Digital Control in Switched Mode Power Converters and FPGA - based Prototyping Prof. Santanu Kapat Department of Electrical Engineering Indian Institute of Technology, Kharagpur

Module - 07 Introduction to Verilog and Simulation Using Xilinx Webpack Lecture - 65 Simulation of Verilog-HDL-based Design using Xilinx Webpack - I

Welcome. So, in this lecture we are going to simulate, we are going to first discuss because, how to simulate you know your Verilog code using the Xilinx ISE simulator. Again, I am saying this is not part of the exam as well as the assignment, even if you do not need to install the software. But, it is just for the interested participant who can simulate.

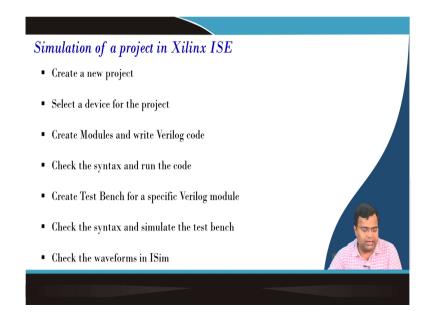
So, this lecture is divided into two parts; in part I, I will be showing some screenshots of how to do that and in part II will be actually going to Verilog Xilinx ISE software and we will write some code there.

(Refer Slide Time: 00:58)



So, here we will first show the step for simulating using the Xilinx ISE simulator and some example case studies using Xilinx webpack wave pack.

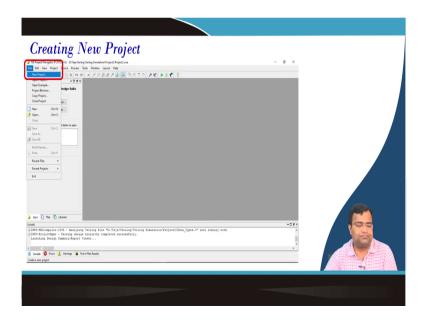
(Refer Slide Time: 01:07)



So, here once you want to simulate in Xilinx ISE. First of all, I am assuming that you know those who are interested you know this for optional Xilinx, you can implement can install it on your computer. And, you know you can check on the web how to install Xilinx ISE. So, we are assuming that you have already installed it.

Now, you can create a project. The first steps are this, you need to create a project then you need to select a device. And, we will be showing what device you are going to use for this hardware demonstration the Xilinx device for this course. Then, we will create some modules and write some Verilog code. Then, check the syntax and run the code and create a test bench for a specific Verilog module. And, we need to check the syntax and simulate the test bench. And, we need to check how to see the waveform using ISim.

(Refer Slide Time: 02:01)



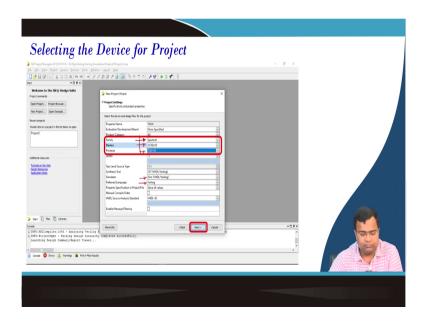
So, this is the first when you install the software. You can see if you go to that, if you click, if you open the software under the file you will create a New Project ok.

(Refer Slide Time: 02:16)

	Project Location		
		- 0 ×	
Int −000 methods with the second sec	Den Frage March Port Angel Character Port Angel Character Port Angel Character Port Angel Character Port Angel Port Angel		
> Start 🚺 Mas 🐧 Ubranies	No Hend Savios type:		
Dist Pit Count Cou	rardhy completed muccessfully. T.,	-06×	

So, in the New Project, if you go this is a screenshot. Then, it will ask you know you have to write something; that means, you can you have to select a folder where you want to keep this project. So, you can create your folder. Then, you have to name you to have to keep the name of this project. So, here the project name is Full Adder ok. Then, you go to the next and do not change anything, this HDL top level should be there.

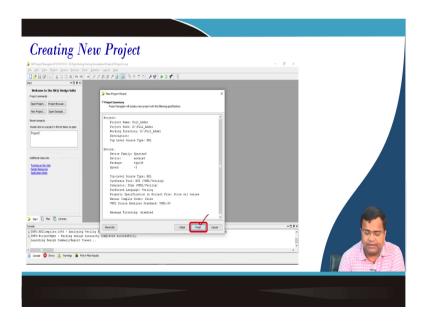
(Refer Slide Time: 02:41)



Then, the next stage will go for this stage where it will ask for the device. So, this thing you do not have to bother with, but here in our course we will be using Spartan6 and the particular device ID is this; that means, XC6SLX9 and the package is TQG1 double 1 double 4. And, then we have to specify the preferred language as the Verilog and the simulator ISim ok.

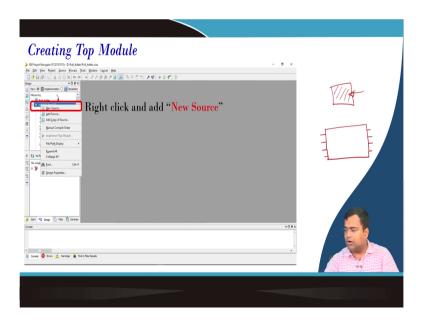
There can be model sim, the third-party software, but we are preferring that whatever Xilinx software is there ISim, that will be used. Then, everything else you keep as it is by default, only you have to specify the Verilog and the ISim and this particular device detail.

(Refer Slide Time: 03:28)



Next, if you go next, it will show the project summary, and then if you finish, if you click this Finish.

(Refer Slide Time: 03:37)



It will create this project. So that means, your new project is created and the new project will come like a device, this ISE ok. And, this device you know I will go in the next you know next part II to show actually in the Xilinx platform.

So, then it will ask for a new source; that means, it will create an ISE-type structure, here with a project name. Then, in this ISE what do you want to implement? Because, as if you are making an ISE, a virtual ISE will have several pins depending upon what exactly you want to implement and how many IO ports are there.

(Refer Slide Time: 04:22)

Creating Top	ensie – Ø X	
	0000 1000 100 A A B B A B A B A B A B A B A B A B A B	
nign D & X View:		
Hearchy ^		
i) 🖸 xclish/9-3tog144		
Empty View	New Source Woard	
The view currently cantains no files.	-	
a using the toubar at left, connands from the Protect nervy, and by	⁴ Select Source Type Select source type, file name and its location.	
using the Design, Files, and		
Ubraries panels.	P ICOTE Gegenter & Architecture Woord	
Use:		
+New Source: To create a	D Verlag Module	
Ro Processes Running	VHCL Module Change	
Vo single design module is selected.	MdCLRuny WdCPuckage	
🖞 🗄 🎾 Design Utilities	VHCL Test Bench Location:	
2	Critical Ander	
	All to prest	
	Here Drfs Cancel	
🖌 Start 🕫 Design 🚺 Files 🚺 Libraries		
areale	+08×	
	A	
	v	
Canade 🧿 Errors 🧘 Warnings 🙀 Find in Files I)	
idd a new source to the project	100	
add a new source to the project		

Then, you have to first add a New Source; that means, we want to write a Verilog code, New Source. Under this source, there are many options; Schematic, then we have to add the Verilog Module that is it. And, if you want to test, there is a test module also there, which will come next. So, under the Verilog module, this module will ask for the Verilog File name.

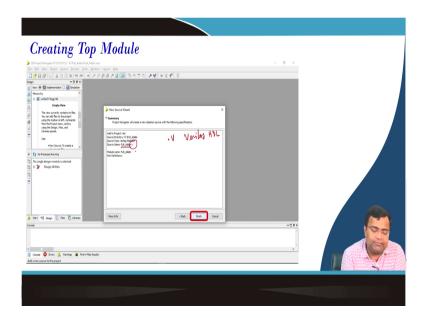
So, again you are making a Full Adder as the File name and it is also under the Location of Full Adder which is a project name. So, next remember the File name can be different. I mean it need not be the same as your project name. But, whatever File name will make, the Verilog dot v file will be created by that name. So, you should not change; that means, whatever you are doing should be consistent.

(Refer Slide Time: 05:06)

	ervise Indone Layout Help	× 12 × 2 1 9			- 0 X	
The second	This Special Ward Code India Special Ward Code India Special Ward Add to we (Ad, data Facilitane	Direction Ro input III input IIII input IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	MSB	X		
Part R\$ Coop) Re () Linver	PercM	inp.d. ∨ inp.d. ∨	Pest >	v		
Grande 🧿 Erver 🔔 Warrings 👰 Prod in Has It dd a new source to the project	taða				, ·	

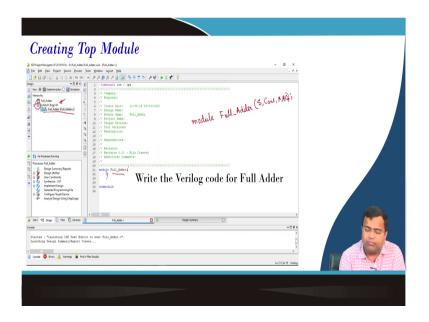
Then, if you go Next, it will ask how many input-output ports you want. But, I think you do not need to do anything, because you will be declaring the port inside the Verilog code. So, you do not leave it is and then go Next.

(Refer Slide Time: 05:21)



If you go Next, then it will ask if can you will see that your file name is Full underscore Adder dot v. So, it is a dot v file, dot v which is a Verilog file, Verilog extension is dot v HDL ok. And, the module name is Full Adder ok and you are also keeping the directory which is D drive in Full Adder in this case.

(Refer Slide Time: 05:48)

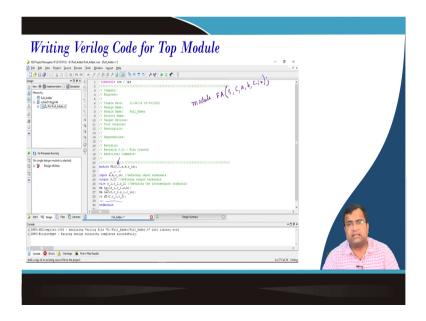


Now, once you it will show. So, this is your project, it will look like an ISE here. You can see under this project you are plugging in one Verilog file which you want to implement like a Full Adder, you want to implement a Full Adder. Once you write by default it will come Full Adder and this block you have to specify; that means, it is like that module, then it will come like a Full underscore Adder.

This is the name of the module. Then, we need to have brackets where in the Full Adder you know you can specify a sum, carry out whatever you want to write A, B then, you can say carry in. So, it is up to you, whatever you want to keep, but by default, since you have not specified in the input-output, it will be left vacant; that means, there is nothing here, you need to fill the gap. Nothing there so, it is vacant. So, you have to fill this gap ok.

So, you can take this here to like this, then end the module. Now, here you will see the time scale 1 nanosecond, that is if you provide any time unit; 10 units, 20 units, it will be in terms of multiplied by 1 nanosecond; that means 20 units corresponds to 20 nanoseconds. And, this 1 picosecond slash suppose it you simulate more precisely decimal, it can go up to 1 picosecond. So, it is more important this value ok.

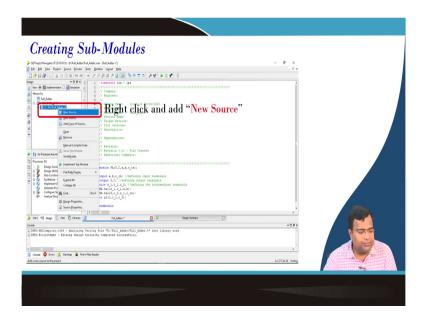
(Refer Slide Time: 07:24)



Next, now you fill up your Verilog code; that means, if you want to write a Full Adder, first you can see the summer, sum block. You know if it is not visible, I am writing here module, then it is the name as the Full Adder. So, here the module name is again different because we have a Full Adder dot v file. But, it is better to maintain the same name rather than name another name. So, we can use the Full Adder here, then S, C, a, b, c in.

Then, we have to define input as a b c in, the output as S and C. Then, we know there are errors there are wire connections between the S bar C bar, c 2, and then Half Adder, we need to take call 2, instantaneous 2 Half Adder and we have to connect accordingly. And, then it will go to or gate to generate the carryout. And, then that is the end of the module.

(Refer Slide Time: 08:40)



Once, you write this code because we have discussed about this Verilog code; then you have to create a New Source under this Full Adder.

(Refer Slide Time: 08:46)

■ The first input of 201010 Dr. 4, 2426. ■ Dr. 100 Dr. 4, 2426. Dr. 4, 2426. ■ Dr. 100 Dr. 4, 2426. Dr. 4, 2426. ■ Dr. 100 Dr. 4, 2426. Dr. 4, 2426. ■ Dr. 100 Dr. 4, 2426. Dr. 4, 2426. ■ Dr. 100 Dr. 4, 2426. Dr. 4, 2426. ■ Dr. 4, 2426. Dr. 4, 2426. Dr. 4, 2426. ■ Dr. 4, 2426. Dr. 4, 2426. Dr. 4, 2426. ■ Dr. 4, 2426. Dr. 4, 2426. Dr. 4, 2426. ■ Dr. 4, 2426. Dr. 4, 2426. Dr. 4, 2426. ■ Dr. 4, 2426. Dr. 4, 2426. Dr. 4, 2426. ■ Dr. 4, 2426. Dr. 4, 2426. Dr. 4, 2426.	Display Display <thdisplay< th=""> <thdisplay< th=""> <th< th=""><th>P C P E C S Monomentary Monomentary</th><th>Norme Norme Norme</th><th>- 0 X</th><th></th></th<></thdisplay<></thdisplay<>	P C P E C S Monomentary Monomentary	Norme	- 0 X	
	30 endeodule	Mare Info	Next > Canol	*	
🍃 Start 🔍 Design 🖒 Files 🚺 Libraries	Full_Addex.v*	Design Summary	0		
Corole	Verilog file "D:/Full_Adder/Full_Adder.	of thes library work		0 # ×	
) INTO FrojectNight - Parking design Console O Brux 4, Warmapa 6, Prod Adda new source to the project	hierarchy completed successfully.	- 1977		Ln.27 Cel 24 Verilog	

There you can have a Half Adder module because we have called Half Adder, but not you have not yet designed it. So, under this Half Adder, you have to create a Half Adder module name. Again, Verilog Module, the name is Half Adder. It must be the same as the module that we have instantiated here.

(Refer Slide Time: 09:04)

1 Characterization 201010 2-24,24424 10 En (2010) 2-24,2444 10 En (2010) 2-24,2444 10 En (2010) 2-24,244 10 En (P C D D C		Oversitie nput nput		M28	159	- 0	× • 0 K	
	C **	Nov 3rb		_	< Bait	Next >	Canol)	
Fort MC Design C) Files C Libraries Corresie	Full Addexv*	0 1	Design Summary						-0#×	
		.v* into library wor	1k					Lis 27 Cel 24	•	

(Refer Slide Time: 09:05)

In Presses Eurorg If Torget Survey Report If Torget Survey Survey Report If Torget Survey Survey Survey Report If Torget Survey Sur	<pre>// Dependencies: // Bergalon 0.01 - File Creat- // Additional Comments: // Additional Comments: /</pre>	Soura Tiper Insign Bandel Soura Tiper (Mar Walder Amer, Ma Anat Ceffortimes					
29	endmodule	Hare Info	< Back Prigh.	Cancel			
tart MC Design C Files C Libraries	Full Adder v*	C Design Summary			>	<u></u>	100
					C # X		100
To LECCOMPLIAT: 145 - Analysing Version fo To LECCOMPLIAT: Fareing design hierarchy oracle O true 1 minimum (A find in Hesitandu new source to the project		" LNG librery work		Ls 27	T Cel 24 Yeilleg		

Then, again we have to create a Half Adder dot v file.

(Refer Slide Time: 09:09)

Creating Su The The The The The The The The The The	densier (Mus) – O X goder Legent Belg - O x	
inter ng Coupe () file () laware () Torono () lamon () laware () Little Terres Colorer () Course () trave () newupe () reaching Course () trave () newupe () reaching	0 4615 88.0%. *	

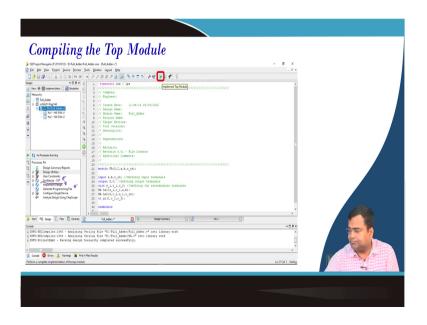
And, it will generate this Half Adder another module.

(Refer Slide Time: 09:13)

Writing Verilog Code for Sub-Modules
Image: State of the state
> form fit boay () files () files () MAJABOL* () E Desp Samery () () HAL
Demonstration of the second se

This Sub Module comes under this Sub Module and they are calling. So, Half Adder module you have a sum out, carry out, a, and b. So, you can write in terms of xor gate and that we have discussed earlier. So that means, two Half Adder now are called instantiate which is why they are coming under one Full Adder.

(Refer Slide Time: 09:33)



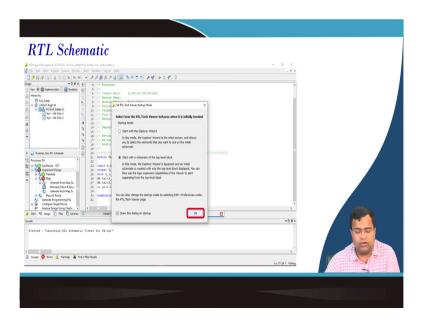
Once, you are done, then you go to implement the top module. You do not need to implement it because you are not using Verilog FPGA implementation right now. You can end up with synthesis because synthesis will have an RTL block and beyond the RTL block it will go for implementation, where it will generate you know, it will map into the target FPGA device. And, final it will generate the bit file which I will be showing when we go to the hardware demonstration. But, here you need to know the RTL logic.

(Refer Slide Time: 10:10)

voject Navigator (P.20131013) - DI-Full		
Edit View Project Source Pr	ven Sen Window Layout Help . P 0 G 🕲 Construint Litter. 🔒 🐊 🦄 🖶 🕐 🔿 🌶 😢 🎽 🖓	4
-0	Ø x 🦞 Con Generator.	
e: () 🗱 Inglenentation () 🚮 Simula Inanchy	for Africal Schematic Vewer 1	
Full Adder	C Terring Analyser . D Technology. Star	1
C xclobr9-3tog144	The second secon	
7 8a1 - HA (HA.) 7 8a2 - HA (HA.)	(g) arout anyte isloas	
Contraction of the second	SmartDiployer •	1 1
	N 15 // uppendentest	
	7% 14 // Revision: 7% 17 // Revision 0.01 - File Created	
	18 // Additional Comments:	
No Processes Running	21 100000000000000000000000000000000000	
CEIDER FA	n 21 module Th(5,C,a,b,c_in); 22	
Synthesize - XST	23 input e,b,c_in: //defining input terminals	
COD Tarolate COD Tarolate COD Map	25 wire # 1,c_1,c_2; //defining the intermediate terminals	
CO Map Bo C Generate Post-Map St.	26 NA hai(si, ci, sh); 27 NA hai(s, ci, si, cin);	
Manually Place & Rou. Generate Post Map St.	20 or g1(C,o_1,o_2);	
B () Place & Route	30 endmodule	
Generate Programming File Configure Target Device	31	
on Analyze Design Using ChipSz.		
t RC Design () Files () Libr		
	*0#	
ted : *Launching #IL Sche	malic Viewer for FA.ngr*.	
)	
sole 🧿 Erors 🧘 Warrings 🖡 Schematic	9) Prodein Pies Reaults Lin 21 Ceil 1 Vesile	
	LN 21 Col 1 Venig	

So that means, then you can see; that means if you implement it will show synthesis is done correctly, then translate. Then, you can see the schematic, there is a tool Schematic Viewer RTL.

(Refer Slide Time: 10:24)

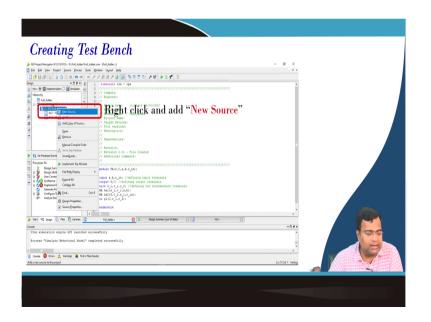


(Refer Slide Time: 10:25)



It will show that it is the Full Adder circuit which consists of 1 Half Adder here like. So, this is I think the Full Adder and Full Adder also have an external gate right? This is Half Adder 1, this is Half Adder 2. So, the first Half Adder will take the two inputs and the second Half Adder take the output of the first Half Adder and the carry-in.

(Refer Slide Time: 10:54)



Then, you can add a New Source, if you want to test it.

(Refer Slide Time: 10:58)

Creating C	# full Addense - [Full Addens] Jush Jindon Lyport Hilp (Res Source Wind R		- 0 x . * x	
> Start #\$ Design) Files) Libraries	6	New Jole	Next > Canal	×	
Consele TSIm assoulation engine GUI Leun Process "Simulate Bahavioral Hoo C Consele O Erver 1 Newage 0 F Add a new survector the project	hed successfully el" completed successfully			C d x A 	

Go to the Verilog Test Fixture; that means, if you right-click here, New Source Test Fixture. Now what; that means, test bench for Full Adder? So, you need to make sure that you are testing the Full Adder, because of the Full Adder circuit you want to test. You can also selectively test the Half Adder circuit, customizable, no problem. So, you have to select which block you have to test. Then, go you have to create a test plan. (Refer Slide Time: 11:25)

Creating Test Bench	
Image: A bit of the state of the s	
Conk	

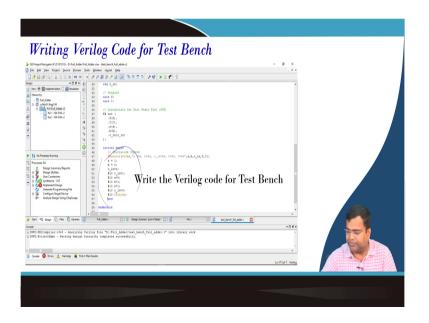
Then, it will associate with Half Adder or Full Adder.

(Refer Slide Time: 11:30)

3 // Company:	- 0 × .	
Annow A	Testers that X Factor of the second of the	
Par Nt Dougn () File () Lineves () All Addes Constel ISIN simulation engine GDI launched successfully	Corpy Servery (and King) () () 98.4	
Process "Samulat Boharisral Hoda" completed worceasfully (Commo O oran 1 more Noter Samuel Note: Protection Results Add area count to the project	v s	

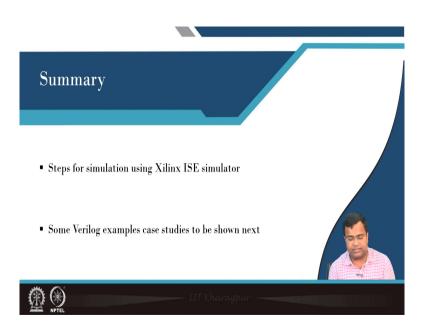
So, if you make it Full Adder, then it will create a Test Bench circuit.

(Refer Slide Time: 11:31)



And, it will link. Now, in this Test Bench circuit, you can make whatever you want ok. And so that means, that is it for the description of how this flow of Verilog. In the next lecture, I will be showing the actual Verilog implementation.

(Refer Slide Time: 11:50)



So, in summary, we have discussed the step for simulation using Xilinx ISE. And, we have also considered an example of a Full Adder circuit and we will be showing the actual Verilog demonstration in the next lecture that is it for today.

Thank you very much.