Intusoft Newsletter

Personal Computer Circuit Design Tools

November 1992 Issue

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

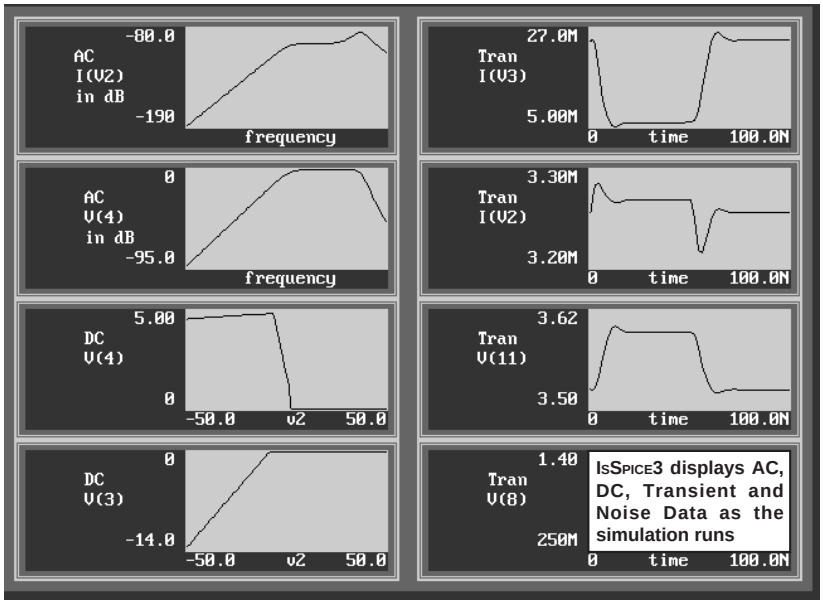
IsSpice3: Simulates More Circuits

The new IsSpice3 simulator contains a wide variety of new models and analysis capabilities. In this newsletter we are going to focus on some of the new problems that IsSpice3 can attack, that your older SPICE programs could not. We will also explore some of the ways in which the new analog behavioral modeling capabilities can dramatically speed up your circuit simulations. IsSpice3 contains new built-in models for:

> GaAs Mesfets, 3 new Mosfet levels (BSIM 1 and BSIM 2, and Level 6), lossy transmission lines using a distributed approach, switches with hysteresis, and several analog behavioral extensions.

We will touch on a few of these topics in this newsletter and the rest in future newsletters.

In This Issue



IsSpice3 displays AC, DC, Transient and Noise Data as the simulation runs

New If-Then-Else Speeds Simulations

With SPICE 2 based programs, a variety of limiters can be built using passive, active and dependent source elements. Figure 1 shows the output of two limiters resulting from an input ramp (WFM.1). One limiter compares two input signals and generates a 0V or 1V output. In the simulation, one input was grounded while a ramp was fed into the other input. A polynomial function was used in the last dependent source to steepen the transition of the output waveform (WFM.3). The second circuit simply limits the input voltage between user specified positive and negative limits.

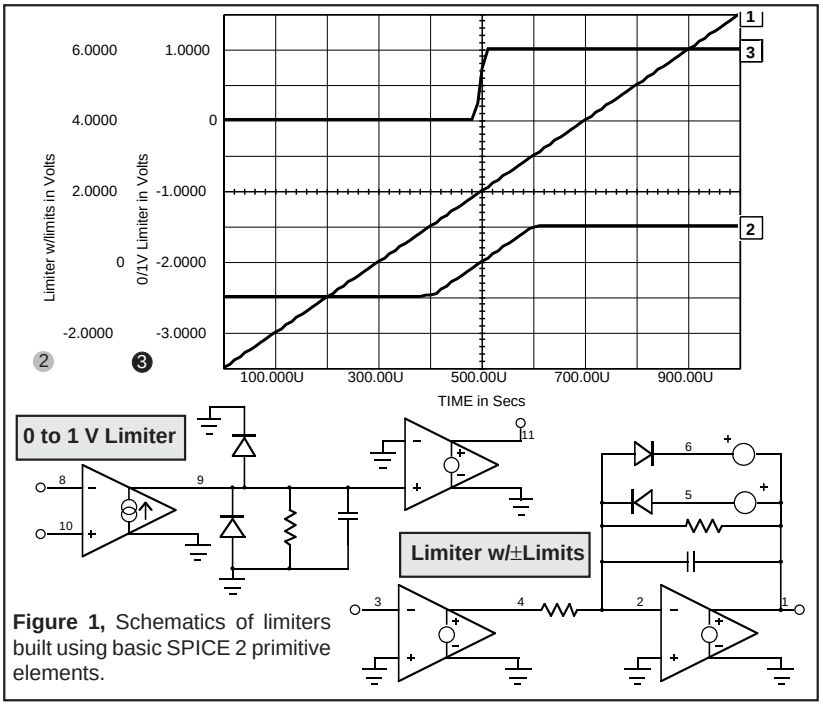


Figure 1, Schematics of limiters built using basic SPICE 2 primitive elements.

While useful, the complexity and nonlinearities introduced by the dependent source polynomials and diodes, cause these limiters to run slowly when compared to the new IsSpice3 limiter.

IsSpice3 program includes a number of new analog behavioral modeling features. One of the most powerful is the If-Then-Else function. This function is implemented using the built-in nonlinear dependent source element, B. A limiter, comparable to those described in Figure 1, can be described as follows:
`B1 2 0 V=V(1) < 0 ? 0 : V(1) > 1 ? 1 : V(1)`. This translates to "If V(1) < 0 then V(2) = 0 else if V(1) > 1 then V(2) = 1 else V(1)=V(2)". Of course, many other limiter variations are possible.

Variable Impedance Limiter

As a practical example, a limiting amplifier was created. In order to realistically model the performance, a variable impedance limiting element must be constructed. This is accomplished using a more complex If-Then-Else expression (Figure 2).

```
X1 6 4 RLIMIT {VMAX=10 VMIN=-.7 RVAL=10K RMIN=1 }

.SUBCKT RLIMIT 1 2 {RMIN=1}
B 1 2 I=V(1,2) > {VMAX} ?
+ {VMAX * (1/RVAL - 1/RMIN) } +  V(1,2)  / {RMIN} :
+ V(1,2) < {VMIN} ? {VMIN * (1/RVAL - 1/RMIN) } +  V(1,2)  / {RMIN} :
+  V(1,2) / {RVAL}
.ENDS
```

X1 RLIMIT

Figure 2, The IsSpice3 netlist describing a variable resistance limiter.

The Rlimit function is evaluated as follows: If the voltage across the limiter is greater than some maximum, VMAX, then the voltage is equal to VMAX * (1/RVAL - 1/RMIN) + V(1,2) / RMIN. If the voltage across the limiter is less than some minimum, VMIN, then the voltage is equal to VMIN * (1/RVAL - 1/RMIN) + V(1,2) / RMIN. Otherwise, the output voltage is equal to voltage across the limiter / RVAL. RVAL is the resistance when the voltage is less than VMAX or greater than VMIN. RMIN is the resistance when the voltage is outside this interval.

Figure 4 shows the transient response of the limiting amplifiers in Figure 3 to an input ramp from 20V to -10V. The value of VMIN is set to .7, a diode drop. For comparison, a zener diode was simulated in place of the variable resistance limiter function.

The variable resistance limiter simulates very quickly and provides the user with the ability to specifically state the resistance limits rather than relying on those produced by the zener model.

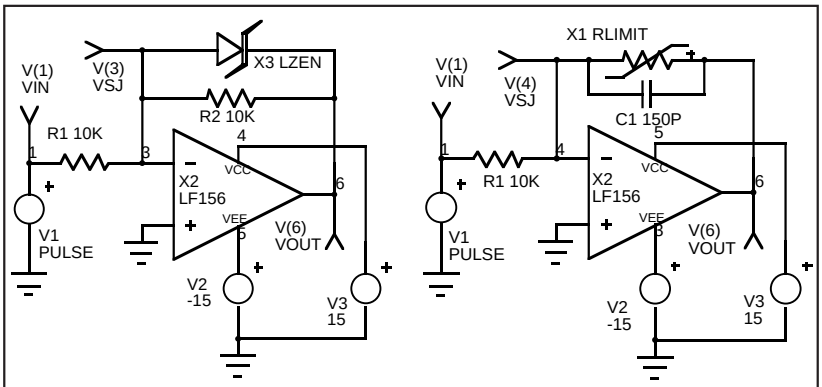


Figure 3, The limiting amplifier circuit on the right uses the new variable resistance limiter to replace the zener diode used in the circuit on the left.

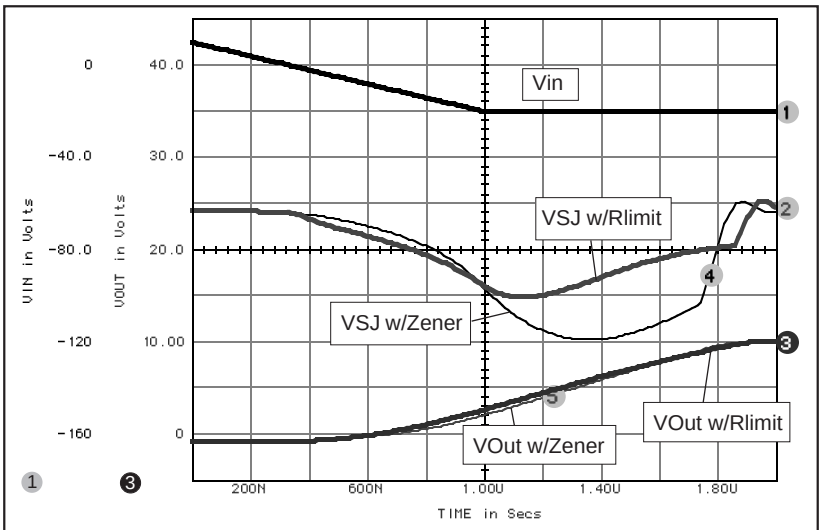
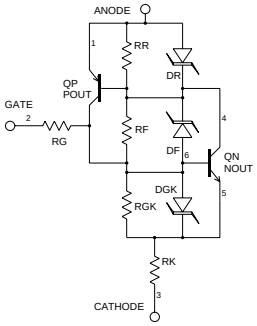


Figure 4, Graph of the input, VSJ, and output waveforms from the limiting amplifier. The thicker waveforms were from the simulation using the new Rlimit subcircuit.

The Fastest SCR/Triac Models Ever

In the February 1992 Intusoft Newsletter, we explored a detailed SPICE model for SCRs. The model, shown left, simulated the majority of higher order effects including the turn-on/turn-off, dv/dt, forward break-over, and reverse and gate characteristics.



Using the new If-Then-Else function, we can construct near ideal SCR and Triac elements. There are many motivations for creating simplified versions. Without the need to get bogged down in the details of snubbing and control circuitry, initial design simulations can be performed much more quickly. The ideality provides greatly increased simulation speed and allows the model to be parameterized in a number of ways.

Figure 5 contains the circuit used to test the SCR and Triac models. The subcircuit netlists are very similar. In the SCR, for example, there are current meters to measure the gate and anode-cathode current (VG/VM), and a switch controlled by the voltage, V(5). This voltage is regulated by an If-Then-Else statement, BSCR. When V(5) is > .2V (VT+VH), the switch is closed, and when V(5) is < 0 (VT-VH) the switch is open. VT and VH are defined in the switch .Model statement. V(5) is set to 2V by BSCR if the gate current, I(VG), is > 50mA and VAK is positive. Otherwise, BSCR will output a voltage equal to the current flowing through the SCR, I(VM). If I(VM) is greater than .2mA, the switch will remain closed. The SCR will turn off at I(VM) = 0 (VT-VH) allowing it to be used with or without snubbing

```
.SUBCKT SCR 1 3 2
* Anode Gate Cathode
VM 4 2
VG 3 2
SW 1 4 5 0 SWITCH
.MODEL SWITCH SW RON=.01 VT=0.1 VH=0.1
BSCR 5 0 V=I(VG) > 0.05 ? V(1,2) > 0 ? 2 : I(VM)
*Switch Control
* If the gate current is > 50mA and the VAK > 0 turn
* on the switch. If SCR current > .2mA the SCR will stay on.
.ENDS
*********
.SUBCKT TRIAC 1 2 3
* MT2 MT1 Gate
VM 4 2
VG 3 2
SW1 1 4 5 0 SWITCH
SW2 1 4 6 0 SWITCH
.MODEL SWITCH SW RON=.01 VT=0.1 VH=0.1
*Quadrant 1 and 2
BT1 5 0 V=ABS(I(VG)) > 0.05 ? V(1,2) > 0 ? 2 : I(VM)
*Quadrant 3 and 4
BT2 6 0 V=ABS(I(VG)) > 0.05 ? V(1,2) < 0 ? 2 : -1*I(VM)
* If the gate current is > 50mA or < -50MA and VMT2-VMT1 is >< 0
* turn on a switch. If current > |.2mA| the TRIAC will stay on.
.ENDS
```

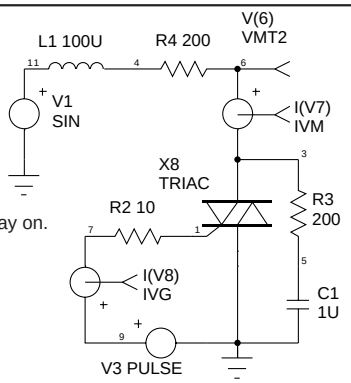


Figure 5, IsSpice3 subcircuit listings for the near ideal SCR and Triac models using built-in switch and If-Then-Else statements.

circuitry. The switch on resistance is set to .01Ω. The off resistance defaults to 1E12Ω. The graph in Figure 6 shows the response of the test circuit using the new SCR and Triac models.

Parameterizing Adds Versatility

By parameterizing the .Model and If-Then-Else statements, the on/off resistance, gate turn-on current (IGT) and holding current (IH) can be specified when the device is used. For example:

```
.Model Switch Sw Ron={RON} Roff={ROFF} VT={IH/2} VH={IH/2}
BSCR 5 0 V=I(VG) > {IGT} ? V(1,2) > 0 ? 2 : I(VM)
```

The gate drive circuitry can also be parameterized allowing the SCR to be controlled using design parameters. For example,

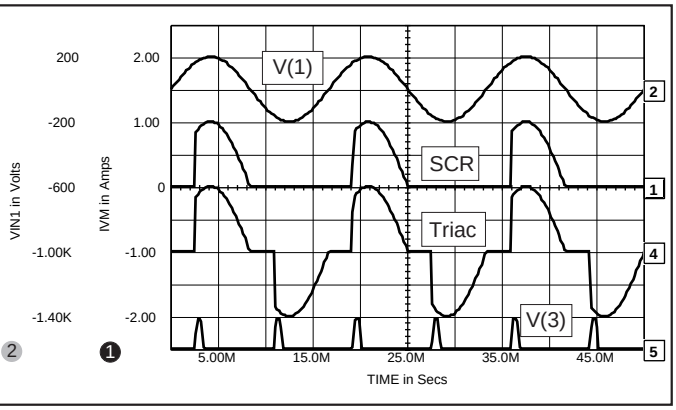


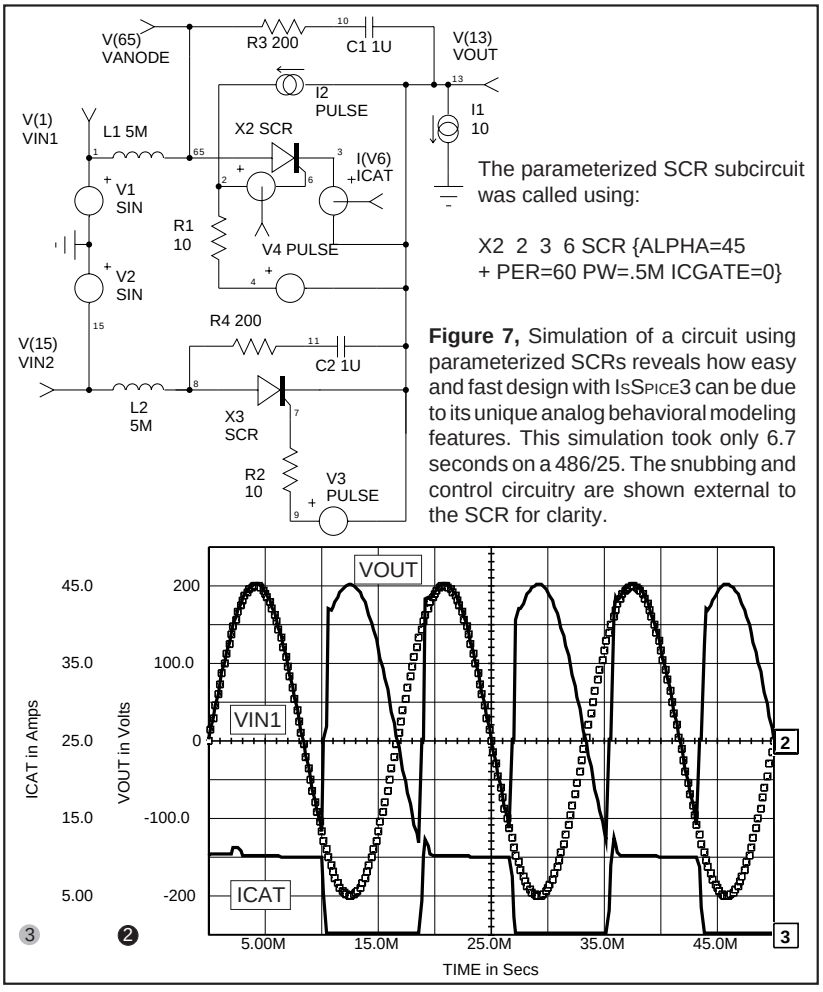
Figure 6, The graph shows the response of the new SCR and Triac models to a series of positive turn-on pulses. A 200V peak sinusoidal input was used.

adding the following statements to the SCR subcircuit,

```
RS 1 5 200
CS 5 2 1UF IC={ICGATE}
RG 3 6 10
VG 6 2 Pulse 0 10 {ALPHA*PER/360} 0 0 {PW} {PER}
```

allows the SCR to be controlled by specifying the frequency of the control signal (PER), the control pulse width (PW), and the point in degrees when the SCR should fire (ALPHA). The snubbing has also been added creating a self contained two terminal SCR.

The parameterized SCR model was tested in the following circuit. For the initial circuit design stages, it was clear that the speed and ease of use of the near ideal SCR far outweighed any problems created by oversimplification.



The parameterized SCR subcircuit was called using:

```
X2  2  3  6 SCR {ALPHA=45
+ PER=60 PW=.5M ICGATE=0}
```

Figure 7, Simulation of a circuit using parameterized SCRs reveals how easy and fast design with IsSpice3 can be due to its unique analog behavioral modeling features. This simulation took only 6.7 seconds on a 486/25. The snubbing and control circuitry are shown external to the SCR for clarity.

IsSpice3 Simulates GaAs Mesfet Circuits

The square law Shichmann-Hodges JFET model used in SPICE 2G programs is not sufficient for modeling the short channel GaAs Mesfets in use today. Even when combined with external diodes, the resulting subcircuit response fails to predict key performance characteristics. Therefore, a GaAs MESFET model, based on the Statz model, has been built into the new IsSpice3 program. This model greatly improves upon JEFT based models with superior results for velocity saturation, doping profile, and charge storage.

The circuit in Figure 8, and output response in Figure 9, show the results of a GaAs Mesfet Nor gate running at 1.4GHz.

```
GaAs Mesfet Nor Gate
.TRAN 10P 1N
*ALIAS  V(2)=VOUT
.PRINT TRAN  V(2)  V(1)
Z1 3 2 2 DEPFET
Z2 2 5 0 ENFET
R1 2 0 20K
C1 2 0 .06PF
Z3 2 1 0 ENFET
VGG 5 0 .1V
V3 3 0 .6
V1 1 0 PULSE .6 .01 50P 50P 50P 300P 700P
*Depletion Mode Device 1ux20u VT=-.5V
.MODEL DEPFET NMF VTO=-.5 BETA=.00214 B=.2 ALPHA=6.725
+ LAMBDA=.483 RD=125 RS=125 CGS=.028PF CGD=.004PF
*Enhancement mode Device 1ux50u VT=.1V
.MODEL ENFET NMF VTO=.1 BETA=.009 B=.2 ALPHA=4.9
+ LAMBDA=.483 RD=50 RS=50 CGS=.07PF CGD=.01PF IS=25E-14
.END
```

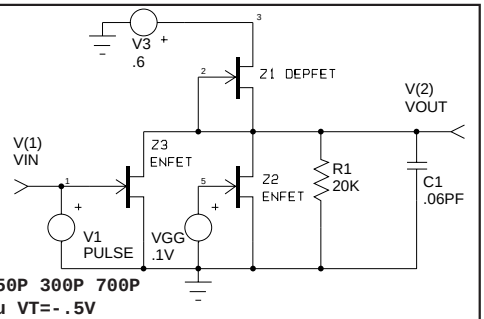


Figure 8, The new IsSpice3 program allows high speed digital and analog GaAs Mesfet based circuits to be examined.

The GaAs MESFET is called in the same way as other active circuit elements. The IsSpice3 syntax starts with the letter Z. The DC characteristics are described by the parameters VTO, B, RD, RS, and Beta, which determine the variation of drain current with gate voltage, Alpha, which determines the saturation voltage, and Lambda, which determines the output conductance. Charge storage parameters are similar to the JFET.

Because of the nonlinear effects included in the model, simulation of complete communication systems, including transient, frequency, and distortion responses is possible.

H. Statz, "GaAs FET Device and Circuit Simulation in SPICE", IEEE Transactions on Electron Devices, V34, #2, 2/1987, pp. 160-69.

A. Parker and D. Skellern, "GaAs Device Modeling for Design and Application", IEEE Intl. Sym. on Circuits and Systems, Singapore, 6/91, pp. 1837-40

S. Sussman-Fort, J. Hantgan, and F. Huang, "A SPICE Mode for Enhancement and Depletion-Mode GaAs Fets", IEE Transactions on Microwave Th. and Tech., Vol. MTT-34 No. 11, 11/1986

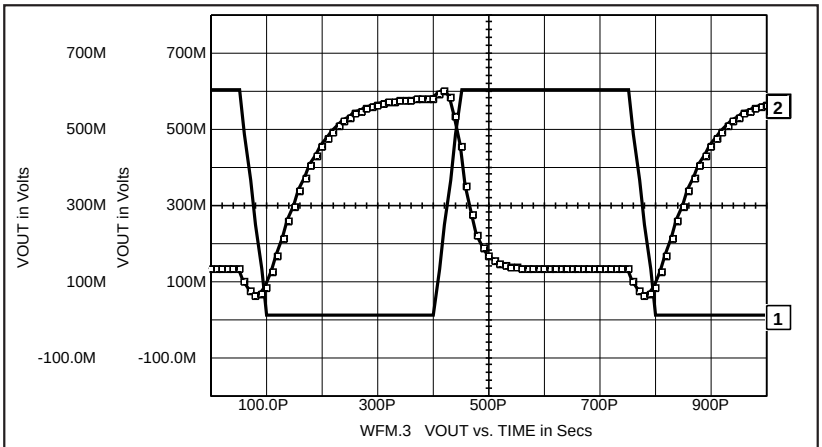


Figure 9, Turn-on and turn-off response of a GaAs Mesfet 2 input Nor Gate.

Taking The Heat Out Of Temperature Analyses

Simulating circuits under a variety of environmental conditions is critical for the success of many applications. Like SPICE 2, IsSpice3 has the ability to simulate an entire circuit at virtually any temperature. In addition, IsSpice3 also provides a new capability. It allows an alternate temperature setting on an individual element that is different from the circuit temperature. This can be extremely useful when determining the reliability of a circuit design in a harsh environment or when an active device in the circuit is subjected to heat stress causing operation to vary.

The following example investigates the effect of temperature variations on the base of a transistor. The first simulation was performed at room temperature, 27°C. The second simulation was performed with the entire circuit at 75°C. The third simulation was performed with the temperature of the transistor set to 125°C and the rest of the circuit at room temperature.

The following syntax was used to make the resistors in the circuit vary with temperature. Notice that in IsSpice3, the temperature coefficients are located in the resistor .Model statement and the temperature is changed in the .OPTIONS statement.

```
.Model Rmod R (TC1=.01 TC2=1E-6)
R19 2 1 30 Rmod
```

The name, “Rmod”, used to reference the .Model statement, occurs on all of the resistor statements in the netlist. Thus, in this simulation, all of the resistors use the same temperature coefficients, although this is not mandatory. The .OPTIONS statement, **`.OPTIONS TEMP=75`**, is used to set the circuit temperature to 75°C.

The syntax for changing the temperature setting on a transistor to 125°C is **`Q10 1 3 0 QN2219 TEMP=125`**.

The results for the three simulations are shown in Figure 10.

Individual temperature settings can be made on virtually all active devices (BJT, Diode, JFET, Mosfets) and resistors. With this feature it is easy to simulate a device at one temperature and the rest of the circuit at another temperature.

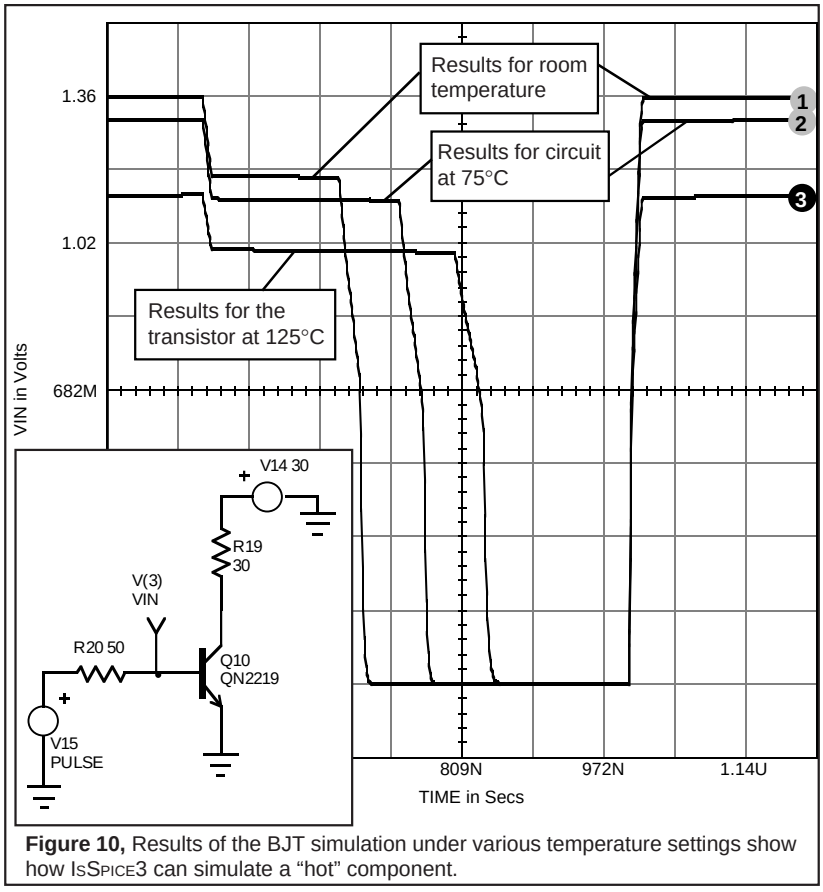


Figure 10, Results of the BJT simulation under various temperature settings show how IsSpice3 can simulate a "hot" component.

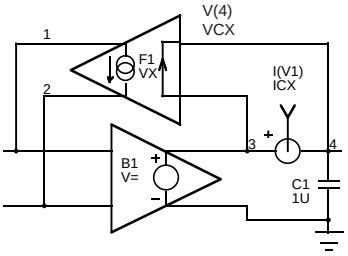
Intusoft Technical Support Expands To Compuserve

For those users wishing to communicate with Intusoft using electronic means, you can look for an Intusoft forum coming soon to Compuserve. We will be posting announcements, product updates, bug fixes, customer requested SPICE models, and other utilities in the forum. The Compuserve system will also be used to transfer customer SPICE files for testing and benchmarking purposes. Further details will be provided in a future newsletter.

New Technique Improves Power Models

The nonlinear dependent source in IsSpice3 allows designers to directly enter arbitrary equations into a model. This feature can be used to make a flexible nonlinear capacitor model. In SPICE 2, creation of a useful nonlinear power Mosfet Cgd capacitance model requires numerous elements, switches, and polynomials. The model presented here dramatically reduces the component and transient iteration count, as well as the simulation time. Assuming we use the topology shown to the left; then a function of voltage, c(v), can be made by making the source function, f(v) = v(1,2) ∗ c(v(1,2) / C1). This result is derived as follows:

capacitance and charge are defined in the following equations: Q = C∗V, where Q is charge, C is capacitance, and V is voltage.



Capacitance is then; C = Q / V = 1/V ∗ ∫ i dt. With i = C1 d f(v) / dt, the proposed IsSpice3 model is then:

Q = ∫ i dt = ∫ C1 d f(v) / dt dt = C1 f(v) = c(v) ∗ v and f(v) = (1 / C1) c(v) ∗ v giving,

B1 3 0 v=v(1,2) ∗ c(v(1,2)) / C1

An MIS Capacitor Model

A Metal - Insulator - Semiconductor (MIS) diode can be used to represent the gate-drain and gate-source diodes in a power Mosfet. The MIS diode consists of the insulator capacitance, CI, in series with the depletion capacitance, CD. If we assume that the voltage across the depletion capacitance is the applied voltage minus the threshold potential; then the model becomes:

$$CD = CJ / [1 - (VD - VT) / VJ]^M$$
$$C = 1 / (1 / CI + 1 / CJ)$$
$$= CO / (CO/CI + [1 - (VD - VT) / VJ]^M)$$

An alternative model has also been developed using a sigmoid function, borrowed from neural network theory, to make an abrupt change in capacitance at the MOS threshold potential, VT.

C =CO / (1 + e^(-M ∗ (VD - VT)); Sigmoid Capacitance

This approach fits the capacitance data and theory presented by various device manufacturers. Note that the constant “M” in the sigmoid model controls the transition sharpness, whereas in the depletion model, M controls the C-V slope in the high voltage region. The results of model tests, shown in Figure 11, compare favorably with vendor data and with the power macro models used

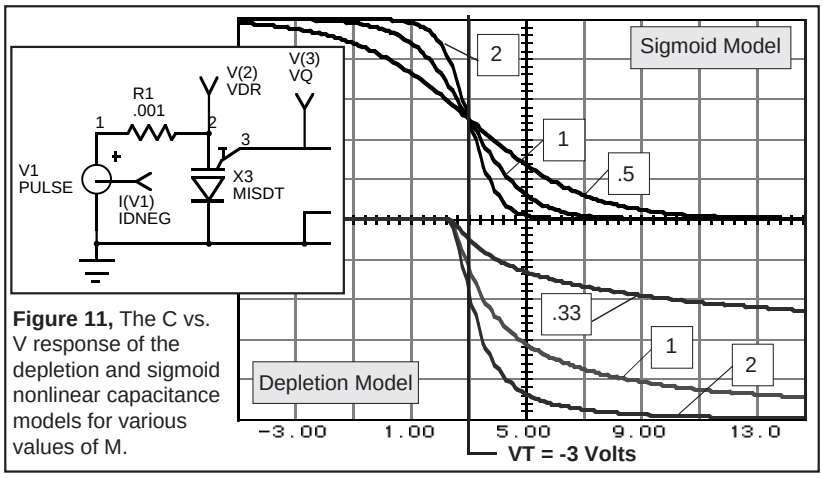


Figure 11, The C vs. V response of the depletion and sigmoid nonlinear capacitance models for various values of M.

in SPICEMOD, the SPICE modeling program. The sigmoid based model, unlike the depletion capacitance model, has no singularities. When a negative number is raised to a non-integer power, the result cannot be expressed as a real number. The iteration process used in ISSPICE3 is such that this situation could occur in the process of finding a numerical solution; causing failure to converge. For this reason, the depletion model should only be used for integer values of M. The graphs illustrate the relative behavior of the models. For M=2, the two models are quite close.

```
.SUBCKT MISD 1       2        3  {M=2 VT=-3}
*           Anode  Cathode  Charge_Test Point
* Sigmoid Model
B2 4 0 V=V(1,2) *1/( + 1 + e^({M} * (V(2,1) + ({VT}))))
V1 4 3 0
C1 3 0 {CO}
F1 1 2 V1 1
.ENDS
*****
.SUBCKT MISD2        2        3  {M=2 VJ=1 VT=-3}
*           Anode  Cathode  Charge_Test Point
* Depletion Model
B1 5 0 V= (1-(V(1,2) - ({VT})) / {VJ}) > .5 ? (1-(V(1,2) - ({VT})) / {VJ}) :  .5
*Note: Zero or negative values in a power expression cannot be evaluated,
*Therefore make sure that V(5) is >.5. Also, V(5) is initialized to zero!!!
B2 4 0 V=V(1,2) *1/({CO/CI} + (V(5) + 1E-13)^({M}))
* Protection against singularities within an If-Then-Else expression.
V1 4 3 0
C1 3 0 {CO}
F1 1 2 V1 1
.ENDS
```

Figure 12, Subcircuit models for the depletion and sigmoid capacitance models.

Adding The Capacitor To A Power Mosfet

Both of the new capacitance models were inserted into the 2N6661 power Mosfet (Table 1). The test circuit in Figure 13 was then used to test the three models. The best performance was obtained using the sigmoid model:

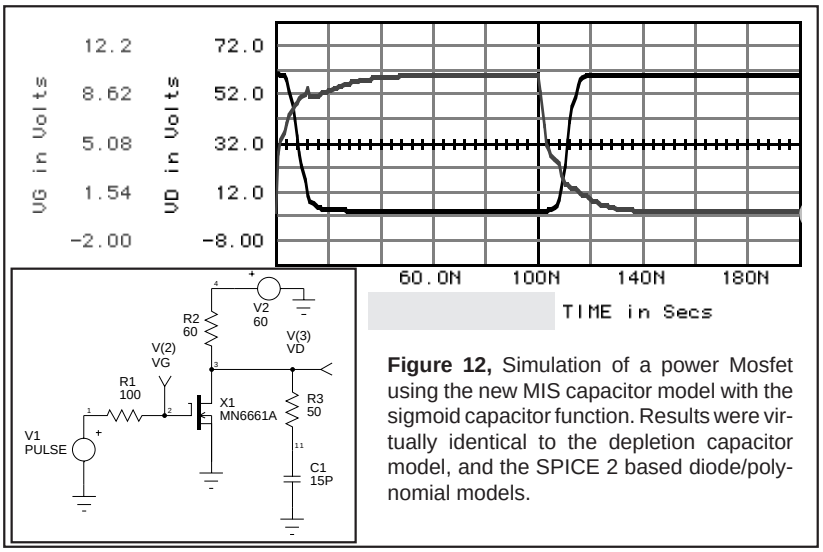


Figure 12, Simulation of a power Mosfet using the new MIS capacitor model with the sigmoid capacitor function. Results were virtually identical to the depletion capacitor model, and the SPICE 2 based diode/polynomial models.

Model	Total Iterations	Rejected time points
Sigmoid	275	21
Depletion	342	16
Standard	321	11

Numerical Overflow Protection

Certain mathematical operations produce undefined results. These results may abort the solution of an expression, preventing IsSpice3 from producing a useful result. The most commonly encountered problems are zero divided by zero, division by zero and power or exp functions that have imaginary roots. In addition, our Macintosh complier generates an exception for zero raised to any non-integer negative number. Whenever a behavioral source function has a variable argument, it is possible to encounter the problem during the intermediate computations that lead to a stable answer. For example, v=1/v(2) may be evaluated for v(2)=0 during an initial condition iteration even though v(2) would never evaluate to zero upon successful completion of the numerical iteration. The “If-Then-Else” syntax does not define an order by which expressions are evaluated; this could result in a singular expression aborting the solution even though its not used to compute the result. The MIS capacitor subcircuit shows how to construct an If-Then-Else statement that avoids this problem and other problems. For division by zero, you can add a small number to the denominator; in the MISD2 subcircuit 1E-13 was used. Zero divided by zero can be handled in a similar fashion - adding a small number to both the numerator and denominator such that their ratio is the value when both terms go to zero.

It is clear that the ability to implement element behavior using functions that are not constrained within the SPICE program is of great value. The model presented here, as well as other variations, can be used to improve all types of power device models.

Note: the CGD, RCG, MCG, ECG, DGD, MDG, and EDG elements have been removed from the sigmoid and depletion subcircuits after adding the proper calls to the MIS capacitor subcircuits (X1, X2). Connections to all models are drain, gate, source.

Standard Capacitance Models

```
.SUBCKT MN6661   10 20 30
M1   1  2  3  3  DMOS L=1U W=1U
RG  20  2  75
RD  10  1  1.49
RDS  1  3  8.9MEG
CGD  4  1  28.3P
RCG  4  1  10MEG
MCG  4  5  2  2  SW L=1U W=1U
ECG  5  2  2  1  1
DGD  2  6  DCGD
MDG  6  7  1  1  SW L=1U W=1U
EDG  7  1  1  2  1
DDS  3  1  DSUB
LS  30  3  7.5N
.MODEL DMOS NMOS (LEVEL=3 VMAX=12.4MEG THETA=.24 VTO=2.4
+ KP=.18 RS=10M IS=160F CGSO=30.4U)
.MODEL SW NMOS (LEVEL=3 VTO=0 KP=9.00M)
.MODEL DCGD D (CJO=18.3P M=.8 VJ=.21)
.MODEL DSUB D (IS=160F RS=0 VJ=.8 M=.4 CJO=65.8P TT=249N)
.ENDS
```

Sigmoid Capacitance Models

```
.SUBCKT MN6661A   10 20 30
*     TERMINALS:  D  G  S
M1   1  2  3  3  DMOS L=1U W=1U
X1 2 3  MISD1 {CI = 15P CO = 15P }
X2 2 1  MISD1 {CI = 15P CO = 15P }
*CgdOv 1 2 3P
RG  20  2  75
RD  10  1  1.49
RDS  1  3  8.9MEG
DDS  3  1  DSUB
LS  30  3  7.5N
RLS 30 3 100
.MODEL DMOS NMOS (LEVEL=3 VMAX=12.4MEG THETA=.24 VTO=2.4
+ KP=.18 RS=10M IS=160F CGSO=30.4U)
.MODEL SW NMOS (LEVEL=3 VTO=0 KP=9.00M)
.MODEL DCGD D (CJO=18.3P M=.8 VJ=.21)
.MODEL DSUB D (IS=160F RS=0 VJ=.8 M=.4 CJO=65.8P TT=249N)
.ENDS
```

Depletion Capacitance Models

```
.SUBCKT MN6661B   10 20 30
*     TERMINALS:  D  G  S
M1   1  2  3  3  DMOS L=1U W=1U
X1 2 3  MISD2 {CI = 15P CO = 15P }
X2 2 1  MISD2 {CI = 15P CO = 15P }
*CgdOv 1 2 3P
RG  20  2  75
RD  10  1  1.49
RDS  1  3  8.9MEG
DDS  3  1  DSUB
LS  30  3  7.5N
RLS 30 3 100
.MODEL DMOS NMOS (LEVEL=3 VMAX=12.4MEG THETA=.24 VTO=2.4
+ KP=.18 RS=10M IS=160F CGSO=30.4U)
.MODEL SW NMOS (LEVEL=3 VTO=0 KP=9.00M)
.MODEL DCGD D (CJO=18.3P M=.8 VJ=.21)
.MODEL DSUB D (IS=160F RS=0 VJ=.8 M=.4 CJO=65.8P TT=249N)
.ENDS
```

Table 1, Shown above are three power Mosfet models using the standard, and new depletion and sigmoid capacitance models. The MISD1 and MISD2 subcircuits are shown in Figure 12. The standard power Mosfet model is implemented in the SPICEMOD modeling program which can create models from data sheet parameters.