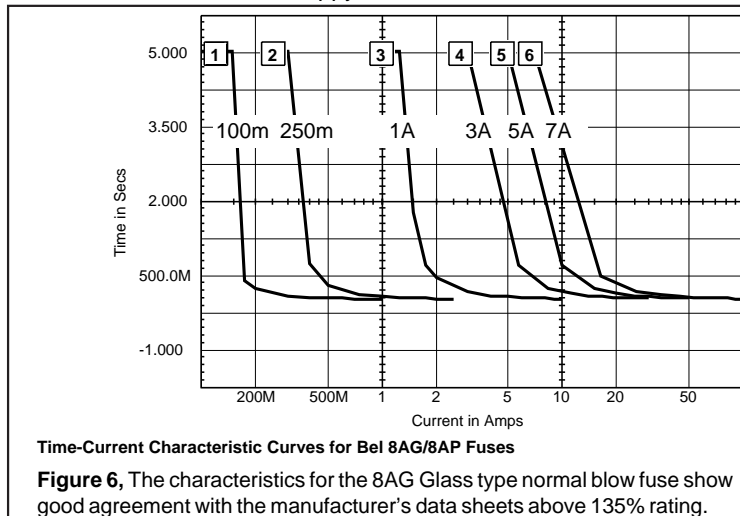
When creating a model for a different fuse family, the first parameter set should be the electrical resistance exponent, RCOLDE. The multiplier in E10 of filament, $\{.1 * I_B^{(RCOLDE)}\}$, times the fuse current, gives the voltage drop of the fuse. An approximate value of RCOLDE can be determined from the cold

resistance vs. IB data by setting the preceding expression equal to the fuse voltage drop (fuse cold resistance*current rating). The IB terms in the conductance polynomial must then be accounted for using the calculated RCOLDE value.

The constants KS and TB, can then be used to fit the manufacturer's curves for the fuse family by adjusting the thermal conductance and capacity. The fuse time to blow vs. current curves tend to be asymptotic at currents near the rating and nonlinear at high currents. KS has the effect of setting the point at which the time vs. current curve becomes linear. The thermal capacity also determines the time to blow, especially at high currents. The TB constant will adjust this time. The IB exponent, HCTE, along with TB, adjusts the spacing between the curves for different ratings. Since the curve spacing is usually nonlinear, the values for TB and HCTE will have to be averaged across the fuse family. If greater accuracy is desired, the HCTE and TB values can be optimized for a single fuse rating. If the fuse composition is known, the polynomial coefficients for the conductance vs. temperature, the electrical resistance vs. temperature coefficient and the fuse melting temperature can also be recalculated.

After these constants are found, the fuse can then emulate any ampere rating in the family simply by calculating the equations in curly braces using the fuse's rated current or by using the PResPICE program's parameter passing feature.

The constants provided in Table 1 set up the subcircuit to model the 8AG Normal blow glass tube series from Bel Fuse Inc. Fuse models for fast acting and slow blow fuses from Littlefuse®, along with the models and subcircuits shown in this application note, can be obtained on floppy disk from Intusoft for \$20.



Circuit and Model Related Problems

Fitting the fuse response at current levels close to the rating (<135%), in addition to those at high current levels, was not possible with the model topology chosen. This is not a critical limitation as simulations of several hours length are highly impractical. Long simulation times can also cause SPICE to trigger the “internal time step to small” message because of the fast time constants associated with the switch that opens the fuse. When the fuse blows, the simulation will appear to slow down to a halt and may abort. The ITL4 and RELTOL values can be adjusted to help the simulation run through the nonlinearity; for example, .OPTIONS ITL4=100 RELTOL=.01. Also, starting the simulation with current flow that exceeds the fuse’s capacity may cause DC convergence and model initialization problems.

As with many data sheets, the ones for fuses leave much to be desired, at least in terms of SPICE modeling. However, the applications engineer at Littlefuse® Inc. was quite helpful and provided detailed electrical data crucial to the model’s development. Calls to other fuse manufacturers revealed a reluctance to state the fuse materials or electrical resistance, thus leaving the designer to guess at a number of important parameters. In any case, with a little trial and error, very accurate results for almost any fuse can be produced with the new Intusoft model, allowing new simulation areas to be explored.

Fuse Model Listings

The PRESPICE subcircuit model for the fuse is shown below. For actual use, the parameters IB, RCLDE, HCTE, TB, and KS should be inserted into the equations in curly braces, evaluated, and replaced with numbers. The constants for the Bel 8AG/8AP fuse family are listed below. The desired ampere rating can then be entered for any fuse in the family.

Table 1, PRESPICE Subcircuit listing for a Fuse	
.SUBCKT FUSE 1 4	R5 9 0 1E12
*Connections Fuse+ Fuse-	R7 11 12 1MEG
*Fuse temperature at node 59	R6 11 59 1MEG
* For 8AG Fuses Use	*Fuse Opening Circuit
* RCLDE=-.3, HCTE=2.0, KS=5.3U, TB=.5	E1 42 0 59 0 -1200 1
*Fuse Filament Circuit	R8 43 44 10K
V1 1 8	V4 43 0 1
H1 14 0 V1 1	Q1 39 45 0 NPN OFF
E10 8 37 POLY(2) 14 0 9 0 0 0 0 0	.MODEL NPN NPN(BR=100 CJC=10PF)
+ {.1*IB^(RCLDE)}	R9 42 45 100MEG
G8 0 59 POLY(2) 14 0 8 37 0 0 0 0 1	R10 43 39 1MEG
R2 59 12 1MEG	Q2 45 39 44 PNP OFF
G2 59 12 59 12 0, {389*IB^(2+RCLDE)*KS},	.MODEL PNP PNP(BR=100 CJC=10PF)
+{-884M*IB^(2+RCLDE)*KS}, {1.76M*IB^(2+RCLDE)*KS},	X1 37 4 39 LSWITCH2
+{-1.48U*IB^(2+RCLDE)*KS}, {484P*IB^(2+RCLDE)*KS}	.SUBCKT LSWITCH2 1 2 3
C1 59 0 {.5M*IB^(HCTE)*TB}	*Connections +- Control
G3 11 12 11 12 0, {389*IB^(2+RCLDE)*KS},	R1 1 2 1E15
+{-884M*IB^(2+RCLDE)*KS}, {1.76M*IB^(2+RCLDE)*KS},	R3 3 0 1E12
+{-1.48U*IB^(2+RCLDE)*KS}, {484P*IB^(2+RCLDE)*KS}	E1 4 0 POLY(1) 3 0 0 0 0 0 0 1
C2 12 0 {.5M*IB^(HCTE)*TB}	R4 4 0 1E6
V3 11 0 300	G1 1 2 POLY(2) 1 2 4 0 0 0 0 0 1MEG
*V3's value is the Ambient Temp. in Deg. K	C1 1 2 10PF
E2 9 0 59 0 3.333M	.ENDS
R3 8 37 1MEG	.END
R4 14 0 1E12	

Undocumented SPICE Options

Many of the SPICE programs compatible with Berkeley SPICE 2G.6, including all versions of IsSPICE, contain several undocumented options that can be quite useful. Two that have been tested and been found to work reliably are the .OPTIONS ITL6 option and the .DC analysis resistor/temperature sweeping options. In this issue, we will focus on the DC convergence assisting option, ITL6.

.OPTIONS ITL6

The IsSPICE .OPTIONS statement includes parameters that allow the user to control the simulator's default settings. One that is currently undocumented is the ITL6 option. ITL6, also known as the source stepping option, can be used to greatly augment the DC convergence capabilities of IsSPICE. If convergence has ever been a hurdle in your simulation activities, you will want to understand and use this option. It has been found to be especially effective in helping circuits containing bipolar devices and/or op-amps to converge. This option is often the main reason behind the claims of improved convergence properties made by many SPICE vendors in spite of the fact that this feature has always been available in Berkeley SPICE and IsSPICE.

The SIMULATING WITH SPICE reference book details a number of procedures that are quite useful in solving virtually all DC convergence problems. These include the use of .NODESET and .IC statements, .TRAN UIC option, OFF keyword, the .OPTIONS parameters, and the pseudo-transient method. The .OPTIONS ITL6 parameter can now be added to the list.

What Is Source Stepping

Source Stepping is an enhancement to the Newton-Raphson iteration method used by SPICE to calculate a circuit's DC operating point. Rather than trying to find the operating point with all the power supplies at full strength, source stepping uses successive approximations based on a series of DC sub-operating point calculations to arrive at the final DC operating point.

When the ITL6 option is present, the operating point calculation proceeds as follows. At the beginning of the simulation, all power sources (voltage and current) are set to a fraction of their actual values and the DC operating point is calculated. All the source values are then incremented slightly and another DC operating point is calculated using the previously calculated operating point as an initial guess. Theoretically, since the sources are only

What Is Source Stepping *(cont'd)*

incremented a small portion of their total value, the initial guess will most likely be very close to the solution for the current operating point. In this manner, the sources are stepped towards their final values and the resulting operating point is calculated.

Source stepping may take more time than the default method, primarily because a DC operating point must be calculated for each value of the incrementing sources. However, there are many cases when the results can be found more quickly using source stepping than by increasing ITL1 (N-R iteration limit), as well as, in instances when the default analysis method fails.

Using The ITL6 Option

Source stepping can be invoked in most SPICE versions (2G.6 compatible) by simply setting the .OPTIONS variable, ITL6, to a value other than 0. The value entered will be used as the Newton-Raphson iteration limit, similar to the ITL1 value, at each step of the source values.

For example, `.OPTIONS ITL6=100`

will cause SPICE to use the source-stepping method with an iteration limit of 100 at each step. Using fewer Newton-Raphson iterations per source step does not necessarily result in faster convergence. The default value for ITL1, equal to 100, is therefore recommended. The default value for ITL6 is 0, which causes `IsPICE` to use the Newton-Raphson method directly. Though easily invoked, source stepping is best used after trying some of the other convergence solving methods such as the increasing of ITL1 and use of the .NODESET statement.

When Source Stepping Doesn't Work

The ITL6 option is not the “end all” solution to DC convergence problems. Many times source stepping will not work well, for example, in circuits that behave in a highly non-linear manner with regard to their operating point. Experience with source stepping has shown, however, that the algorithm works quickly when it is successful.

In such a complex program as SPICE, it is inevitable that new features and methods are continuously uncovered. As convergence tends to be a recurring problem for circuit simulation it is comforting to find a new tool right at our disposal.

[1] RCG Research, Inc. Inside Spice Newsletter, Jan-Feb. 1990

[2] U.C. Berkeley SPICE Version 2G User's Guide Aug. 10 1981