Introduction to SPICE simulator

Department Of Electrical Engineering
Indian Institute Of Technology Dharwad

October 21, 2021

What is this SPICE all about?

- Stands for Simulation Program with Integrated Circuit Emphasis
- Used for simulating circuits for verifying performance and optimization
- Powerful simulator with industry standards
- Supports different types of analysis like AC, DC, TRAN, PZ, OP etc.

Why SPICE

- Circuits get complex with inreasing number of transistors
- Analytically solving is impractical
- Even for small circuits in scaled technologies the number of parameters are far too many
- Example
    - The BSIM4 MOS model has about 39 technology dependent parameters!!

SPICE Code Format

- The SPICE code includes following sections:
  TITLE LINE
  CIRCUIT DESCRIPTIONS
  .MODEL STATEMENTS
  ANALYSIS COMMANDS
  OUTPUT COMMANDS
  .END
- The **title line** and is ignored by the simulator. This line is printed verbatim on the output screen
- Comment lines should be preceeded with an *.
- Multiple files can be included using **.include** directive. These other files may be subcircuits or model files for custom devices.

SPICE Code Format

- The “**analysis commands**” include indications for transient, DC , AC analysis.
- the “**output command**” must begin with **.control** directive and terminates with **.endc** directives.
- Commands not written in the **.control** block should be preceeded with a “**.**”
- The “**output commands**” are used for saving or ploting or printing outputs.
- The program file should end with a **.end** directive.

Circuit Description

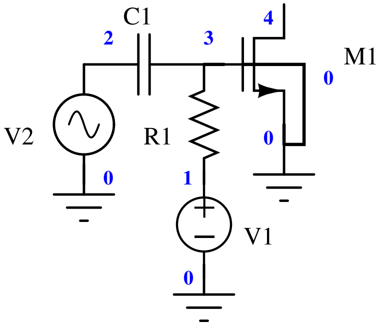
- The **CIRCUIT DESCRPITION** in SPICE is called “netlist”, consisting of element and source description.
- Element discription format:
  <letter><name> <n1> <n2>... [modelname] [parvals]
- Elementary components in native spice are
  - R* => resistor
  - C* => capacitor
  - L* => inductor
  - SW* => switch
  - K* => coupled mutual inductor
  - M*=> MOSFET
    nodes: [DRAIN -> GATE -> SOURCE-> SUBSTRATE]
  - Q*=> BJT
    nodes: [COLLECTOR->BASE->EMITTER->SUBSTRATE]

Circuit Description

- Elementary sources :
  Independent sources
  - V<name> <n+> <n-> [type] <val>
  - I<name> <n+> <n-> [type] <val>

  Dependent sources
  - <letter><name> <nout+> <nout-> <nc+> <nc-> <gain>
    G* => linear voltage controlled current source
    E* => linear voltage controlled voltage source
    F* => linear current controlled current source
    H* => linear current controlled voltage source
    Nonlinear dependent sources are also available (look at references if interested)

Circuit Description: Example



```
R1 3 1 1K
C1 2 3 1p
M1 4 3 0 0 cmosn W=2um L=0.18um
V1 1 0 dc 5V
V2 2 0 sin(0 1V 1K 0 0)
```

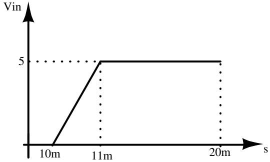
Analysis Commands

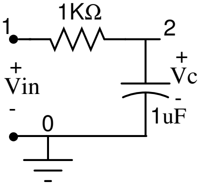
- DC analysis for DC transfer characteristics
  - .DC source_name vstart vstop stepsize
- AC analysis for frequency response
  - .AC DEC/OCT/Lin NP fstart fstop
  - .AC DEC 10 10 100M
- Trasnient analysis for time response
  - .TRAN stepsize stoptime starttime

Output Commands

- Plot voltages and currents
  - plot V(1) V(2) I(3) Vid#branch
  - plot V(3) vs V(4)
- Save vectors in file
  - set hcopydevtype=postscript
    hardcopy filename.ps V(1) V(2) ..
- Print voltages and currents
  - print V(vout) V(1)

Example:RC Circuit transient analysis





```
RC Transient Response
*resistor connected between nodes 1 and 2
r1 1 2 1k
*capacitor connected between nodes 2 and 0
c1 2 0 1u
*piecewise linear input voltage
vin 1 0 pwl (0 0 10ms 0 11ms 5v 20ms 5v)
```

Example:RC Circuit transient analysis

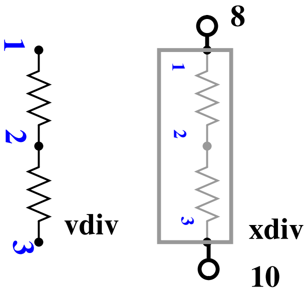
```
*transient analysis for 20ms, step size 0.02ms
.tran 0.02ms 20ms
*defining the run-time control functions
.control
run
*plotting input and output voltages
plot v(1) v(2)
.endc
.end
```

Example:RC Circuit AC analysis

For the same R-C circuit let us do the small-signal AC analysis.

```
RC Circuit Frequency Response
r1 1 2 1k
c1 2 0 1u
*Specifying an AC source with zero dc
vin 1 0 dc 0 ac 1
*AC analysis for 1 Hz to 1MHz, 10 points per decade
.ac dec 10 1 1Meg
.control
run
*Magnitude dB plot for v(2) on log scale
plot vdb(2)
*Phase degrees plot for v(2) on log scale
plot vp(2)
.endc
.end
```

Subciruits

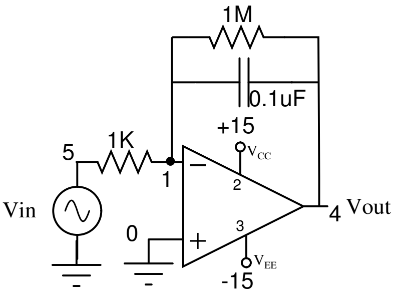


- Regular circuits can be described using repeated instantiations of elemental cells
- Subcircuits description
  - .subckt nodes sname
  - ...netlist
  - .ends sname

    .subckt vdivide 1 3
    R1 1 2 10K
    R2 2 3 5K
    .ends vdivide
- Subcircuits are identified by X* port list instance name
- X* nodes sname

  xdiv 8 10 vdiv

Example:Integrator using OPAMP-741



```
Differentiator using op-amp 741
*Including the predefined op-amp subcircuit file
.include ua741.txt
*Connections as mentioned in subcircuit file
x1 0 1 2 3 4 UA741
r1 5 1 1k
c1 4 1 0.1u
rf 4 1 1Meg
```

Example:Integrator using OPAMP-741

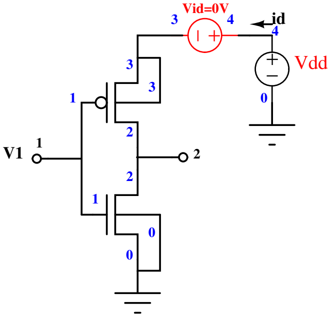
```
vcc 2 0 dc 15v
vee 3 0 dc -15v
*Giving a sinusoidal input
vin 5 0 sin (0 1v 1k 0 0)
.tran 0.02ms 6ms
.control
run
plot v(5) v(4)
.endc
.end
```

.MODEL STATEMENTS

- Devices in a circuit are defined by same set of device model parameters.
- These parameters are defined on a separate **.model** line and assigned a unique model name.

  model mname type(pname1=pval1 pname2=pval2 . .)
- Model parameters written a file can also be inculded using **.include** command.

Example: CMOS Inverter



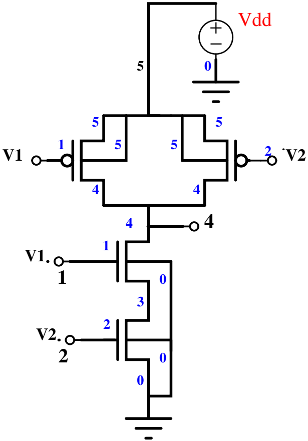
```
DC analysis
* include model file
.include tsmc_spice_model.txt
*Element instantiation
M1 2 1 0 0 cmosn w=2um l=0.18um
M2 2 1 3 3 cmosp w=5um l=0.18um
*supply voltages , input sources
vdd 4 0 dc 1.8V
vin 1 0 0.0 pulse (0 1.8v 0 0 0 5us 10us)
*dummy voltage to measure current
vid 4 3 dc 0v
```

Example: CMOS Inverter

```
*perform DC analysis
.dc Vin 0 1.8 0.1
.control
run
plot v(2)
plot v(1)
plot i(vid)
plot v(2) vs v(1)
*save the plots in eps file
set hcopydevtype=postscript
hardcopy invplots.eps i(vid)
.endc
.end
```

Example: CMOS Nand

```
Nand transient analysis
* include model file
.include tsmc_spice_model.txt
*Element instantiation
m1 3 2 0 0 cmosn w=2um l=0.18um
m2 4 1 3 0 cmosn w=2um l=0.18um
m3 4 1 5 5 cmosp w=5um l=0.18um
m4 4 2 5 5 cmosp w=5um l=0.18um
*supply voltages,input sources
vdd 5 0 dc 1.8v
v1 1 0 pulse (0 1.8v 0 1ns 1ns 5us 10us)
v2 2 0 pulse (0 1.8v 0 1ns 1ns 2.5us 5us)
```

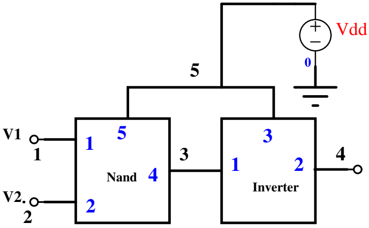


Example: CMOS Nand

```
*perform transient analysis
.tran 0.02us 10us
.control
run
plot v(4)
plot v(1) v(2)
.endc
.end
```

Example: CMOS AND gate using subcircuits

```
AND transient analysis
* NAND subcircuit
 .subckt inv 1 2 5
.include tsmc_spice_model.txt
m1 2 1 0 0 cmosn w=2um l=0.18u
m2 2 1 5 5 cmosp w=2um l=0.18u
.ends inv
* NAND subcircuit
.subckt nand 1 2 4 5
.include tsmc_spice_model.txt
m1 3 2 0 0 cmosn w=2um l=0.18u
m2 4 1 3 0 cmosn w=2um l=0.18u
m3 4 1 5 5 cmosp w=2um l=0.18u
m4 4 2 5 5 cmosp w=2um l=0.18u
.ends nand
```



Example: CMOS AND gate using subcircuits

```
*instantiate the subcircuits
xnand 1 2 3 5 nand
xinv 3 4 5 inv
vdd 5 0 dc 1.8v
V1 1 0 pulse (0 1.8v 0 1ps 1ps 5us 10us)
V2 2 0 pulse (0 1.8v 0 1ps 1ps 2.5us 5us)
*specify analysis
.tran 0.02us 10us
.control
run
plot v(4)
plot v(1) v(2)
.endc
.end
```