ECE1388 Tutorial

Mixed Signal Simulations within Virtuoso AMS Environment

Contents

1 IC6 2

1.1 Environment Setup for UG Machines 2

1.2 Design and Testbench Setup 2

1.3 Hierarchy Editor Setup 2

1.4 Netlisting and Simulation 4

1.4.1 Simulator and analysis setup 4

1.4.2 Digital-Analog connection setup 6

1.5 Design and Testbench Setup 8

The intention of this tutorial is to get you familiar with Virtuoso AMS environment and simulator in order to netlist, compile, elaborate, and simulate a system containing analog, digital and mixed-signal components. The AMS simulator provides both the Spectre and the UltraSim solvers and you can switch between the two as your design evolves.
In this tutorial, we will design an inverter using transistors and verilog separately. We will use the following test bench in both cases where some of the inverters are simulated using verilog simulator and some are simulated using spectre simulator at the same time.

1 IC6

1.1 Environment Setup for UG Machines

1. Change to your local project directory

2. Execute following three lines in the exact order in terminal before starting Cadence session.

```
source /CMC/tools/CSHRCs/Cadence.IC617
source /CMC/tools/CSHRCs/Cadence.XCELIUM
setenv LD_PRELOAD /usr/lib/x86_64-linux-gnu/libstdc++.so.6
```

3. Start cadence session using following command.

```
startCds -t ic6-crn65gp
```

Ignore the following error message in CIW and terminal ”ERROR: ld.so: object ’/usr/lib/x86_64-linux-gnu/libstdc++.so.6’ from LD_PRELOAD cannot be preloaded (wrong ELF class: ELFCLASS64): ignored.”

4. Open cds.lib and following two lines at the end of the file.

```
SOFTINCLUDE $AMSHOME/tools/affirma_ams/etc/connect_lib/cds.lib
DEFINE sample $CDSHOME/tools/dfII/samples/cdslib/sample
```

5. Refresh library manager from *CIW* → *File* → *Refresh* and make sure you see two new libraries, *connectLib* and *sample*

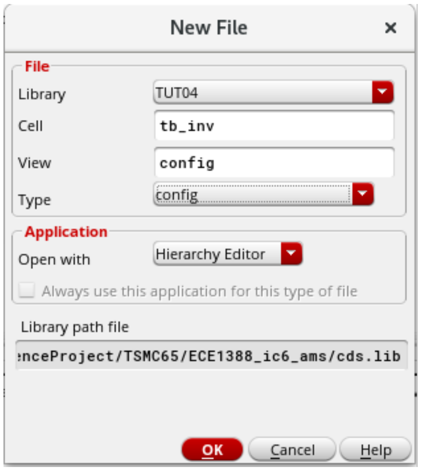
1.2 Design and Testbench Setup

Design the testbench and modules as mentioned in section 1.5 of this tutorial.

1.3 Hierarchy Editor Setup

1. Close all windows except *CIW* window.

2. In the library manger, create new view for *tb_inv* cell as below



New File

File

Library: TUT04

Cell: tb_inv

View: config

Type: config

Application

Open with: Hierarchy Editor

□ Always use this application for this type of file

Library path file: nceProject/TSMC65/ECE1388_ic6_ams/cds.lib

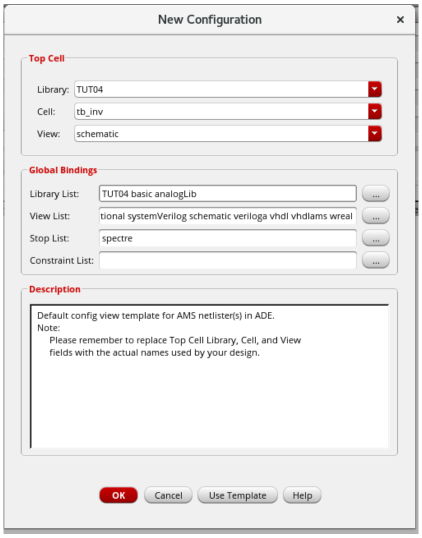
3. New configuration window for hierarchy editor will open.

4. Click on *Use Template*.

5. Choose AMS from the drop-down menu.

6. Add *TUT04 analogLib basic* to *Library List* under *Global Bindings*.

7. Make sure *Top Cell Library, Cell, and View* fields are correctly filled as *TUT04 tb_inv schematic* respectively.



Top Cell

Library: TUT04

Cell: tb_inv

View: schematic

Global Bindings

Library List: TUT04 basic analogLib

View List: tional systemVerilog schematic veriloga vhdl vhdlams wreal

Stop List: spectre

Constraint List:

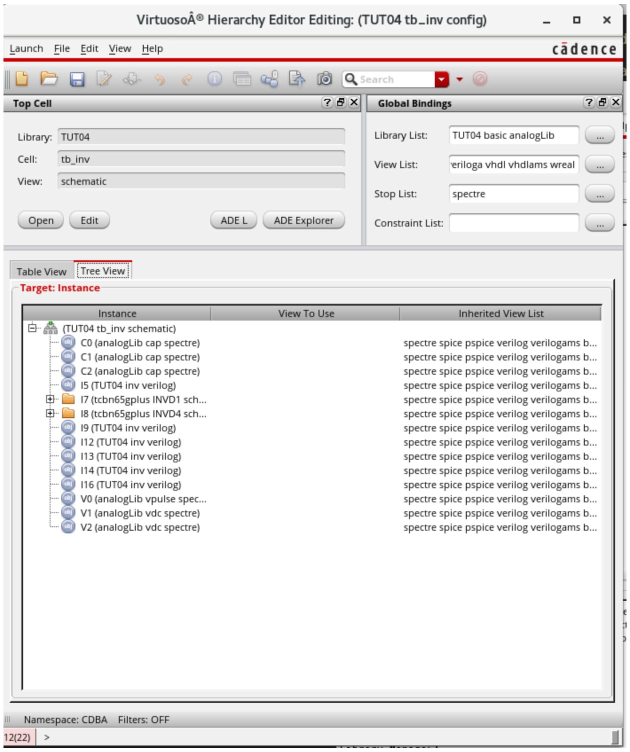
Description

Default config view template for AMS netlister(s) in ADE.
Note:
  Please remember to replace Top Cell Library, Cell, and View
  fields with the actual names used by your design.

8. Click OK

9. *Hierarchy editor*(HED) window will be populated.

10. Click on *Tree* in HED window. This will show tree view of all the instances of in your top cell.



Top Cell

Library: TUT04

Cell: tb_inv

View: schematic

Global Bindings

Library List: TUT04 basic analogLib

View List: eriloga vhdl vhdlams wreal

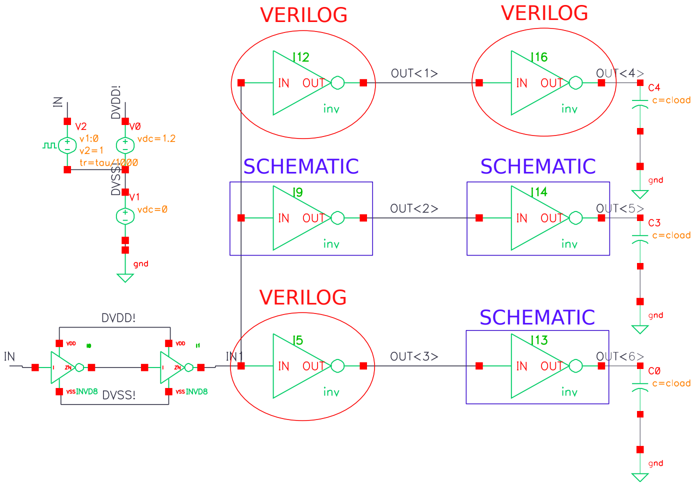
Stop List: spectre

Constraint List:

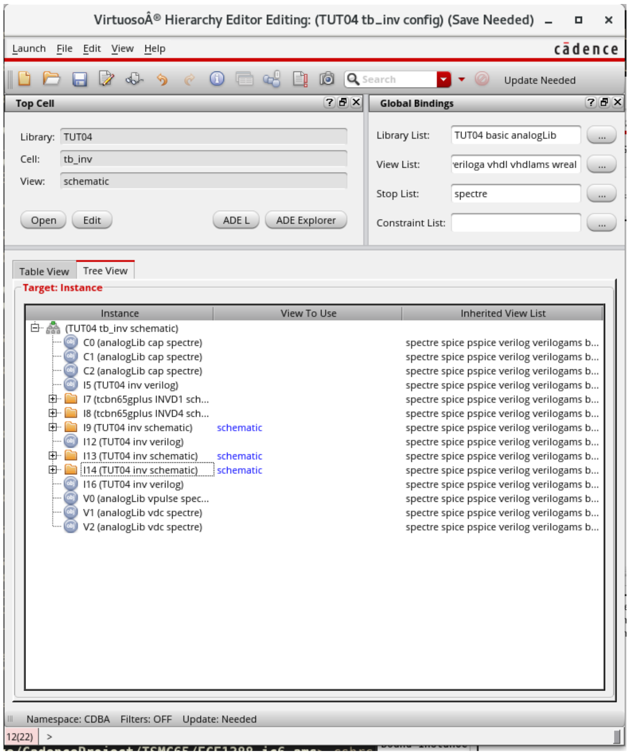
Target: Instance

Instance	View To Use	Inherited View List
(TUT04 tb_inv schematic)		
C0 (analogLib cap spectre)		spectre spice pspice verilog verilogams b...
C1 (analogLib cap spectre)		spectre spice pspice verilog verilogams b...
C2 (analogLib cap spectre)		spectre spice pspice verilog verilogams b...
I5 (TUT04 inv verilog)		spectre spice pspice verilog verilogams b...
I7 (tcbn65gplus INVD1 sch...		spectre spice pspice verilog verilogams b...
I8 (tcbn65gplus INVD4 sch...		spectre spice pspice verilog verilogams b...
I9 (TUT04 inv verilog)		spectre spice pspice verilog verilogams b...
I12 (TUT04 inv verilog)		spectre spice pspice verilog verilogams b...
I13 (TUT04 inv verilog)		spectre spice pspice verilog verilogams b...
I14 (TUT04 inv verilog)		spectre spice pspice verilog verilogams b...
I16 (TUT04 inv verilog)		spectre spice pspice verilog verilogams b...
V0 (analogLib vpulse spec...		spectre spice pspice verilog verilogams b...
V1 (analogLib vdc spectre)		spectre spice pspice verilog verilogams b...
V2 (analogLib vdc spectre)		spectre spice pspice verilog verilogams b...

11. In this example we will simulate three blocks with verilog model and three blocks with schematic/spectre model as shown below



12. In HED window assign *schematic* as a view to use for the instance highlighted using blue box above. After assigning all the instances HED should look as follows


VirtuosoÂ® Hierarchy Editor Editing: (TUT04 tb_inv config) (Save Needed)

Library: TUT04
Cell: tb_inv
View: schematic

Library List: TUT04 basic analogLib
View List: 'eriloga vhdl vhdlams wreal
Stop List: spectre
Constraint List:

Instance	View To Use	Inherited View List
(TUT04 tb_inv schematic)		
C0 (analogLib cap spectre)		spectre spice pspice verilog verilogams b...
C1 (analogLib cap spectre)		spectre spice pspice verilog verilogams b...
C2 (analogLib cap spectre)		spectre spice pspice verilog verilogams b...
I5 (TUT04 inv verilog)		spectre spice pspice verilog verilogams b...
I7 (tcbn65gplus INVD1 sch...		spectre spice pspice verilog verilogams b...
I8 (tcbn65gplus INVD4 sch...		spectre spice pspice verilog verilogams b...
I9 (TUT04 inv schematic)	schematic	spectre spice pspice verilog verilogams b...
I12 (TUT04 inv verilog)		spectre spice pspice verilog verilogams b...
I13 (TUT04 inv schematic)	schematic	spectre spice pspice verilog verilogams b...
I14 (TUT04 inv schematic)	schematic	spectre spice pspice verilog verilogams b...
I16 (TUT04 inv verilog)		spectre spice pspice verilog verilogams b...
V0 (analogLib vpulse spec...		spectre spice pspice verilog verilogams b...
V1 (analogLib vdc spectre)		spectre spice pspice verilog verilogams b...
V2 (analogLib vdc spectre)		spectre spice pspice verilog verilogams b...

Namespace: CDBA Filters: OFF Update: Needed

12(22) >

Save and close all the windows except *CIW and Library Manager*

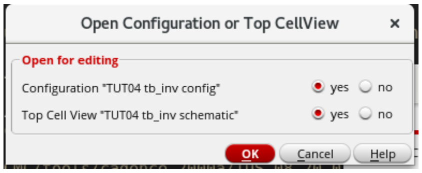
1.4 Netlisting and Simulation

We will perform simulation using AMS simulator.

1.4.1 Simulator and analysis setup

1. Open *TUT04* → *tb_inv* → *config*

2. A new window as shown below opens asking to open either of Configuration and Top cell view or both of them. Click yes for both of them.



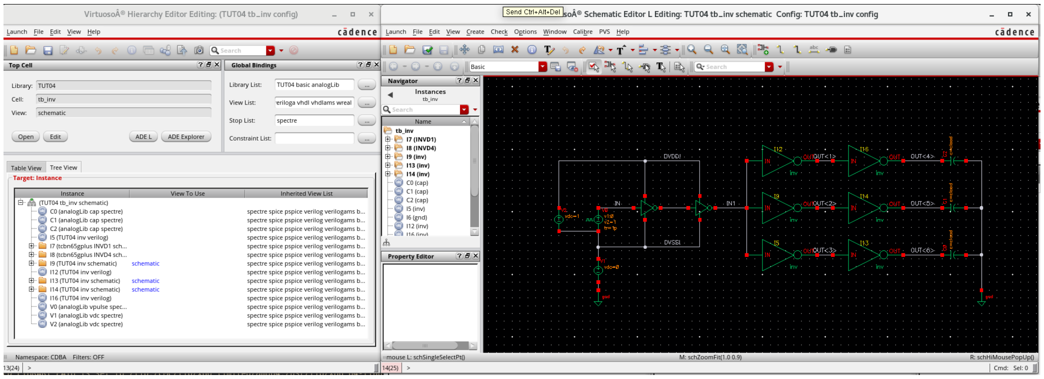
Open Configuration or Top CellView

Open for editing

Configuration "TUT04 tb_inv config" ● yes ○ no

Top Cell View "TUT04 tb_inv schematic" ● yes ○ no

3. Now you can see the Cadence Hierarchy Editor and the schematic of the design.



VirtuosoÂ® Hierarchy Editor Editing: (TUT04 tb_inv config)

Top Cell

Library: TUT04

Cell: tb_inv

View: schematic

Global Bindings

Library List: TUT04 basic analogLib

View List: eriloga vhdl vhdlams wreal

Stop List: spectre

Constraint List:

Target: Instance

Instance	View To Use	Inherited View List
(TUT04 tb_inv schematic)		
C0 (analogLib cap spectre)		spectre spice pspice verilog verilogams b...
C1 (analogLib cap spectre)		spectre spice pspice verilog verilogams b...
C2 (analogLib cap spectre)		spectre spice pspice verilog verilogams b...
I5 (TUT04 inv verilog)		spectre spice pspice verilog verilogams b...
I7 (tcbn65gplus INVD1 sch...		spectre spice pspice verilog verilogams b...
I8 (tcbn65gplus INVD4 sch...		spectre spice pspice verilog verilogams b...
I9 (TUT04 inv schematic)	schematic	spectre spice pspice verilog verilogams b...
I12 (TUT04 inv verilog)		spectre spice pspice verilog verilogams b...
I13 (TUT04 inv schematic)	schematic	spectre spice pspice verilog verilogams b...
I14 (TUT04 inv schematic)	schematic	spectre spice pspice verilog verilogams b...
I16 (TUT04 inv verilog)		spectre spice pspice verilog verilogams b...
V0 (analogLib vpulse spec...		spectre spice pspice verilog verilogams b...
V1 (analogLib vdc spectre)		spectre spice pspice verilog verilogams b...
V2 (analogLib vdc spectre)		spectre spice pspice verilog verilogams b...

Namespace: CDBA Filters: OFF

13(24) >

soÂ® Schematic Editor L Editing: TUT04 tb_inv schematic Config: TUT04 tb_inv config

mouse L: schSingleSelectPt()

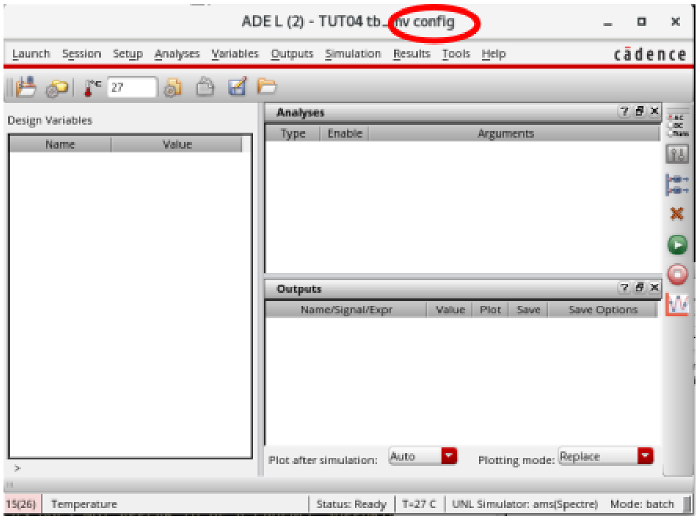
M: schZoomFit(1.0 0.9)

R: schHiMousePopUp()

14(25) >

Cmd: Sel: 0

4. Open the ADEL from Hirearchy editor by click ADEL button or from virtuoso schematic editor by doing *Launch* → *ADEL*. Make sure the design view is config. Otherwise correct it in *Setup* → *Design*.



ADE L (2) - TUT04 tb_inv config

Design Variables

Name	Value

Analyses

Type	Enable	Arguments

Outputs

Name/Signal/Expr	Value	Plot	Save	Save Options

Plot after simulation: Auto

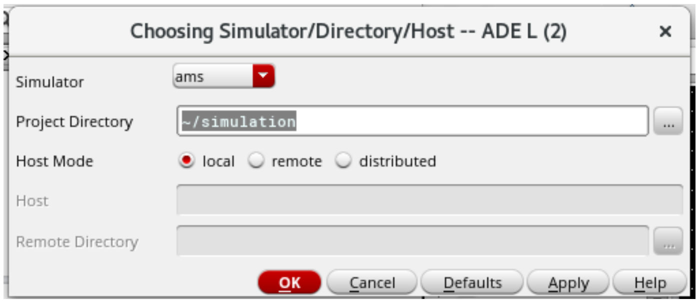
Plotting mode: Replace

15(26) Temperature

Status: Ready | T=27 C | UNL Simulator: ams(Spectre) Mode: batch

Launch Session Setup Analyses Variables Outputs Simulation Results Tools Help
cādence
27

5. Make sure the simulator is ams from Setup → Simulator/Directory/Host in the ADE window.



Simulator: ams

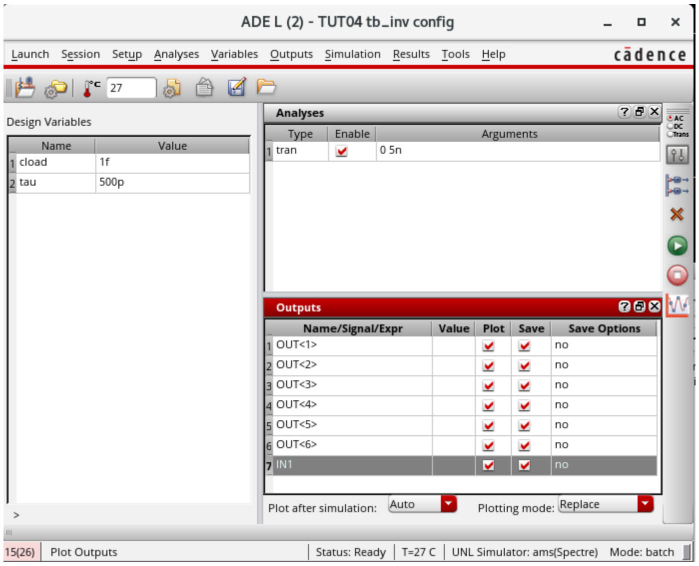
Project Directory: ~/simulation

Host Mode: ● local ○ remote ○ distributed

Host:

Remote Directory:

6. Make the time period of the vpulse to be $500ps$ with 50% duty cycle. Save and plot all 6 outputs $OUT[1:6]$ and input node $IN1$. Set transient analysis for $5ns$.



Design Variables

	Name	Value
1	cload	1f
2	tau	500p

Analyses

	Type	Enable	Arguments
1	tran	✓	0 5n

Outputs

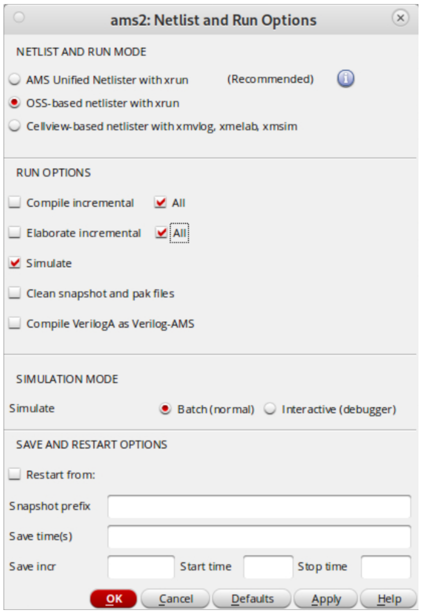
	Name/Signal/Expr	Value	Plot	Save	Save Options
1	OUT<1>		✓	✓	no
2	OUT<2>		✓	✓	no
3	OUT<3>		✓	✓	no
4	OUT<4>		✓	✓	no
5	OUT<5>		✓	✓	no
6	OUT<6>		✓	✓	no
7	IN1		✓	✓	no

Plot after simulation: Auto   Plotting mode: Replace

15(26) Plot Outputs Status: Ready T=27 C UNL Simulator: ams(Spectre) Mode: batch

1.4.2 Digital-Analog connection setup

1. Click *Simulation* → *Netlister and Run Options* and make sure settings match as follows.



NETLIST AND RUN MODE

○ AMS Unified Netlister with xrun (Recommended)

● OSS-based netlister with xrun

○ Cellview-based netlister with xmvlog, xmelab, xmsim

RUN OPTIONS

□ Compile incremental ☑ All

□ Elaborate incremental ☑ All

☑ Simulate

□ Clean snapshot and pak files

□ Compile VerilogA as Verilog-AMS

SIMULATION MODE

Simulate ● Batch (normal) ○ Interactive (debugger)

SAVE AND RESTART OPTIONS

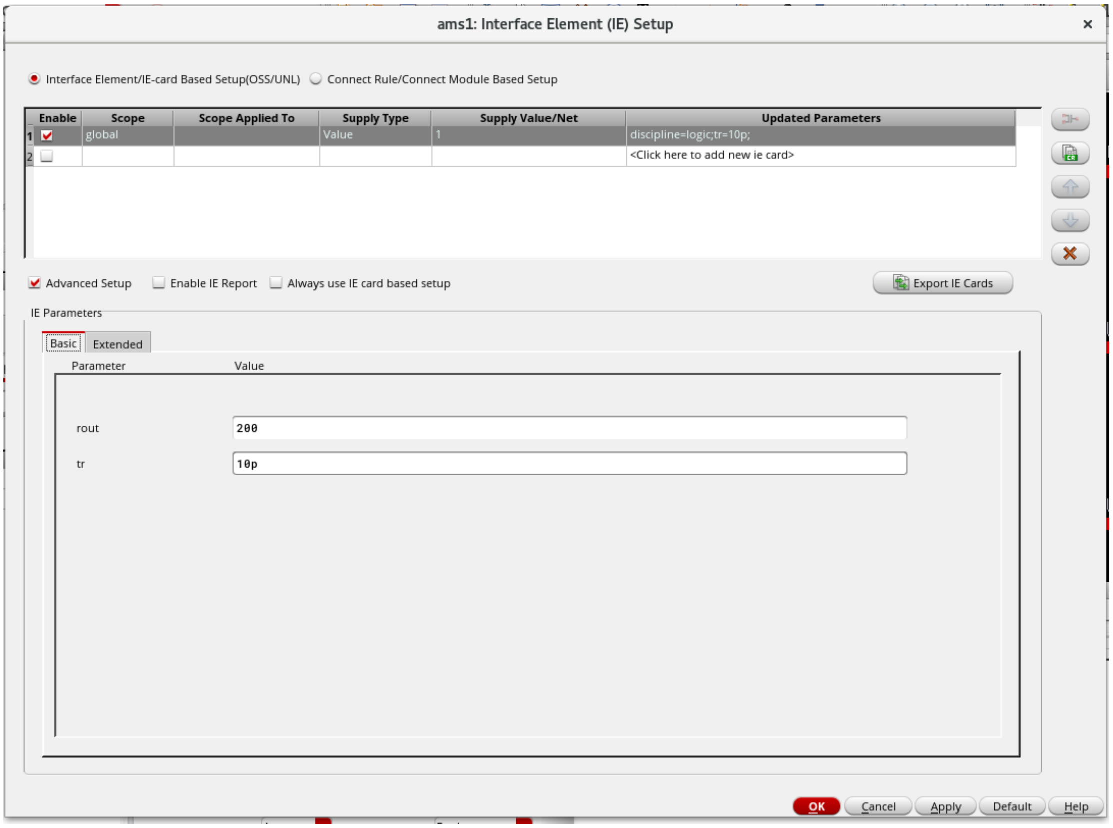
□ Restart from:

Snapshot prefix

Save time(s)

Save incr   Start time   Stop time

2. Open *Setup* → *Connect Rules*
   The digital 0 and 1s need to be converted to analog levels in order to be able to simulate the design



ams1: Interface Element (IE) Setup

● Interface Element/IE-card Based Setup(OSS/UNL) ○ Connect Rule/Connect Module Based Setup

	Enable	Scope	Scope Applied To	Supply Type	Supply Value/Net	Updated Parameters
1	☑	global		Value	1	discipline=logic;tr=10p;
2	☐					<Click here to add new ie card>

☑ Advanced Setup ☐ Enable IE Report ☐ Always use IE card based setup

IE Parameters

Basic | Extended

Parameter	Value
rout	200
tr	10p

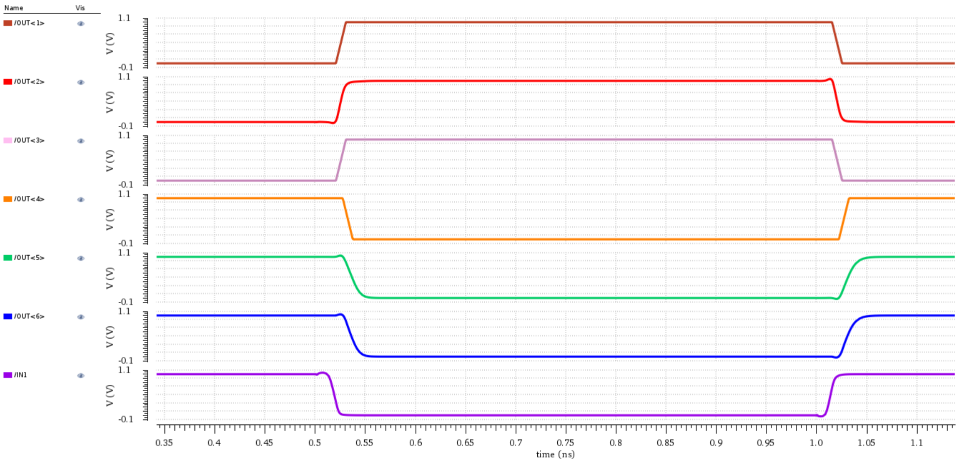
3. Change *Supply value/Net* from 1.8 to 1.

4. Make sure first row is selected and then click on *Advanced Setup* option.

5. Change rise time, *tr*, to 10*p*

6. Click OK and close the window.

7. The design is ready for simulation now. Click on *Simulation* → *Netlist and Run* to simulate. A sample output is provided below.

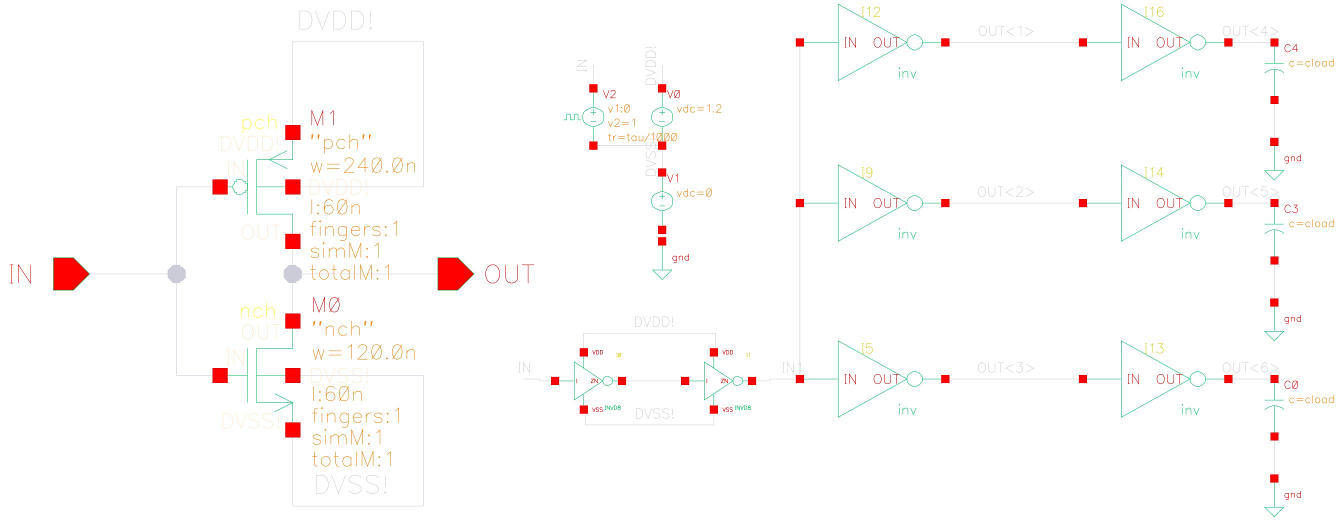


1.5 Design and Testbench Setup

Create following library and cellviews.

1. **Library:** Make new library named *TUT06*. Attach the library to existing technology and choose tsmcN65 from drop-down menu.

2. **Cellview → inv → schematic:** Make a unit sized inverter schematic in library *TUT04* as shown as shown in figure 1a with terminals *IN, OUT,* and global power connections *DVDD!* and *DVSS!* (No separate pins for power).
NOTE: Also create a symbol view from this schematic. This symbol will have only two terminals, *viz. IN* and *OUT*.



(a) inv

(b) tb_inv

Figure 1: Inverter testbench

3. **Cellview → tb_inv → schematic:** Create a testbench as shown in figure 1b

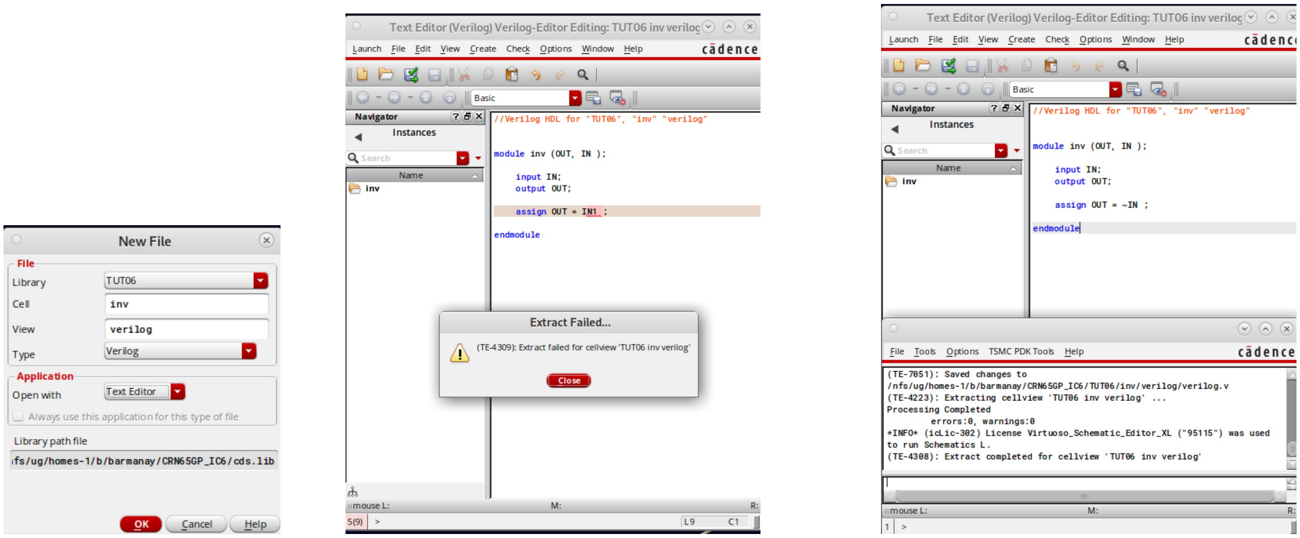
   NOTE: Before proceeding to the next step make sure you are able to simulate the *tb_inv* without any problems.

4. **Cellview → inv → verilog:** Here we assume that symbol view(It doesn't matter schematic is present or not) is already present in the *inv* cell. Create verilog cellview as shown in figure 2a.

   (a) Text editor(in this case it is vim) will open with with io ports predefined as per in symbol.

   (b) Write the code. Save and close the file.

   (c) if there are errors in the code the pop-up as shown in 2b will appear.



Library: TUT06
Cell: inv
View: verilog
Type: Verilog
Open with: Text Editor
Library path file: ıfs/ug/homes-1/b/barmanay/CRN65GP_IC6/cds.lib

(a) Inv verilog cellview

```
//Verilog HDL for "TUT06",  "inv" "verilog"


module inv (OUT, IN );

    input IN;
    output OUT;

    assign OUT = IN1 ;

endmodule
```

(TE-4309): Extract failed for cellview 'TUT06 inv verilog'

(b) Wrong verilog code

```
//Verilog HDL for "TUT06",  "inv" "verilog"


module inv (OUT, IN );

    input IN;
    output OUT;

    assign OUT = ~IN ;

endmodule
```

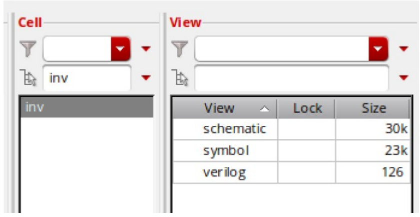
```
(TE-7051): Saved changes to
/nfs/ug/homes-1/b/barmanay/CRN65GP_IC6/TUT06/inv/verilog/verilog.v
(TE-4223): Extracting cellview 'TUT06 inv verilog' ...
Processing Completed
        errors:0, warnings:0
*INFO* (icLic-302) License Virtuoso_Schematic_Editor_XL ("95115") was used
to run Schematics L.
(TE-4308): Extract completed for cellview 'TUT06 inv verilog'
```

(c) Error-free verilog code

Figure 2: Verilog cellview for inverter(NOT gate)

(d) Open the file and fix the code. Save and close the file. If you do not see any pop-ups, check *CIW* window and make sure there are no errors.

5. At the end of this section your TUT06 library should look something like as below



View	Lock	Size
schematic		30k
symbol		23k
verilog		126