

Gheorghe Asachi Technical University of Iasi



Faculty of Electronics, Telecommunications and Information Technology

Title of Discipline:

Computer-Aided Analysis of Electronic Circuits

Laboratory 2

Bachelor : Telecommunication Technologies and Systems

Year of Study: 2

Lab instructors: Iolanda Alecsandrescu Felix Diaconu



TESTAS ES.

www.study.tuiasi.ro

www.learning.tuiasi.ro

Computer-Aided Analysis of Electronic Circuits

Laboratory 2 PSpice simulator – basic concepts

What is SPICE?

SPICE is a general-purpose circuit-simulation program for nonlinear DC, nonlinear transient and linear AC analysis. As outlined above, it solves the network equations for the node voltages. The program is equally suited to solves linear as well as nonlinear electrical circuits.

CAAEC Laboratory

What is SPICE?

Circuits can contain:

- resistors, capacitors, inductors, mutual inductors;
- □ independent voltage and current sources;
- dependent voltage and current sources;
- □ transmission lines;
- □ the most common semiconductor devices:
 - diodes
 - bipolar junction transistors (BJTs)
 - junction-field effect transistors (JETs)
 - metal-oxide-semiconductor field effect transistors (MOSFETs)
 - metal-semiconductor FETs GaAs (MESFETs)
 - special devices and integrated circuits

Any general-purpose circuit simulation program must to give the following three basic solutions: bias point (OP), frequency response (AC) and transient response.

DC/AC analysis

- **The DC analysis** part of the program computes the bias point of the circuit with capacitors disconnected and inductors short-circuited. SPICE uses iterations to solve the nonlinear network equations; nonlinearities are due mainly to the nonlinear current-voltage (I-V) characteristics of semiconductor devices.
- The AC analysis mode computes the complex values of the node voltages of a linear circuit as a function of the frequency of a sinusoidal signal applied at the input. For nonlinear circuits, such as transistor circuits, this type of analysis requires the small-signal assumption; that is, the amplitude of the excitation sources are assumed to be small compared to the thermal voltage for BJTs ($V_{in} \ll V_{th} = 25$ mV, for small distortions). Only under this assumption can the nonlinear circuit be replaced by its linearized equivalent around the DC bias point.

Transient analysis

• The transient analysis mode computes the voltage waveforms at each node of the circuit as a function of time. This is a large-signal analysis: no restriction is put on the amplitude of the input signal. Thus the nonlinear characteristics of semiconductor devices are taken into account.

More types of analysis, associated with the above three basic simulation modes, are available in (P)SPICE.

Why use SPICE?

SPICE is a great tool for learning electronics. You can increase your understanding of circuits as you play and tinker with them.

- **Experiment!** Modify the circuit and see what happens! How long does it take? Change a resistor value and see the effect on a circuit in seconds.
- Ideally, we would actually build and test actual circuits to understand all of its behaviors. However, you would need breadboards, components and time to wire the circuit.

Why use SPICE?

Actual circuits also require expensive equipment like power supplies, signal generators and oscilloscopes. It may be difficult to physically breadboard every circuit you encounter.

You can spend hours building an actual circuit and only get a simple concept from it, whereas, SPICE provides the insight in minutes. SPICE can be your "virtual" breadboard.

Even if you have a short time to spare, you can cover several circuit principles and applications.

PSPICE?

- PSPICE is a mixed analog/digital electrical circuit simulator meaning PSpice can calculate the behavior of analog-only, mixed analog/digital and digital-only circuits with speed and accuracy.
- **PSpice** analog and digital algorithms are build into the same program. Hence, mixed analog/digital circuits can be simulated with tightly-coupled feedback loops between the analog and digital sections, without any performance degradation.
- **PSpice** calculates "voltage" and "currents" of the analog components and nodes, and "states" of the nodes connected to digital components. The results are formulated into meaningful graphical traces, tables and plots for further analysis.

Numeric Value and Expression

Literal numeric values are written in standard floating point notation.

PSpice assumes default units for numbers described component values and electrical quantities. However, values can be scaled by following the number by the appropriate scale suffix as shown in Table 1.

Numeric values can also be indirectly represented by parameters (PARAM) Numeric values and parameters can be used together to form arithmetic expressions. PSpice expressions may incorporate the intrinsic functions (ABS, SQRT, EXP, LOG, LOG10, PWR, SIN, COS, TAN, ATAN, TABLE, LIMIT) and user-functions (FUNC).

Scale suffixes for numeric values

Name	Scale	Symbol
tera	10^{+12}	T
giga	10 ⁺⁹	G
mega	10 ⁺⁶	MEG
kilo	10 ⁺³	К
milli	10 ⁻³	М
micro	10 ⁻⁶	U
nano	10 ⁻⁹	N
pico	10 ⁻¹²	Р
femto	10 ⁻¹⁵	F

© www.etti.tuiasi.ro

www.study.tuiasi.ro

Elements, Models and Nodes

Elements and models/macromodels description represents the core of the circuit description.

An element statement (in netlist file) contains connectivity information (circuit topology from schematics) and either explicity or by reference to a model/subcircuit name (from a models library), the value of the defined element.

Model (MODEL)/macromodel (SUBCKT) statements are necessary for defining the parameters of complex elements: all semiconductor devices and many ICs.

CAAEC Laboratory

Model type

	Type(Name)	
Device		Model Type
GaAs MESFET Transistor	В	GASFET
Capacitor	С	CAP
Diode	D	D
Voltage-controlled voltage source	E	Without model
Current-controlled current source	F	Without model
Voltage-controlled current source	G	Without model
Current-controlled voltage source	Н	Without model
Current source	I	Without model

CAAEC Laboratory

Model type

TTTTT	т	
JFE1 transistor	J	NJF-for n channel
		PJF-for p channel
Inductor coupling	K	CORE
Inductor	L	IND
MOSFET transistor	м	NMOS-for n channel
		PMOS-for p channel
BJT tranzistor	Q	NPN- for npn
		PNP-for pnp
Resistor	R	RES
Voltage-controlled switch	S	VSWITCH
Transmission line	Т	TRN
Voltage source	V	Without model
Current-controlled switch	W	ISWITCH
IGBT	Z	NIGBT
(Insulated-Gate Bipolar Transistor)		
Miscellaneous (OpAmp,	Х	Subcircuits (generic
Comparator, Voltage Reference,		form):SUBCKT
Quartz Crystal, etc)		

Table 2. Analog Devices Summary

www.study.tuiasi.ro

The following *conventions* must be observed in the circuit definitions:

- A circuit must always contain a ground node, which must always be number 0.
- Every node in the circuit must have at least two elements connected to it; the only exceptions are the nodes of unterminated transmission lines.
- Every node in the circuit must have a DC path to ground. In DC, capacitors represent open circuits and inductors represent shorts. This requirement prevents the occurrence of floating nodes, for which the program cannot find a bias point.

- Because standard SPICE2G.6 uses modified nodal analysis to solve for both node voltages and currents of voltage-defined elements, such as voltage sources and inductors, <u>two restrictions</u> must be observed:
- the circuit cannot contain a loop of voltage sources or inductors
- it cannot contain a cut-set of current sources or capacitors.

The former is disallowed due to Kirchhoff's voltage law and the latter due to Kirchhoff's current law.

Any violation of the above restrictions results in an error message and termination of the SPICE program. The possible error messages and corrective actions are:

1. "Floating Nodes"

During read-in, *PSpice* checks the topology of the circuit. One of the checks done is to make sure that there are no floating nodes. If there are floating nodes, *PSpice* will indicate a read-in error on the screen and the output file will contain a message similar to the following:

ERROR: *Node X is floating*.

CAAEC Laboratory

Conventions

This means that there is no DC path to ground from node X.

A DC path is one that draws current, for instance a path through resistors, inductors and transistors. There are several paths that do not draw current and that do not count:

 The two ends of a transmission line do not have a DC connection between them: in the following example node 5 has a connection to node 0 (ground): T1 5 0 4 8 Z0=75 td=20ns

 Voltage-controlled sources do not have a DC connection to their controlling nodes, so these sources do not draw current from their controlling nodes. In the following examples, node 5 has a connection to ground:

EGAIN 5 0 4 8 100

GA 50480.8

The two sides of a capacitor have no DC connection between them. In the following example, node 5 has no DC path to ground:

C5 5 0 0.1u

In all these cases the solution is straightforward: connect the floating circuitry to ground with a resistor (usually of large value, 1MEGohm).

Conventions 2. "Voltage Source/Inductor Loop"

Another topology check that is done on each circuit is to make sure that there are no loops with zero resistance. If there are, PSpice will indicate a read-in error on the screen and the output file will contain a message similar to the following:

ERROR: Voltage loop involving Vx

This means that the circuit has a loop of zero resistance components, one of which is Vx.

2. "Voltage Source/Inductor Loop"

The zero resistance components in PSpice are: independent voltage sources (V), inductors (L), voltage-controlled voltage sources (E), and current-controlled voltage source (H). Examples of such loops are:

a.) Vin 3 0 10V

Vs 305V

- b.) V1 3 5 15V
 - L1 3 5 10u
 - E1352710

Note that it makes no difference whether the values of the voltage sources are 0 or not. Having a zero resistance loop means that the program will need to divide 10V (or any value of voltage) by 0, which is impossible.

In all these cases the solution is straighforward: add a series resistance to at least one component in the loop. Choose the resistor's value to be small enough so that it does not disturb the operation of the circuit. However, to avoid exceeding the dynamic range of the double-precision arithmetic used in PSpice, it is recommended not going bellow 1micro-ohm. To be more accurate, choose a value that approximates the actual parasitic resistance of the component.

3. "Voltage-Controlled Sources"

During the read-in and checking part of a run you may get an error which prints the message in the output file:

ERROR: Less than 2 connections at node X

The inputs to voltage-controlled sources are not considered connections during this check. This is because these inputs are ideal inputs and draw no current (they have infinite impedance).

If this occurs, the solution is: connect a very large resistor (a Gohms, say) from the source's input to ground. This will satisfy the topology check and, if the resistor is large enough it will not affect the circuit's behavior.

Convergence problems

The DC sweep, bias point calculation and transient analysis all use iterative algorithms.

These algorithms start with a set of node voltages and for each iteration they calculate a new set of node voltages which hopefully are closer to a solution of Kirchhoff's voltage and current laws. *In other words, an initial guess is used and successive iterations are supposed to converge to the solution.*

If the iterations do not converge onto a solution, then the analysis fails. The DC sweep skips the remaining points in the sweep. Failure of the bias-point calculation prevents other analyses (AC, sensitivity, etc) which depend on it from being done. The transient analysis skips the remaining time.

© www.etti.tuiasi.ro

www.study.tuiasi.ro

Convergence problems

PSpice is successful in analyzing most circuits.

Considerable effort has gone into eliminating problems which impeded the progression of circuit analysis.

However, there are no guarantees. If an analysis fails for our circuit, the company give us few suggestions for DC Sweep analysis, Bias Point and Transient Analysis.

Applications

- Performing the analysis of a simple circuit using PSpice (simulation flow). The circuit will be provided.
 - *Edit the circuit file* containing the netlist and commands (.cir)
 - Start simulation
 - View output file (.out) for error messages (if any)
 - Correct errors and restart the simulation
 - View simulation results
 - Inspect output file for bias point or other simulation results (ex: Fourier analysis)
 - Use **Probe** waveforms processor to represent waveforms or expressions and to perform measurements.