EE434 ASIC & Digital Systems

Design Compiler

Dae Hyun Kim daehyun@eecs.wsu.edu

1

Overview

- Synthesis is converting given HDL source codes into a netlist.
- Synthesis software
 - Synopsys Design Compiler
 - Cadence RTL Compiler

How to Run Synopsys Design Compiler (DC)

- Run the following command:
 > source /net/ictools/sh/synopsys.sh
- Run DC:
 - > design_vision

Library Files

- Download the following file into your working directory:
 - http://eecs.wsu.edu/~ee434/Labs/tut-dc.zip
- Unzip it
 - unzip tut-dc.zip
- You will see the following files:
 - NangateOpenCellLibrary_typical.db
 - Standard cell library for synthesis
 - NangateOpenCellLibrary.v
 - Verilog source code for standard cells
 - and some more files

Run DC

- Run DC.
 - > design_vision
- Set libraries.
 - > set link_library {NangateOpenCellLibrary_typical.db}
 - > set target_library {NangateOpenCellLibrary_typical.db}
- Read source files.
 - > read_file -format vhdl {nand2.vhd}
 - (Make sure that there is no error)
- Set the top-level module you want to synthesize.
 - > current_design myNand2
- Compile.
 - > compile -exact_map -map_effort high

Compile

- "Compile" actually synthesizes a given circuit and generates a netlist.
- It also optimizes your netlist during synthesis.
- Once it's done, let's visualize the netlist in DC.
- Click myNand2 in the "Logical Hierarchy" window.

	esign Visio	on - Top	Level.1 (r	myNand2) -	[Hier.1]										
	ile <u>E</u> dit	<u>V</u> iew	<u>S</u> elect	<u>H</u> ighlight	List	<u>H</u> ierarchy	<u>D</u> esign	<u>A</u> ttributes	S <u>c</u> hematic	Timing	<u>T</u> est	<u>P</u> ower	<u>W</u> indow	Help	
	; 📙 🎒	🗗	0.2	Q @ Q		🔹 🚖 🛛 🖸		- 🎫 🏨 🛛		∎eee ∭m	yNand	2 🔻			
	Logical	Hierarc	hy	Cells (H	ierarch	ical)									
18	D=	=> myl	Vand2	Cell Na	me	Ref Nam	Ref Name Cell Path			Dont Touch					
(i)				1											
Đ,															
Q															

Schematic Viewer

- You can see the I/O ports or the schematic (netlist) of the module.
- Click the following button to see the I/O ports of the module (symbol view).

	🖳 Design Vision - TopLevel.1 (myNand2) - [Hier.1]														
Đ _e ,	<u>F</u> ile	<u>E</u> dit	<u>V</u> iew	<u>S</u> elect	<u>H</u> ighligh	t L <u>i</u> st	<u>H</u> ierarchy	<u>D</u> esigr	<u>A</u> ttributes	S <u>c</u> hematic	<u>T</u> iming	<u>T</u> est	<u>P</u> ower	<u>W</u> indow	Help
	ê 📙	6	8	0.2	Q 🛈 🤅) () [1 🔹 🚖 🗗 🔁		🛏 👪 📗		list ∏m	yNand	12 🔻		
	Logical Hierarchy					Hierarc	hical)								
4	[™] D==> myNand2				Cell N	ame	Ref Name	e	Cell Path	Dont To	Dont Touch				



Schematic Viewer

- Close the symbol view.
- Click the following button to see the schematic of the module.

	🚰 Design Vision - TopLevel.1 (myNand2) - [Hier.1]														
₽ !	<u>F</u> ile	<u>E</u> dit	<u>V</u> iew	<u>S</u> elect	<u>H</u> ighlig	ht L <u>i</u> st	<u>H</u> ierarch	/ <u>D</u> esign	<u>A</u> ttributes	S <u>c</u> hematic	<u>T</u> iming	<u>T</u> est	Power	<u>W</u> indow	Help
	<u> </u>	6	8	0.2	Q 🔞 (0.0	*	D 🖸 🛛	🗠 🎫 🔣		₽	yNand	2 🔻		
	Lo	gical I	Hierarc	hy	Cells	(Hierarc	hical)								
18	_	==> myNand		==> myNand2 Cell Name		Ref Na	me	Cell Path	Dont To	Dont Touch					



Save

- Save the synthesized netlist.
 - > write -format verilog -output nand2_syn.v
 - (Notice that we are using Verilog for the synthesized netlist.)
- Open the synthesized netlist in a text editor and see how it looks like.



• Exit

> exit (or quit)



- Repeat the same synthesis procedure to synthesize a full adder.
- Use "fa.vhd".
- The top-level module name is "myFA".
- Use "compile -exact_map -map_effort low"
- Symbol





• Schematic



Simulation

- Save the synthesized netlist.
 > write –format verilog –output fa_syn.v
- Exit.
- Now, we will simulate the synthesized full adder using ModelSim.

Simulation

- Run ModelSim (you need to source /net/ictools/sh/mentormodelsim.sh).
- Create a new project and add existing files.
 - NangateOpenCellLibrary.v
 - fa_syn.v
 - fa_sim.vhd
- Compile all of them.
- Run until 12ns.



• Results

Messages										
🕀 🔶 :test_myfa:g_in	000	000	001	010	011	100	101	110	111	000
	0									
🔶 :test_myfa:g_co	0									