MACROMODELING WITH SPICE

Prentice-Hall International Editions

J. ALVIN CONNELLY
PYUNG CHOI

Macromodeling with SPICE

J. Alvin Connelly
Georgia Institute of Technology, Atlanta, Georgia

Pyung Choi
Kyungpook National University, Korea

江苏工业学院图书馆
藏书章

Prentice-Hall International, Inc.

This edition may be sold only in those countries to which
it is consigned by Prentice-Hall International. It is not to
be re-exported and it is not for sale in the U.S.A., Mexico,
or Canada.

© 1992 by Prentice-Hall, Inc.
A Simon & Schuster Company
Englewood Cliffs, New Jersey 07632

All rights reserved. No part of this book may be
reproduced, in any form or by any means,
without permission in writing from the publisher.

The author and publisher of this book have used their best efforts in preparing this book. These efforts include the development, research, and testing of the theories and programs to determine their effectiveness. The author and publisher make no warranty of any kind, expressed or implied, with regard to these programs or the documentation contained in this book. The author and publisher shall not be liable in any event for incidental or consequential damages in connection with, or arising out of, the furnishing, performance, or use of these programs.

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

ISBN 0-13-544537-X

Prentice-Hall International (UK) Limited, *London*
Prentice-Hall of Australia Pty. Limited, *Sydney*
Prentice-Hall Canada Inc., *Toronto*
Prentice-Hall Hispanoamericana, S.A., *Mexico*
Prentice-Hall of India Private Limited, *New Delhi*
Prentice-Hall of Japan, Inc., *Tokyo*
Simon & Schuster Asia Pte. Ltd., *Singapore*
Editora Prentice-Hall do Brasil, Ltda., *Rio de Janeiro*
Prentice-Hall, Inc., *Englewood Cliffs, New Jersey*

Foreword

Simulation of electronic circuits has replaced breadboarding as a means of verifying and analyzing the performance of complex circuits or systems. Unfortunately, simulation is only as good as the models used to represent the components of the circuit or system which is being simulated. The well-known and extensively applied simulation program SPICE was based on efficient and accurate models for electronic devices, including the pn junction diode, the bipolar junction transistor, and the field-effect transistor. Although these device models have become very sophisticated in their capability and efficiency, the number of devices contained in today's electronic circuit or system has far outstripped the ability of SPICE-like simulators.

One solution to the limitations of SPICE-like simulators is to raise the level of the model from the device level to a higher level. Two categories of higher level models are available. One uses SPICE primitives, and the other uses SPICE primitives plus other simulator attributes not found in most SPICE-like simulators. The first category, known as macromodeling, is the subject of this book. The second category is simply called higher level modeling or behavioral modeling. The advantage of macromodeling is that it is compatible with SPICE-like simulators and does not require that the user learn a new simulator.

Macromodels are not new. They were first introduced in 1974 by Boyle, Cohn, Pederson, and Solomon. As computational power increased, the demand for macromodels did not grow. However, the size and complexity of electronic circuits has rapidly outstripped the increases in computational power, resulting in a crisis in simulation when using device-level models. It is not unusual to spend days on a dedicated high-speed computer in an attempt to simulate a complex circuit or system. If the simulation time becomes unreasonable, then the complex circuits or systems are partitioned and simulated individually. This method has the disadvantage of not being able to model high-level behavior, such as stability or timing.

The crisis in simulation has led to a renewed interest in macromodeling. This timely book offers one solution to the inability to simulate large circuits and systems adequately. It is the first book devoted totally to the subject of macromodels. A methodology for the development of macromodels is given, along with many examples. A complete library of drop-in macromodels along with programs for automatically generating macromodels, is included. The application of these macromodels to design, analyze, and test a phase locked loop illustrates how macromodels can be used to obtain a simulation of a complex electronic system. The phase locked loop is impractical to simulate at the device level.

The authors have extensive experience in the development of macromodels and their application to complex circuits and systems. The macromodels presented have been applied and debugged in the classroom. The authors also include many useful techniques for optimizing the use of SPICE. The subject material in this book will be useful to both practicing engineers and students using SPICE. This book will be useful to those who know nothing about macromodeling as well as those who are familiar with macromodeling. It will give the reader an increased capability of using SPICE based on knowing the proper level of modeling to be used in a given simulation. The authors are to be commended on their effort and for making an important contribution to the field of electronic circuit and system simulation.

Phillip E. Allen,
Schlumberger Professor of Electrical Engineering
Georgia Institute of Technology
Atlanta, Georgia

此为试读，需要完整PDF请访问：www.ertongbook.com

Preface

This book will be useful for individuals having an intermediate to advanced knowledge of SPICE and its various derivative circuit analysis programs. Electrical engineering students enrolled in upper-level courses as well as graduate students represent the primary academic market. Practicing professional engineers engaged in circuit and system design and analysis also will find this book very useful. In both cases it is assumed that the reader has some previous experience with the SPICE circuit analysis program. This familiarity can come from having used SPICE under the guidance of the built-in instruction manual included on the program available from Berkeley. Other good references for the beginning SPICE user are *SPICE A Guide to Circuit Simulation and Analysis Using PSpice* by Paul W. Tuinenga (Prentice Hall, 1988), *Computer-Aided Circuit Analysis Using SPICE* by Walter Banzhaf (Prentice Hall, 1989), and *SPICE for Circuits and Electronics Using PSpice* by Muhammad H. Rashid (Prentice Hall, 1990). Many new textbooks have included chapters and/or appendices covering introductory aspects of SPICE utilization.

This book has the following unique aspects, which intermediate and advanced SPICE users will find very helpful. The first chapter defines the symbols and notation used throughout the book. Then subcircuits are explained and used to develop an operational amplifier macromodel. An example is given of how to use subcircuit macromodels in an active filter. Finally, some special tricks are presented for extending the built-in capabilities of SPICE by sweeping temperature and component values as well as modeling zener diodes. Chapter 2, "Macromodeling Methodology," describes the techniques that were used in the development of the "tool kit" of macromodels for many circuit and system functions. These techniques can be extended by readers to customize their particular macromodels by either modifying one of the given models from the later chapters, or creating a new macromodel using the techniques presented. This chapter guides the reader through the creation and development of an initial macromodel that approximates either experimental results or specification sheet parameters. Next, a simple, easy-to-use macromodeling technique for piecewise linear circuits is explained and illustrated. Behavioral-level macromodels are discussed and examples are given for both continuous-time (s-domain) analog filters and discrete-time (z-domain) sample data filters.

Chapter 3, "Drop-in Macromodels," is a collection of over thirty behavioral, functional, and commercial macromodels that have been developed, simulated, tested, and verified. The macromodels are called *drop-in* because they can easily be used singly or in groups to perform virtually any analog signal-processing function. These macromodels can be used much as functional blocks or modules of laboratory equipment are used to simulate and evaluate the performance of circuits and systems. Using these drop-in macromodels with various test signals allows a circuit and/or system designer rapidly and easily to determine gains, impedance levels, noise, distortion, spectral content, and the like as functions of time and frequency without solving equations or performing tedious calculations. Furthermore, the designer can simulate a large system using the macromodels to identify which block or stage is the critical one for limiting the overall performance. Then a more complete, device-level model can be inserted for this critical block while continuing to use the other drop-in macromodel functions. The tool kit of available drop-in behavioral macromodels includes the following: summer, multiplier, divider, squarer, ideal transformer, square root extractor, triangle-to-sine converter, phase shifter, integrator, differentiator, absolute value, peak detector, frequency multiplier, frequency divider, frequency adder and subtractor, phase detector, delay line, hysteresis (Schmitt trigger) simulator, sample-hold, pulse-width modulator, amplitude modulator, log amplifier, Nth root extractor, voltage and current noise sources including 1/f frequency effects, random noise generator, voltage controlled oscillator (for sine, square, and triangle waveforms), and a frequency counter. Functional macromodels are given for voltage comparators and operational amplifiers. The description given for each drop-in macromodel includes the functional block diagram, input–output equations, design equations and methodology, a circuit diagram, the SPICE input control file, and at least one practical example of each macromodel with actual simulation results.

Chapter 4 explains how several of the behavioral drop-in macromodels are interconnected to form an APLL (Analog Phase Locked Loop). A phase detector, VCO (voltage controlled oscillator), and low-pass filter are used to produce the complete APLL macromodel. The functional block macromodels are verified with simulations for various test signals.

In Chapter 5, an APLL system is designed to model the NE565 integrated circuit phase locked loop. Simulation results are compared with experimental measurements to show that the PLL accurately and efficiently models system operation in all four regions of operation: unlocked state, locked mode, acquisition, and loss of lock regions. To verify its practicality, a FSK (frequency shift key) network is simulated.

Chapter 6 describes a computer program written in BASIC that permits the user to select from a menu a macromodel to be constructed . The menu contains nine macromodels: analog phase locked loop, continuous time domain transfer function, peak detector, sample-and-hold circuit, Schmitt trigger, noise source, piecewise linear circuit, discrete time domain transfer function, and an amplifier. Once the user selects the desired model, the program prompts for the specific performance information required. Once all questions have been answered, the program creates a SPICE subcircuit macromodel file having the required topology, elements, and component values. Diskettes are included so readers can rapidly, accurately, and automatically generate macromodels using an IBM or compatible personal computer. Chapter 6 provides a complete listing for all nine macromodels which have been programmed for automatic computer generation plus the menu program. The chapter contains a complete listing of the source programs for these nine models. Users having some familiarity with BASIC programming can easily adapt and modify these programs to include many more of the models covered in the book or can extend these computer programs to their own specialized macromodels.

Any serious SPICE user no doubt has experienced frustration with convergence problems in SPICE. Appendix A explains an orderly procedure to follow for correcting the bias point and transient analysis convergence problems often encountered. Also, a listing of error messages and their interpretation is given.

The PROBE feature available in PSpice versions from MicroSim Corporation is a very powerful tool that allows quick and accurate determination of many key circuit performance indexes. The regular mathematical functions in PROBE are explained in Appendix B along with some special graphical tricks for plotting key asymptotic lines and curves. Another useful feature of the PROBE command is the ability to merge two or more PROBE plots from two separate simulations into a single graph. In short, this appendix shows how to extend the capabilities of the postprocessor PROBE function.

The last section, Appendix C, is intended as a quick reference for the four parts of the SPICE program—that is, the elements, devices, subcircuits, and control statements. Figures and examples are provided to help the reader quickly locate and specify the correct format for each of these parts of SPICE.

Many of the macromodels presented in this book have been used in a variety of classes at Georgia Tech. The s- and z- domain macromodels have enabled students studying operational amplifiers and active and switched-capacitor filters to simulate easily complicated and involved behavioral transfer functions. Graduate students in a Frequency Synthesizers course have been able to simulate and better understand how a phase locked loop functions in all four regions of operation. Furthermore, they have been able easily to change the type and characteristics of the low-pass filter and to determine the important effects this has on the loop dynamics, the lock and tracking ranges, and the overall stability. Students in a Low Noise Electronic Design course have been able to model amplifier noise performance using the included macromodels. Also, they can quickly determine the noise bandwidth and the equivalent input and output noise levels in their circuits using the integration capabilities of PROBE with PSpice. The multipliers, mixers, phase shifters, and so on, have been used to demonstrate various modulation techniques in communication courses. It is hoped and expected that there will be virtually no end to the applications for these macromodeling tools in the hands of clever students, professors, and practicing engineers.

ACKNOWLEDGMENTS

It is not possible to enumerate all those whose assistance contributed to completing this book, but there are some who deserve special acknowledgment. We wish to express thanks to the many students who helped in the development of this book. Special thanks go to Ahmad Dowlatabadi, Fred Pruner, Katherine Taylor, Jim Caravella, Tim Holman, Mark Thrower, and Craig Hollabaugh. Appreciation is expressed to fellow faculty members at the Georgia Institute of Technology, notably Doctors P. E. Allen, M. Brooke, G. Casinovi, R. K. Feeney, D. R. Hertling, and W. E. Sayle. The encouragement and assistance of many people, especially Dr. Roger P. Webb, was essential to the success of the book. We also recognize the help and cooperation of Doctors E. J. Kennedy of the University of Tennessee and Eugene Chenette of the University of Florida.

The guidance and leadership of our editor, Elizabeth Kaster, is gratefully acknowledged, and the suggestions of the many reviewers are very much appreciated, especially Doctors Paul D. Stigall of the University of Missouri (Rolla) and James E. Morris of SUNY, Binghamtom. We also appreciate the thoroughness of our production editor, Irwin Zucker. The financial support of the Semiconductor Research Corporation is certainly recognized.

We gratefully acknowledge the support, patience, and understanding of our wives, Mary Nelle and Yeong Ae.

J. Alvin Connelly
Pyung Choi

Contents

此为试读，需要完整PDF请访问： www.ertongbook.com

Appendices

1

Introduction

1.1 HISTORY OF SPICE

SPICE is a powerful, general-purpose circuit analysis program that simulates analog circuits. SPICE (an acronym for Simulation Program with Integrated Circuit Emphasis) is by far the most popular analog circuit simulation program being used today by both practicing engineers and students. SPICE was developed by the Integrated Circuit Group of Electronics Research Laboratory and the Department of Electrical Engineering and Computer Sciences at the University of California, Berkeley, California in the late 1960s and was released to the public in 1972.[1] The person credited with originally developing SPICE is Dr. Lawrence Nagel, whose Ph.D. thesis describes the algorithms and numerical methods used in SPICE.

Over the years, SPICE has gone through many upgrades. The most significant came with SPICE2, in which the kernel algorithms were upgraded to support advanced integrated system methods, many of which relate to IC performance. SPICE2 has virtually replaced SPICE1 as the SPICE choice. SPICE2 has been ported to numerous types of mainframe computers and personal computers, and to various operating systems. Because the development of SPICE2 was supported using public funds, this software is in the public domain, which means it may be used freely by any U.S. citizen. SPICE2 has become the industry standard and is simply referred to as SPICE. It is a large (over 17,000 lines of FORTRAN source code), powerful, and extremely versatile industry-standard program for circuit analysis and IC design.

Recently SPICE2 was upgraded to SPICE3. In the newer version, the program was converted from FORTRAN to C for easier portability. Also, several devices were added to the program library, such as a varactor, semiconductor resistor, and lossy RC transmission line models. However, the kernel algorithms were not changed, and the added components are not too significant because SPICE2 can simulate all the devices built into SPICE3 using external device modeling techniques.

Today there are more than 35 SPICE derivative programs, many known by associated acronyms such as HSPICE and RAD-SPICE (from Meta-Software), IG-SPICE (from A. B. Associates), I-SPICE from NCSS timesharing), PSpice (from MicroSim), IS_Spice (from Intusoft), SLICE (from Harris), ADVICE (from AT&T Bell Laboratories), Precise (from Electronic Engineering Software), and ASPEC (from Control Data Corporation).[2] In 1984 MicroSim introduced two versions of SPICE, called PSpice, that run on an IBM personal computer. The educational or student version of PSpice is distributed free by MicroSim. This has made SPICE available to vast numbers of students and has caused a rethinking of the way classes and laboratories are being taught. The student version of PSpice that is PC-based is limited to circuits with approximately 10 or fewer transistors. However, the professional (or production) version can simulate a circuit with up to 200 bipolar transistors or 150 MOSFETs.

SPICE and other simulation programs will be around and will be widely used for many years to come. Students will find SPICE an important tool for learning circuit analysis and design, and for testing electronic circuits in ways not easily done in most college laboratories.

1.2 SPICE CAPABILITIES

SPICE contains built-in models for passive elements such as resistors, capacitors, inductors, and transmission lines; semiconductor devices such as diode, bipolar transistor, junction field-effect transistor (JFET), and metal oxide semiconductor field-effect transistor (MOSFET); independent voltage and current sources, and dependent voltage and current sources including linear and nonlinear types (VCVS, VCCS, CCVS, and CCCS). By including control lines in an input file, SPICE can be made to perform many kinds of analyses of a circuit, as follows:

1. Nonlinear DC analysis, which determines the DC operating point of the circuit
2. Linear small signal analysis, which calculates the frequency response of the circuit
3. Transient analysis, which determines the response as a function of time over a specified time interval
4. Small signal DC transfer function analysis of a circuit from a specified input to a specified output (input resistance, output resistance, and transfer function)
5. DC small signal sensitivity analysis of one or more specified output variables with respect to every parameter in the circuit
6. Distortion analysis with AC analysis
7. Noise analysis with AC analysis, which determines the equivalent output and input noise at specified output and input nodes
8. Fourier analysis of an output variable when done with a transient analysis
9. Temperature analysis[3]

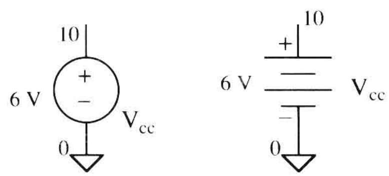
1.3 LIST OF SYMBOLS AND NOTATION

Independent sources. General form for voltage and current sources

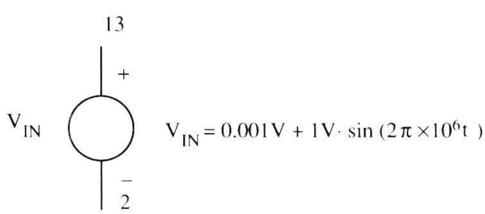
```
Vxxxxxxx  N+  N-  <<DC>  DC/Tran Value>
                  <<AC> <ACMAG<AC Phase>>>

Iyyyyyyy  N+  N-  <<DC>  DC/Tran Value>
                  <<AC> <ACMAG<AC Phase>>>
```

Here are some examples.

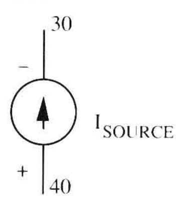
```
VCC  10  0  DC  6
```




```
VIN  13  2  DC  0.001  AC  1  SIN(0 1 1MEG)
```

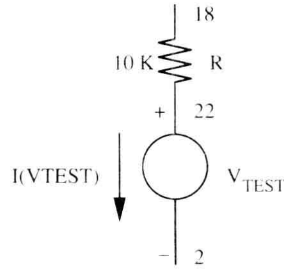



```
ISOURCE  40  30  DC  1A  AC  1MA
```



Note: Positive current is assumed to flow from the positive node, through the source, to the negative node. A current source of a positive value will force current to flow out of the N+ node, through the source, and into the N− node. Voltage sources, in addition to being used for circuit excitation, are the ammeters for SPICE. Zero-valued voltage sources may be inserted into the circuit for the purpose of measuring current.

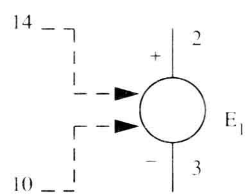
They will have no effect on circuit operation because they represent short circuits. To find the AC current through a sample resistor use the following commands:




```
VTEST   22   2   DC   0V   AC 0V
.PRINT   AC   I(VTEST)
```

Dependent sources. Linear Voltage Controlled Voltage Source (VCVS)

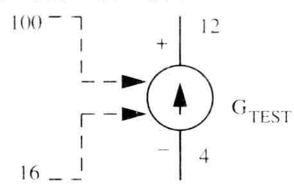
```
Exxxxxxx   N+   N-   NC+   NC-   VALUE
E1   2   3   14   10   1.5
```



$E_1 = 1.5[V(14) - V(10)] = 1.5[V(14,10)]$

Linear Voltage Controlled Current Source (VCCS)

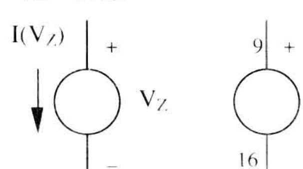
```
Gxxxxxxx   N+   N-   NC+   NC-   VALUE
GTEST   4   12   100   16   8.4
```



$G_{TEST} = 8.4[V(100) - V(16)]$
$= 8.4[V(100,16)]$

Linear Current Controlled Voltage Source (CCVS)

```
Hxxxxxxx   N+   N-   VNAME   VALUE
HSOURCE   9 16   VZ   0.5K
```



$H_{SOURCE} = 0.5\ K\ I(V_Z)$

此为试读，需要完整PDF请访问：www.ertongbook.com