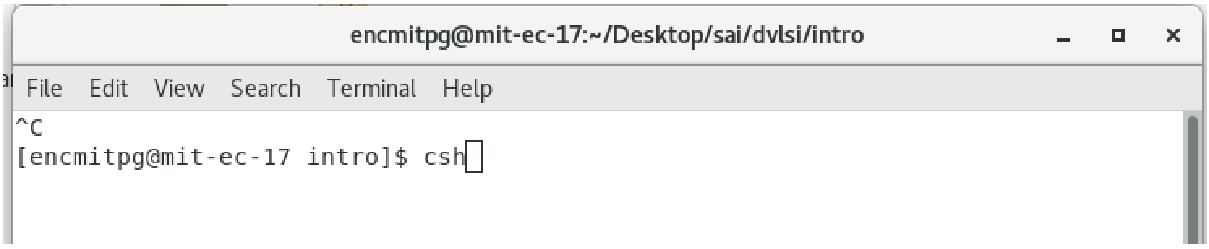
## *Creating a Workspace:*

- In Desktop Create a folder and name it as Digital_VLSI (give folder name without any space).
- Create a new sub-directory and name it as andgate for the design and open a terminal from the sub-directory.

## Functional Simulation:

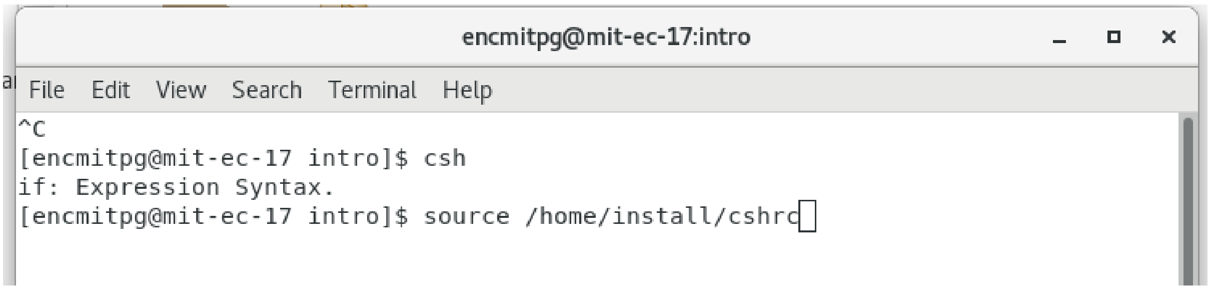
Invoke the cadence environment by type the below commands

- `csh` (Invokes C-Shell)
- `source /home/install/cshrc` (mention the path of the tools)


encmitpg@mit-ec-17:~/Desktop/sai/dvlsi/intro
File Edit View Search Terminal Help

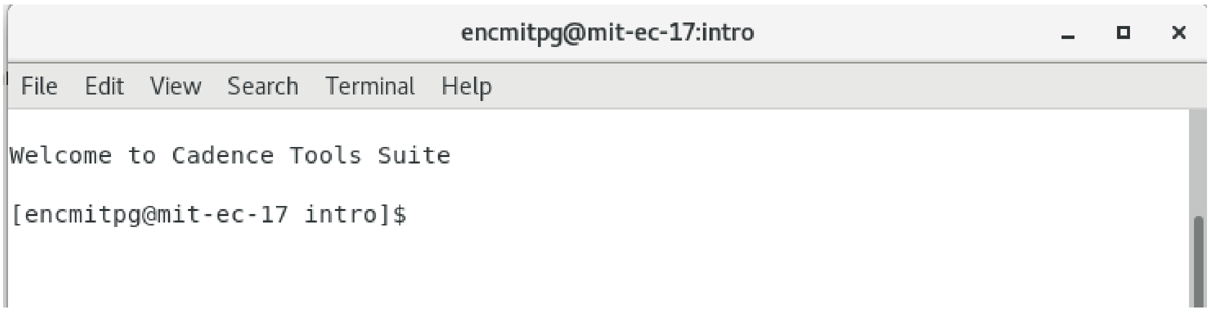
```
^C
[encmitpg@mit-ec-17 intro]$ csh
```

Fig. 1: csh



```
^C
[encmitpg@mit-ec-17 intro]$ csh
if: Expression Syntax.
[encmitpg@mit-ec-17 intro]$ source /home/install/cshrc
```

Fig. 2: source /home/install/cshrc



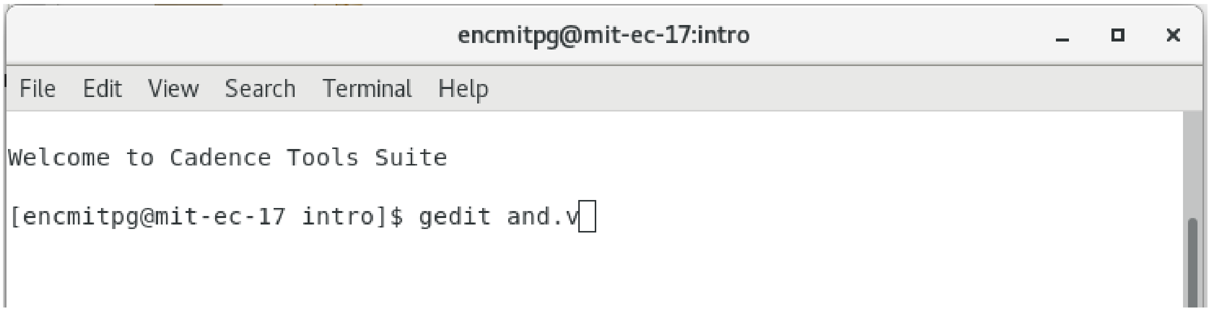
```
Welcome to Cadence Tools Suite

[encmitpg@mit-ec-17 intro]$
```

Fig. 3: Cadence Tools Suite

## Creating Source Codes

- In the Terminal, type `gedit <filename>.v`



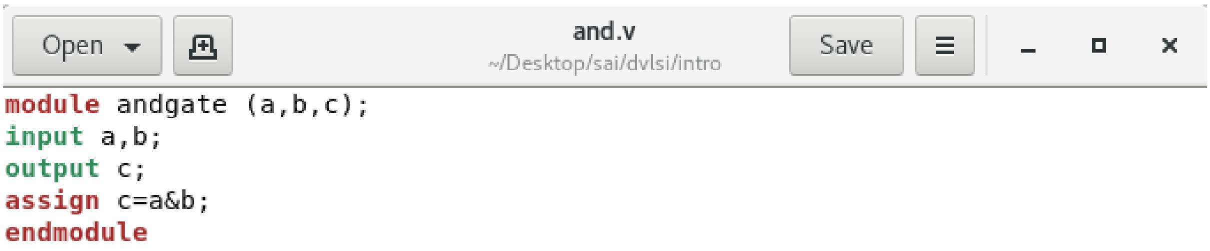
```
Welcome to Cadence Tools Suite

[encmitpg@mit-ec-17 intro]$ gedit and.v
```

Fig. 4: Verilog file creation

- A blank document opens up into which the following source code can be typed down.
- Use Save option or Ctrl+S to save the code or click on the save option from the top most right corner and close the text file.

## Source Code:

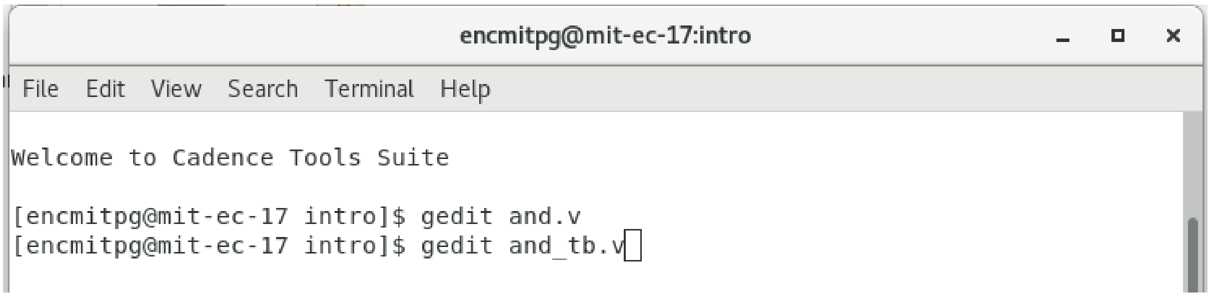


```
module andgate (a,b,c);
input a,b;
output c;
assign c=a&b;
endmodule
```

Fig. 5: source code

## Creating Test bench:

- Similarly, create test bench using `gedit <filename_tb>.v` to open a new blank document.



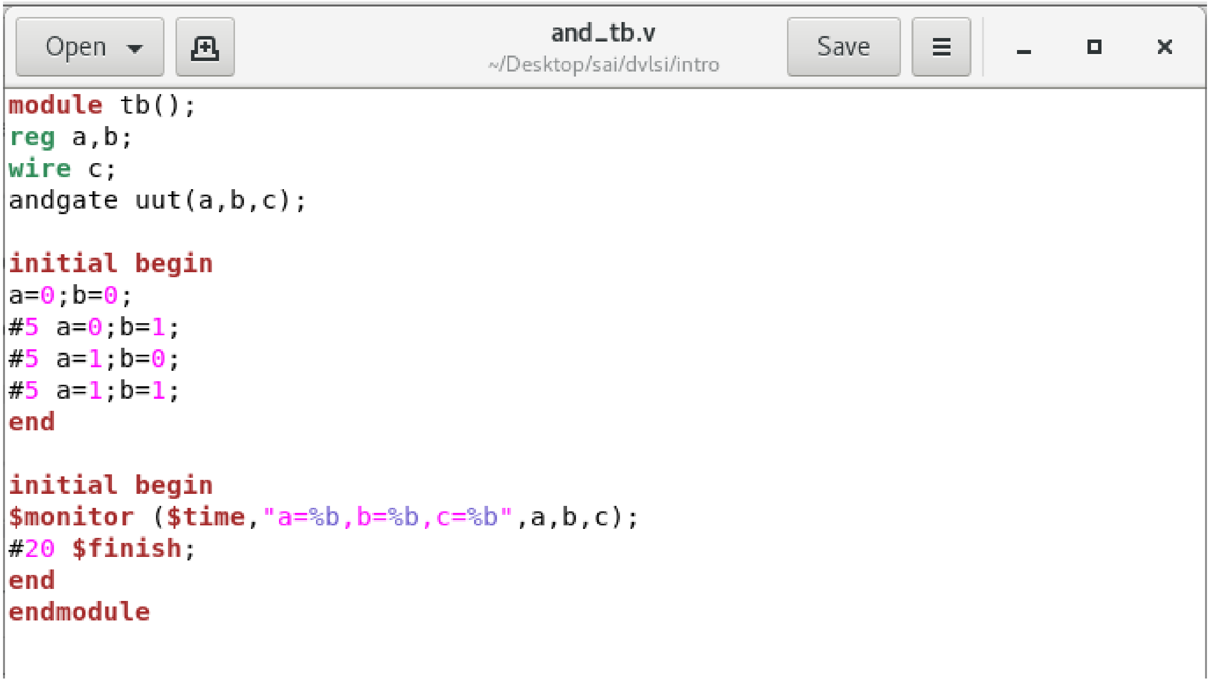
```
Welcome to Cadence Tools Suite

[encmitpg@mit-ec-17 intro]$ gedit and.v
[encmitpg@mit-ec-17 intro]$ gedit and_tb.v
```

Fig. 6: testbench file creation

- Click on the Save option and it will look like the below window and then close the file.

Testbench code:



```
module tb();
reg a,b;
wire c;
andgate uut(a,b,c);

initial begin
a=0;b=0;
#5 a=0;b=1;
#5 a=1;b=0;
#5 a=1;b=1;
end

initial begin
$monitor ($time,"a=%b,b=%b,c=%b",a,b,c);
#20 $finish;
end
endmodule
```

Fig. 7: testbench code

To Launch Simulation tool

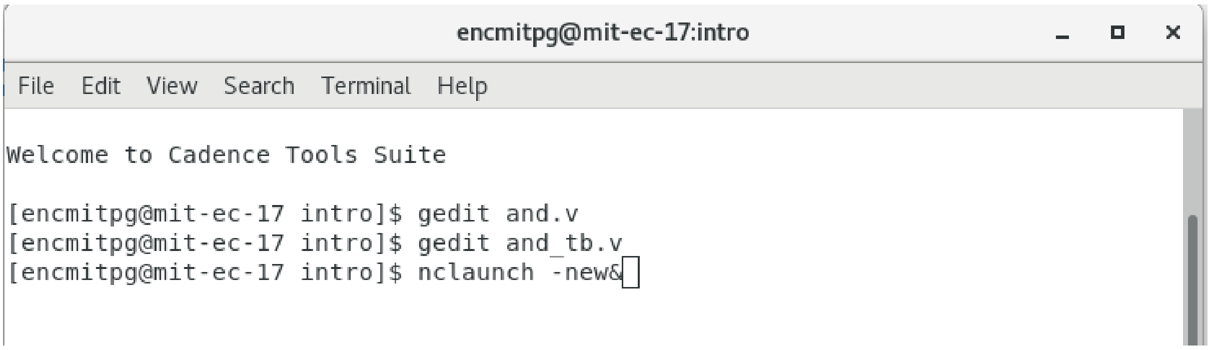
- `nclaunch -new&`

// “-new” option is used for invoking NCVERILOG for the first time for any design

- `nclaunch&`

// On subsequent calls to NCVERILOG

- It will invoke the nclaunch window for functional simulation we can compile, elaborate and simulate it using Multiple.

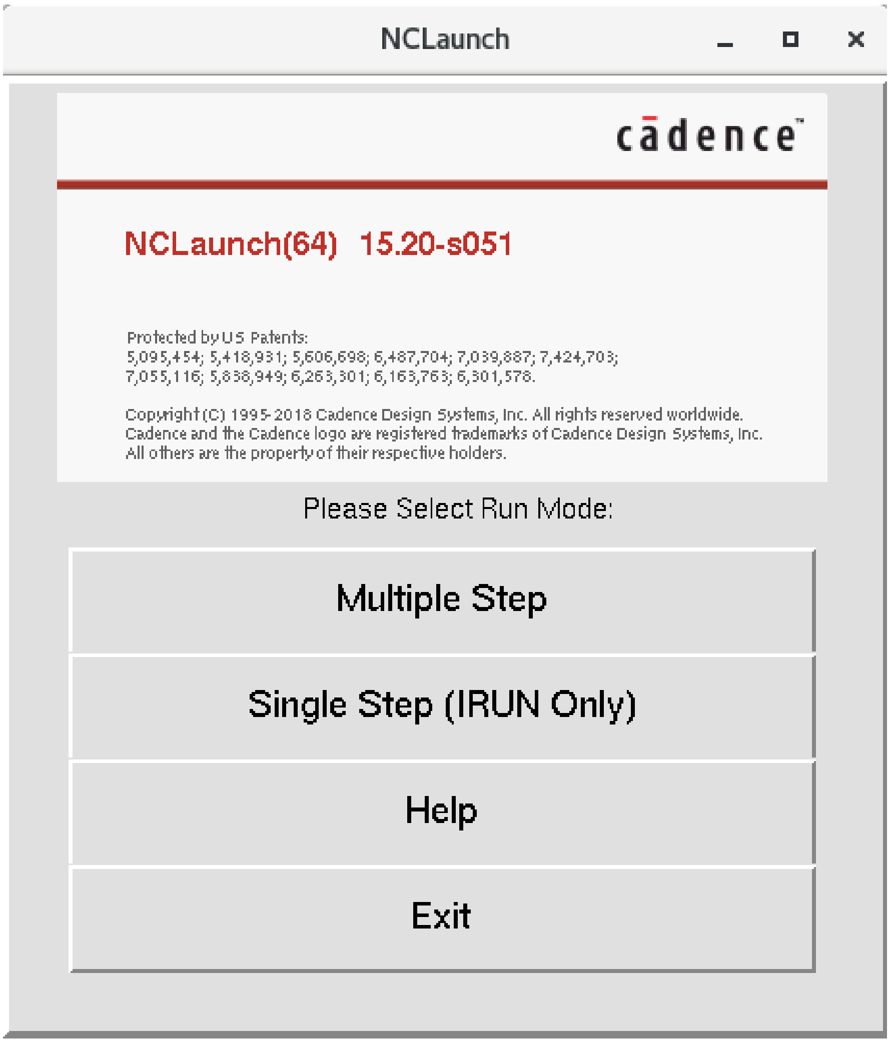


```
Welcome to Cadence Tools Suite

[encmitpg@mit-ec-17 intro]$ gedit and.v
[encmitpg@mit-ec-17 intro]$ gedit and_tb.v
[encmitpg@mit-ec-17 intro]$ nclaunch -new&
```

Fig. 8: nclaunch command

- Select Multiple Step and then select “Create cds.lib File” as shown in below figure.



NCLaunch(64) 15.20-s051

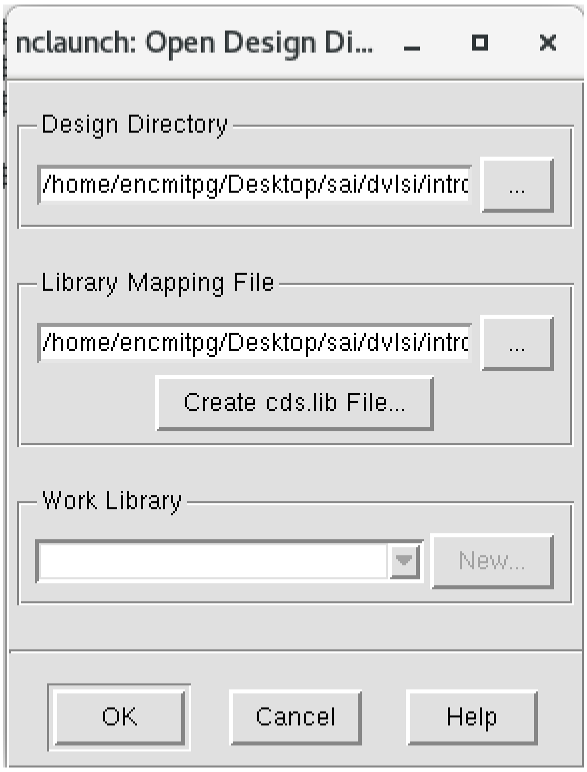
Protected by US Patents:
5,095,454; 5,418,931; 5,606,698; 6,487,704; 7,039,887; 7,424,703;
7,055,116; 5,838,949; 6,263,301; 6,163,763; 6,301,578.

Copyright (C) 1995-2018 Cadence Design Systems, Inc. All rights reserved worldwide.
Cadence and the Cadence logo are registered trademarks of Cadence Design Systems, Inc.
All others are the property of their respective holders.

Please Select Run Mode:

Fig. 9: nclaunch setup selection

- Click the cds.lib file and save the file by clicking on Save option.



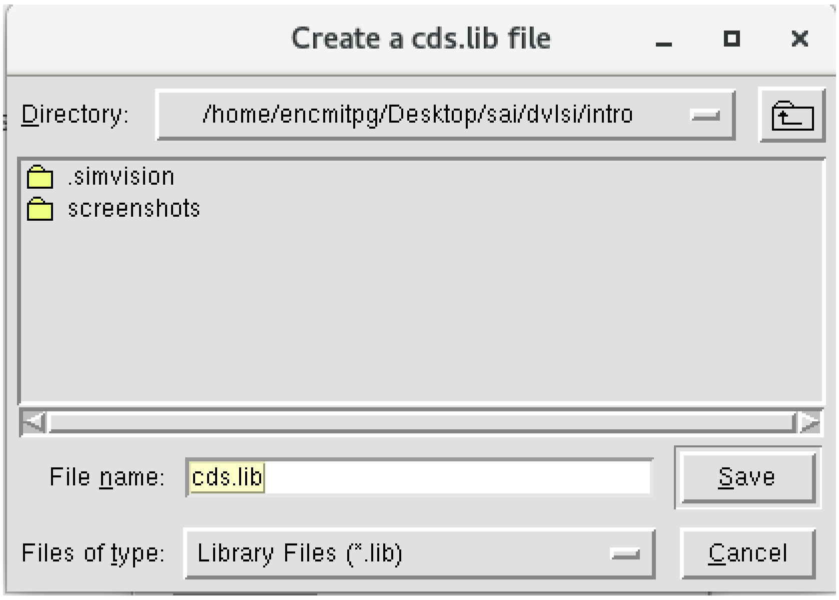
Design Directory: /home/encmitpg/Desktop/sai/dvlsi/intrc

Library Mapping File: /home/encmitpg/Desktop/sai/dvlsi/intrc

Work Library:

Fig. 10: nclaunch after selecting multiple steps

- Save cds.lib file and select the correct option for cds.lib file format based on the HDL Language and Libraries used.



Directory: /home/encmitpg/Desktop/sai/dvlsi/intro

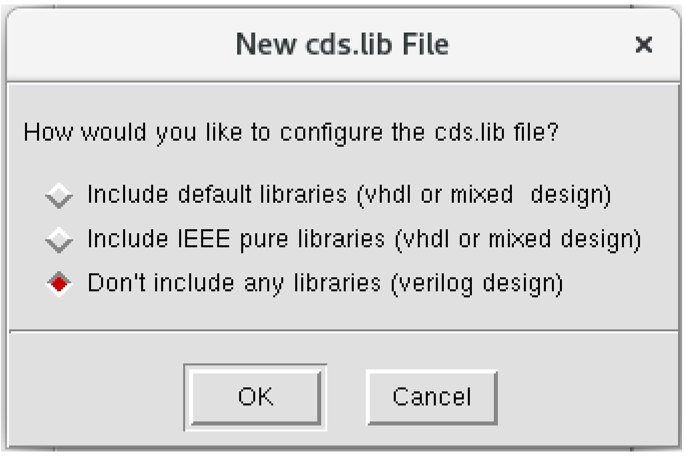
.simvision
screenshots

File name: cds.lib

Files of type: Library Files (*.lib)

Fig. 11: Create cds.lib file and save

- Select “Don’t include any libraries (verilog design)” from “New cds.lib file” and click on “OK” as in below figure.



New cds.lib File

How would you like to configure the cds.lib file?

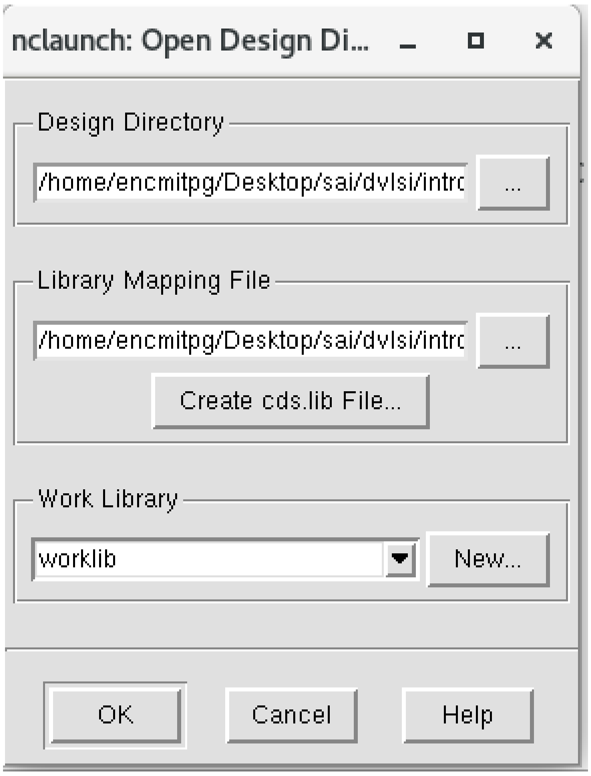
◇ Include default libraries (vhdl or mixed  design)

◇ Include IEEE pure libraries (vhdl or mixed design)

◆ Don't include any libraries (verilog design)

Fig. 12: Don’t include libraries

- We are simulating verilog design without using any libraries
- A Click “OK” in the “nclaunch: Open Design Directory” window as shown in below figure.



nclaunch: Open Design Di...

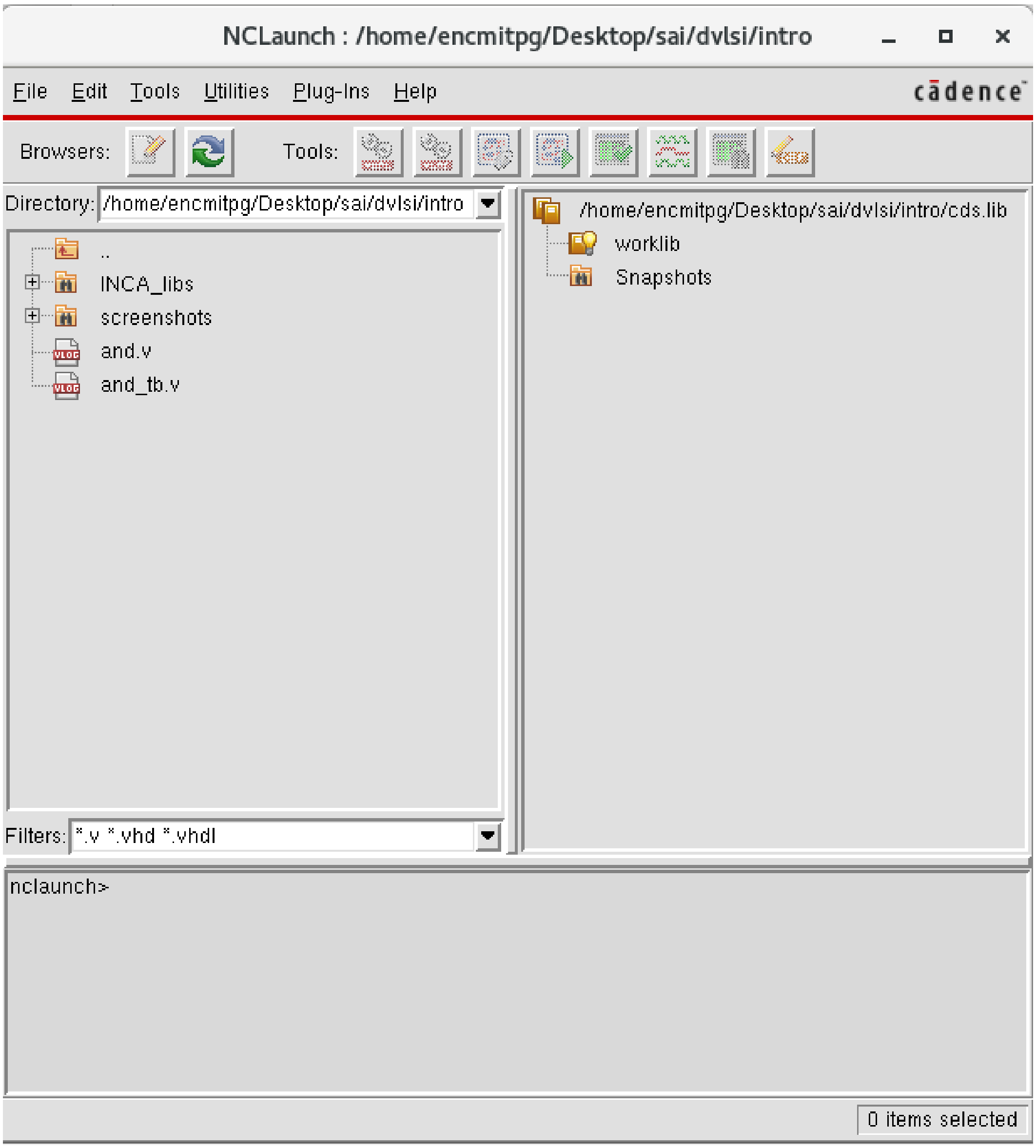
Design Directory: /home/encmitpg/Desktop/sai/dvlsi/intrc

Library Mapping File: /home/encmitpg/Desktop/sai/dvlsi/intrc

Work Library: worklib

Fig. 13: After saving cds.lib file

- A ‘NCLaunch window’ appears as shown in figure below
- Left side you can see the HDL files. Right side of the window has worklib and snapshots directories listed.
- Worklib is the directory where all the compiled codes are stored while Snapshot will have output of elaboration which in turn goes for simulation.


NCLaunch : /home/encmitpg/Desktop/sai/dvlsi/intro
File Edit Tools Utilities Plug-Ins Help
cādence
Browsers:
Tools:

Directory: /home/encmitpg/Desktop/sai/dvlsi/intro

..
INCA_libs
screenshots
and.v
and_tb.v

Filters: *.v *.vhd *.vhdl

/home/encmitpg/Desktop/sai/dvlsi/intro/cds.lib
worklib
Snapshots

nclaunch>

0 items selected

Fig. 14: nclaunch window

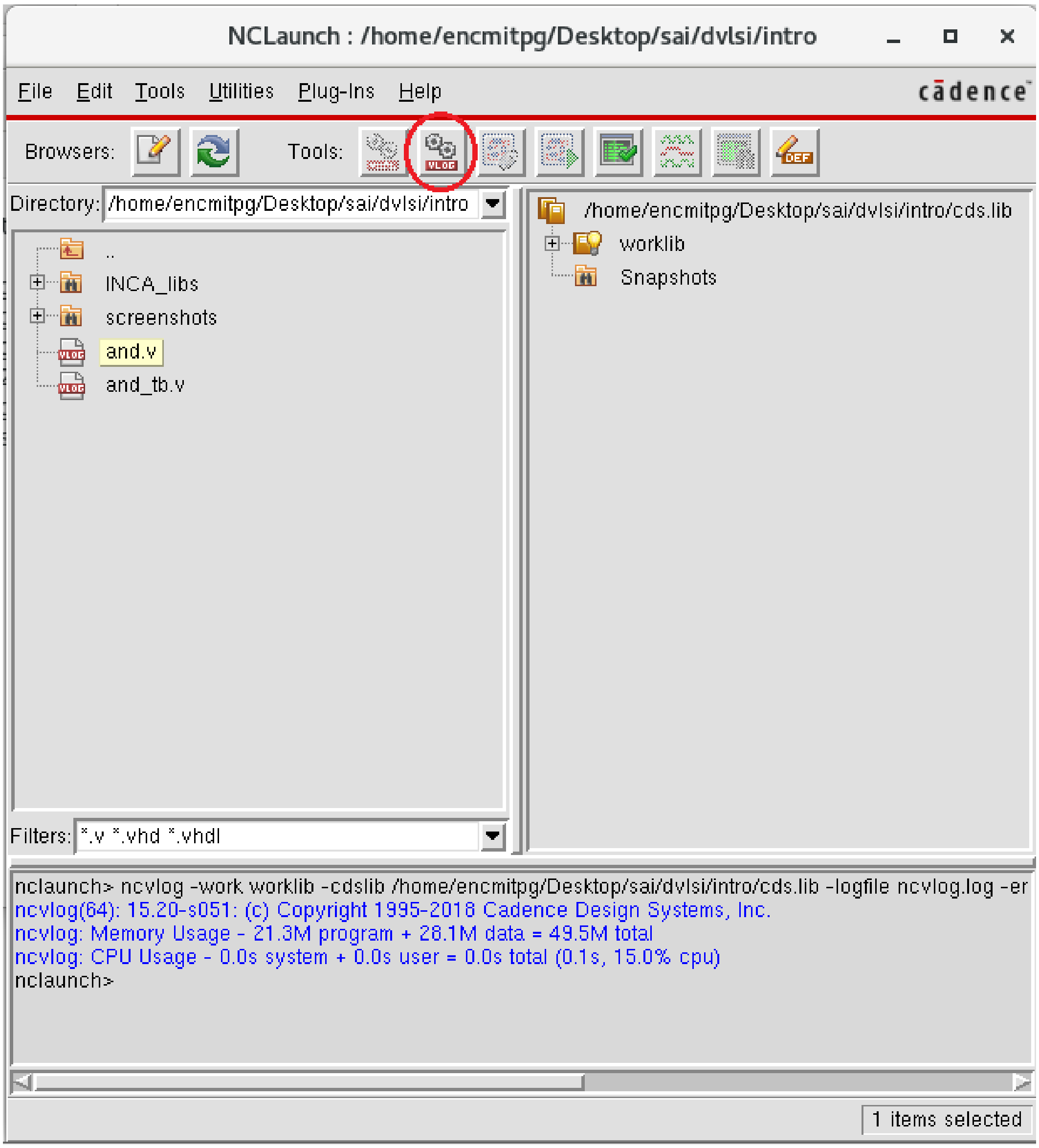
To perform the function simulation, the following three steps are involved Compilation, Elaboration and Simulation.

**Step 1:** Compilation: Process to check the correct Verilog language syntax and usage
Inputs: Supplied are Verilog design and test bench codes
Outputs: Compiled database created in mapped library if successful, generates report else error reported in log file

Steps for compilation:

1. Create work/library directory (most of the latest simulation tools creates automatically).
2. Map the work to library created (most of the latest simulation tools creates automatically).
3. Run the compile command with compile options.

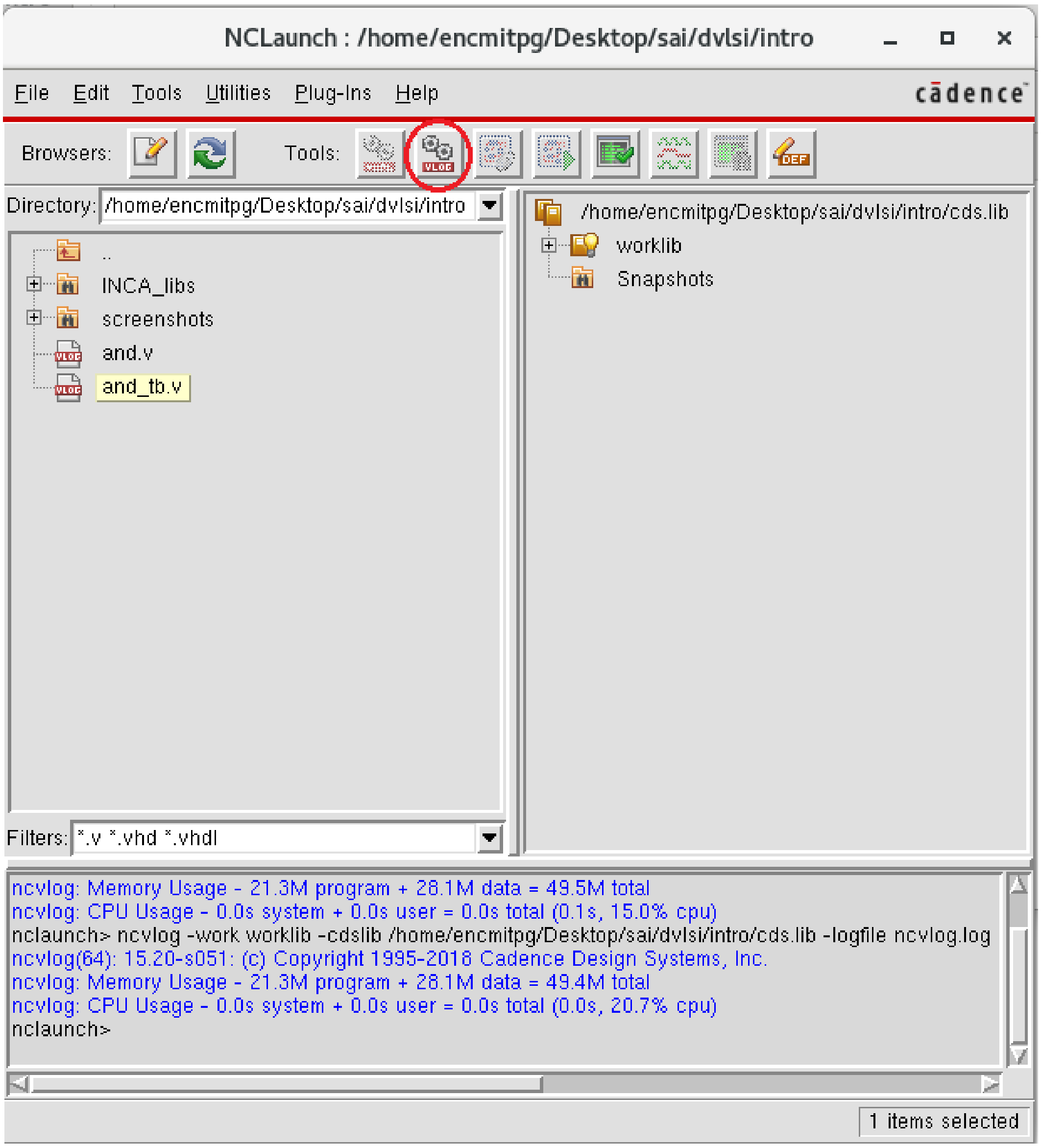
- Left side select the file and in Tools: launch verilog compiler with current selection will get enable. Click it to compile the code
- Worklib is the directory where all the compiled codes are stored while Snapshot will have output of elaboration which in turn goes for simulation


NCLaunch : /home/encmitpg/Desktop/sai/dvlsi/intro
File Edit Tools Utilities Plug-Ins Help
cādence
Browsers:
Tools:
Directory: /home/encmitpg/Desktop/sai/dvlsi/intro
..
INCA_libs
screenshots
and.v
and_tb.v
/home/encmitpg/Desktop/sai/dvlsi/intro/cds.lib
worklib
Snapshots
Filters: *.v *.vhd *.vhdl
1 items selected

```
nclaunch> ncvlog -work worklib -cdslib /home/encmitpg/Desktop/sai/dvlsi/intro/cds.lib -logfile ncvlog.log -er
ncvlog(64): 15.20-s051: (c) Copyright 1995-2018 Cadence Design Systems, Inc.
ncvlog: Memory Usage - 21.3M program + 28.1M data = 49.5M total
ncvlog: CPU Usage - 0.0s system + 0.0s user = 0.0s total (0.1s, 15.0% cpu)
nclaunch>
```

Fig. 15: select Verilog code and launch Verilog compiler

- After compilation it will come under worklib you can see in right side window.
- Select the test bench and compile it. It will come under worklib. Under Worklib you can see the module and test-bench.


NCLaunch : /home/encmitpg/Desktop/sai/dvlsi/intro
File Edit Tools Utilities Plug-Ins Help
cadence
Browsers:
Tools:
Directory: /home/encmitpg/Desktop/sai/dvlsi/intro
..
INCA_libs
screenshots
and.v
and_tb.v
/home/encmitpg/Desktop/sai/dvlsi/intro/cds.lib
worklib
Snapshots
Filters: *.v *.vhd *.vhdl
1 items selected

```
ncvlog: Memory Usage - 21.3M program + 28.1M data = 49.5M total
ncvlog: CPU Usage - 0.0s system + 0.0s user = 0.0s total (0.1s, 15.0% cpu)
nclaunch> ncvlog -work worklib -cdslib /home/encmitpg/Desktop/sai/dvlsi/intro/cds.lib -logfile ncvlog.log
ncvlog(64): 15.20-s051: (c) Copyright 1995-2018 Cadence Design Systems, Inc.
ncvlog: Memory Usage - 21.3M program + 28.1M data = 49.4M total
ncvlog: CPU Usage - 0.0s system + 0.0s user = 0.0s total (0.0s, 20.7% cpu)
nclaunch>
```

Fig. 16: select Verilog code and launch Verilog compiler

- The cds.lib file is an ASCII text file. It defines which libraries are accessible and where they are located. It contains statements that map logical library names to their physical directory paths. For this Design, you will define a library called “worklib”.

**Step 2:** <u>Elaboration</u>: To check the port connections in hierarchical design

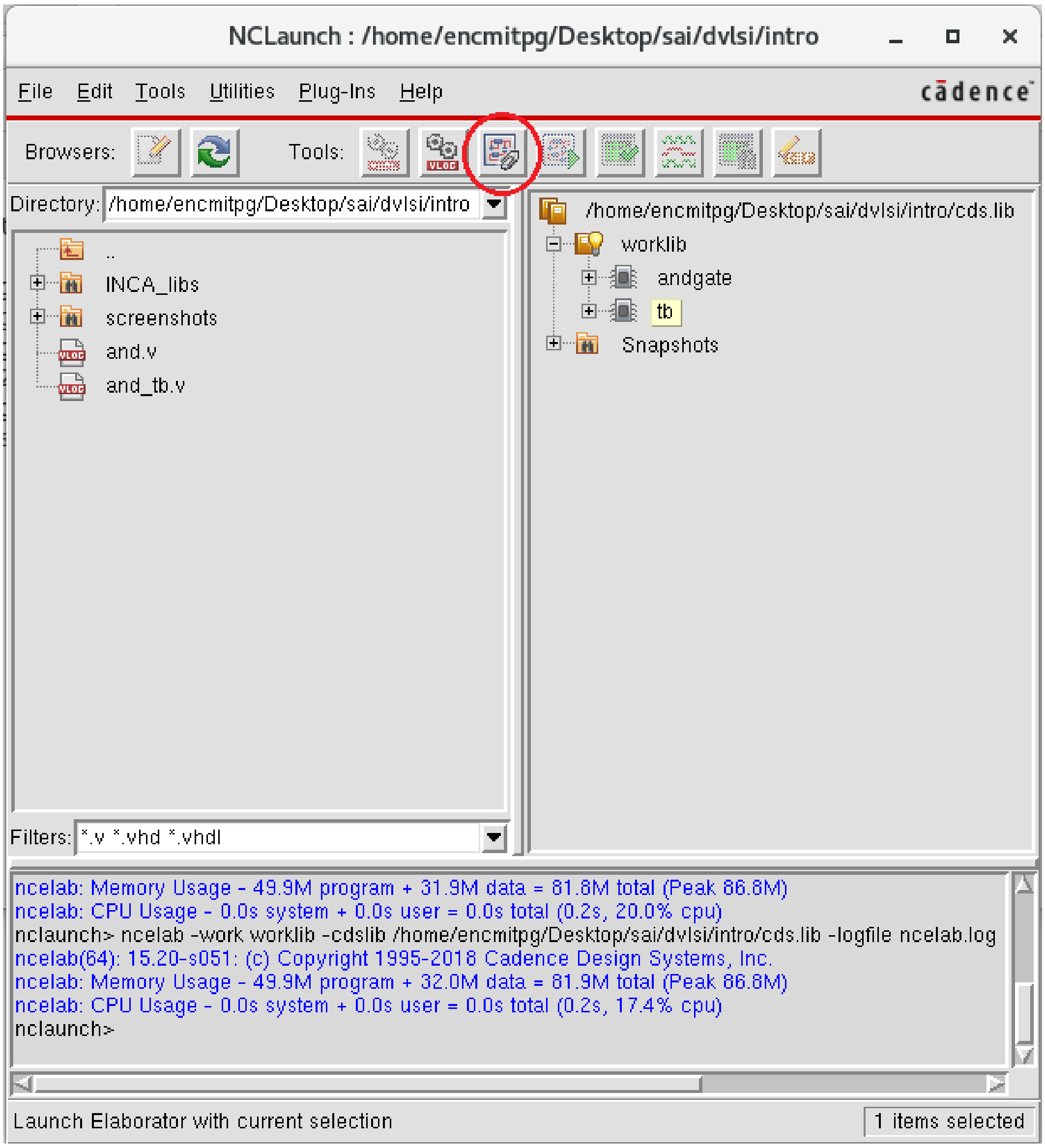
<u>Inputs</u>: Top level design/test bench Verilog codes

<u>Outputs</u>: Elaborate database updated in mapped library if successful, generates report else error reported in log file

Steps for elaboration – Run the elaboration command with elaborate options

1. It builds the module hierarchy.
2. Binds modules to module instances.
3. Computes parameter values.
4. Checks for hierarchical names conflicts.
5. It also establishes net connectivity and prepares all of this for simulation.

- After elaboration the file will come under snapshot. Select the test bench and elaborate it.


NCLaunch : /home/encmitpg/Desktop/sai/dvlsi/intro
File Edit Tools Utilities Plug-Ins Help
cādence
Browsers:
Tools:
Directory: /home/encmitpg/Desktop/sai/dvlsi/intro
..
INCA_libs
screenshots
and.v
and_tb.v
Filters: *.v *.vhd *.vhdl
/home/encmitpg/Desktop/sai/dvlsi/intro/cds.lib
worklib
andgate
tb
Snapshots

```
ncelab: Memory Usage - 49.9M program + 31.9M data = 81.8M total (Peak 86.8M)
ncelab: CPU Usage - 0.0s system + 0.0s user = 0.0s total (0.2s, 20.0% cpu)
nclaunch> ncelab -work worklib -cdslib /home/encmitpg/Desktop/sai/dvlsi/intro/cds.lib -logfile ncelab.log
ncelab(64): 15.20-s051: (c) Copyright 1995-2018 Cadence Design Systems, Inc.
ncelab: Memory Usage - 49.9M program + 32.0M data = 81.9M total (Peak 86.8M)
ncelab: CPU Usage - 0.0s system + 0.0s user = 0.0s total (0.2s, 17.4% cpu)
nclaunch>
```

Launch Elaborator with current selection

1 items selected

Fig. 17: Elaboration Launch Operation

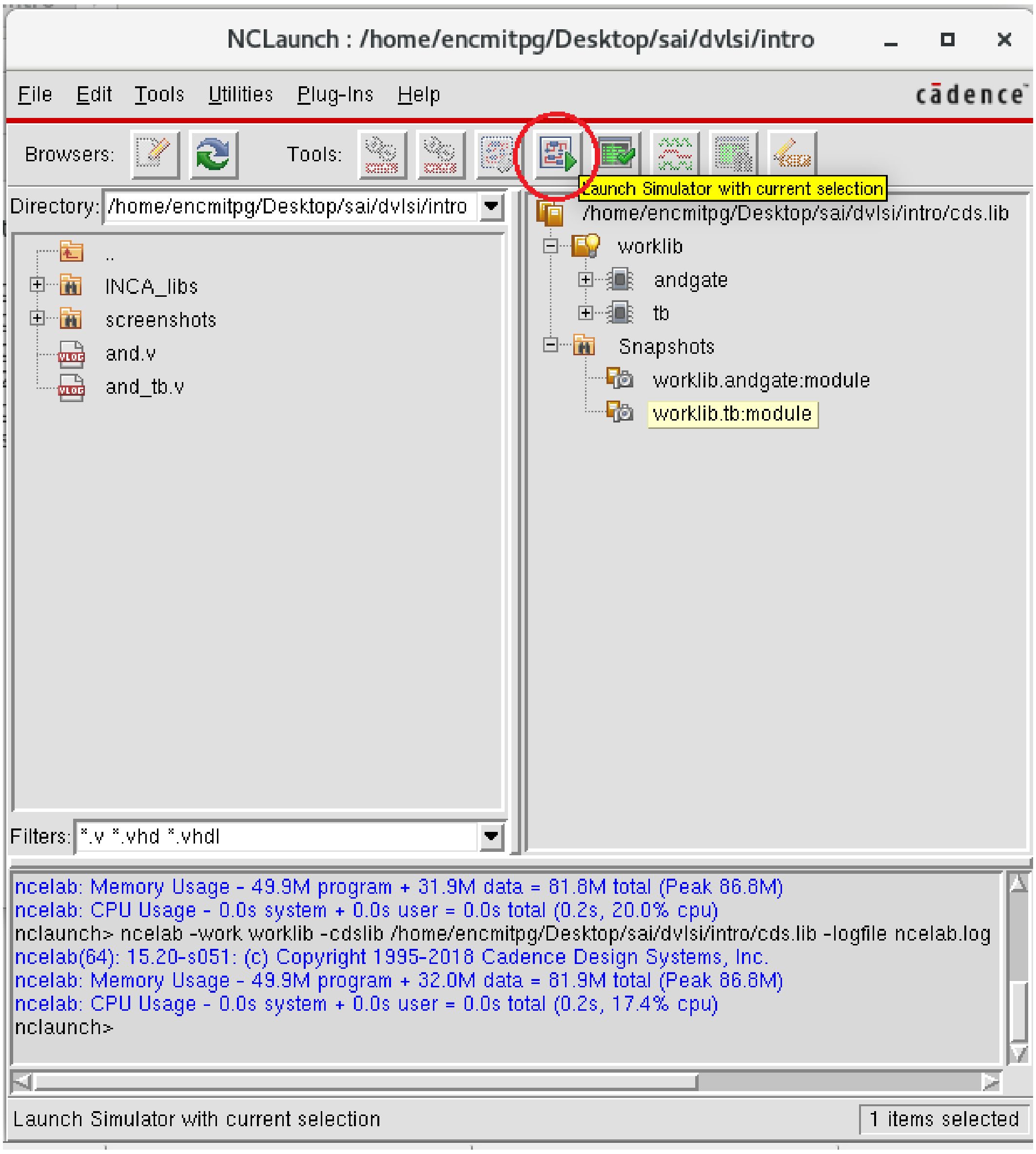
**Step 3:** Simulation: Simulate with the given test vectors over a period of time to observe the output behaviour.

Inputs: Compiled and Elaborated top level module name.

Outputs: Simulation log file, waveforms for debugging. Simulation allows to dump design and test bench signals into a waveform.

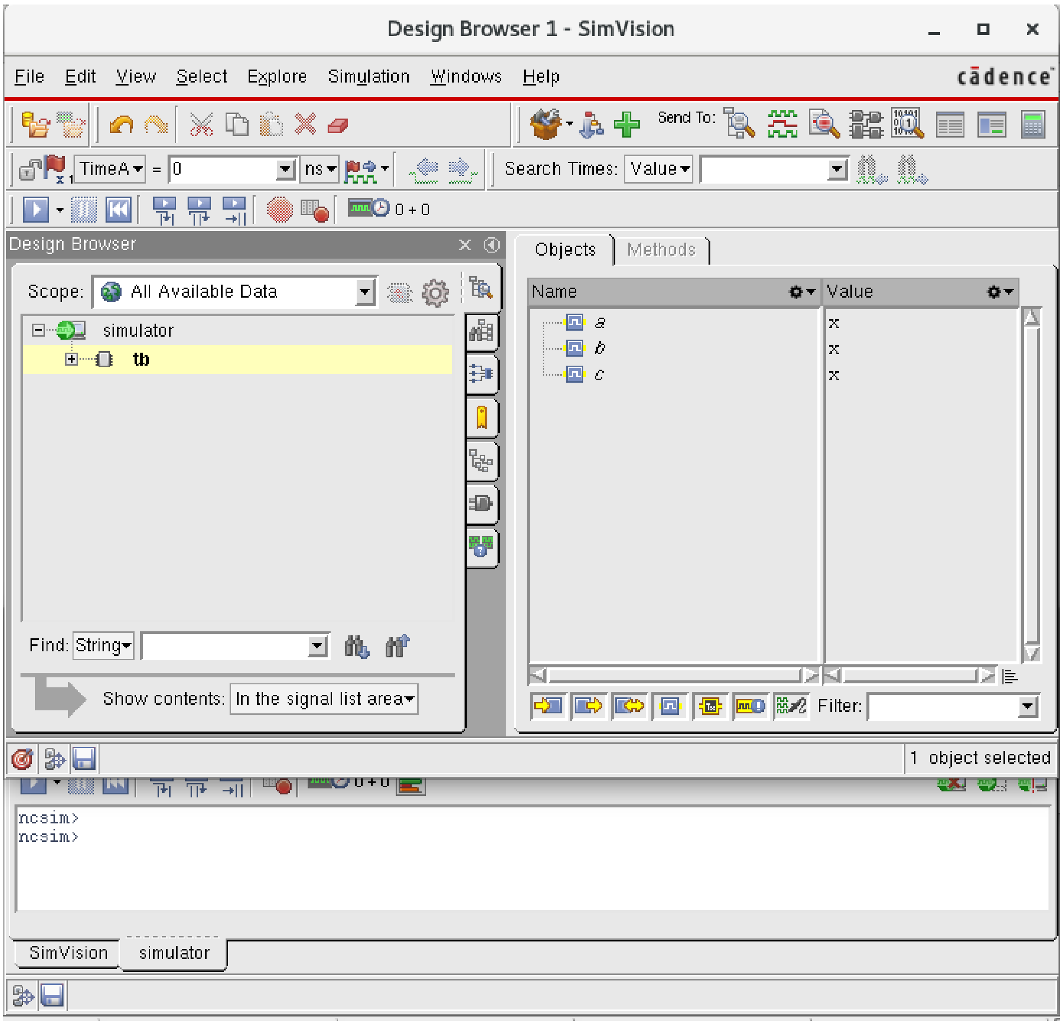
Steps for simulation – Run the simulation command with simulator options.

- Instead of nclaunch, design file and testbench can be run using single irun command.
  - `irun and.v and_tbv -access +rwc -gui`


NCLaunch : /home/encmitpg/Desktop/sai/dvlsi/intro
File Edit Tools Utilities Plug-Ins Help
cādence
Browsers:
Tools:
Launch Simulator with current selection
Directory: /home/encmitpg/Desktop/sai/dvlsi/intro
..
INCA_libs
screenshots
and.v
and_tb.v
/home/encmitpg/Desktop/sai/dvlsi/intro/cds.lib
worklib
andgate
tb
Snapshots
worklib.andgate:module
worklib.tb:module
Filters: *.v *.vhd *.vhdl
Launch Simulator with current selection
1 items selected

```
ncelab: Memory Usage - 49.9M program + 31.9M data = 81.8M total (Peak 86.8M)
ncelab: CPU Usage - 0.0s system + 0.0s user = 0.0s total (0.2s, 20.0% cpu)
nclaunch> ncelab -work worklib -cdslib /home/encmitpg/Desktop/sai/dvlsi/intro/cds.lib -logfile ncelab.log
ncelab(64): 15.20-s051: (c) Copyright 1995-2018 Cadence Design Systems, Inc.
ncelab: Memory Usage - 49.9M program + 32.0M data = 81.9M total (Peak 86.8M)
ncelab: CPU Usage - 0.0s system + 0.0s user = 0.0s total (0.2s, 17.4% cpu)
nclaunch>
```

Fig. 18: Simulation Launch Operation



Scope: All Available Data

Show contents: In the signal list area

Filter:

| Name | Value |
|---|---|
| a | x |
| b | x |
| c | x |

```
ncsim>
ncsim>
```

Fig. 19: In SimVision window, right click on the tb module and select “send to simulation window”

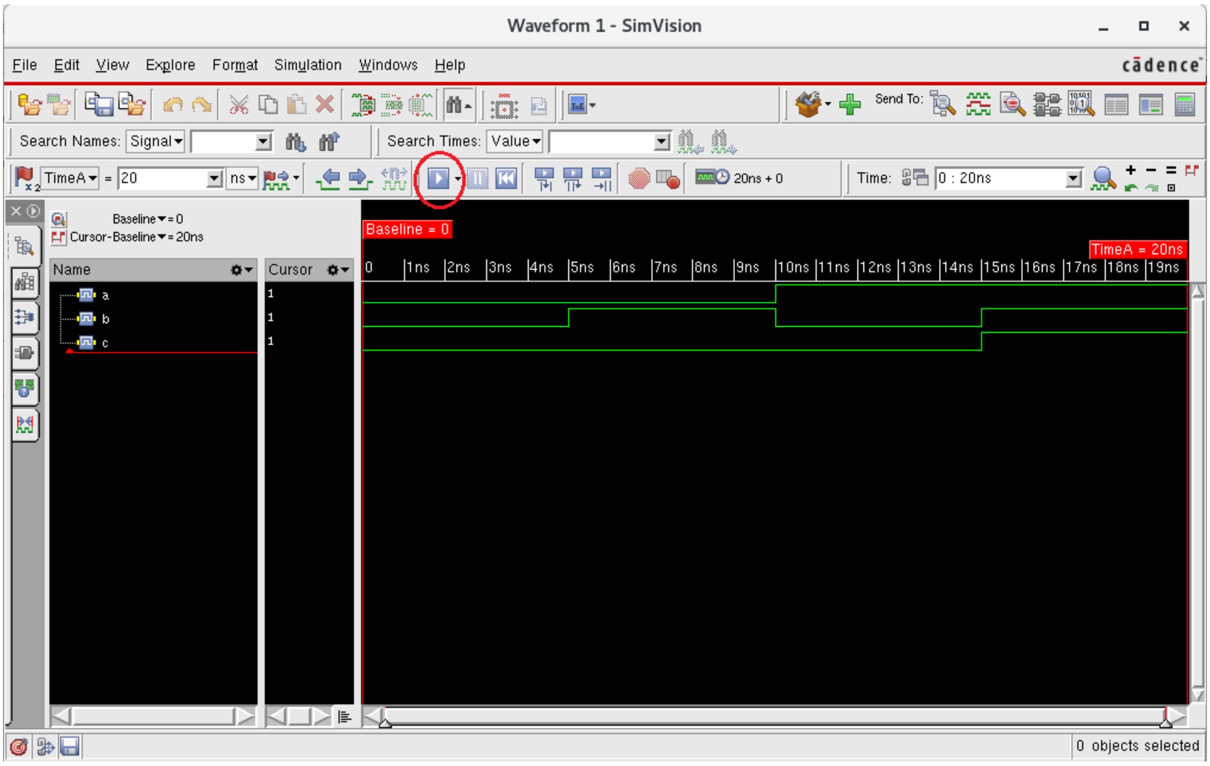
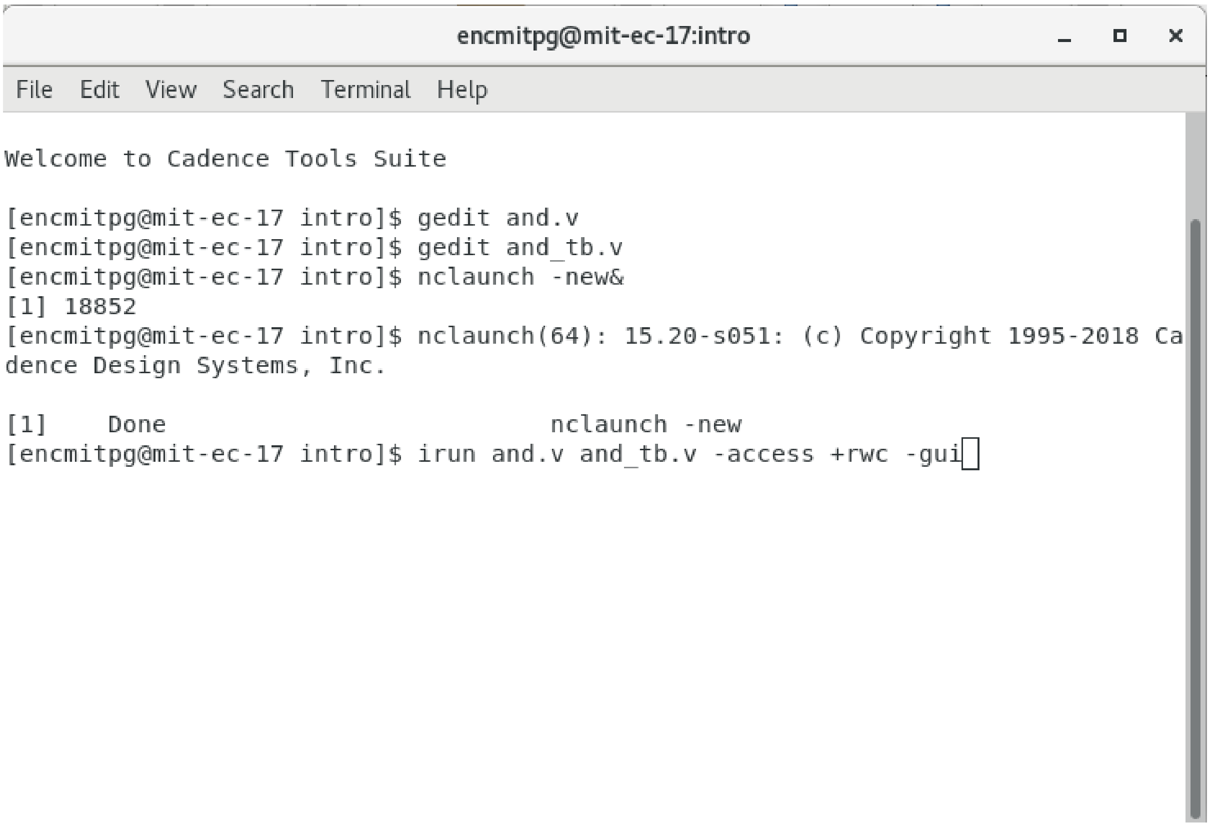


Fig. 20: Click on Run simulation to get waveforms



```
Welcome to Cadence Tools Suite

[encmitpg@mit-ec-17 intro]$ gedit and.v
[encmitpg@mit-ec-17 intro]$ gedit and_tb.v
[encmitpg@mit-ec-17 intro]$ nclaunch -new&
[1] 18852
[encmitpg@mit-ec-17 intro]$ nclaunch(64): 15.20-s051: (c) Copyright 1995-2018 Ca
dence Design Systems, Inc.

[1]    Done                          nclaunch -new
[encmitpg@mit-ec-17 intro]$ irun and.v and_tb.v -access +rwc -gui
```

Fig. 21: irun single command to open SimVision