COMPUTER-AIDED CIRCUIT ANALYSIS USING SPICE

WALTER BANZHAF

COMPUTER-AIDED CIRCUIT ANALYSIS USING SPICE

WALTER BANZHAF

Ward College of Technology
University of Hartford
West Hartford, Connecticut

Prentice Hall, Englewood Cliffs, New Jersey 07632

Library of Congress Cataloging-in-Publication Data

Banzhaf, Walter.
Computer-aided circuit analysis using SPICE / Walter Banzhaf.
p. cm.
Bibliography: p.
Includes index.
ISBN 0-13-162579-9
1. Electric circuit analysis—Data processing. I. Title.
TK454.B33 1989
621.3815'3'0285—dc19 88-22459
CIP

Editorial/production supervision: *Mary Carnis*
Cover design: *Wanda Lubelska*
Manufacturing buyer: *Robert Anderson*
Page layout: *Karen Noferi*

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

ISBN 0-13-162579-9

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*

PREFACE

This book was written with the express purpose of making it possible (and easy) for anyone in the electronics field to start using SPICE to analyze electric or electronic circuits in a very short time. SPICE is simple to use when you have learned its rules and syntax. Learning those things was challenging for me and for others I spoke with, since a text on the topic didn't exist. My engineering technology students have been using SPICE on a mainframe computer for years; they were able to master its use by following examples which they received in class. Each example contained a schematic diagram of the circuit to be analyzed, the input file which described the circuit to SPICE, and the output file created by SPICE with a discussion of its contents. After making a few of the classical errors (e.g. typing a letter "O" when the digit "0" (zero) was intended), most students were quite comfortable doing analyses completely on their own.

During what was to be a leisurely one-semester sabbatical leave, I had planned to write a small manual on SPICE for use by students at Ward College of Technology of the University of Hartford. Discussions with Greg Burnell, Editor-in-Chief, Electronic Technology, at Prentice-Hall convinced me that there was a need for a text on circuit analysis using SPICE that would appeal to two types of readers.

The first type is engineering and engineering technology students, for whom this book could be a supplemental text for any circuits or electronics course. In addition to learning classical methods of circuit analysis and design, they also need to learn how to perform circuit analysis on computers in order to be prepared for industry. One added benefit of simulating circuit behavior on a computer is that students can gain great insight into circuit behavior which would be otherwise unattainable due to the sheer tedium of the mathematical operations necessary. For example, a student having spent the better part of an hour analyzing a transistor amplifier with a BJT whose beta is 80 is not likely to repeat the exercise with the beta changed to 40 just to satisfy intellectual curiosity. With SPICE, repeating the analysis would take only a few seconds. Also, by using computer simulations to check the work they do by hand students gain confidence in their own analytical abilities.

The second type includes practicing engineers, technologists and technicians who are already competent at circuit analysis and design and want to learn how to use SPICE. This book will help them to use the computer to confirm results obtained by hand, to save time in the laboratory and to go from a concept to a working prototype with minimum delay. Although SPICE (or variations of it) can be found in a large number of universities and industries, it is still not easy to learn to use it due to the lack of a text with sufficient clear examples.

Thus, to answer a clear need, the book you are looking at was developed (the sabbatical was not altogether leisurely). It has a large number of examples which illustrate nearly all the capabilities of SPICE. A first year student may be able to use only the DC and AC analysis examples at first. As that student's knowledge of electronics advances, more and more of the book will be appropriate and useful. Conversely, a practicing engineer, technologist or technician may wish to jump to the more advanced topics right away. It is advisable for that reader at least to skim through the preliminary chapters to see first the pitfalls (not many, but potentially fatal) and second the extensive capabilities of SPICE.

Chapter 1 is an introduction to computer-aided circuit analysis, including a brief history and some insight into how it is done by a computer. The history of SPICE is covered in Chapter 2, emphasizing why such a tool was needed for integrated circuit development.

The specific steps needed to run SPICE are presented in Chapter 3, including the pitfalls you may encounter when creating input files. Chapter 3 also shows how to describe resistors, capacitors, self and mutual inductors, and independent voltage and current sources to SPICE. The actual "how to do it" process is shown in Chapter 4, which includes four example analyses which illustrate the capabilities of SPICE. After reading Chapter 4 the reader should be able to begin using SPICE (assuming you have access to a mainframe computer with SPICE on it, or to a personal computer and a PC version of SPICE).

Linear dependent (controlled) sources are explained in Chapter 5. Chapter 6 is an examination of ten kinds of analysis of which SPICE is capable, containing examples of DC, AC, transient, operating point, transfer function, sensitivity, distortion, noise, Fourier and temperature analysis. How to convince SPICE to make graphs and tables of analysis results is covered in Chapter 7.

The large topic of semiconductors and their SPICE models is shared between two chapters. How to describe semiconductors to SPICE is covered in Chapter 8, while an introduction to changing semiconductor models follows in Chapter 11.

Chapter 9 presents the use of subcircuits in SPICE input files. Transmission lines are examined in Chapter 10, including examples showing time-domain reflectometry and steady-state AC analysis.

The part of the book that ties it all together, Chapter 12, presents 30 examples of SPICE analysis. Each example contains a schematic diagram of a circuit to be analyzed, an input file submitted to SPICE for analysis, the results in an output file created by SPICE and a discussion of key points illustrated by the example.

Five appendices present other SPICE features, how to model transformers and switches, nonlinear (polynomial) dependent sources, a bibliography and sources of SPICE-based circuit analysis software.

Each example in the text can be considered to be a practice problem, and it is expected that readers will have no shortage of circuits to analyze from courses being taken (if students) or from daily work (if engineers, technologists or technicians). Only by applying SPICE to your own circuits will you achieve a mastery of SPICE for your purposes.

I am grateful to all those at the University of Hartford who supported this effort and helped make it happen. This

certainly includes my students from whom I learn so much about learning. Greg Burnell and Mary Carnis of Prentice Hall have earned my respect and gratitude for their gentle but effective guidance throughout the writing of the text. The comments provided by James Morris of T.J. Watson School of S.U.N.Y., David O'Brien of Wentworth Institute of Technology, and Carl Zimmer of Arizona State University were most helpful. Without the patience and help of my wife, Mattie (an eagle-eyed proofreader), and my children Amy and Jeremy I could not have written this book. Thank you all.

Walter Banzhaf
Ward College of Technology
University of Hartford
West Hartford, Connecticut

CONTENTS

PREFACE ix

1 INTRODUCTION TO COMPUTER-AIDED CIRCUIT ANALYSIS 1

1.1 The Need for Computer-Aided Circuit Analysis 1

1.2 ECAP Is Developed 2

1.3 After ECAP 2

1.4 How It's Done 3

Chapter Summary, 4

2 HISTORY OF AND INTRODUCTION TO SPICE 5

2.1 Acknowledgment 5

2.2 The Need for CAD with Integrated Circuits 5

2.3 SPICE Background Information 7

Chapter Summary, 8

3 **GROUND RULES OF SPICE & BASIC ELEMENT LINE RULES** 9

3.1 Introduction 9

3.2 Process for Using SPICE 10

3.3 Rigid Rules, Handy Hints and Useful Things to Remember 13

3.4 Rules for Element Lines (R, C, L & K) 15

3.5 Independent Sources 19

Chapter Summary, 29

4 **FIRST ATTEMPTS AT ANALYSIS, WITH SAMPLE CIRCUITS** 30

4.1 Introduction 30

4.2 The First Circuit - 3 Resistors, 1 Battery 30

4.3 How to Run SPICE on a Mainframe Computer 34

4.4 How to Avoid Wasting Paper 35

4.5 The Second Circuit - 3 Resistors, Two Batteries 36

4.6 The Third Circuit - Parallel Resonant Tank, AC Analysis 41

4.7 The Fourth Circuit - RC Circuit with Pulse Input 45

Chapter Summary, 49

Problems 49

5 **LINEAR DEPENDENT SOURCES** 51

5.1 Introduction 51

5.2 Linear VCCS 52

5.3 Linear VCVS 54

5.4 Linear CCCS 56

5.5 Linear CCVS 58

Chapter Summary, 59

6 TYPES OF ANALYSES 60

6.1 Introduction 60

6.2 DC Analyis Using .DC Control Line 62

6.3 AC Analysis Using .AC Control Line 62

6.4 Transient Analysis Using .TRAN Control Line 64

6.5 Operating Point Analysis Using .OP Control Line 65

6.6 Transfer Function Analysis Using .TF Control Line 66

6.7 Sensitivity Analysis Using .SENS Control Line 67

6.8 Distortion Analysis Using .DISTO Control Line 67

6.9 Noise Analysis Using .NOISE Control Line 68

6.10 Fourier Analysis Using .FOUR Control Line 69

6.11 Analysis at Different Temperatures Using .TEMP Control Line 70

Chapter Summary, 71

Problems 71

7 FORMATTING SPICE OUTPUT: PRINTING TABLES AND GRAPHS 77

7.1 Introduction 77

7.2 Use of the .PRINT Control Line 78

7.3 Use of the .PLOT Control Line 81

7.4 Interpreting SPICE Output Information 84

Chapter Summary, 95

8 SEMICONDUCTORS IN SPICE **96**

8.1 Introduction 96

8.2 Describing Diodes to SPICE 97

8.3 Describing Bipolar Junction Transistors to SPICE 100

8.4 Describing Junction Field-Effect Transistors to SPICE 103

8.5 Describing MOS Field-Effect Transistors to SPICE 105

Chapter Summary, 110

9 SUBCIRCUITS **111**

9.1 The Need for Subcircuits in SPICE Input Files 111

9.2 How to Use Subcircuits 112

Chapter Summary, 118

10 TRANSMISSION LINES **119**

10.1 Transmission Lines Are Lossless? 119

10.2 Transmission Line Element Line 120

10.3 Equivalency Between Transmission Delay and Electrical Length 121

10.4 Sample Transmission Line Problems 123

Chapter Summary, 135

11 HOW TO CHANGE SEMICONDUCTOR MODELS **141**

11.1 Why Change Semiconductor Models? 141

11.2 Changing the SPICE Diode Model 142

11.3 Changing SPICE Transistor Models 146

Chapter Summary, 150

12 SAMPLE CIRCUITS **152**

12.1 Why Sample Circuits? 152

12.2 No Analysis Specified 153

12.3 DC Analysis 160

12.4 AC Analysis 166

12.5 Transient Analysis 193

12.6 .TEMP, .FOUR, .TF, .OP, .DISTO, .SENS & .NOISE CONTROL LINES 223

Chapter Summary, 258

ANSWERS TO PROBLEMS 260

APPENDIX A: OTHER CONTROL LINES 267

APPENDIX B: TRANSFORMERS AND SWITCHES 274

APPENDIX C: NONLINEAR (POLYNOMIAL) DEPENDENT SOURCES 287

APPENDIX D: BIBLIOGRAPHY 296

APPENDIX E: SOURCES OF SPICE-BASED CIRCUIT ANALYSIS SOFTWARE 299

INDEX 303

Chapter 1

INTRODUCTION TO COMPUTER-AIDED CIRCUIT ANALYSIS

1.1 THE NEED FOR COMPUTER-AIDED CIRCUIT ANALYSIS

Circuit analysis is a necessary part of circuit design. Once a design for a circuit has been determined, the soundness of the design must be tested to ensure that the circuit does indeed perform as required. Often this involves testing for DC operating point and performance with signal applied, over a range of DC supply voltages, input signal levels, and temperatures. The time-honored way to do this was to build a prototype of the circuit, send it off to the laboratory, and invest large amounts of time and money in putting the circuit through its paces, in the hope that it would perform as desired.

Even if it did, one did not know how the circuit would perform with active devices whose parameters ranged considerably from a nominal value. For example, a small signal transistor typically has a DC current gain, or beta, around 100. The values of beta one might encounter in a lot of acceptable transistors shipped by a reputable manufacturer could range from 40 to 250. A prudent circuit designer would have to test the circuit operation by building many test circuits, using transistors at both ends of the beta range, and putting all of them through their paces. If there were a way to simulate accurately and with confidence the performance of electronic circuits without having to build them, the development cost and time to bring a new circuit into production could be reduced. This is why a great deal of effort was put into simulation of circuit performance by computers.

1.2 ECAP IS DEVELOPED

Computer-aided circuit analysis first became popular in the mid 1960s when the computer program ECAP (Electric Circuit Analysis Program) was developed and made available. This software for mainframe computers (since that's all there were at the time) made it possible for an engineer to analyze an electronic circuit without having to write equations defining the circuit. ECAP was a major accomplishment by programmers at IBM Corporation. Once the program user described the circuit using nodal notation, the program itself wrote the circuit equations and solved them.

Prior to the development of ECAP, digital computers had been in use for years for solving circuit equations. But the tedium involved in writing those equations for all but the most simple circuits was substantial. And once the equations were written by an engineer and solved by a computer, any change to the original circuit design necessitated the rewriting of the circuit equations.

1.3 AFTER ECAP

ECAP was followed by several similar programs, which offered a variety of improvements to ECAP. Among these were SPECTRE, TRAC, NET and CIRCUS. This book is about SPICE, which is the most popular computer program in the world today for predicting the behavior of electronic circuits.

It is not necessary to understand the algorithms and mathematical techniques that are utilized by SPICE to analyze circuits in order to be successful at solving circuit problems with SPICE, any more than one must have a working knowledge of internal combustion engines to drive a car. However, it is vitally important for anyone using SPICE (or any other computer analysis/design program) to be a competent practitioner in the field. Only in that way can one check the correctness and reasonableness of the answers by estimation using sound engineering judgement and/or by working a sample problem through by conventional means.

Also, for some of the trickier kinds of circuit analysis involving circuits which have positive feedback (oscillators, Schmitt triggers) or when non-linear circuit elements are present, a little insight into how SPICE does things can be useful.

1.4 HOW IT'S DONE

If you do want to know how circuit analysis is performed by a computer, a brief look at some of the techniques used by SPICE may give you some insight into the process. A vastly more detailed reference is L. Nagel's thesis, "SPICE2: A Computer Program to Simulate Semiconductor Circuits." Information on ordering a copy can be found in Appendix E. A condensed explanation of how SPICE works is nicely presented by W. Blume in an article in BYTE magazine, referenced in Appendix D.

Circuits are described to SPICE by use of an input file, which lists each circuit element (resistor, capacitor, inductor, voltage and/or current source, semiconductor device) and indicates how each is connected using node numbers. In addition, there are lines in the input file which designate the frequency of sources, temperature, the types of analyses to be done and how the analysis results are to be presented.

SPICE reads the input file and figures out all the circuit elements connected to each node. It then uses Kirchhoff's current law to create a system of equations for the circuit, where the voltages at each node are the unknowns, and the admittance (inverse of impedance) of each branch connecting two nodes are the known quantities. This is a form of nodal analysis. Of course, before this can be accomplished SPICE has to know the model for each semiconductor in the circuit (it does) and determine the specific values for each parameter in the model.

What then results is a system of simultaneous equations, whose size is determined by the number of nodes and circuit elements in the circuit; this group of equations is made into an admittance matrix. In order to solve this matrix, a technique called the Newton-Raphson method is used. Nagel's thesis not only describes how all the algorithms of SPICE work but also explains alternate methods which were considered, tried and ultimately rejected. While it makes for interesting reading, as stated earlier it is not necessary to have a particle of insight into the analysis methods used, especially for the person just learning SPICE.

CHAPTER SUMMARY

Computer-aided circuit analysis saves time and money in the process of circuit design.

ECAP, developed by IBM Corporation, was the first widely-available computer program for analyzing electronic circuits. It wrote and solved the equations for a circuit described to it by nodal notation. ECAP ran on mainframe computers, since personal computers did not exist.

A large number of programs were developed after ECAP; the most popular is SPICE.

You can use SPICE to predict circuit behavior without having an understanding of how the algorithms in the program function. However, it is important to be competent in electronic circuit analysis in order to verify results of SPICE.

Chapter 2

HISTORY OF AND INTRODUCTION TO SPICE

2.1 ACKNOWLEDGMENT

SPICE was developed by the CAD (Computer-Aided Design) Group of the University of California, Berkeley, and is the sole property of the Regents of the University of California. The author is grateful to the University of California for granting permission to use excerpts from its Electronics Research Laboratory memoranda in this book.

2.2 THE NEED FOR CAD WITH INTEGRATED CIRCUITS

During the early 1970s, the number of active devices on integrated circuits (ICs) being developed grew dramatically; SSI (small-scale integration, characterized by about 20 transistors on each IC) led to MSI (medium-scale integration, with about 100 transistors per IC). (Specific quantitative definitions of SSI, MSI, LSI and VLSI vary; the transistor counts shown here are representative.) Later came LSI (large-scale integration) and VLSI (very large-scale integration), in which there were literally thousands of transistors