Intusoft Newsletter

intusoft *Personal Computer Circuit Design Tools*

Intusoft Goes Macintosh ! !

Intusoft announces a price performance breakthrough with IsSpice/MAC and PreSpice/MAC, two new circuit analysis programs for the Macintosh. With IsSpice/MAC, an enhanced version of Spice (Berkeley compatible), you can now run Spice simulations on your Macintosh II, and Mac/SE computers. The full set of Spice primitives are supported and full AC, DC, and Transient analyses are available. Simulation status is continuously displayed allowing the user to interactively see how the simulation is progressing. IsSpice will take input from many common schematic entry programs including the soon to be released SpiceNet/MAC. PreSpice/MAC brings intusoft's extensive part libraries to the MAC. Included are models for common components like transistors, diodes, and opamps, as well as a wide variety of hard to model components. Utilities for model library usage and equation based modeling are included with a special netlist editor. The editor comes with complete Spice syntax help on-line making it easier than ever to generate IsSpice input files. The ICAPS menu drive system and the Macintosh interface greatly facilitate learning and the use of Spice, making it easier to use than most DOS based versions. Coupled with the ability to transport graphics and cut and paste between windows, data can be easily integrated into other MAC applications. IsSpice/MAC and PreSpice/MAC will be available for shipping September 1, 1989. IsSpice includes the popular Spice reference book "Simulating With Spice", a small model library, and several benchmark/tutorial circuits.

In This Issue

1 New IsSpice and PreSpice For The Macintosh

2 **Neural Networks and Spice (Part II)**

7 Modeling Lasers with IsSpice and PreSpice

10 Obtaining More Spice Models

SpiceNet and IntuScope Planned for the Mac

Versions of the popular schematic entry and post processing programs, SpiceNet and IntuScope, are planned for release later in the

Intusoft Goes MAC, *Continued on page 14-10*

Using SPICE For Neural Network Development
(Part II)

In this second part to our application note on Neural Network development we will review the basic motivation and SPICE models developed in part 1. Then we will extend the concepts to show how the learning process is initialized and how the learned weights can be remembered.

Motivations For Using SPICE

Neural Networks, based on the Perceptron with the Rumelhart back propagation algorithm, can be used to solve a wide range of pattern recognition problems. Software is available from SAIC [1] and others to model these networks using PC's with special accelerators. Designing a Neural Network requires several steps:

Definition of an Interconnect Architecture
Training the Network
Building a Special Purpose Circuit

Many projects never get to the last step; they are simply looking for new theoretical insights. In some ways, the computer has bred a new science; experimental mathematics, where new directions of study are first found fruitful by simulation. The motivation for SPICE simulation, on the other hand, is the actual circuit realization. Once the architecture and training methods are decided, then actual circuit realization involves a series of compromises and approximations to the theoretical equations. Moreover, there may be a sensitivity to initial conditions that could dramatically alter the computer predictions when actual hardware is substituted.

The SPICE simulation is used to make a more realistic estimate of the effects of these approximations. The models presented here are theoretically correct and can be used as a comparative baseline. Used in this manner, these models can evaluate the performance of new circuit designs. They can also be used to first gain confidence in the simulation by reproducing results developed elsewhere.

Review of the Basic Elements

Two SPICE subcircuits were developed, the Axon and Cell. While biological names are used; it should be noted that the intent is not to simulate biological activity. There are some remarkable behavioral similarities; however, the architectural similarities simply are

not there. The Axon, whose symbol is shown below, performs the weighted summation and propagates the training function backwards through the hidden layers. The Cell, shown in the same sketch, performs the threshold function and the back propagation window function. Notice that we don't have an analog of the biological synapse. Synaptic firing keeps the biological nets from being connected at the same time, turning it into a sampled data arrangement. In the Perceptron, the back propagation function is recursive, or synchronously clocked. The time quantization in the feedback path, rather than the feed forward path is a dramatic difference between this Neural Network and the biological function.

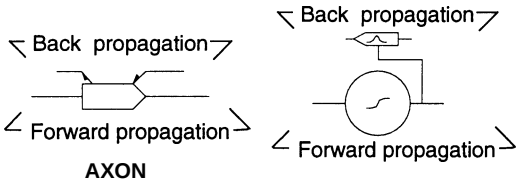


Figure 1, The Cell performs the threshold logic summation and the back propagation window function, while the Axon performs the interconnect and weight generation.

Recursive Summation - Memory

The training process requires that the summing weights be altered and remembered. In our last newsletter this was represented with Z transforms using the SPICE transmission line for the time delay. There are pros and cons to this approach. On the negative side is the long initialization time required for SPICE transient simulations when multiple transmission lines are present. On the other hand, the Z transform representation is a valid continuous time model enabling an AC analysis to be performed. This could be quite useful in evaluating stability as a function of the learning gain constant, initial conditions, or intermediate states.

SPICE transmission lines cannot be initialized. To get a non-zero initial weight, the initial weights are summed using the zero order term of the summing amplifier shown in Figure 2 of last months' newsletter. Another approach to forming the recursive summation is to use the switched capacitor circuit shown in Figure 2. The

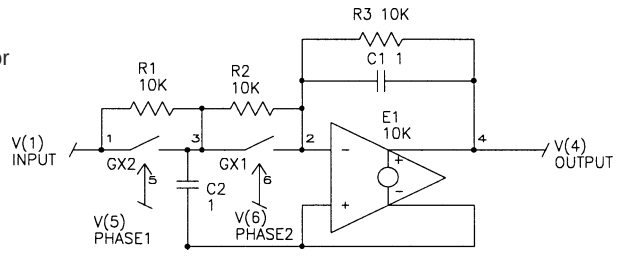


Figure 2, A Switched Capacitor arrangement performs the memory functions in place of the continuous time unit delay element

Recursive Summation - Memory *(continued)*

2 phase clocks are generated in the main circuit as shown in the exclusive-or example, Figure 3.

An Exclusive-Or Test Circuit

Testing of the models was performed using the exclusive-or circuit. This is a 2 layer circuit that has received much attention. The original Perceptron work laid dormant for 10 years because there was no back propagation algorithm; severely limiting the application of this network. The importance of the exclusive-or in pattern recognition lies in its close approximation of multiplication; a process fundamental to correlation or maximum likelihood detection. For example, the digital exclusive-or is used extensively as a phase detector in phase locked loops.

Axon and Cell Model Changes

Running the models developed last month exposed some errors and SPICE convergence problems. The Axon weight should have been multiplied by its input, and the input should not have been back propagated through the multiplier, GB. In the previous article we used a fixed threshold; for future work we will want to adaptively determine threshold along with the weights. To do this, the internal Cell threshold is made zero, and an extra Axon is added that is connected to a unit valued input. Both the EXP (exponential) and Divide models were changed from their library configurations to improve convergence. These changes limit the range of operation to reasonable values for the neural simulations. The new listings are shown in Table 1. Figure 4 shows the revised Axon model.

Exclusive-Or Test Results

We experimented with various gains and input-output values using a special purpose program. It was found that low gain (η<.5) circuits were slow to converge. When the gain was raised, the network would frequently get stuck with the output staying near zero or one. Reducing the output logic levels from the .1 and .9 values suggested by Rummelhart to .3 and .7 allowed a higher gain to be used. It is clear that the initial operation at high gain is chaotic (unstable); a biological analogy of electroshock therapy may be appropriate. The convergence time was reduced from the 600 training sets reported by Rummelhart to 140 with a gain of 7.5 by using output levels of .3 and .7.

We then ran an IsSPICE simulation to see if we could reproduce the results. Training of the network using SPICE was accomplished

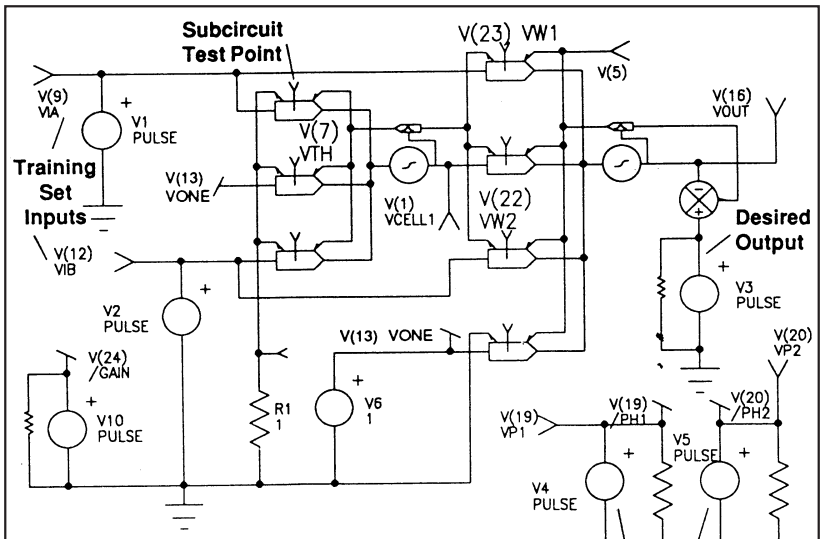


Figure 3, A 2 layer exclusive-or circuit using the ***intusoft*** Neural Network models. Axon subcircuit test points are used to report the learned weights.

by presenting a series of training sets to the network until the weights became stabilized. Each training set consisted of a complete set of binary inputs (00,01,10,11) along with the desired output response (.3,.7,.7,.3). The IsSpice and IntuScope data results for several of the weights is shown in Figure 5. Notice that the ripple period is at the training set frequency and its magnitude is proportional to the error. While the time to reach a stable solution is approximately the same, the path is different, suggesting the expected sensitivity to initial conditions for high gain. Even this simple circuit ran very slowly, suggesting that Spice should be used only for final circuit design validation and not for developing the network architecture.

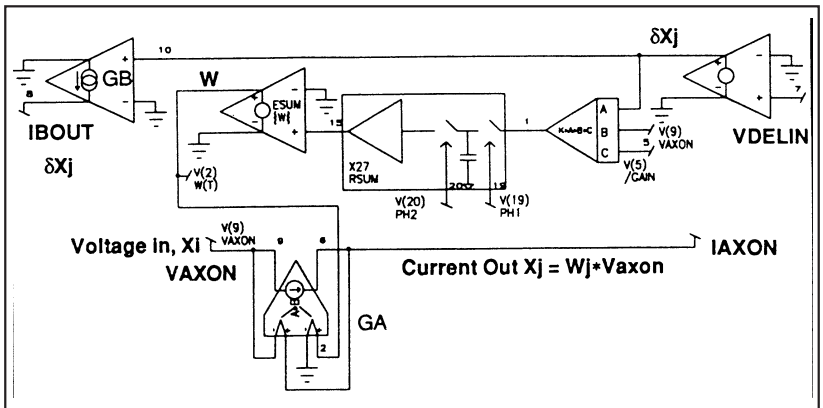


Figure 4, The updated Axon subcircuit using new SC integrator element

Bibliography

[1] SAIC: Science Applications International Corp, 4275 Campus Point Court, San Diego, Ca. 92121, (619)546-6290

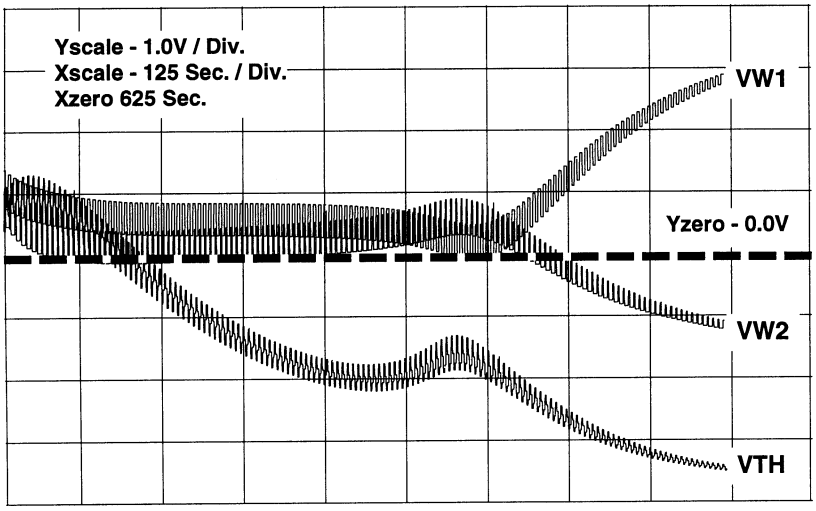


Figure 5 - The axon weights vs. time graph shows the settling of the weights and reduction in error as the number of training sets are processed. The simulation took approximately 9.33 hours and ran through 140 training sets using a Compaq Deskpro 386/20.

Table 1, SPICE Listings for Neural Net Elements

```
.SUBCKT CELL 1 2 11 3
* 2 Output Y or Xi
* 1 IJ - Current Summation
* 11 Idelin - Back Propagation In
* 3 Vdelout 0 Back Propagation Out
RX2 2 0 1K
RX9 9 0 1K
RX7 7 0 1K
RX8 8 0 1K
RX63 63 0 1K
RX12 12 0 1K
RX5 5 0 1K
RX6 6 0 1K
RX10 10 0 1K
RX3 3 0 1K
X5 8 7 2 DIVIDEA
EX6 5 0 9 0 0 -1
EX7 7 0 POLY(2) 6 0 8 0 0 1 1
V2 8 0 1
H1 9 0 VJ 1
VJ 1 0
EX12 12 0 POLY(2) 63 0 2 0 0 0 0 0 1
EX14 10 0 POLY(2) 2 0 8 0 0 -1 1
H2 63 0 V6 1
V6 11 0
EX26 3 0 POLY(2) 10 0 12 0 0 0 0 0 1
X4 5 6 EXPA
.ENDS
*Revised EXP Function with
*Limited Output Range
.SUBCKT EXPA 1 2
*Vout = EXP(Vin)
R 1 0 1K
E 3 0 1 0 .0258642
V 3 4
RD 4 0 1E8
D 4 0 DIODE
C1 4 0 .0025U IC=0
.MODEL DIODE D(IS=1U RS=10K )
I 0 4 -1U
H 2 0 V 1MEG
.ENDS
.SUBCKT DIVIDEA 1 2 4
*Revised Divide function with
*Limited Output Range
* V4 = V1 / V2
R1 1 0 1MEG
R2 2 0 1MEG
R4 4 0 1MEG
G1 0 3 1 0 1
C1 3 0 .001 IC=0
G2 3 0 POLY(2) 2 0 3 0 0 0 0 0 1
R3 3 0 100
E1 4 0 3 0 1
.ENDS
**********
.SUBCKT AXON 9 6 7 8 2 5 100 101
*5 Gain, Hidden Node
*2 Weight, Subcircuit Testpoint
*8 Ibout - Back Propagation Out
*9 Vaxon - Voltage Input
*6 Iaxon - Current Summation In
*7 Vdelin - Back Propagation Out
*100 - Clock, Phase 1
*101 - Clock, Phase 2
** Previous Z Transform UTD **
*ETD1 30 0 2 0 1
*RTI 30 0 1K
*T1 30 0 1 0 Z0=1 TD=1
*RTD1 1 0 1
*EX7 2 0 POLY(2) 1 0 3 0 {W} 1 1
****New SC Element
XSUM 3 30 100 101 RSUM
ESUM 2 0 30 0 {W} -1
RESUM 2 0 10K
*********
RX3 3 0 1K
RX10 10 0 1K
GA 9 6 POLY(2) 2 0 9 6 0 0 0 0 1
GB 0 8 10 0 1
E1 10 0 7 0 1
*3 Input Multiplier Correction
EX3 3 0 POLY(3) 10 0 5 0 9 0 (14 Zeros) 1
.ENDS
********
.SUBCKT RSUM 1 4 5 6
*Switched Capacitor Circuit for
*Recursive Summation
*Integrator, Gain -1/S
*1 Input
*4 Output
*5 Phase 1 Clk
*6 Phase 2 Clk
GX1 3 2 POLY(2) 3 2 6 0 0 0 0 0 10K
R1 1 3 10K
R2 3 2 10K
E1 4 0 0 2 10000
C1 2 4 1 IC=0
R3 2 4 10K
C2 3 0 1 IC=0
GX2 1 3 POLY(2) 1 3 5 0 0 0 0 0 10K
.ENDS
```

Modeling Lasers With IsSpice and PreSpice

Laser diodes are finding widespread use in consumer electronics and various communications applications where they are employed in fiber-optic transmission systems. The PreSpice model library contains a macro model for a GaAlAs laser diode, specifically, the Hitachi HL7801E. The model is adaptable to other laser diodes, however. Simulation of the output power of a laser with temperature is an important but difficult feature to simulate. Laser burnout can be caused by increases in optical output power which are accompanied by decreases in temperature. Therefore, a novel laser model was developed including a monitor diode that monitors the change in optical power with temperature, making it useful for a variety of real life simulations.

Laser Diode Operation

The laser diode subcircuit can be seen in Figure 1. The shaded areas indicate external components used only for testing. The diode D1 (DLASER) provides the VI characteristics of the laser and will exhibit variations in operating point vs. temperature as shown in Figure 2. You can use the .TEMP command in IsSpice to alter the laser's operating temperature. This diode may be used alone for analyses where other circuit effects are being investigated and the power output of the diode is not important. The energy gap, EG, saturation temperature exponent, XTI, and

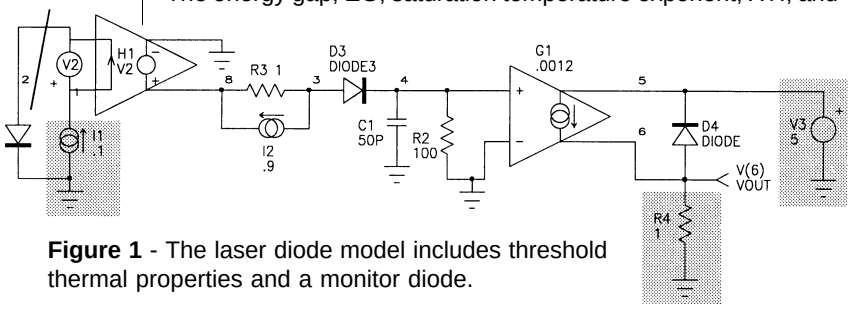


Figure 1 - The laser diode model includes threshold thermal properties and a monitor diode.

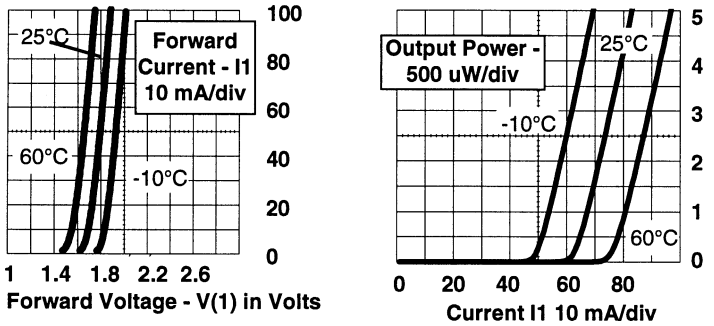


Figure 2 - Forward Voltage and Optical Output Power vs. Forward Current graphs show the laser diodes operating characteristics.

Laser Diode Operation *(continued)*

saturation current, IS are adjusted to provide the correct VI response vs. temperature. The rest of the subcircuit model is used to generate the proper monitor diode current. The normal SPICE diode produces an increase in current flow with temperature. To generate a negative temperature coefficient response for the optical power, the arrangement of H1, R3, I2, and the diode D3, is used. The H1 multiplying constant, R3 temperature coefficient, I2 current and diode D3 are all used to establish the monitor diode current vs. forward current response. The laser diode response is much faster when the diode is turned on then when it is turned off. C1 and R2 control the impedance of the model and thus the response performance. In this case, when the diode is in the low power region, the RC time constant sets the response time at 5NS. At higher bias levels the RC impedance becomes negligible and the response time of the diode decreases. The capacitance of the diode D4 is setup to mimic the transient performance of the real monitor diode. The conversion factor used in G1, .0012, converts the optical power analog into the proper scaling units for monitor diode current. With this configuration, the monitor diode can be used to control the biasing of the laser diode as seen in the SAMPLE.CIR simulation.

Using The Laser Diode Model

The model stored in the PRESPICE library is incorrect and requires some minor changes in order for it to be used. The netlist shown in Figure 4 is correct and describes the changes. In addition, in order to properly use SPICENET, the laser diode symbol must be altered. The current laser diode symbol incorrectly lists the pin order as (Laser diode) Anode, Cathode, (Monitor diode) Cathode, Anode. It should be (Laser Diode) Anode Cathode, (Monitor Diode) Anode, Cathode (as correctly stated in PRESPICE chapter 12).

To change the SPICENET Laser Symbol

1. First display and select the old laser diode symbol
2. "Break" the Symbol (F2 B)
3. Select F3 Pins (F3 P), to get the pin cursor
4. Place pins on the symbol corresponding to the order (Laser Diode) Anode Cathode, (Monitor Diode) Anode, Cathode.
5. Perform the "to Symbol" command (F2 S)
6. Fill in the subcircuit definition menu as follows:

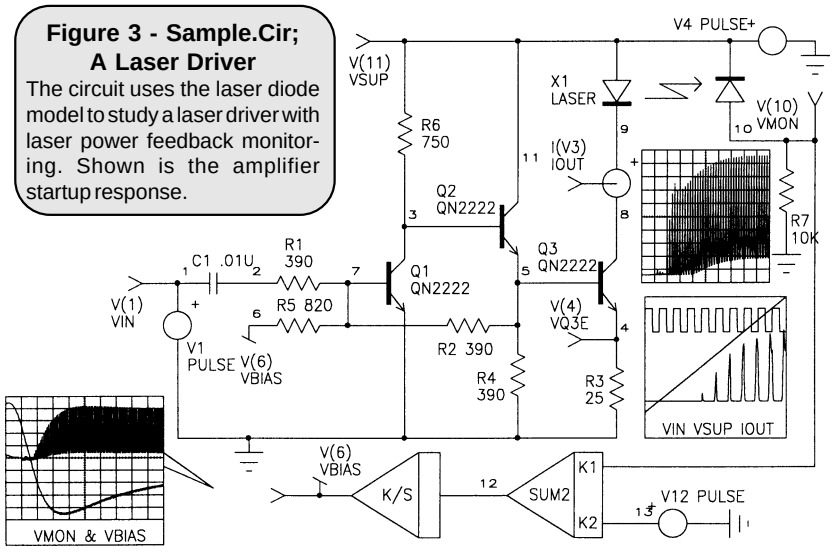
Name	Laser
Params	
Library	Device2

7. Select OK to finish and save the corrected symbol.

A Sample Laser Driver

For those of you who were wondering what the “SAMPLE.CIR" really does, its time to find out. The sample circuit, Figure 3, is actually a laser driver/amplifier. The monitor diode and bias circuitry is designed to prevent destruction of the laser in the event that one of the power supplies is lost or interrupted. Using the simple diode model the various electrical aspects of the circuit can be simulated. The schematic below shows the laser driver utilizing the complete laser diode model. The output power of the laser is summed and integrated with a bias voltage element. This voltage then controls the bias on the amplifier.

Figure 3 - Sample.Cir; A Laser Driver
The circuit uses the laser diode model to study a laser driver with laser power feedback monitoring. Shown is the amplifier startup response.



Laser Diode Models

```
.SUBCKT LASER  2  3 (11  10)
D1 1 3 DLASER
.MODEL DLASER D(RS=1.5 EG=2.8
+ XTI=3 N=1 IS=1E-30)
H1 7 0 V1 24
D2 8 9 DIODE3
.MODEL DIODE3 D(EG=0 )
R1 9 0 100
G1 10 11 9 0 .0012
D3 (11  10) DIODE
.MODEL DIODE D(CJO=5PF)
R3 7 8 1 TC=.0085
I2 8 7 .9
C1 9 0 50P
V1 2 1
.ENDS
```

SPICENET Laser Symbol Connections
(Laser Diode) Anode Cathode,
(Monitor Diode) Anode, Cathode

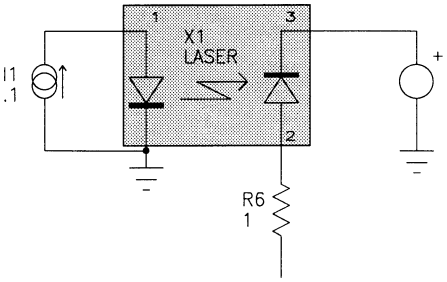


Figure 4 - Shown above is the correct subcircuit netlist for the laser diode model.The circled numbers have been reversed from the original listing in the DEVICE2.LIB file.

Intusoft Goes MAC, *Continued from page 14-1*

year. SpiceNet and IntuScope will be specially altered to take advantage of the special features in the MAC environment. Both programs will not be simply ported to the MAC as some other vendors have done with their PC versions. A newsletter describing the new programs will be released as soon as they are ready.

Obtaining Spice Models

Power Mosfet Models

Motorola, Literature Distribution Center 602-994-6561
Models for TMOS devices (Free)

LCF Enterprises, Paul Finman 805-499-5562
Models for RF Power Mosfets

Digital Logic Families and Input/Output Stages

(For simulating daughter card/backpanel interconnects)
RDC Consulting, Robert Cutler 213-456-5325

Opamps

Linear Technology, Marketing Dept. 408-432-1900
Models for the LT1013/1014 (Free)

PMI, Literature Department 408-562-7470
Models for the OP42, OP64, and OP400 (Free)

Harris, Gloria Simpson 407-724-3739
Models for the HA5190 and HA2539 (Free)

ComLinear, Sales Support Engineering 303-226-0500
Models for CLC series 400, 401, 501, 231, 220, 205, 206 (Free)

Texas Instruments, Literature 214-997-3389
Models for over 20 different devices (Free)

Modeling Vendors

Future CAD Inc, 408-279-3552

Silvaco Data Systems, Bob Vatter 408-988-2862