

Monte Carlo Simulation in Hspice

(Last updated: Sept. 21, 2008)

Monte Carlo Tutorial

This tutorial was created to document the steps needed to run Monte Carlo simulations in batch mode within Hspice.

Setup

1. In your "cadXX" directory (e.g., cad1, cad2, etc.), make a new directory called "mc_hspice".

```
% mkdir mc_hspice
```

2. Open up "icfb" and *Analog Environment* from your testbench, as if you were going to simulate the testbench, as usual.
3. Load the correct state that you would like to use for Monte Carlo simulations. Make sure that all of your stimuli (input pulse functions, global DC sources, etc.) and your variables are set to the correct values (for example, setting the "outcap" variable to 25fF from CAD's 1 & 2).
4. Within the *Analog Environment* window, execute the following command

Simulation->Netlist->Create

5. A new window should pop-up with the new netlist and the full path for the netlist is listed in the title bar of the window. It will resemble the path below:

```
/tmp/Hspice_[uniquename]/[CELL_NAME]/hspiceD/schematic/netlist/input.ckt
```

6. Copy this netlist file to your new "mc_hspice" directory.

```
% cp /tmp/Hspice_[uniquename]/[CELL_NAME]/hspiceD/schematic/netlist/input.ckt mc_hspice/
```

7. Also copy the "_graphical_stimuli.scs" file to your "mc_hspice" directory.

```
% cp /tmp/Hspice_[uniquename]/[CELL_NAME]/hspiceD/schematic/netlist/_graphical_stimuli.scs mc_hspice/
```

8. Change into your "mc_hspice" directory and open the new "input.ckt" file in a text editor.

```
% cd mc_hspice
% nedit input.ckt &
```

9. First, add "gnd!" to the list of .GLOBAL nets if it is not included.

```
Change      .GLOBAL vdd!
To          .GLOBAL vdd! gnd!
```

10. Add the following **.PARAM** line following the **.GLOBAL vdd! gnd!**

```
.PARAM mc_global=1
```

11. Change the `.TRAN` line to the following (substituting your own STEP and STOP times):

Change **`.TRAN 100e-12 10e-9 START=0.0`**
 To **`.TRAN 100e-12 10e-9 SWEEP MONTE=100`**

Also substitute in the number of Monte Carlo runs that you'd like to run. In this example, we're running 100 Monte Carlo simulations.

12. Modify the `.OPTION` commands that are included.

Modify **`.OPTION`**
 `+ ARTIST=2`
 `+ DCSTEP=1`
 `+ INGOLD=2`
 `+ MEASOUT=1 ...`

Remove the "`+ ARTIST=2`", "`+ PARHIER=LOCAL`", and "`+ PSF=2`" lines that follow the `.OPTION` line.

Add the following lines after the `.OPTION` line:

`+ POST=1`
`+ METHOD=GEAR`
`+ MEASDGT=6`

The resulting set of `.OPTION` lines should now look like:

`.OPTION`
`+ METHOD=GEAR`
`+ MEASDGT=6`
`+ DCSTEP=1`
`+ INGOLD=2`
`+ MEASOUT=1`

13. Modify the `.include` line that includes the `"_graphical_stimuli.scs"` file and point it to your local copy.

Change
`.include "/tmp/Hspice_[username]/[CELL_NAME]/hspiceD/schematic/netlist/_graphical_stimuli.scs"`
 To
`.include "_graphical_stimuli.scs"`

14. Before the `.END` statement at the end of the "input.ckt" file, add some print statements to output any node voltages or currents.

`.PRINT V(node1) V(node2) I(node3)`
 For example, for CAD1 we would add:
`.PRINT V(in) V(out)`

15. After the added `.PRINT` statements, add your measurement statements to automatically calculate delay or other useful metrics.

`.MEASURE TRAN delayf TRIG V(in) VAL=0.6 RISE=1 TARG V(out) VAL=0.6 FALL=1`
`.MEASURE TRAN delayr TRIG V(in) VAL=0.6 FALL=1 TARG V(out) VAL=0.6 RISE=1`

`.MEASURE TRAN risetime TRIG V(out) VAL=0.12 RISE=1 TARG V(out) VAL=1.08 RISE=1`
`.MEASURE TRAN falltime TRIG V(out) VAL=1.08 FALL=1 TARG V(out) VAL=0.12 RISE=1`

`.MEASURE TRAN avgsupply_i AVG I(v0)`

.MEASURE TRAN avgsupply_p PARAM='-avgsupply_i * 1.2'

Note that the first two measurement statements measure the falling and rising propagation delay (50% in -> 50% out) between the "in" node and the "out" node. In Hspice, V(**node**) represents the voltage at node, **node**. The next two statements measure the 10%->90% rise and fall times for the output signal. The final two statements are one way to measure the average current coming out of your power supply source, "v0". The "avgsupply_i" statement first measures the average current (over the whole simulation) going into node "v0" (notice this value is negative in the resulting .mt0 file since positive current represents current flowing into a node, while our supply current will be sourcing current). The last statement then multiplies the average supply current by the supply voltage, 1.2V, to get average total power.

16. Save and Close "input.ckt" file.

17. Run hspice from the command line on your new testbench using the following command (substituting the cell's name for **CELL_NAME**):

```
% hspice -i input.ckt -o CELL_NAME.lis
```

18. A number of files should have been created in your "mc_hspice" directory. The three most interesting files will be:

```
CELL_NAME.lis
CELL_NAME.mt0
CELL_NAME.tr0
```

CELL_NAME.lis will tell you if the spice simulation completed or if you had any warnings or errors.

If there weren't any warnings and the command line reported:

```
HSPICE job input.ckt completed.
```

then

CELL_NAME.mt0 will contain all of the .MEASURE results that you created along with any parameters that were changed due to the Monte Carlo simulation (in this example, "delayf", "delayr", "risetime", "falltime", "avgsupply_i" and "avgsupply_p" would be included in the **CELL_NAME**.mt0 text file for each of the 100 Monte Carlo runs).

CELL_NAME.tr0 contains all of the transient simulation data generated and you can graphically analyze your simulations by typing a command like:

```
% ezwave CELL_NAME.tr0 &
```

OR

```
% c_scope &
```

Other Useful Techniques/Hints

1. You may need to setup the correct software environment variable to run “hspice”. You can add the following line to your .cshrc file.

```
setenv SW_HSPICE 2006.09-sp1
```

2. Use the **.OPTION SEED=x** option to set the Monte Carlo seed the same across different computers (otherwise, SEED will be generated randomly for every different Monte Carlo run).

After **.OPTION** line, add:

```
+ SEED=x
```

where **x** is some number between 1 and 259200. The default value is “1” if it is not specified.

3. The Hspice documentation can be found at:

```
/usr/caen/bin/hspice-2006.09-sp1/docs
```

Some useful files include:

```
/usr/caen/bin/hspice-2006.09-sp1/docs/hspice-sa.pdf - Simulation & Analysis Guide
```

```
/usr/caen/bin/hspice-2006.09-sp1/docs/hspice-cmdref - Hspice Command Reference
```