Start Project Navigator

1. Start new Project Navigator from "Start Menu."

🥭 Start					Fig (1)
		<i>u</i>			
		77 🙆	Software Update Center		
All Programs 🔸 🐋	MSN		Readme		
	Adobe Acrobat 6.0 Professional	150	Project Navigator		
Tour Windows > 🛃	Acrobat Distiller 6.0		Documentation	F	
	Xilinx ISE 8.1i	• 🛅	Accessories	F	
Wizard	PrintMe Internet Printing				
Elected Calling	Nero	- - -			
🛛 🏹 Windows Messe 🛅	ModelSim SE 6.1c				
	Microsoft Visual C++ 6.0	→ [¹]			
🛛 💙 Windows Media 🛅	Microsoft Office	▶ nd			

Create New Project

- 1. File ->New Project
- 2. In the dialog page, enter project named and location (somewhere on your network drive, i.e. Z:\. The type of the top-level module should be **HDL**.
- 3. If you want to use schematic to implement your design (Lab1), choose your project type on the dialog shown as Fig (2).

Property Name	Value	
Product Category	All	
Family	Spartan2	
Device	×C2S50	
Package	TQ144	
Speed	-6	
Top-Level Source Type	HDL	
Synthesis Tool	XST (VHDL/Verilog)	1
Simulator	ISE Simulator (VHDL/Verilog)	
Enable Enhanced Design Summary		
Enable Message Filtering		
Display Incremental Messages		

4. If you want to use Verilog to implement your design (Lab2~Lab5), choose your simulator value as "**Modelsim-SE Verilog**", shown as Fig (3).

Property Name	Value	
Product Category	All	
Family	Spartan2	
Device	xc2s50	
Package	TQ144	
Speed	-6]
Top-Level Source Type	HDL	ſ
Synthesis Tool	XST (VHDL/Verilog)	1
Simulator	Modelsim-SE Verilog	<u> </u>
Enable Enhanced Design Summary	v	-
Enable Message Filtering		
Display Incremental Messages		

Create New Schematic

1. Create New Source->Schematic.

2. Select the symbols for your schematic from the panel on the left.

	203.00		×	
Categories				
< All Sumhols>				
Symbols				
and16			-	
and2				
and2b1				
and2b2				
and3			-	
Symbol Name Filter			_	
Drientation				
Rotate 0			-	
	Symbol Info			
📸 Snapshots	Libraries	👷 Symbols		D ' ()
				Fig (5

3. Add -> I/O Maker to add the input/output pins of your schematic.

🚾 Xilinx - ISE - Z:\testt\testt.ise - [s1.sc	h]
File Edit View Project Source Process	Add Tools Window Help
🗋 🆻 🖥 🖉 😓 📑 🗭 🕗 🗄	🤅 Wire 🛛 🖓 🕅
6800 🖓 🕨 🤟 🛍	- Net Name Ctrl+D
	Bus Tap Ctrl+B
Categories	✓ I/O Marker Ctrl+G
< All Sumbols>	Symbol Ctrl+M
Symbols	Instance Name Ctrl+J
and16 and2 and2b1 and2b2 and3 Symbol Name Filter	Arc Circle Line Ctrl+L Rectangle Text Ctrl+T

4. Double clicking on the I/O label can edit the property of each I/O pin.

bject Properti						
ategory ∃-Nets Ia ⊒-1/0 Markers		Net View and edit the	Attribute attributes of the	95 e selected net	s	
ia	Name	Value	Visible		New	
	Name	a		Add	E 0 T	
	PortPolarity	Input	-	Add	Edit I rai	
					Delete	·
	4			}		
		K Can	icel	Apply	Help	
		K Can		Apply	Help	
0 2 0 4 0		K Can		Apply	Help	
		K Can		Apply		
a		K Can		Apply	Help	
a				Apply		
a		K Can		Apply		
a		K Can		Apply		c
a b		K Can		Apply	Help	
a b				Apply	Help	C

<u>Create New Verilog Module</u>

1. Create New Source->Verilog Module.

📧 New Source Wizard - Select Source Type	
 IP (Coregen & Architecture Wizard) Schematic State Diagram Test Bench WaveForm User Document Verilog Module Verilog Test Fixture VHDL Module VHDL Library VHDL Package VHDL Test Bench 	File name: Adder Location: Z:\Test
More Info	<pre></pre>

2. Set Module I/O port.

Port Name	Direction	Bus	MSB	LSB	
3	input	- I	3		0
)	input	- I	3		0
cin	input	•			
sum	output	V	3		0
cout	output	_			
	input				
	input	–			
	input	•			
	input	_			
	input	•			
	input	-			

Create Test Bench Waveform

1. Create New Source->Test Bench Waveform.

 New Source Wizard - Select Source Type BMM File IP (Coregen & Architecture Wizard) MEM File Schematic Implementation Constraints File State Diagram Test Bench WaveForm User Document Verilog Module Verilog Test Fixture VHDL Module VHDL Library VHDL Package VHDL Test Bench 	File name: test1 Location: Z:\Test	
	Add to project	
More Info	< Back Next > Cancel	Fig (10

- 2. Select a source you want to associate with the test bench.
- 3. Enter the test pattern and the expected output waveforms.



Fig (11)

Run Simulation

1. Change the mode to "Behavioral Simulation."



2. If you are using schematic to design, double click on "*Simulate Behavioral Model*" to run the simulation, shown as Fig (13)



3. If you are using Verilog to design, double clicking on "*Simulate Behavioral Model*" will call *ModelSim* to run the simulation, shown as Fig (14).



Download Design to FPGA Board

1. Change the mode back to "Synthesis/Implementation."



2. Create New Source->Implementation Constraints File.

🚾 New Source Wizard - Select Source Type	×
Image: Second structure with the second structure struc	File name: const Location: Z:\Test
More Info	< Back Next > Cancel
	Fig (16)

3. Select the Verilog file you want to associate with the constraints file.

4. Double click on "Assign Package Pins".



5. Enter the pin number for each FPGA pin on the left side of the window.

🛔 Design Ob	ject List - I/O I	pins				×
I/O Name	I/O Direction	Loc	Bank	1/0 St	d. Vr	•
a<0>	Input	P30	BANK			1
a<1>	Input	P31	BANK			
🔁 a<2>	Input	•				
🗖 a<3>	Input					8
🗖 b<0>	Input					
🗖 b<1>	Input					
<mark>□</mark> b<2>	Input	-				
🗖 b<3>	Input					
📑 cin	Input					
📑 cout	Output					_
n sum∠Ω>	Outout					-
• I						
# Group	1/O Direction	Loc		/0 Std.	Vref	٧c
🔁 4 a 👘 🛛	Input					
🔁 4 b 🛛 🛛	nput					
🔁 4 sum 🛛	Dutput					

6. To find out which pins of the board you can use for external logic refer to the XSA50 manual. As you can see below, some of these pins can be used as general-purpose I/O with the bitstream.

(18)

Configuration Pins (30*, 31*, 37, 38*, 39*, 44*, 46*, 49*, 57*, 60*, 62*, 67*, 68*, 69, 72, 106, 109, 111): These pins are used to load the Spartanll FPGA with a configuration bitstream. Some of these pins are dedicated to the configuration process and cannot be used as general-purpose I/O (37, 69, 72, 106, 109, 111). The rest can be used as general-purpose I/O after the FPGA is configured. If external logic is connected to these pins, you may have to disable it during the configuration process. The DONE pin (72) can be used for this purpose since it goes to a logic high only after the configuration process is completed.

Free Pins (77*, 78*, 79*, 80*, 83*, 84*, 85*, 86*, 87*): These pins are not connected to any other devices on the XSA Board so they can be used without restrictions as generalpurpose I/O through the prototyping header.

- 7. When entering a pin number always insert a "P" before the number.
- 8. Synthesize and implement the design. Perform all the steps that are check-marked in Fig (19).



9. The ".bit" file for FPGA board configuration is ready.