# **CADENCE TUTORIAL**



# San Diego State University, Department of Electrical and Computer Engineering

Amith Dharwadkar and Ashkan Ashrafi

# Contents

1)	Introduction	3
2)	Connecting to the Volta server	4
	Creating a work directory	
4)	RTL Simulation	8
5)	NETLIST Simulation	12
6)	Layout Design	16

#### ABSTRACT

This tutorial is aimed at introducing a user to the CADENCE tool. It gives step by step approach to performing a RTL simulation, gate level synthesis/simulation and finally layout design using SOC ENCOUNTER's auto place and route with TSMC 0.13 µm standard cell library. The tutorial however does not discuss installation and environment setup for CADENCE. The entire tutorial is organized into five chapters beginning with connecting to Volta server on which CADENCE resides. It then explains RTL simulation, gate-level synthesis, post-synthesis simulation and layout design using encounter.

#### Connecting to the Volta server

There are two ways of connecting to the Volta server on which CADENCE is installed.

Connecting remotely – X-Win32 or Secure Shell Client can be used to establish a connection to the Volta server. X-Win32/SSH client can be downloaded from SDSU college website - <u>http://scc.sdsu.edu/downloads.php</u>.

SSH client installation steps are given here. X-Win32 can be installed in the same manner and a connection established with the Volta server.

Upon SSH client installation, click on the SSH client icon to invoke SSH. From the pop-up pane click on the File menu tab and select 'Quick Connect' option to connect to the Volta server.



Specify the hostname and the username and click on Connect button.

HostName – volta.sdsu.edu

Username and Password – Enter the username and password provided to you.

- Connecting from a school computer The second approach to connect to the VOLTA server is to use the LINUX machines available on-campus. Login to the Linux machines using the username and password provided to you. Invoke the command prompt. Verify if a connection has been established with the Volta server. If the connection is not established, execute the following command at the prompt,
  - 1. Type **ssh -X volta.sdsu.edu** at the command prompt.
  - 2. Enter username and password to login to the Volta server.

#### Creating a work directory

The first step in using CADENCE effectively is to create a work directory/folder. A work directory enables the user to organize the code files. The user can create folder for every project that is implemented on CADENCE. The following steps need to be followed in order to create a work folder,

• Ensure that a folder called **cadence** is available in your home directory. Every user on the Volta server is assigned a home directory to which he/she has exclusive access. The home directory generally has the same name as the Username. Following commands could be used to verify the presence of cadence directory/folder in the home directory.

# >> cd /home/student/<Enter username> >> ls

See if the cadence folder is present in this folder.

- Change to cadence directory using the command
   > cd cadence
- This directory should have the following three files for CADENCE to compile verilog files without any errors. The files are,
  - 1. Compile bgx.scr
  - 2. Tpd013n2.v
  - 3. Tcb013ghp.v

🕮 2:volta.sdsu.edu - default - SSH Secure Shell	
File Edit View Window Help	
🔲 🖨 🐧 🔎 🎉 🛍 💼 🙀 💭 🎭 🤣 🛠	
S Ouick Connect Profiles	
-rwxrwxrwx 1 dharwadk student 18086 May 16 19:05 ac_shell.cmd	
-rwxrwxrwx 1 dharwadk student 13614 May 16 19:05 ac_shell.log	
-rwxrwxrwx 1 dharwadk student 3640 Apr 13 2009 Add4.vhd	
-rwxrwxrwx 1 dharwadk student 4456 Apr 13 2009 ButterFly16.vhd	
-rwxrwxrwx 1 dharwadk student 2179 Apr 13 2009 CarryGenFinal.vhd	
-rwxrwxrwx 1 dharwadk student 2645 Apr 13 2009 CarryGen.vhd	
-rwxrwxrwx 1 dharwadk student 111 Feb 24 12:47 cds.11b	
-rwxrwxrwx 1 dharwadk student 3505 Mar 2 20:56 #compile bgx.scr#	
rw-rr 1 dharwadk student 6185 May 16 19:05 compile bgx.sor drwxrwyrwx 3 dharwadk student 4096 Feb 24 00:17 meree	
Grwarwarwa 2 dharwadk student 4096 FeD 24 0017 counter trad	
drwxrwxrwx 2 dharwadk student 4096 her 22 15:15 mail tea drwxr-xr-x 6 dharwadk student 4096 her 16 18:40 DualTree	
Grwxr-xr-x 6 Gharwadk Student 4096 May 16 18:40 Dualree	
drwarwarwar 5 dharwadk student 4096 Apr 13 2009 pri	
-rwxxwxrwx 1 dharwadk student 49426 Apr 13 2009 FFT main.vhd	
warwarwa 1 dharwada student 4959 Apr 13 2009 FFTest.vhd	
dwxrwxwx 4 dharwadk student 4096 Mar 1 19:31 Thirds on the	
-rwxrwxrwx 1 dharwadk student 2667 Jan 23 2009 gds2 encounter.map	
-rwxrwxrwx 1 dharwadk student 67 Feb 23 2009 hdl.var	
dwarwarwa 3 dharwadk student 4096 Feb 23 2009 THEATING	
-rwarwarwa 1 dharwadk student 4884 Apr 13 2009 multiplier.vhd	
-rwxrwxrwx 1 dharwadk student 2041 Mar 1 17:38 ncelab.log	
-rwarwarwa 1 dharwadk student 0 Mar 2 17:34 nclaunch.key	
-rwxrwxrwx 1 dharwadk student 134 Feb 25 00:12 ncsim.key	
-rwxrwxrwx 1 dharwadk student 623 Feb 25 00:12 ncsim.log	
-rwxrwxrwx 1 dharwadk student 276 Mar 1 17:40 ncvhdl.log	
-rwxrwxrwx 1 dharwadk student 211 Mar 2 14:54 ncvlog.log	
drwxr-xr-x 3 dharwadk student 4096 May 14 19:42 practice	
drwxrwxrwx 3 dharwadk student 4096 Feb 22 16:23 NA work	
-rwxrwxrwx 1 dharwadk student 894041 Jan 23 2009 tcb013ghp.v	
-rwxrwxrwx 1 dharwadk student 4929 Jan 26 21:54 temp	
-rwxrwxrwx 1 dharwadk student 13853 Mar 1 18:37 test.txt	
drwxrwxrwx 24 dharwadk student 4096 Apr 14 11:35 thesis	
drwxrwxrwx 24 dharwadk student 4096 May 7 02:50 thesis_veri	
drwxrwxrwx 24 dharwadk student 4096 May 7 02:10 Thesis veri	
drwxr-xr-x 24 dharwadk student 4096 May 13 23:30 thesis_ver3	
-rwxrwxrwx 1 dharwadk student 3819 May 13 16:16 tompile pgx.scr	
rrwxrwxrwx 1 dharwadk student 13698 Jan 23 2009 tpd013n2.v	
drwxrwxrwx 2 dharwadk student 4096 Feb 23 2009	
arwarwarwa 2 anarwada student 4096 FeD 24 2009 owni drwarwarwa 2 dharwada student 4096 FeD 25 0011 www.anar	
GIVETRAL 2 GRATHACK Student 4090 FED 2 00111 metric and 5 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
• Lograndi Lograndi Anna Anna Anna Anna Anna Anna Anna Ann	
/ Connected to volta.sdsu.edu	SSH2 - aes128-cbc - hmac-md5 - nc 178x42
Contected to voitalsusaleau	55112 - BESTED-CDC - NMBC-MOD - NC 170X42

**Compile\_bgx.scr** is a script that **bgx\_shell** uses in order to compile the user created Verilog files. The compile\_bgx.scr script contains module path and all other details necessary for

generating an object file. The remaining two files contain delay information for TSMC 0.13um standard cells that would be used in our design.

Notes - If these files are not present then they need to be copied into cadence folder.

- Once the cadence folder is created and all the necessary files copied, the work folder has to be created in the cadence directory. The work directory will house all the Verilog files and test benches. The following command can be used to do the same.
- >> **mkdir counter** // this creates a folder named counter
- >> **cd counter** // Changes the directory to counter.



• Now create a folder named **encounter** in the work directory. This folder will house files that are necessary for proper execution of the SOC ENCOUNTER tool. The SOC ENCOUNTER tool could be used to generate the layout for compiled/synthesized Verilog/netlist file.

Commands to be executed for creating the encounter folder,

>> mkdir encounter
>> cd encounter

*Note – Before executing these commands ensure that you are in the work directory (which is "Counter" in the current example).* 

- Copy the files 1) encounter.conf 2) encounter.tcl 3) encounter\_power.tcl 4) gds2\_encounter.map into these folder. These files are necessary for proper functioning of the ENCOUNTER tool and should be made available to you before you start working with Cadence.
- Also copy 1) Tpd013n2.v 2) Tcb013ghp.v into encounter folder. These contain delay information and definition of TSMC 0.13um standard cells/gates. These files are used when verifying the synthesized code using the Verilog-XL compiler.
- Copy both the Verilog as well as the testbench file into the encounter folder. You could use a SSH client to copy the already created files or create new Verilog and testbench files using a VI editor.

*Note* – *Steps to Create verilog and testbench files are beyond the scope of this tutorial.* 

S20 [uolta]/home/student/dharwadk/caderoce/Counter/wncounter) III gds2_encounter.onf encounter.tcl encounter_power.tcl Counter.v Counter_TB.v tcb013ghp.v tpd013n2.v         rup=rr=1 dharwadk student       358 Har 2 16:50 Counter.JB.v         rup=rr=1 dharwadk student       358 Har 2 16:50 Counter.JB.v         rup=rr=1 dharwadk student       358 Har 2 16:50 Counter.JB.v         rup=rr=1 dharwadk student       364 Te 2 16:50 Counter.JB.v         rup=rr=1 dharwadk student       364 To 41:50 encounter.poorf         rup=rr=1 dharwadk student       365 Be 17 04:59 encounter.poorf         rup=rr=	

#### **RTL Simulation using "Verilog-XL" compiler**

A good design practice dictates that a designer first carry out RTL simulation in order to validate the design and then eventually move over to gate level synthesis. A RTL simulation lets us know if the behavior of the component is as desired.

In order to carry out RTL simulation we can use either

- 1) Verilog-XL compiler.
- 2) NCVERILOG and NCSIM(simvision).

This tutorial describes the use of Verilog-XL compiler of CADENCE in order to carry out RTL simulation. The following command has to be executed to invoke the compiler,

#### >> verilog main\_file.v test\_bench.v

The user has to pay attention when specifying the files names. The files have to be specified in a particular order such that the lower-level modules are compiled before the higher-level modules. The testbench would be the last item to be compiled. Upon execution of the above statement the simulation results are displayed on the terminal. (For the simulation results to be displayed on the terminal the designer has to include "display" or "monitor" tasks in the testbench).

X	xterm	
	14 [volta]/home/student/dharwadk/cadence/Counter/encounter/ verilog Counter.v Counter_TB.v Tool: VERILOG-XL 08.20.001-p Har 1, 2010 17:07:20	
	Copyright (c) 1995-2004 Cadence Design Systems, Inc. All Rights Reserved. Umpublished — rights reserved under the copyright laws of the United States.	
	Copyright (c) 1995-2004 UNIX Systems Laboratories, Inc. Reproduced with Permission. compiler	
	THIS SOFTWARE AND ON-LINE DOCUMENTATION CONTAIN CONFIDENTIAL INFORMATION AND TRADE SECRETS OF CADENCE DESIGN SYSTEMS, INC. USE, DISCLOSURE, OR REFORMACTION FRONTIBITED WITHOUT THE PRIOR EXPRESS WRITTEN PERMISSION OF RESTRICTED RIGHTS LEGEND	
	Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(1) of the Rights in Technical Data and Computer Software classes at DFRRS 252,227-7013 or subparagraphs (c)(1) and (2) of Commercial Computer Software — Restricted Rights at 40 CFR 52,227-19, as applicable.	
	Cadence Besign Systems, Inc. 955 River Daks Parkway San Jose, California \$5124	
	For technical assistance please contact the Cadence Response Center at 1-877-CIS-4911 or send email to supportBoadence.com	
	For more information on Cadence's Verilog-XL product line send email to talkv@cadence.com	
	Compiling source file "Counter_v" Compiling source file "Counter_IB.v" Highest level modules: Counter_IB	
	0       : Count value is 0000         50       : Count value is 0001         150       : Count value is 0001         250       : Count value is 0001         350       : Count value is 0001         450       : Count value is 0010         450       : Count value is 0110         550       : Count value is 0111         550       : Count value is 0110	
	850 : Count value is 0111 950 : Count value is 1000 1050 : Count value is 1001 1150 : Count value is 1010	
	L12 "Counter_TB.v": #Finish at simulation time 12000 0 simulation events (use sprofile or +listcounts option to count) CPU time: 0.0 secs to compile + 0.0 secs to link + 0.0 secs in simulation End of Tool: VERLUG-M. 06.20.001P. Har 1.2010 [17:07:21	
	15 [volta]/howe/student/dharwadk/cadence/Counter/encounter>	

#### GATE LEVEL SYNTHESIS

Gate level synthesis involves implementing the behavior of the circuit (described by a Verilog model) using standard gates. In gate level synthesis, the Verilog file is synthesized into a netlist file which includes standard gates and their delays. In any design process the simulation of this gate-level netlist will eventually shows the success or failure of the design. Along with the netlist file the synthesis script also generates timing-report, area report and power report. The timing report is a critical piece of information. It gives details about critical path which indicates whether the circuit meets the timing/frequency requirements.

Steps to carry out gate level synthesis,

The **compile\_bgx.scr** script is used to generate the gate level netlist. The following changes need to be made to the script in order to generate the gate level netlist file(which will have a .vh extension) and .sdc file which would contain the timing information for standard gates present in the netlist file. The .vh and the .sdc files are eventually used by SOC ENCOUNTER to generate the layout.

- Specify the Verilog file containing the definition of the top-level module.
- Specify the correct directory.
- Specify the name of the top-level module.
- Specify the frequency for which the design has to be implemented. Based on the frequency information provided, the synthesis tool tries to optimize the design so as to meet the frequency requirements.

** Compile Script for Cadence BuildGates */ ** TSMC 01500 Standard Cell Library */	
et verilog RTL */	
** tox_shell -f compile_bgx.scr */	
#* Dr. Rahkan Rahrafi SUSU Department of ECE */ #* Jan. 23rd. 2009 */	
<pre># Hit verilog files, separated by spaces set mg_verilog_files {,//Counter/encounter/Counter.v} set mg_report_path {,//Counter/encounter/&lt;</pre>	
<pre>#set mg_verilog_files {./fulladd/encounter/fulladdmain.v} #set mg_vert_path {./fulladd} # Tor_imay Module # Tor_imay Module</pre>	
set mg_toplevel_module Counter	
<pre># The name of the clock pin. If no clock-pin # exists pick anything set wg_clock.pin clk</pre>	
# Target frequency in MHz for optimization set my_clock_freq_MHz 100	
<pre># Delay of input signals (Clock-to-Q, Package etc.) set my_input_delay_ns 0</pre>	
<pre># Reserved time for output signals (Holdtime etc.) set my_output_delay_ns 0</pre>	
No charges necessary beyond this point	
set TSMCHOME \$erv(TSMCHOME)	
read_tlf \$TSNCHOME/digital/Front_End/timing_power/tcb013ghp.211a/tcb013ghptc.tlf	
read_verilog_fwg_verilog_files	
set_global target_technology tcb013ghptc set_global fix_multiport_nets true	
set_global hdl.verilog.out_unconnected.style full set.global hdl.verie.mull.line.port.mase.false	
set_global hdl_tree_height_reduction true set_global hdl_common_subexpression_elimination true	
set_global aware_adder_architecture fcla # set_global aware_mux_dissolve_size 14	
	-
"compile_bgx,scr" 85L, 3039C         38,0-1           37, 7, 7, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,	Тор

Note – It is advisable to spend some time trying to understand the compile\_bgx.scr script. This script has details about the netlist file, .sdc file and timing, power and area reports. It

specifies the path where the reports are generated. The .vh and .sdc files generated on running this script have to be copied into "encounter" folder in the work directory for the SOC ENCOUNTER tool to generate the layout.

• Additional changes have been made to the script to copy the netlist and .sdc file into encounter folder. In addition to that the reports generated are directly copied into encounter folder. Add the following lines of code to the script.



The compile bgx.scr script is executed using the command,

#### >> bgx\_shell -f compile\_bgx.scr

M xterm		
45 [volta]/home/student/dharwadk/cadence> bgx_shell -f compile_	bgx.scr	
	•	
* Copyright (c) Cadence Design Systems, Inc. 1994 - 2005. * All rights reserved.		
* Cadence acknowledges trademarks or registered trademarks of * other organizations for their respective products and * services.	• • •	
<ul> <li>This programs contains confidential, trade searct information of Caderoe Beign Systems, Inc. Any reproduction, use or disclosure of this program, including any attempt to obtain a humarreadable version of this program, uthout the express, prior written consent of Cadence Design Systems, Inc. is strictly prohibited.</li> </ul>	- - - - - - -	
U.S. GOVERNMENT END USERS: This program is a "commercial item" as that term is defined at 49 C.F.R. 2.101 (0CT 1995), consisting of "commercial computer software" and "commercial 48 B.F.R. 12.212 (SEPT 1985), Consistent with 48 C.F.R. 212 and 48 C.F.R. 227.2020 - through 227.2024 (JUNE 1955), all "U.S. Government End Users acquire the program with only those rights set forth herein."	8 9 9	
Cadence Design Systems, Inc. 2655 Seely Avenue San Jose, CA 95134, USA bax_shell v05,17-901341 Feb 5 2007 03;22;26		
* Options -BGX -License	*	
* Host volta Mon Mar 01 18:29:19 PST 2010	* *	
* Loading command line script 'compile_bgx.scr'	•	
_		

	rm													
kter	m			Path	Grou	p Opt	ions I	Report				1		
		1 I.		1	Pts I	Sla	sk I I	Critic ndpoin	ts I S	lack	1	- i		
i	defa	dt. L		1	1		1		/9	+inf	0.00	i –		
		1	none	asing	dept	h for	slew	propag	ation.	Rese	tting	timing	<tof< td=""><td>T-52</td></tof<>	T-52
				Path	Grou	p Opt	ions	Report				i –		
				rt   	All   Pts	Targ Sla	st   sk	Critic Indpoin	al I M ts I S	orst lack	I TEFS			
	defa	lt İ		- i	i		- i		/9 i	+inf	i 0.00			
		1	ncre	asing	dept	h for	slew	propag	ation.	Rese	tting	timing	<tof< td=""><td>T-52</td></tof<>	T-52
								Report						
		1 I.		- I	Pts	Sla	sk I I	Critic Indpoin	ts I S	lack	1	1		
								0						
	+							unter						
								slack						
				÷					+		+-			
	÷		·					op. <t< td=""><td></td><td></td><td></td><td></td><td></td><td></td></t<>						
								Report				-+		
I Pa	athGri	up I	Effo	rt I	A11	Targ	et I	Critic	al I U	orst	I TEFS			
	defa	lt		1	I			0	/9	+inf	1 0.00	i		
				Path	Grou	p Opt	ions I	Report				i –		
Pa	athGro	up	Effo	rt	All   Pts	Targ Sla	st   sk	Critic	al I M ts I S	orst lack	I TEFS	i i		
		+-		+-	+		+		/9		+			
۱		1	forc lar usir COMM	e -mi 1 18: 9 0:0 0N-07	nimiz 29:23 :1 Re 6>.	e mul 2010 al ti	tiple we. C	reclai output	-prio	rity .	area f	inishe	hig d at	ı Mon

• Observe that files 1) Counter.vh 2) Counter.sdc are generated and placed in the encounter folder. Counter.vh is the netlist file where as Counter.sdc is the timing file which has all the timing details for the standard gates present in the netlist file.

## **NETLIST Simulation**

This chapter discusses simulation of the netlist file. It is also known as post-synthesis simulation. There are two approaches to doing this. We can either use the Verilog-XL compiler or use NCVERILOG/NCSIM. Chapter 2 talked about using Verilog-XL for carrying out RTL simulation. The same procedure could be used for simulating the netlist file. However, we will have to include TSMC 0.13um standard cell library files tcb013ghp.v and tpd013n2.v when invoking the compiler. These files contain delay information and definition for standard gates present in the netlist file. The command to invoke the Verilog-XL compiler to compile and simulate the netlist file is,

#### >> verilog tcb013ghp.v Counter.vh Counter\_TB.v

Observe the output and see of it tallies with the desired output.

The second method to simulate the netlist file is to use NCVERILOG/NCSIM. This approach is explained in detail so as to familiarize users with NCVERILOG. The GUI for NCVERILOG can be invoked using the command

#### >> nclaunch -new&

This command has to be executed from the encounter subfolder. –**new** option is entered when invoking NCVERILOG for the first time for any given project(in this case it would be Counter project). On subsequent calls to NCVERILOG the following command has to be used

#### >> nclaunch &

m 179 [volta]/home/student/dharwadk/cadence/Counter/encounter>_cd		Ŀ	- 0 X
77 [ulta]/home/student/dharwadd/cadence/Lownter/encounter> ed 80 [uolta]/home/student/dharwadd/cadence/cunter> ed 81 [uolta]/home/student/dharwadd/cadence/ed 1: No such file on directory. 82 [uolta]/home/student/dharwadd/cadence> hclaunch -mewk. 21 [15618]			
82 [volta]/home/student/dharwadk/cadence> nclaunch -new& 2] 13616 83 [volta]/home/student/dharwadk/cadence> nclaunch: 08,20-p001; (c	a) Comminist 1995-2009 Cadence Tenice Sustano Teo		
(	NCLaunch		
	cãdence		
	NCLaunch 08.20-p001		
	Protected by U.S. Palints: 5,058,545,5410,931;56,668,098;6,467,704;7,039,867; 7,075116;5,838,946;56,851,911;6;163,755;5,5101;570.		
	Copyright ICC1 1995-2006 Cadence Design Systems, Inc. All rights reserved worldwide. Cadence and the Cadence logo are registered tradinations of Cadence Design Systems, Inc. All others are the property of their respective holders.		
	Please Select Run Mode:		
	Multiple Step		
	Single Step (IRUN Only)		
	Help		
	Exit		

Select **Mutiple Step** option. Click on **Create cds.lib file** button and press **OK**. This will setup the environment for compiling and simulating the netlist file using NCVERILOG. The same approach could be followed for RTL simulation. The only difference between RTL and NETLIST simulation are the files that are compiled and simulated. In netlist simulation the netlist file(.vh) file is compiled and simulated where in RTL simulation the Verilog file is compiled and simulated.

The figure below highlights various regions in the tool that need to be explored by the user. When compiling/simulating a Verilog/netlist file care has to be taken while selecting the files. The order of selection is important. Always start with the file that has the lower-level components/modules and gradually move onto the files containing higher level modules. The testbench should always be the last file to be selected.

# *Note* – *The files in the left pane of the window may not be arranged in order. It is the job of the user to select the files in the right order and subject them for compilation/simulation.*

Upon selecting the files, click on the **Tools** menu item and select the **Verilog Compiler** option. Press **OK** in order to compile the netlist file and generate a **worklib** folder in the right pane. On expanding the **worklib** folder in the right pane users should be able to see two links. One carries the name of the top module and another is the testbench module.

Select/Click on the testbench module link in the right pane and select the **ELABORATE** option from the **Tools** menu. Once the elaboration process is complete, in the right pane under the **snapshot** folder a snapshot is created for the testbench module.

xterm		
478 [volta]/home/student/dharwadk/cadence/Counter/er		
	NCLaunch : /home/student/dharwadk/cadence/Counter/encounter	
	<u>File Edit T</u> ools <u>U</u> tilities <u>P</u> lug-Ins	Help
	Browsers: 🛃 😰 Tools: 🔬 🌺 🛄 ⋞ 醥 🖼 😭	
	Directory: Jharwadk/cadence/Counter/encounter 🔄 👘 🛷 CKNXD6	
	E S CKXOR2D0	
	🕀 🗀 waves.shm	
	Counter.v CKXOR2D2	
	E Counter.vh	
	CMPE22D1	
	Moun.v CMPE22D2	
	Moun TB.y	
	E S CMPE32D2	
	tod013n2.v E Ø CMPE42D1	
	🗈 🖓 CMPE42D2	
	🗎 🕀 🖓 Counter	
	E Counter_TB	
	UCAP	
	🗈 🛷 DCAP16	
	🗎 🕀 🛷 DCAP32	
	🗎 🕀 🐨 🛷 DCAP4	
	🗈 🛷 DCAP64	
	📃 💮 🐨 🛷 DCAP8	
	Filters: ".v ".vhd ".vhd ".vh	
	ncvlog: 08.20-p001: (c) Copyright 1995-2008 Cadence Design Systems, Inc.	
	ncvlog: Memory Usage - 8.6M program + 9.1M data = 17.7M total ncvlog: CPU Usage - 0.0s system + 0.3s user = 0.3s total (0.4s, 83.6% cpu)	
	Inclaunch> ncelab -work worklib -cdslib /home/student/dharwadk/cadence/Counter/enco	unter/cds.lib -log
	ncelab: 08.20-p001: (c) Copyright 1995-2008 Cadence Design Systems, Inc.	
	ncelab: Memory Usage - 18.7M program + 13.6M data = 32.3M total ncelab: CPU Usage - 0.0s system + 0.0s user = 0.1s total (0.1s, 60.3% cpu)	
	nclaunch>	<b>Q</b>
		1 items selected
		I Itellio selected

The next step is to simulate the compiled netlist file and observe the waveforms. NCSIM is used for simulation. Simulation is used to observe the waveforms. In order to invoke simulation select the snapshot and click on the arrow button as shown in the figure below.



The Simvision tool, upon being invoked, opens in a new terminal. From the left pane select the module for which the waveforms need to be generated. From the right pane highlight all the nets for which waveforms need to be generated. Right click and select "Send to waveform window".

🚉 Design Browser 1 - SimVision	
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>S</u> elect Explore Sim <u>u</u> lation <u>W</u> indows <u>H</u> elp	cādence
🗗 🐕 📷 🖉 🗞 🗅 🛍 🗙 🖉	🗳 - 🔈 💠 Send To: 🗽 🚝 🚉 🎆 📰 🔳
🐨 🎘 2 TimeA 🕶 = 158 🔽 ns 🔻 👯 🖌 🛫 💁 🛛 Search Times: Value 🕶 🔛 🏨	. 11.
💽 - 🔟 📧   🐺 🐺 🛄   🛑   📼 318ns + 0	
Design Browser × ④ Name -	Value (as recorded) -
Browse: S All Z S Options	0 -> 1 'h 2 -> 'h 3
Simulator	Selected Items 1
	Bookmark This View
	Cut
	Сору
	Paste Delete
8	
	Trace Driving Logic Trace Loading Logic
	Send to Waveform Window Send to Source Browser
	Send to Schematic Tracer
	Send to Memory Viewer
	Send to new
	Break on Change
	Create Force Release Force
Leaf Filter:	Deposit Value
Show contents: In the sig	Create Probe
Show contents: In the sig	Describe
Ø	3 objects selecter

This will invoke the waveform window. In order to observe the waveforms run the simulation by typing the command

>> run

in the simulation console.



#### Layout Design

We started with RTL simulation, then synthesized the Verilog code into gate-level netlist, performed post-synthesis simulation and finally verified the behavior of the circuit by generating waveforms. The last step in the design is to auto place and route the standard gates. We use ENCOUNTER tool for this purpose. The **encounter.tcl** script invokes the ENCOUNTER tool and does auto place and route. The **encounter.tcl** script reads the **encounter.conf** which has all the details about layouting (number of metal layers, VDD, GND). The **encounter.conf** file has to be modified so as to refer to the synthesized netlist file.



Before invoking the SOC ENCOUNTER tool ensure that gds2\_encounter.map file is present in the encounter folder. The SOC ENCOUNTER tool can be invoked using the command,

>> encounter -init encounter.tcl

	cādence"	
SoC Encounter™ RTL-to-GDSII System		
RTE-to-oDoil System	version 8.1	
Presected by US patients: 0.101.070; 0.240.002; 0.200.700; 0.203.770; 0.405.740; 0.6412.730; 0.6112,742; 0.6117,741; 0.610 0.611; 0.604; 0.612; 0.612; 0.612; 0.701; 716 0.610; 0.612; 0.612; 0.612; 0.701; 716 0.610; 1.40; 0.612; 0.610; 0.701; 701; 0.71 7.024,644; 7.7202; 0.702; 0.701; 701; 701; 701; 701; 7.024,644; 7.702; 0.702; 0.701; 701; 701; 701; 701; 7.024,644; 7.702; 0.702; 0.701; 701; 701; 701; 701; 7.024; 0.71; 7.702; 0.701; 701; 701; 701; 701; 701; 701; 701;	1: 0.640.0011.6,070.103; 0.702.013; 0.702.013; 0.702.016; 1: 0.702.013; 0.702.016; 1: 0.901.233; 7.010.7906; 1: 7.223.110; 7.202.070; 1: 7.223.110; 7.201.024; waarvad woodbuilds waarvad woodbuilds	
Cadence Design Systems, Inc. All others are the property of th	eir respective holders.	

The figure given below is that of the layout generated by the auto place and route functionality of SOC ENCOUNTER tool for our circuit.



On the left side is the summary report obtained by clicking on the "summary report" button on the top. The summary report carries details about area utilization. The encounter.tcl script also generates a timing report bearing the name timing\_final.rep. This has details of the critical path, slack time etc. The post synthesis (generation of the netlist file using compile\_bgx.scr script) too resulted in a timing report getting generated. Upon comparing both the results it can be observed that the report generated by SOC ENCOUNTER will incorporate delays introduced due to interconnections and parasitic capacitances along with propagation delays through standard gates.

2 / # C.   # # B. B. B.		<i>√</i> :			
🕺 Quick Connect 📄 Profiles					
Slack Time Clock Rise Edge Beginpoint Arrival Time Instance   Arc   count_reg_0   CP ^ - y   count_reg_0   CP ^ - y   i_10   Sl v -> 21	V   DFQD1 ^   INR2D0   SDFKCNQD0	Delay   ;                 0.189     0.122     0.000	Arrival   Re Time   0.000   0.189   0.311   0.311	equired   Time	ENCOUNTER TOOL timing report
count_reg_1   SI ^					
+					27, 0−1 Bot  SSH2 - aes128-cbc - hmac-md5 - nr  174x14   20   0   0   0   0
+		-			
mected to volta.sdsu.edu m capacitance unit   1.00 pF resistance unit   1.00 kDm 2ath 1: HET Setup Check with Pin C There for the capacity () the provide t	unt_reg_1/CP	edge of 'vo	elk'		SSH2 - aes128-cbc - hmac-md5 - nr 174x14