Spectre simulations from the command line/LP
2018-10-02/Version 1.1 for Cadence 6 on heffalump

How to run a Spectre simulation from the command line

In this document I show how you can run the simulation program Spectre from the shell command line and view the resulting data using the waveform viewer ViVa, ViVa is the default waveform viewer in Cadence 6.

The Spectre input file

Write the input file in a text editor. I usually use emacs but any text editor will do. Spectre input files usually have the file-name extension `.scs`. If you are to run your simulation on `heffalump` it may be good idea to also create the input file on `heffalump` since you are not allowed to copy files to and from `heffalump.` But the file you submit with your prelab we will copy for you.

Note that if you use a text editor on a Windows computer it may not work directly under Unix since Windows and Unix use different end-of-line characters (carriage return + linefeed under Windows and only linefeed under Unix).

Spice is the most well-known simulation program and by default Spectre uses Spice syntax. To switch to the less ancient Spectre syntax you have to give the command:

```
simulator lang=spectre
```

in your input file. The lines after that command will be interpreted as Spectre syntax. In Spice the first line in the input file is a header, so do not put any command there, because then it will not be read. So the first lines in your file could look like these:

```
//This is a header
simulator lang=spectre
```

Note that comments start with `//`. It is good practice to comment your input!

You need to identify the model libraries that are to be used in your simulation and also which corner to use for the models. We have put the commands that include all these file in one file the usual directory:

```
/usr/local/cad/course/MCC092/Y2018/modelsTT.scs
```

So in your spectre input file you need merely to write this line at the beginning of the input file:

```
include "/usr/local/cad/course/MCC092/Y2018/modelsTT.scs"
```

After that comes a section where you specify your circuit topology, the so-called netlist. In this lab you only have to use transistors, resistors and capacitors plus voltage sources to generate the input signals and constant voltages.

Spectre simulations from the command line/LP
2018-10-02/Version 1.1 for Cadence 6 on heffalump

In the netlist you can use parameters for example

```
parameters supplyv=1.2
```

defines a parameter called `supplyv` which you can use later. You can also use expressions in all places where numerical values are expected. This feature is very useful when you want to change one parameter and have the scaling of several other depend on that parameter.

Here are some examples of lines from a netlist:

```
// An inverter with 20X sizes

xp1  ( out1 in vdd vdd ) psvtlp w=20*0.2 l=0.06
xn1  ( out1 in 0 0 ) nsvtlp w=20*0.1 l=0.06

// Connected to a pi link

c11  (out1 0) capacitor c=7.5f
r1   (out1 out2 ) resistor r=110
c12  ( out2 0 ) capacitor c=7.5f

//Connected to another 20X inverter

xp2  ( out3 out2 vdd vdd ) psvtlp w=20*0.2 l=0.06
xn2  ( out3 out2 0 0 ) nsvtlp w=20*0.1 l=0.06

// Connected to a second pi link

c21(out3 0) capacitor c=7.5f
r2  (out3 out4) resistor r=110
c22  (out4 0) capacitor c=7.5f

// Set VDD with a DC source

vvdd ( vdd 0 ) vsource dc=supplyv

// Set the input with a pulse source

vin  ( in 0 ) vsource type=pulse val0=0 val1=supplyv delay=0 rise=10p
fall=10p width=90p period=200p
```

Each instance of a component must be on its own line. The line starts with an instance name, then comes the names of the circuit nodes that the component is connected to (possibly surrounded by optional parentheses), then the component name, for example `capacitor` or `vsource` , and then values for any parameters.

Finally, you need to specify the analysis you want to run. Here it is the transient analysis that is used:

```
analysis1 tran step=1p stop=0.3n method=gear2only
```

By default all node voltages at the top level are saved. If you want to save any currents you have to use the `save` statement. If you use any save statement you need to specify **all** waveforms you want to save, also the circuit nodes that were previously saved by default. This is because in large circuits one does not want to save the waveforms for all circuit nodes since it would make the output file too large.

Hierarchy in the netlist

To create a hierarchy in the netlist one uses subcircuits. Here are the definitions of two subcircuits that will be handy in lab 4:

```
subckt xinv (in out vdd)
     parameters x=2
     xp1 (out in vdd vdd) psvtlp w=x*0.2 l=0.06 //Parameters in um!
     xn1 (out in 0 0) nsvtlp w=x*0.1 l=0.06     //Parameters in um!
ends xinv

subckt piwire (n1 n2)
     parameters cap=15f res=0.000000001 // A very small resistance
     cap1 (n1 0) capacitor 0.5cap
     res (n1 n2) resistor res
     cap 2 (n2 0) capacitor 0.5*cap
ends piwire
```

Within the parenthesis are the node names for the nodes connecting to the subcircuit. On the next line are the input parameters to the subckt. If no value is supplied for a parameter when the subcircuit is instantiated, the default value, given in subcircuit definition, is used. Any node names or instance names used inside a subcircuit are local to that subcircuit.

Here the two subcircuits above are instantiated:

```
inv1 (n1 n2 vdd) xinv x=16 //An X16 inverter
segment1 (n2 n3) piwire cap=1f res=0.1
```

First comes an instance name, which has to be unique at that level, then the nodes that are connected to the subcircuit, then the name of the subcircuit, and finally values for any input parameters of the subcircuit. The parentheses around the node names are optional but helpful I think.

Running Spectre

We have made a command that you can use which sets up all the paths etc. for you to run spectre. It is available on `heffalump`. You have to log onto `heffalump` using VNC just as in previous labs. As usual you should first change directory to the directory where you want to run Spectre. For example like this:

```
> cd MCC092/cadence/lab4
```

To run the Spectre simulation you issue this command at the command line:

```
> /usr/local/cad/course/MCC092/Y2018/spectre ckt.scs &
```

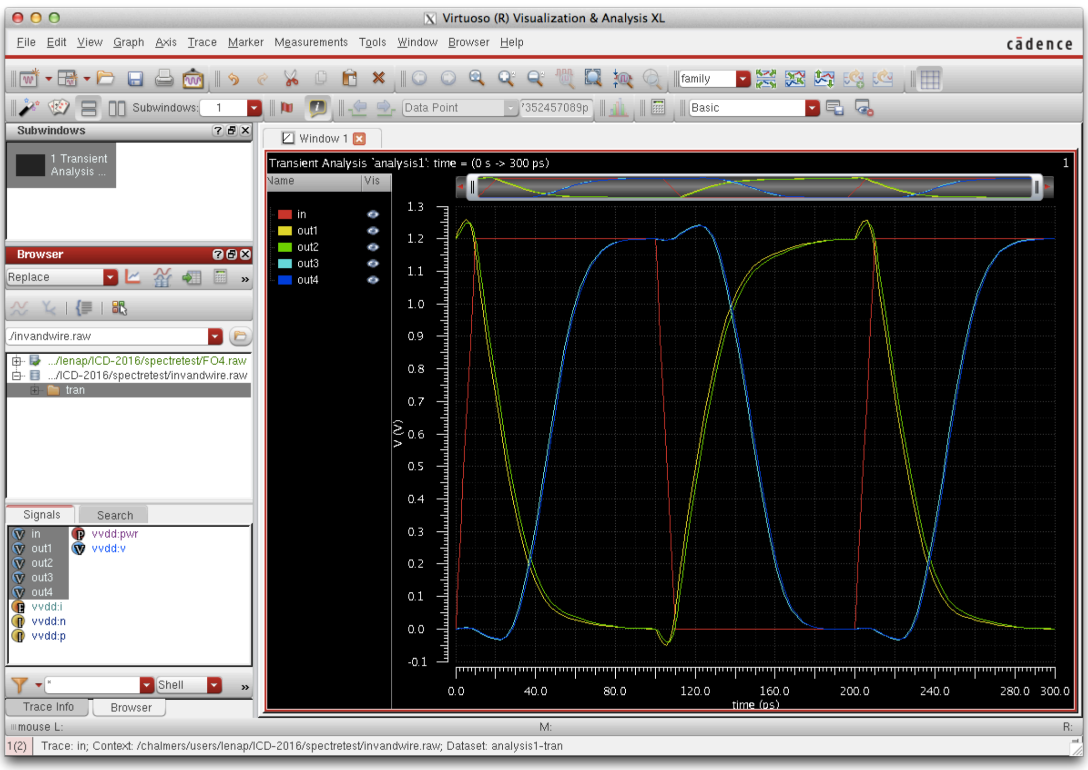
if `ckt.scs` is the file that contains your Spectre input file. Here the `&` sign at the end causes Spectre to run in the background so that you can use the command line for other things. If your input file is parsed correctly, you will see some output about the simulation statistics. If not, you will see some error messages. In that case, locate on what line the errors were and try to fix them in you text editor and rerun.

Viewing the result

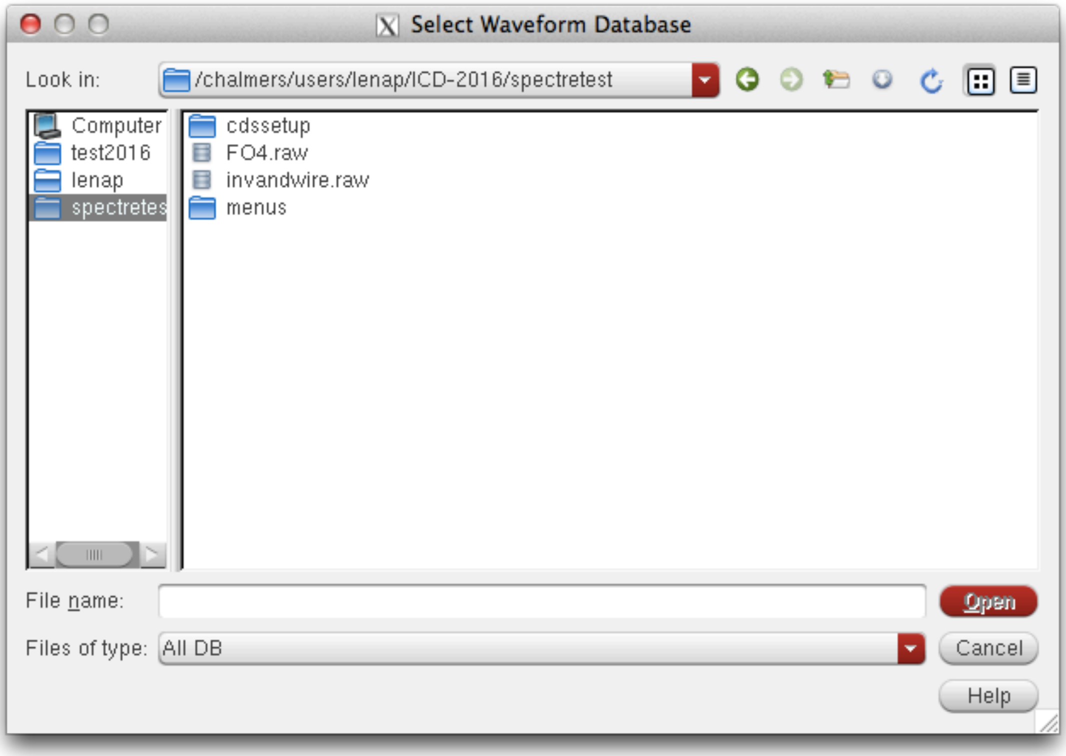
If your simulation concluded successfully, you can use the waveform viewer ViVa to view the results. Then you write this command while in the directory in which you ran Spectre:

```
> /usr/local/cad/course/MCC092/Y2018/viva ckt.raw &
```

Cadence is started also when you invoke only ViVa, so this step usually takes quite some time. First a small window is shown where you are asked about using the ADE XL license for ADE L. After that the usual waveform-viewer window appears. You have to specify where the results file is located by choosing **File->Open Results** in the upper left corner. Select the file called .raw. Then to the left the signals folder appear. For example called tran if you ran a transient analysis. If you doubleclick on the folder icon you will see symbols for the signals in the lower left corner. You can specify the signals you want to plot by doubleclicking on the signal as you see in the screenshot below.



If you want to switch file click on the folder icon and this window appears where you can switch the .raw file you are currently seeing in the lower left corner:



Select the file you want to view and click on the "Open" button to view the signals from the current raw file.

Loading new data from same input file

You do not have to restart ViVa after you have rerun Spectre which is really good since it takes quite some time to invoke ViVa. The easiest way to load new data is to use the Folder icon as shown above. One has to be careful to close the waveform window or use "Replace" before plotting new curves, or it may get quite confusing.

References

Spectre Simulator User Guide. Cadence 2004.

Kenneth S. Kundert & Paul R. Gray: The Designer's guide to SPICE and Spectre. Kluwer Academic Publishers, 1995. Particularly Appendix B Spectre Netlist Language. This book is available as an e-book from the Chalmers library.