

Digital Systems Design with FPGAs

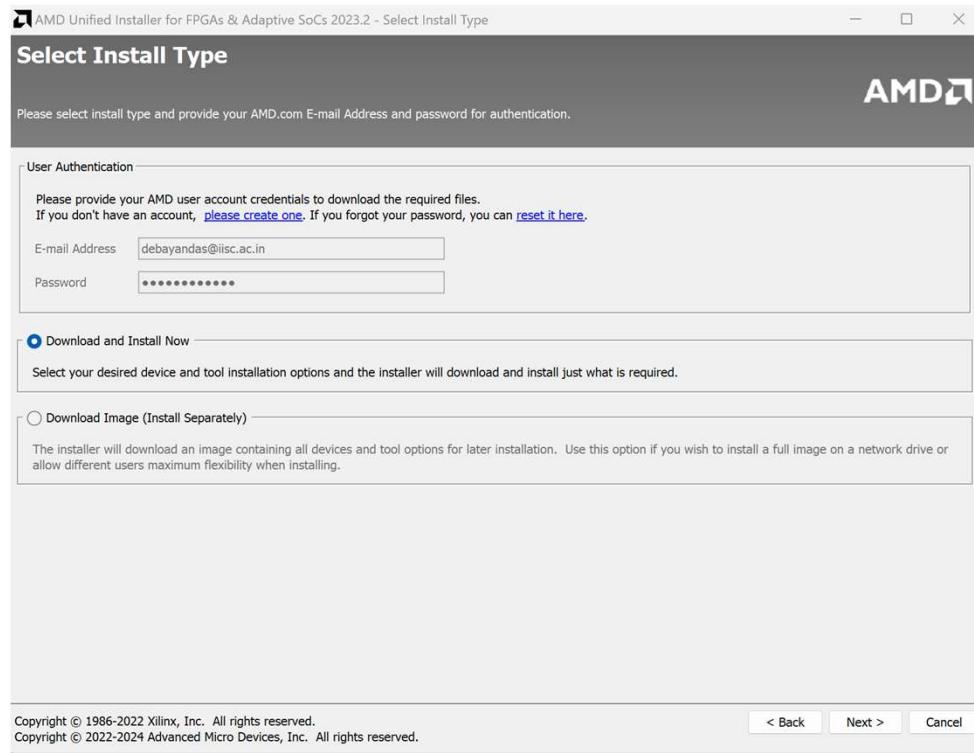
Vivado Implementation & FPGA Programming

Debayan Das
DESE

Indian Institute of Science

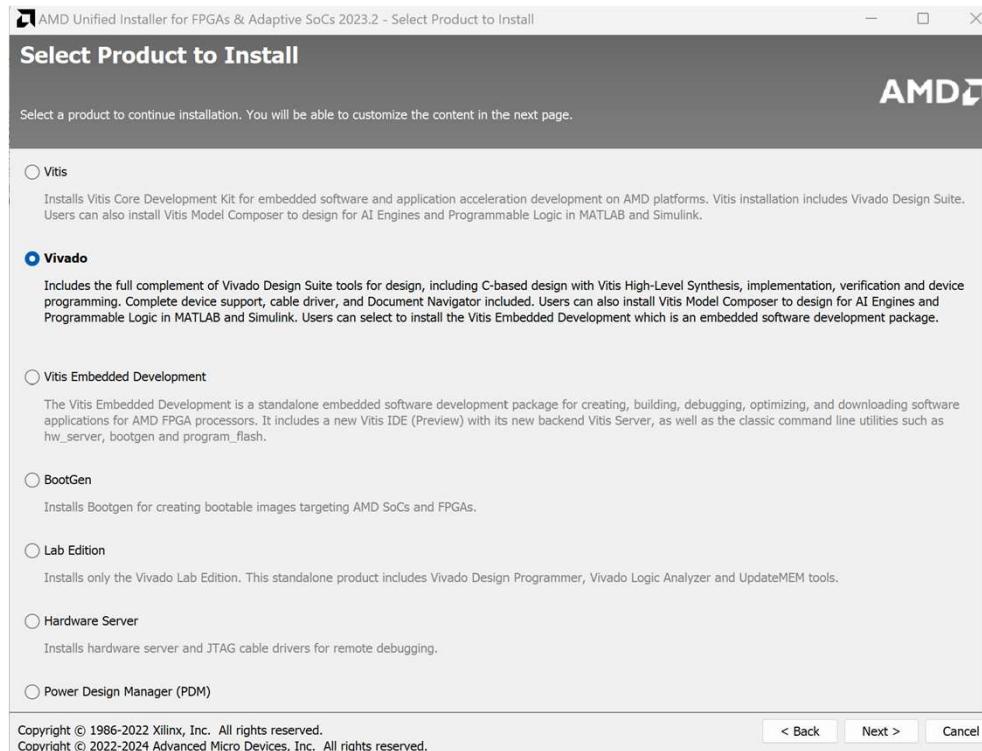


Installing Xilinx Vivado





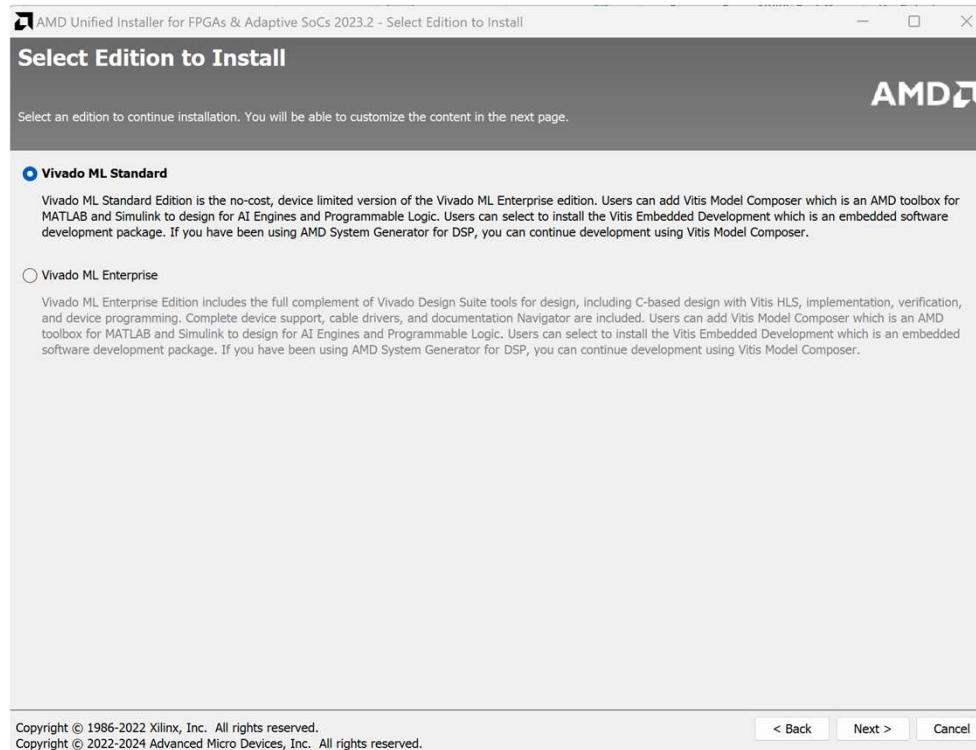
Installing Xilinx Vivado



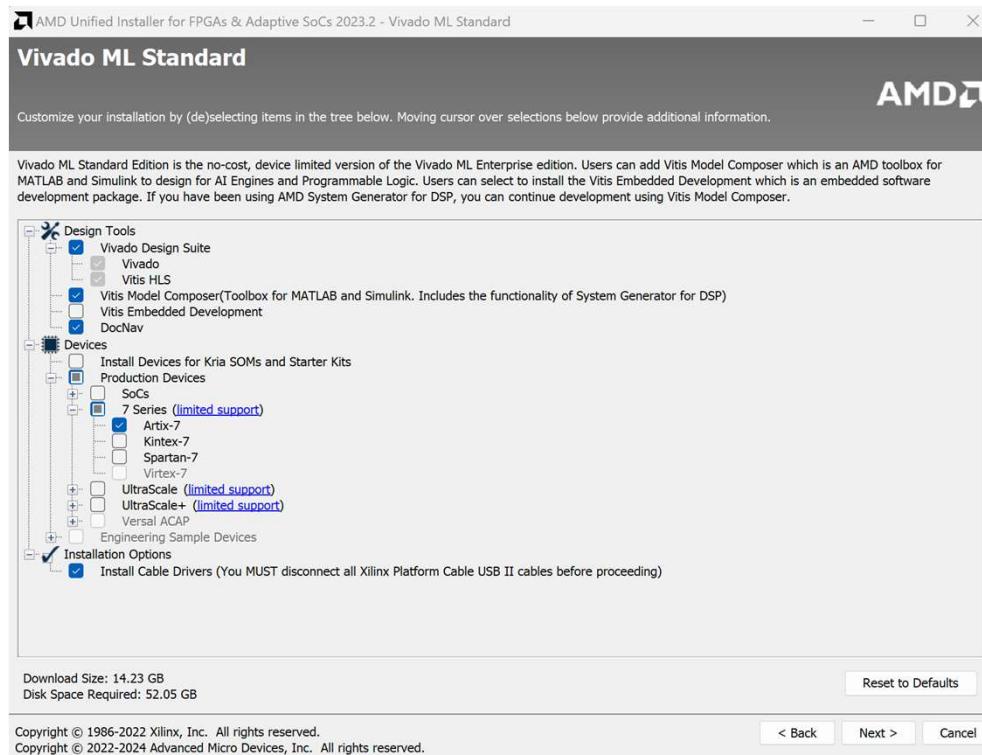
The Lab edition should also work for the labs



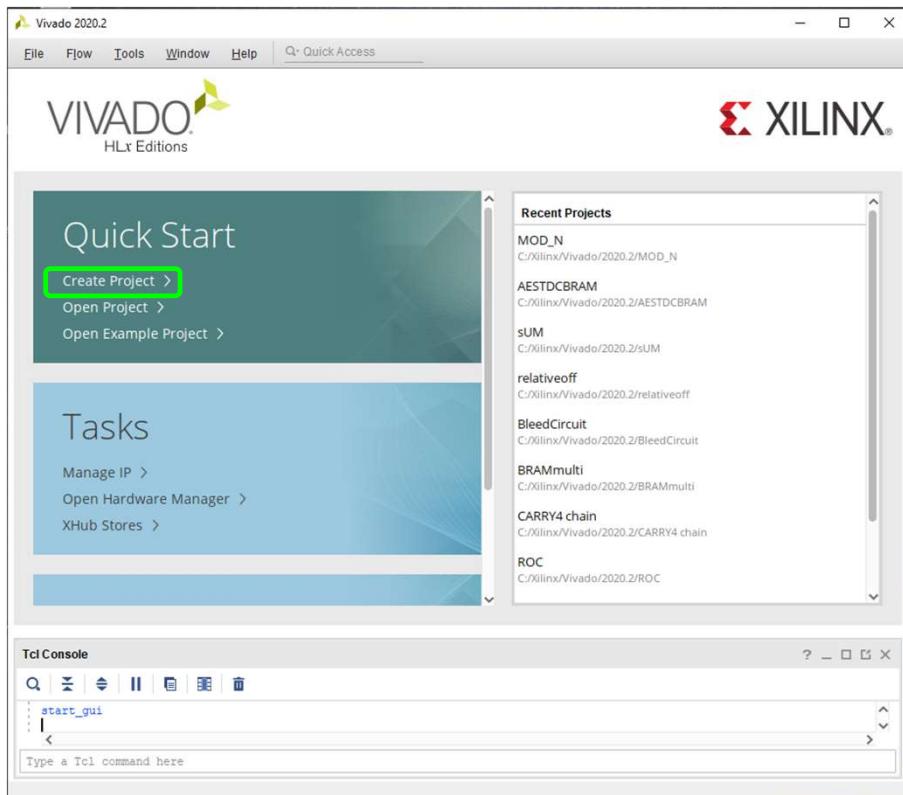
Installing Xilinx Vivado



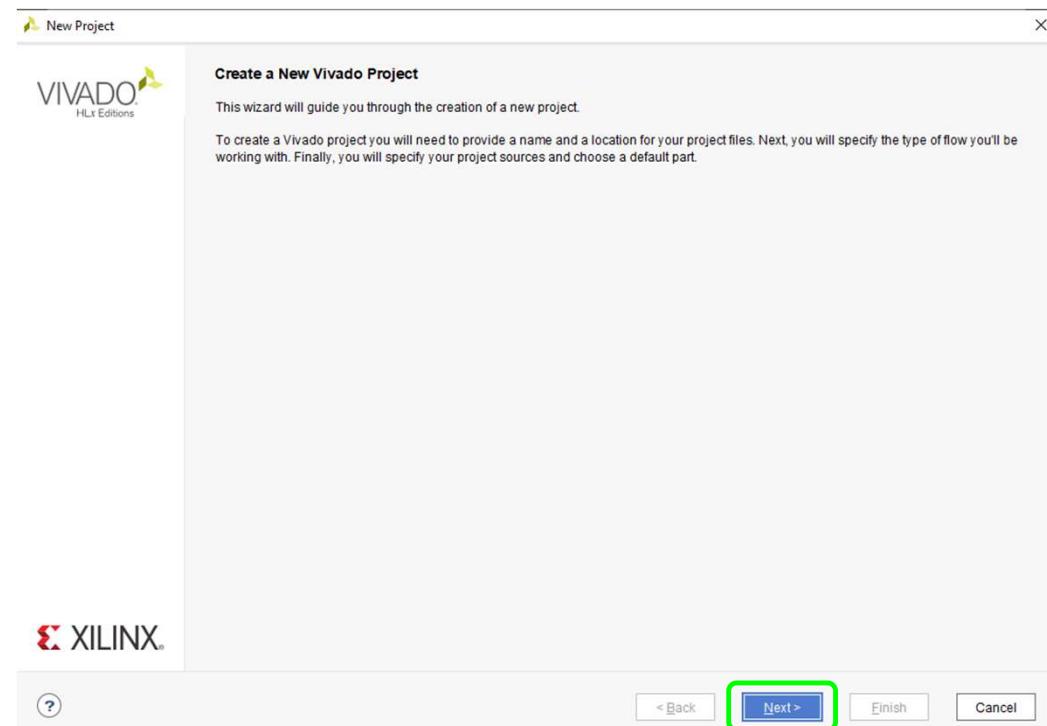
Installing Xilinx Vivado



Vivado: Creating a new Project



STEP 1 : Launch Vivado and Select 'Create Project'



STEP 2 : Click 'Next'



Vivado: Creating a new Project



New Project

Project Name
Enter a name for your project and specify a directory where the project data files will be stored.

Project name: MODN

Project location: C:/Xilinx/Vivado/2020.2

Create project subdirectory

Project will be created at: C:/Xilinx/Vivado/2020.2/MODN

? < Back Next > Finish Cancel

**STEP 3 : Enter 'Project Name'
and Click 'Next'**

New Project

Project Type
Specify the type of project to create.

RTL Project
You will be able to add sources, create block designs in IP Integrator, generate IP, run RTL analysis, synthesis, implementation, design planning and analysis.
 Do not specify sources at this time
 Project is an extensible Vivit platform

Post-synthesis Project
You will be able to add sources, view device resources, run design analysis, planning and implementation.
 Do not specify sources at this time

I/O Planning Project
Do not specify design sources. You will be able to view part/package resources.

Imported Project
Create a Vivado project from a Synplify, XST or ISE Project File.

Example Project
Create a new Vivado project from a predefined template.

? < Back Next > Finish Cancel

**STEP 4 : Click 'Next' after
verifying the default setup**



Vivado: Creating a new Project



The screenshot shows two windows of the Vivado 'New Project' wizard. The left window is titled 'Default Part' and lists various Xilinx parts. A search bar at the top has 'xc7a35tcpg' entered, resulting in four matches. The part 'xc7a35tcpg236-1' is highlighted with a green box. The right window is titled 'New Project Summary' and displays the selected part as 'xc7a35tcpg236-1'. It also shows the project name 'MODN' and other settings like Speed Grade -1. Both windows have a 'Finish' button at the bottom.

Default Part
Choose a default Xilinx part or board for your project.

Parts | Boards

Reset All Filters

Category: All Family: All

Search: Q: xc7a35tcpg (4 matches)

Part	I/O Pin Count	Available IOBs	LUT Elements	FlipFlops	Block RAMs	Ultra RAMs	DSPs	Gb Transceivers	GTPE2 T
xc7a35tcpg236-3	236	106	20800	41600	50	0	90	2	2
xc7a35tcpg236-2	236	106	20800	41600	50	0	90	2	2
xc7a35tcpg236-2L	236	106	20800	41600	50	0	90	2	2
xc7a35tcpg236-1	236	106	20800	41600	50	0	90	2	2

New Project Summary

A new RTL project named 'MODN' will be created.

The default part and product family for the new project:
Default Part: xc7a35tcpg236-1
Product: Artix-7
Family: Artix-7
Package: cpg236
Speed Grade: -1

XILINX To create the project, click Finish

**STEP 5 : Enter Board to be used
and Click 'Next'
For Basys-3 : xc7a35tcpg-1**

**STEP 6 : Review and
Click on 'Finish'**



Create new design



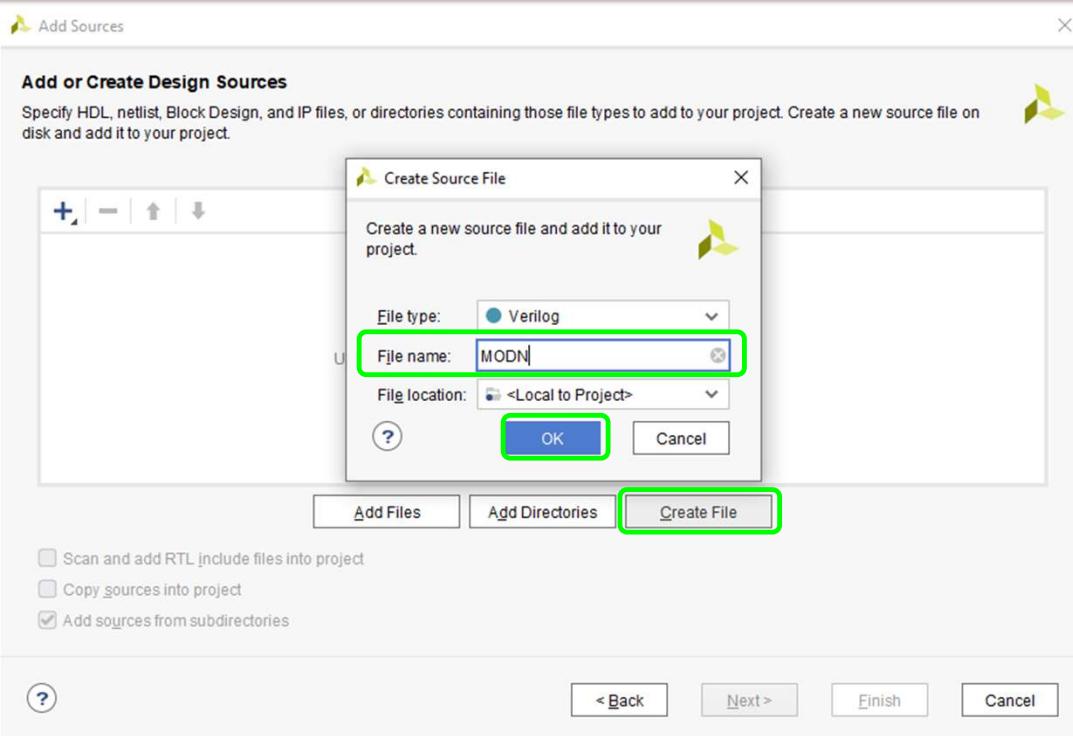
The screenshot shows the Vivado 2020.2 Project Manager interface. The left sidebar contains sections for PROJECT MANAGER, IP INTEGRATOR, SIMULATION, RTL ANALYSIS, SYNTHESIS, and IMPLEMENTATION. The PROJECT MANAGER section is expanded, showing options like Settings, Add Sources, Language Templates, and IP Catalog. The central area displays the PROJECT MANAGER - MODN window. The Sources tab is active, showing a tree view with Design Sources, Constraints, Simulation Sources (containing sim_1), and Utility Sources. A green box highlights the '+' icon in the toolbar above the tree view. Below the tree view is a Properties panel with the message "Select an object to see properties". At the bottom is a Design Runs panel showing synthesis and implementation runs.

STEP 1 : Click on '(+)' Add' symbol after creating a project

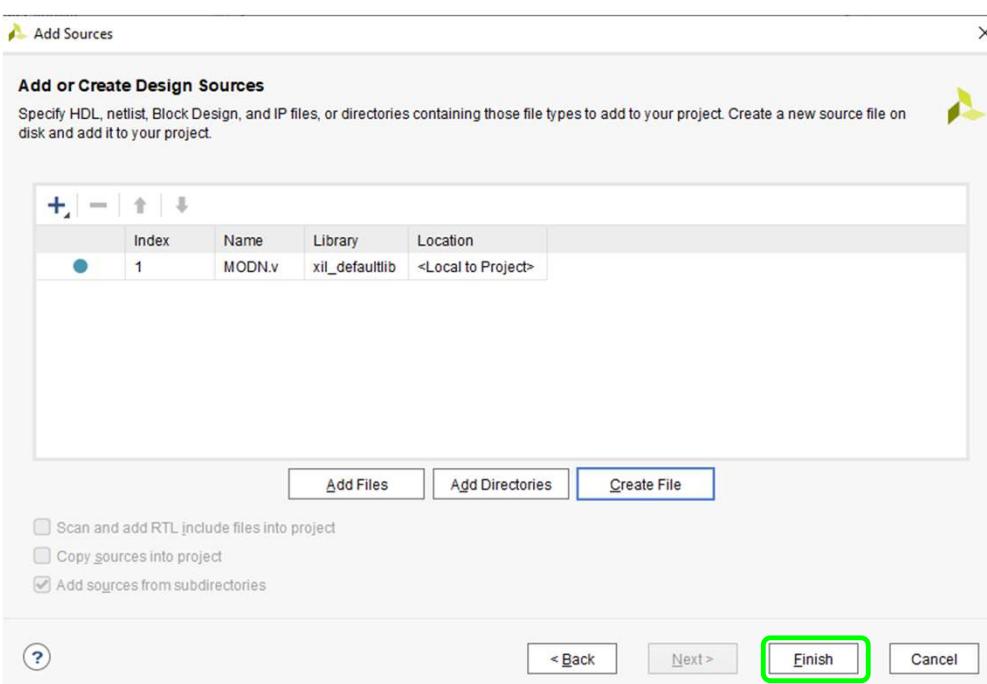
The screenshot shows the 'Add Sources' wizard in Vivado. The title bar says 'Add Sources'. The main content area is titled 'Add Sources' with the sub-instruction 'This guides you through the process of adding and creating sources for your project'. It contains two radio buttons: 'Add or create constraints' (unchecked) and 'Add or create design sources' (checked). Below these are fields for Project name (set to 'sim_1'), Project location, Product family, Project part, Top module name, Target language, and Simulator language. On the right, there are 'Settings' and 'Edit' tabs. At the bottom are buttons for '?', '< Back', 'Next >', 'Finish', and 'Cancel'. A green box highlights the 'Next >' button.

STEP 2 : Select 'Add or Create Design Sources' to create a Verilog Design Source File And Click 'Next'

Create new design

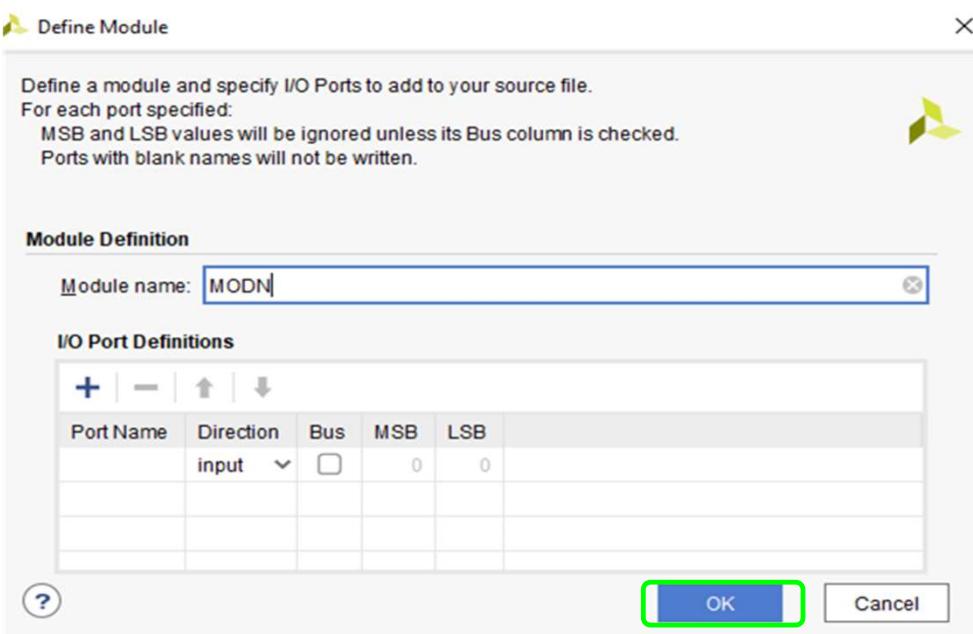


**STEP 3: Select 'Create File' and then add the 'file/module name'
Click 'OK'**

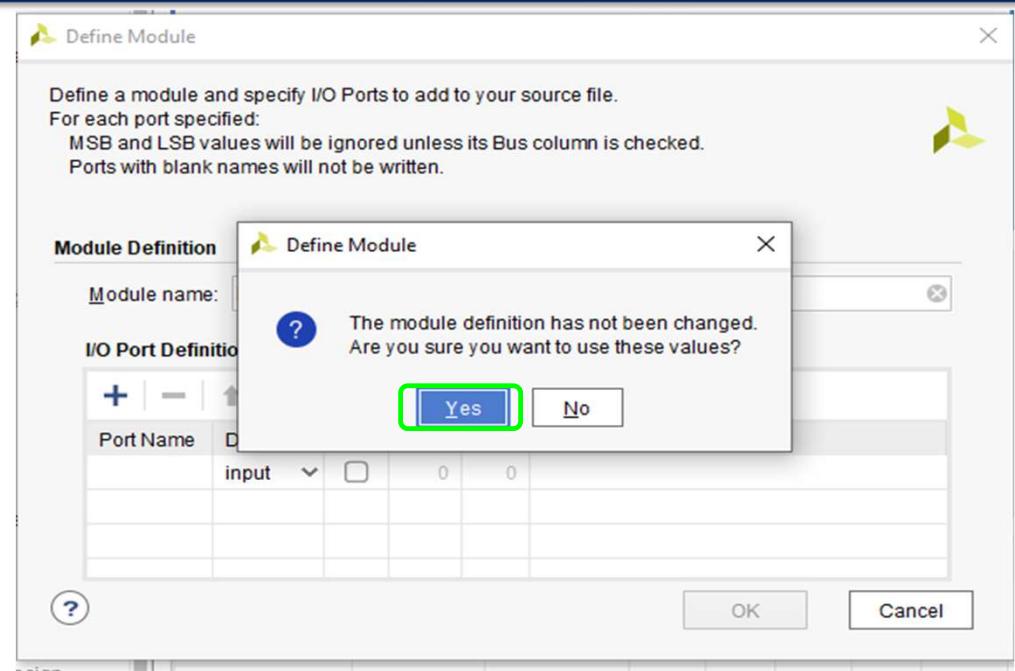


STEP 4 : Click on 'Finish'

Create new design

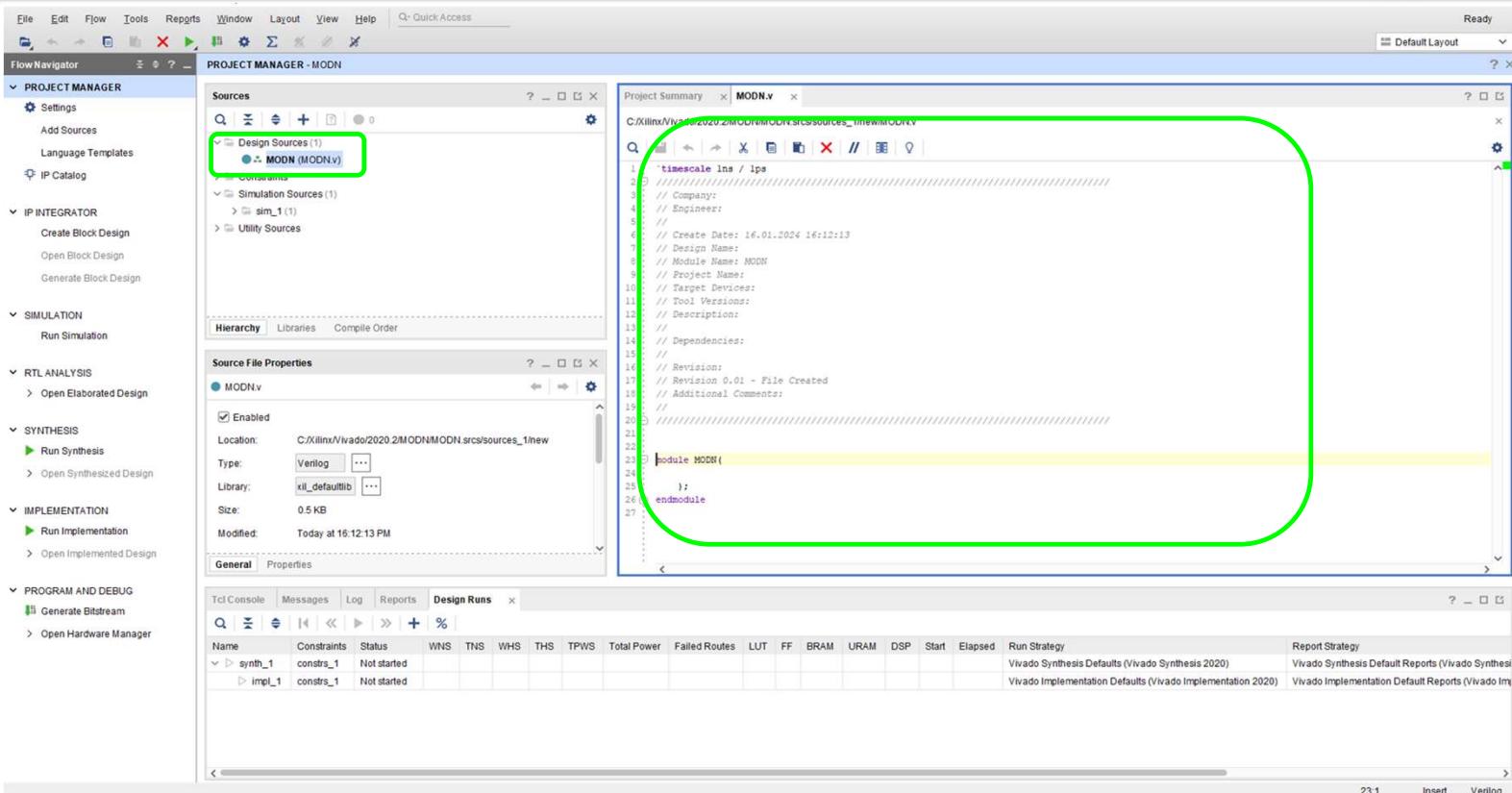


STEP 5 : Default window pops up
Verify 'Module name'
Click 'OK'



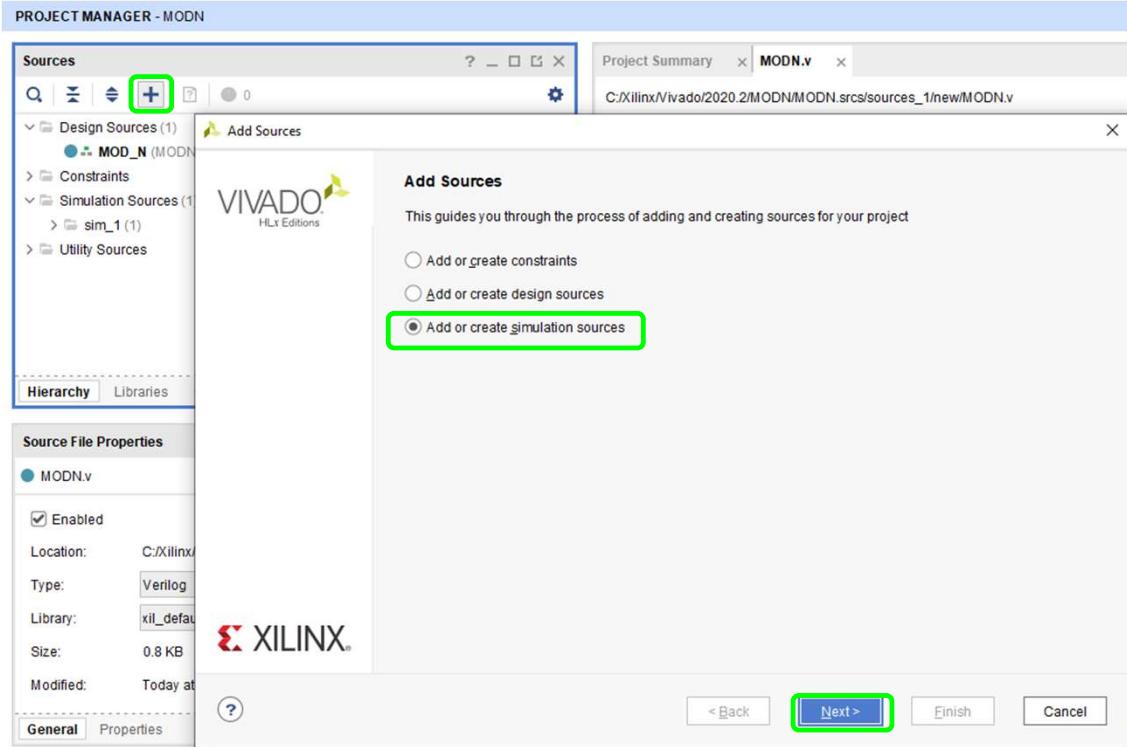
STEP 6 : Click 'Yes'

Create new design

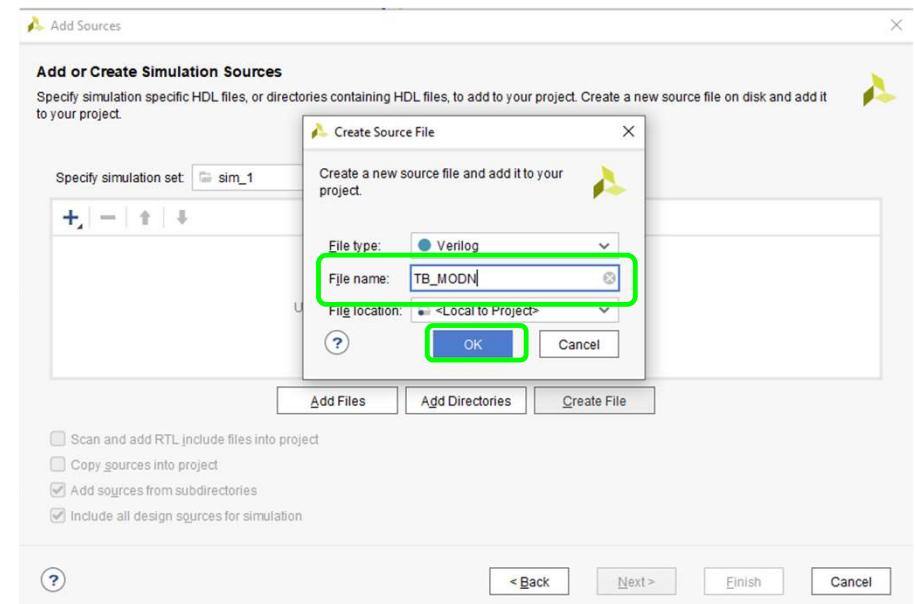


STEP 7 : On Selecting 'Design Sources' your module file appears with a default definition of module you can add your verilog code here

Create new design testbench

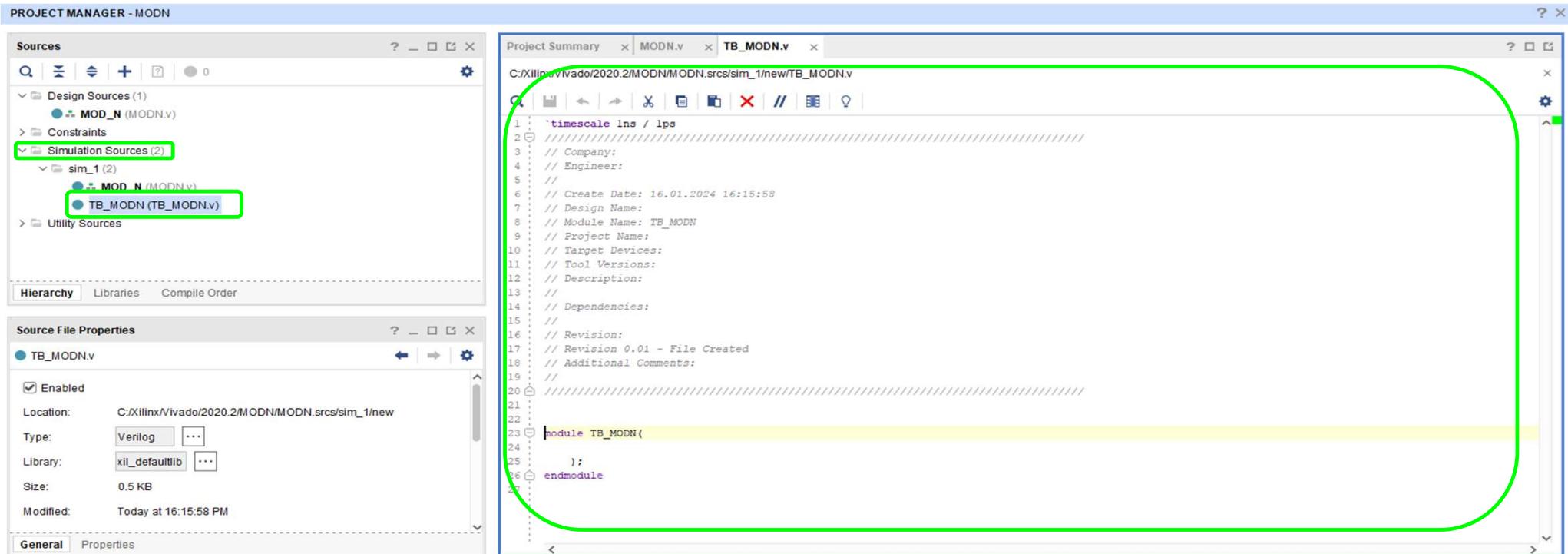


**STEP 8 : Click on ‘(+) Add’ symbol after adding your module code
 Select ‘Add or Create Simulation Sources’ to create a test bench
 Click ‘Next’**



**STEP 9 : Enter your ‘Test Bench File name’
 Click ‘OK’
 POP up windows similar to STEP 4-6 will open
 navigate the same way**

Create new design testbench



The screenshot shows the Xilinx Vivado Project Manager interface. On the left, the 'PROJECT MANAGER - MODN' window displays the 'Sources' tree. Under 'Design Sources', there is a file named 'MOD_N (MODN.v)'. Under 'Simulation Sources', there are two files: 'sim_1 (2)' which contains 'MOD_N (MODN.v)' and 'TB_MODN (TB_MODN.v)'. A green box highlights 'TB_MODN (TB_MODN.v)'. Below the tree, tabs for 'Hierarchy', 'Libraries', and 'Compile Order' are visible. On the right, the 'Project Summary' window shows the code for 'TB_MODN.v' with a yellow highlight around the module definition:

```

1 `timescale 1ns / 1ps
2 ///////////////////////////////////////////////////////////////////
3 // Company:
4 // Engineer:
5 //
6 // Create Date: 16.01.2024 16:15:58
7 // Design Name:
8 // Module Name: TB_MODN
9 // Project Name:
10 // Target Devices:
11 // Tool Versions:
12 // Description:
13 //
14 // Dependencies:
15 //
16 // Revision:
17 // Revision 0.01 - File Created
18 // Additional Comments:
19 //
20 //
21 //
22 //
23 module TB_MODN(
24 );
25 endmodule
26

```

**STEP 10 : In ‘Simulation Sources’ you can find your Test Bench with a Default Module Syntax
You can edit/add your Test Bench Code here**



भारतीय विज्ञान संस्थान



Simulating the TB

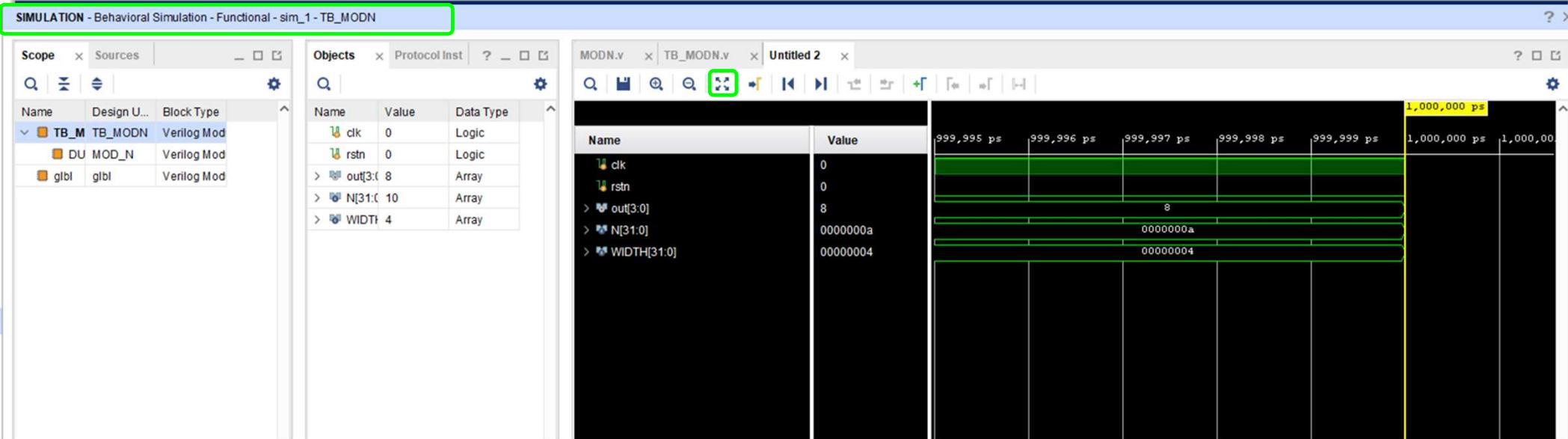
The screenshot shows the Vivado 2020.2 interface with the following details:

- Left Sidebar (PROJECT MANAGER):** Shows sections for Settings, Add Sources, Language Templates, IP Catalog, IP INTEGRATOR, Create Block Design, Open Block Design, Generate Block Design, and **SIMULATION**. The **Run Simulation** button is highlighted with a green box.
- Source Editor (Top Right):** Displays the **TB_MODN.v** file content, which is a behavioral testbench for a module named MOD_N. It includes parameters N=10 and WIDTH=4, and logic for a DUT (clk, rstn, out).
- Design Runs Table (Bottom Right):** Shows a table with columns: Name, Constraints, Status, WNS, TNS, WHS, THS, TPWS, Total Power, Failed Routes, LUT, FF, BRAM, URAM, DSP, Start, Elapsed, Run Strategy, and Report Strategy. It lists two runs: synth_1 and impl_1, both in the 'Not started' status.
- Taskbar (Bottom):** Includes icons for Start, Search, Task View, File Explorer, Mail, Edge, Google Chrome, and a local file icon. It also shows system information like 19°C Haze, ENG, 16:19, and 16-01-2024.

In 'Simulation'
Click on 'Run Simulation'
Click on
'Run Behavioral Simulation'

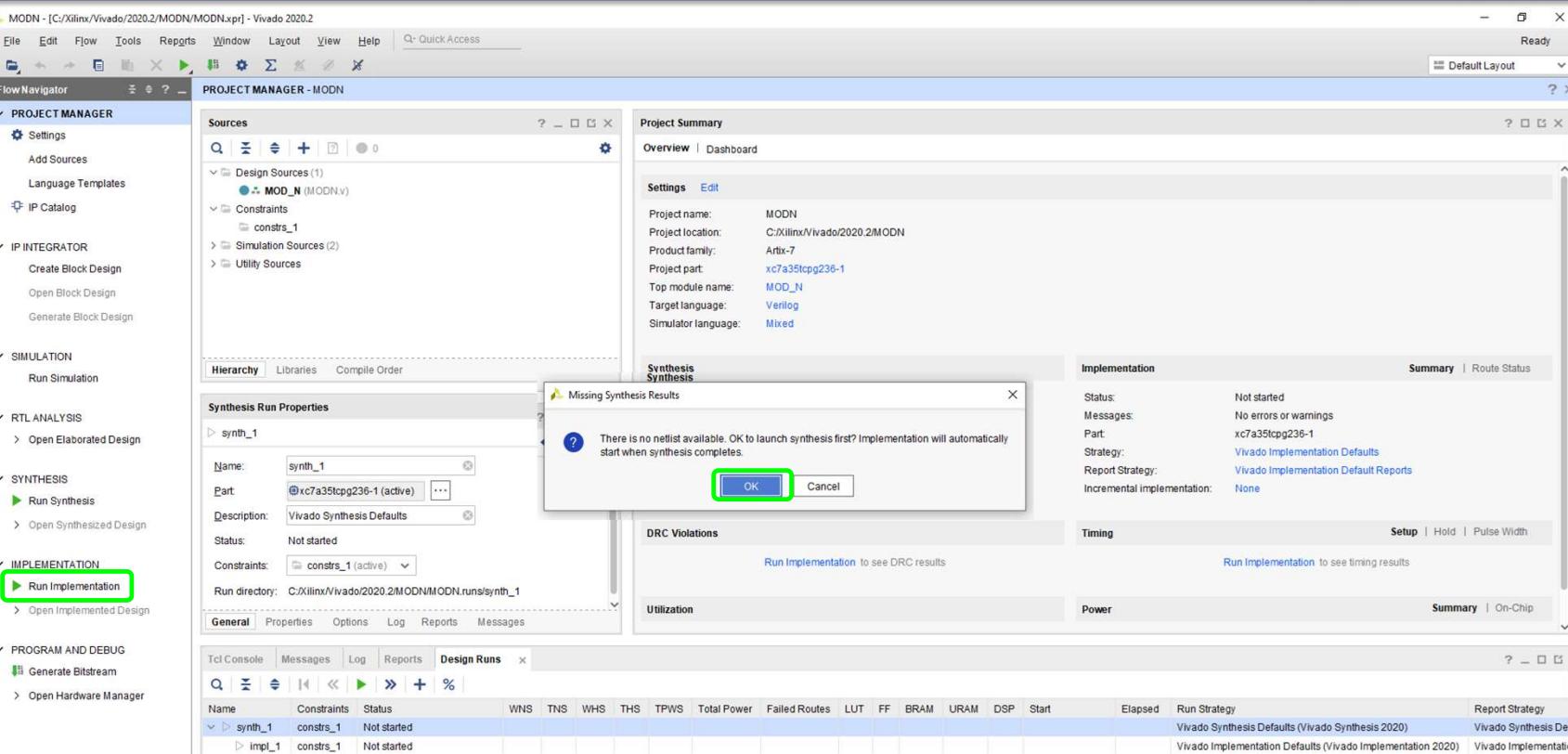


Observe the timing waveforms



'Simulation Window' Pops up where you can view your waveforms based on your Testbench
To make graph Fit to the Screen Click on the 'outward arrows' highlighted

Implementation



The screenshot shows the Vivado 2020.2 Project Manager interface. The 'Implementation' tab is selected. A pop-up window titled 'Missing Synthesis Results' is centered over the main window. It contains the message: 'There is no netlist available. OK to launch synthesis first? Implementation will automatically start when synthesis completes.' Below the message are two buttons: 'OK' (highlighted with a green box) and 'Cancel'. In the background, the Project Summary and various implementation tabs like Timing, Power, and Utilization are visible.

STEP 1 : After

'Simulation'

Click on 'Run Implementation'

If you have not done 'Synthesis' yet you will see a pop up Click 'OK' and continue

You may run synthesis separately before implementation if you want

Synthesis will convert the RTL code to the netlist.

Implementation tool will take the netlist as input and does optimization, placement and routing.

Implementation

Implementation Completed

Implementation successfully completed.

Next

- Open Implemented Design
- Generate Bitstream
- View Reports

Don't show this dialog again

OK **Cancel**

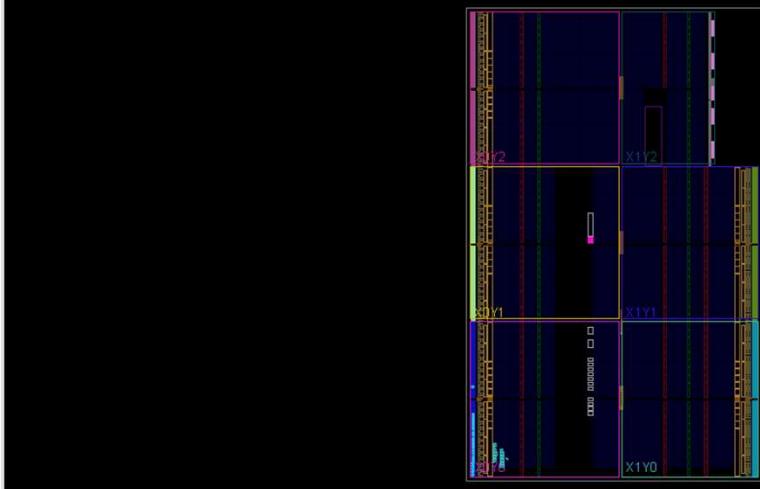
IMPLEMENTED DESIGN - xc7a35tcpg236-1

Sources Netlist Synthesis Run Project ? □ X

Project Summary Device

N MOD_N

- > Nets (189)
- > Leaf Cells (160)



Tcl Console Messages Log Reports Design Runs DRC Methodology Power Timing

Design Timing Summary

Setup	Hold	Pulse Width
Worst Negative Slack (WNS): NA	Worst Hold Slack (WHS): NA	Worst Pulse Width Slack (WPWS): NA
Total Negative Slack (TNS): NA	Total Hold Slack (THS): NA	Total Pulse Width Negative Slack (TPWS): NA
Number of Failing Endpoints: NA	Number of Failing Endpoints: NA	Number of Failing Endpoints: NA
Total Number of Endpoints: NA	Total Number of Endpoints: NA	Total Number of Endpoints: NA

There are no user specified timing constraints.

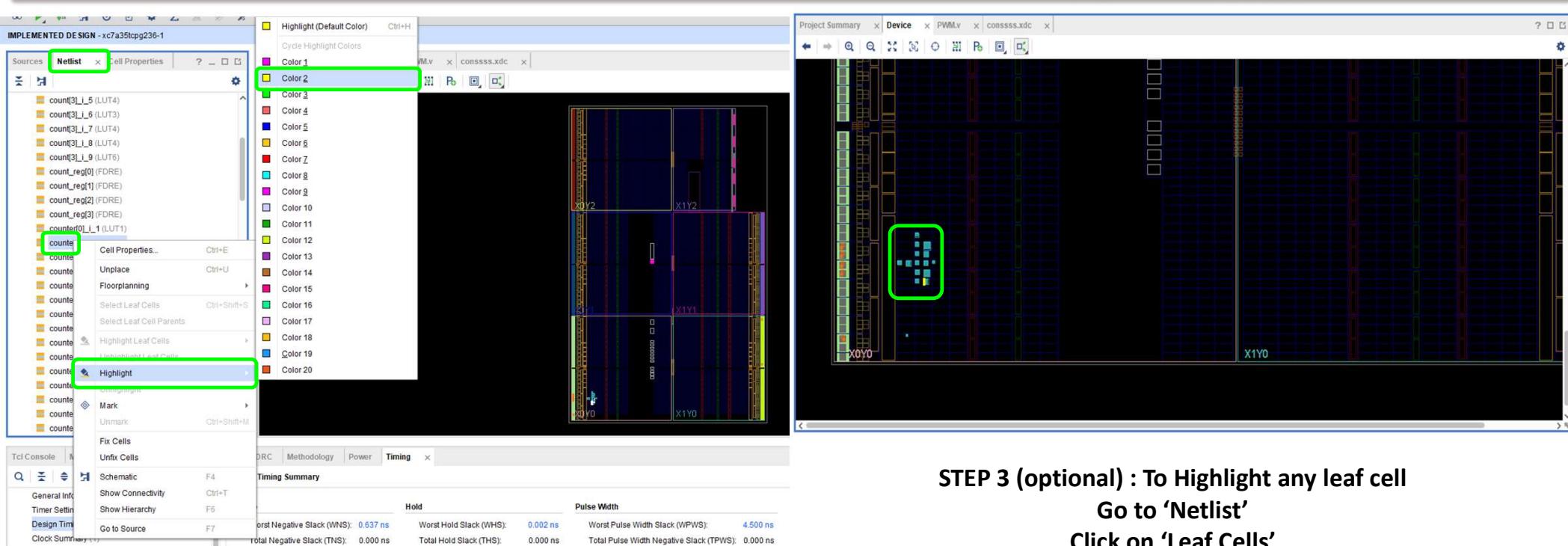
Timing Summary - impl_1 (saved)

STEP 2 : After implementation a window will pop up as shown above

Click on 'Open Implemented Design'

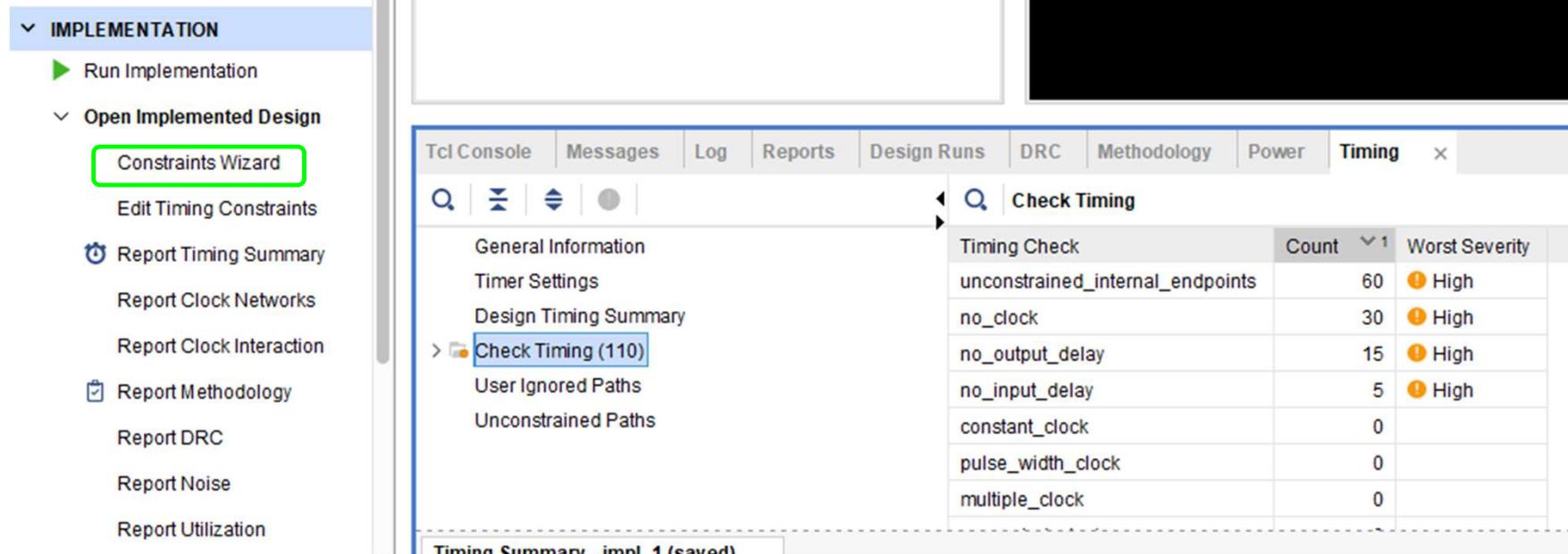
You can select 'Device' you can see the nets and leaf cells of your design

Implementation



STEP 3 (optional) : To Highlight any leaf cell
Go to 'Netlist'
Click on 'Leaf Cells'
Right Click on 'Leaf cell to be highlighted'
Hover on 'Highlight'
Select color with which it must be highlighted

Clocks



The screenshot shows the Vivado IDE interface. On the left, the 'IMPLEMENTATION' tab is selected in the navigation bar. Under 'Open Implemented Design', the 'Constraints Wizard' option is highlighted with a green border. The main workspace displays the 'Timing' tab of the 'Check Timing' report. The report table lists various timing constraints and their counts and severities:

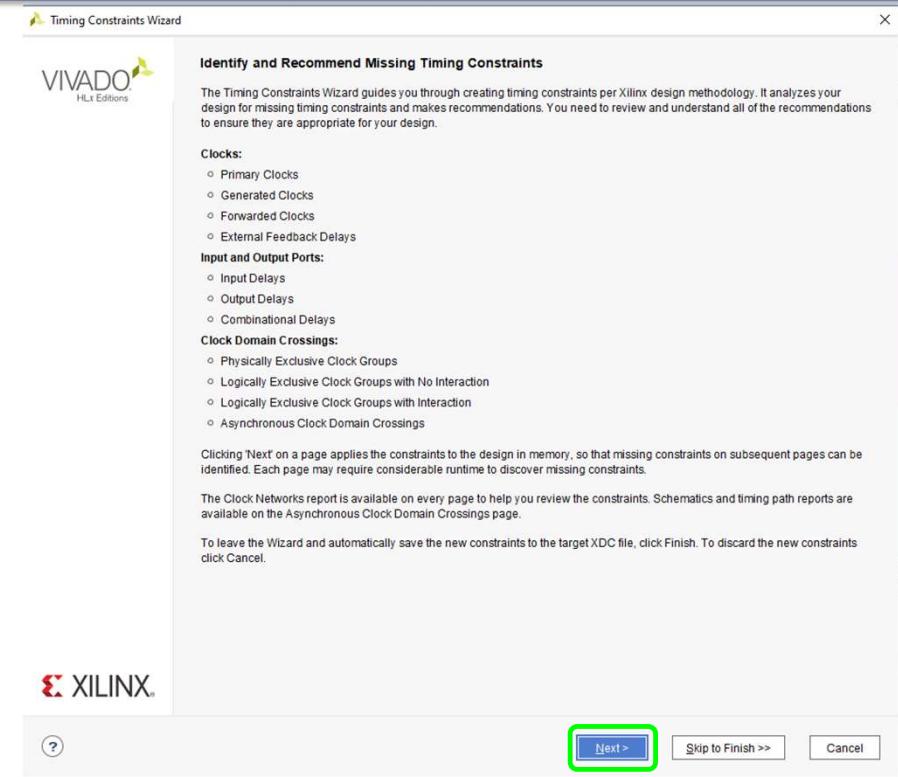
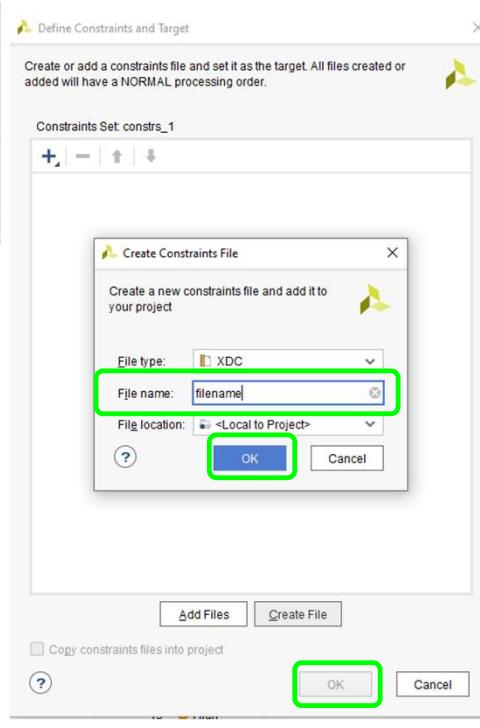
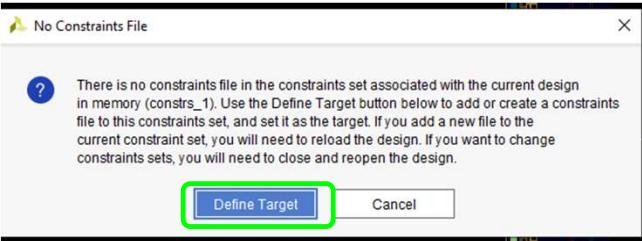
Timing Check	Count	Worst Severity
unconstrained_internal_endpoints	60	!! High
no_clock	30	!! High
no_output_delay	15	!! High
no_input_delay	5	!! High
constant_clock	0	
pulse_width_clock	0	
multiple_clock	0	

If Clock is not defined well, i.e, you have just mapped the internal Clock PIN of FPGA. You might see these warnings and won't be able to extract a Timing Report
 To Define a Clock Signal properly follow these steps

STEP 1: Click on 'Constraints Wizard'

<https://docs.xilinx.com/r/2021.2-English/ug903-vivado-using-constraints/Timing-Constraints-Wizard>
<https://www.xilinx.com/video/hardware/using-vivado-timing-constraint-wizard.html>

Clocks



STEP 2: Click on 'Define Target'

This pop up only appears if you have not generated a constraint file yet

Create a 'Constraint File'

Click 'OK'

STEP 3: Click on 'Next'

Clocks

Timing Constraints Wizard

Primary Clocks

Primary clocks usually enter the design through input ports. Specify the period and optionally a name and waveform (rising and falling edge times) to describe the duty cycle if not 50%. [More info](#)

Recommended Constraints

Object	Name	Frequency (MHz)	Period (ns)	Rise At (ns)	Fall At (ns)	Jitter (ns)
<input checked="" type="checkbox"/>	clk	100.000	10.000	0.000	5.000	

Constraints for Pulse Width Check Only

Object	Name	Frequency (MHz)	Period (ns)	Rise At (ns)	Fall At (ns)	Jitter (ns)
<input type="checkbox"/>						

Tcl Command Preview(1)

```
create_clock -period 10.000 -name clk -waveform {0.000 5.000} [get_ports {clk}]
```

Next > **Next >** **Skip to Finish >>** **Cancel**

**STEP 4: Select ‘Frequency (MHz)’
And enter the frequency value
Rest all will be automatically filled
Click on ‘Next’**

Timing Constraints Wizard

Input Delays

Input delays describe relative phase between reference clocks (usually board clocks) and input signals at the FPGA boundary. Inaccurate input delay values can make timing fail and affect implementation quality of results. [More info](#)

Recommended Constraints

Interface	Clock	Synchronous	Alignment	Data Rate and Edge
<input checked="" type="checkbox"/>	N[*]	clk	System	Edge
<input checked="" type="checkbox"/>	rstn	clk	System	Edge

Delay Parameters

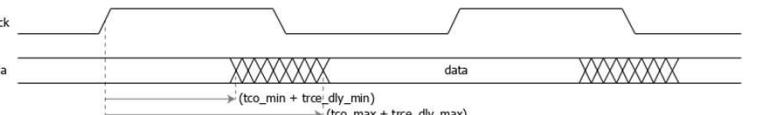
clock period:	10 ns
_min:	0.1 ns
_max:	0.1 ns
_dly_min:	0.1 ns
_dly_max:	0.1 ns

Rise Max = tco_max + trce_dly_max
Rise Min = tco_min + trce_dly_min

Tcl Command Preview(4)

Existing Set Input Delay Constraints (0)

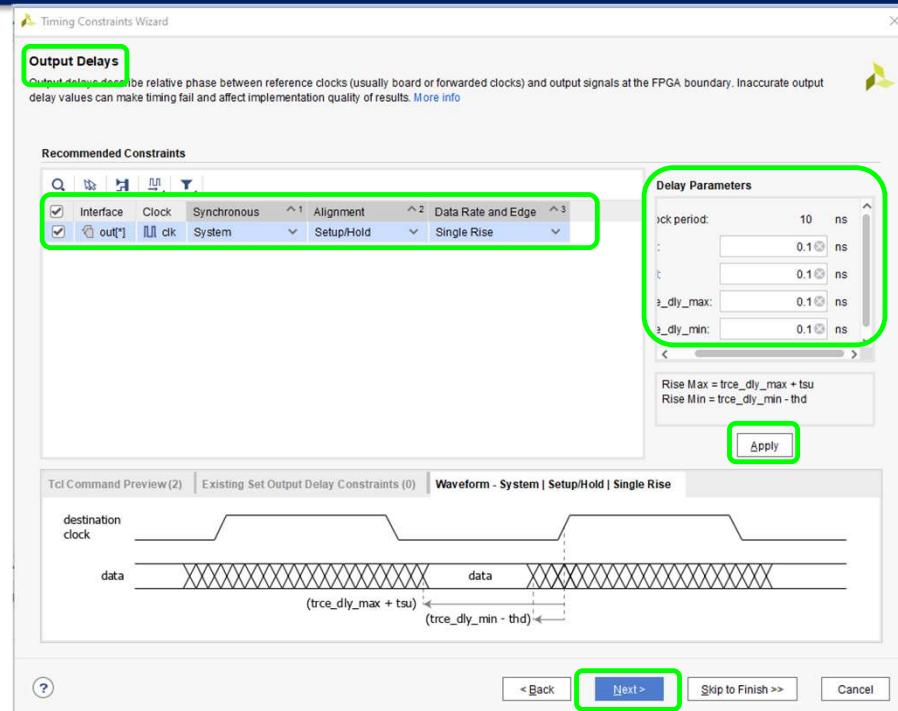
Waveform - System | Edge | Single Rise



Next > **Next >** **Skip to Finish >>** **Cancel**

**STEP 5: Input Delays
In ‘Delay Parameters’
Enter all the Delay values for each of the Interface
Click ‘Apply’
Click on ‘Next’**

Clocks



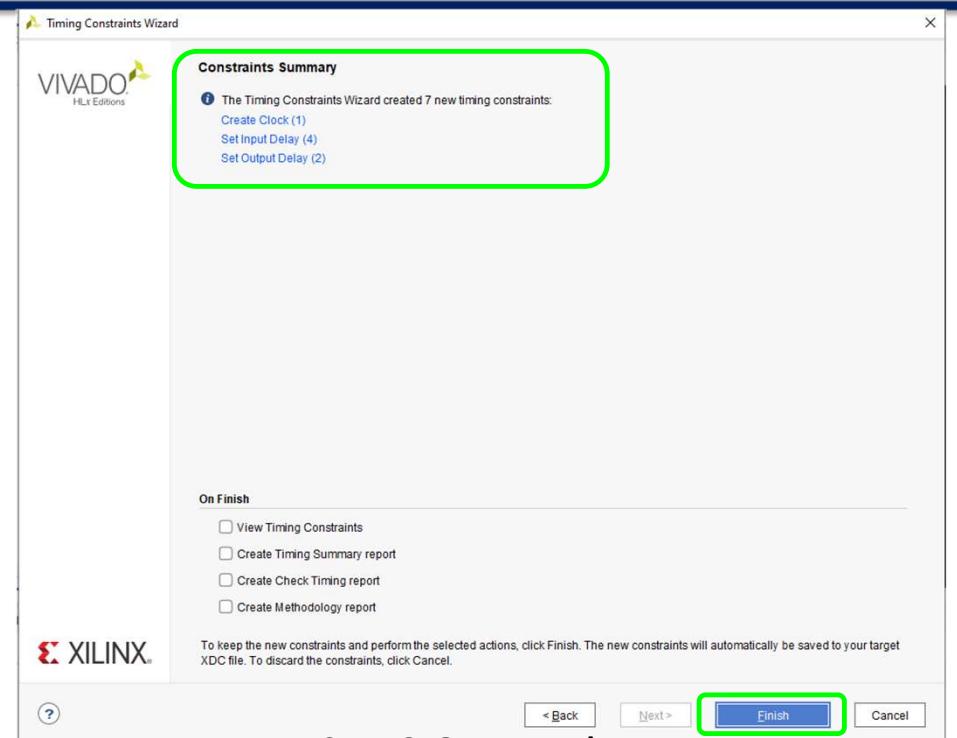
STEP 6: Output Delays

In 'Delay Parameters'

Enter all the Delay values for each of the Interface

Click 'Apply'

Click on 'Skip to Finish'



STEP 6: Output Delays

Verify the constraints added

Click on 'Finish'

Now if you re-run Implementation warnings shown before will disappear



I/O Planning - 1



Flow Navigator

IMPLEMENTED DESIGN - xc7a35tcpg236-1

Sources Netlist Constraint Set Project

Language Templates IP Catalog

IP INTEGRATOR Create Block Design Open Block Design Generate Block Design

SIMULATION Run Simulation

RTL ANALYSIS Open Elaborated Design

SYNTHESIS Run Synthesis Open Synthesized Design

IMPLEMENTATION Run Implementation Open Implemented Design

- Constraints Wizard
- Edit Timing Constraints
- Report Timing Summary
- Report Clock Networks
- Report Clock Interaction
- Report Methodology
- Report DRC
- Report Noise
- Report Utilization
- Report Power
- Schematic**

Tcl Console Messages Reports Design Runs DRC Methodology Power Timing

Project Summary MODN.v Schematic

158 Cells 22 I/O Ports 189 Nets

Hierarchy Libraries Compile Order

INFO: [Netlist 29-28] Unisim Transformation completed in 0 CPU seconds
INFO: [Project 1-479] Netlist was created with Vivado 2020.2
INFO: [Project 1-570] Preparing netlist for logic optimization
Reading XDEF placement...
Reading placer database...
Reading XDEF routing.
Read XDEF File: Time (s): cpu = 00:00:00 ; elapsed = 00:00:00.023 . Memory (MB): peak = 2123.840 ; gain = 0.000
Restored from archive | CPU: 0.000000 secs | Memory: 0.000000 MB |
Finished XDEF File Restore: Time (s): cpu = 00:00:00 ; elapsed = 00:00:00.023 . Memory (MB): peak = 2123.840 ; gain = 0.000

Type a Tcl command here

STEP 1: Click on 'Schematic' in 'Open Implemented Design' Select 'I/O Ports'



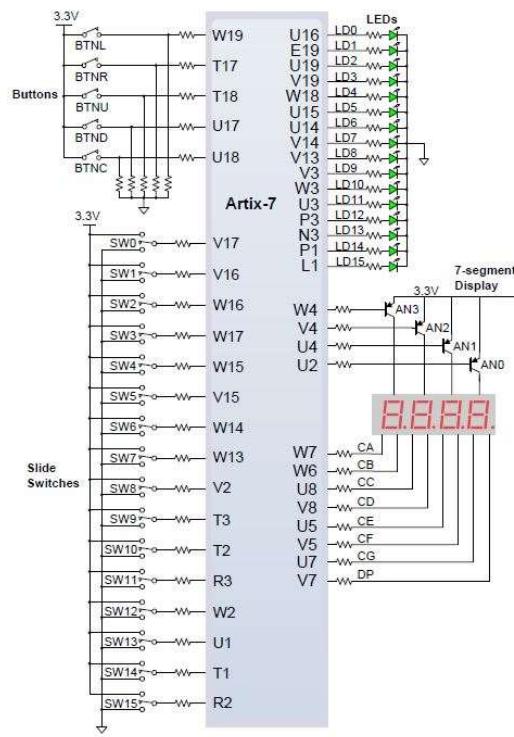
I/O Planning - 1

The screenshot shows the Xilinx Vivado IDE interface for an implemented design named "xc7a35tcpg236-1". The top menu bar includes File, Edit, Flow, Tools, Reports, Window, Layout, View, Help, and a Quick Access search bar. A status bar at the top right indicates "write bitstream Complete". A green box highlights the "I/O Planning" dropdown menu in the top right corner.

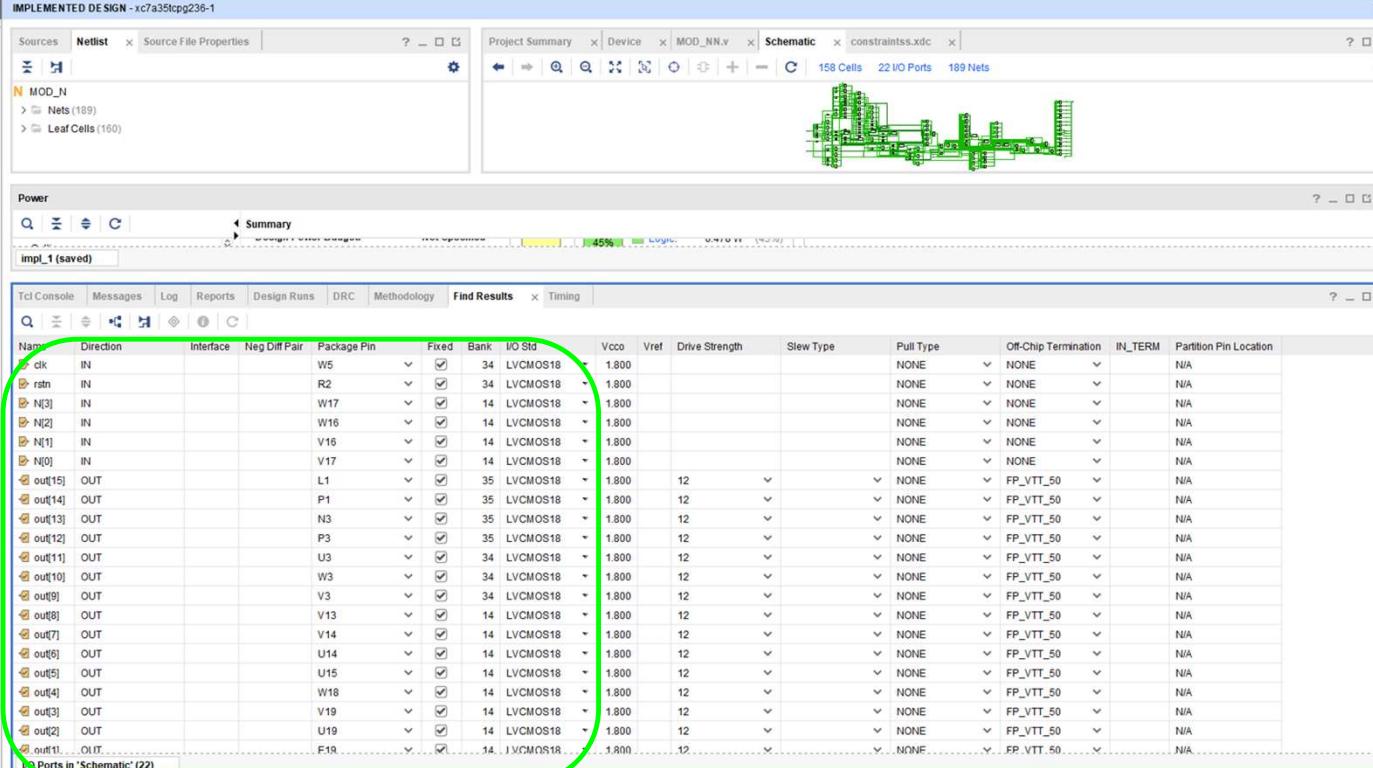
The main workspace displays a "Sources" panel on the left containing "Design Sources", "Constraints", "Simulation Sources", and "Utility Sources". The "Device" panel on the right shows a detailed I/O pin map for a device package, with various pins labeled (e.g., X1Y2, X1Y1, X0Y0, X1Y0). A "Properties" panel below the sources shows details for a selected clock source named "clk_IBUF_BUFG". The bottom navigation bar includes tabs for Tcl Console, Messages, Log, Reports, Design Runs, Utilization, Timing, DRC, Power, Methodology, Package Pins, and I/O Ports.

You may do the IO mapping anytime after the simulation step by selecting the I/O Planning from the dropdown menu at the top

Basys 3 I/O



I/O Planning - 1



The screenshot shows the Xilinx Vivado IDE interface with the following sections:

- Flow Navigator:** On the left, showing various project management options like IP Catalog, IP Integrator, Simulation, RTL Analysis, Synthesis, Implementation, and Program and Debug.
- Implemented Design - xc7a35tcpg236-1:** The main workspace displays the Schematic view with a complex circuit diagram.
- Netlist:** Shows the source file properties for the current design.
- Schematic:** Shows the schematic representation of the design.
- Power:** Shows power analysis details.
- Find Results:** A table showing I/O port constraints. A green circle highlights the 'Fixed' column, which is checked for most ports. The table includes columns for Name, Direction, Interface, Neg Diff Pair, Package Pin, Fixed, Bank, IO Std, Vcco, Vref, Drive Strength, Slew Type, Pull Type, Off-Chip Termination, IN_TERM, and Partition Pin Location.

Name	Direction	Interface	Neg Diff Pair	Package Pin	Fixed	Bank	IO Std	Vcco	Vref	Drive Strength	Slew Type	Pull Type	Off-Chip Termination	IN_TERM	Partition Pin Location	
clk	IN			W5	<input checked="" type="checkbox"/>	34	LVCMS018	-	1.800			None	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	N/A
rstn	IN			R2	<input checked="" type="checkbox"/>	34	LVCMS018	-	1.800			None	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	N/A
N[3]	IN			W17	<input checked="" type="checkbox"/>	14	LVCMS018	-	1.800			None	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	N/A
N[2]	IN			W16	<input checked="" type="checkbox"/>	14	LVCMS018	-	1.800			None	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	N/A
N[1]	IN			V16	<input checked="" type="checkbox"/>	14	LVCMS018	-	1.800			None	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	N/A
N[0]	IN			V17	<input checked="" type="checkbox"/>	14	LVCMS018	-	1.800			None	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	N/A
out[15]	OUT			L1	<input checked="" type="checkbox"/>	35	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[14]	OUT			P1	<input checked="" type="checkbox"/>	35	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[13]	OUT			N3	<input checked="" type="checkbox"/>	35	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[12]	OUT			P3	<input checked="" type="checkbox"/>	35	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[11]	OUT			U3	<input checked="" type="checkbox"/>	34	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[10]	OUT			W3	<input checked="" type="checkbox"/>	34	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[9]	OUT			V3	<input checked="" type="checkbox"/>	34	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[8]	OUT			V13	<input checked="" type="checkbox"/>	14	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[7]	OUT			V14	<input checked="" type="checkbox"/>	14	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[6]	OUT			U14	<input checked="" type="checkbox"/>	14	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[5]	OUT			U15	<input checked="" type="checkbox"/>	14	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[4]	OUT			W18	<input checked="" type="checkbox"/>	14	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[3]	OUT			V19	<input checked="" type="checkbox"/>	14	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[2]	OUT			U19	<input checked="" type="checkbox"/>	14	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50
out[1].....OUT				F19	<input checked="" type="checkbox"/>	14	LVCMS018	-	1.800	12	>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	None	<input checked="" type="checkbox"/>	FP_VTT_50

STEP 2:

Select the 'Package Pin'
 Tick box of 'Fixed' field
 Select respective 'I/O Standard'

Repeat this for all I/O ports in design

Save after selecting all fields
 It will automatically generate a constraint file

I/O Planning - 1 and BITSTREAM

Screenshot of Xilinx Vivado Design Suite interface showing I/O planning and bitstream generation.

The left sidebar shows the Flow Navigator with the following sections expanded:

- IMPLEMENTATION** (highlighted)
- PROGRAM AND DEBUG**

The main window displays the following components:

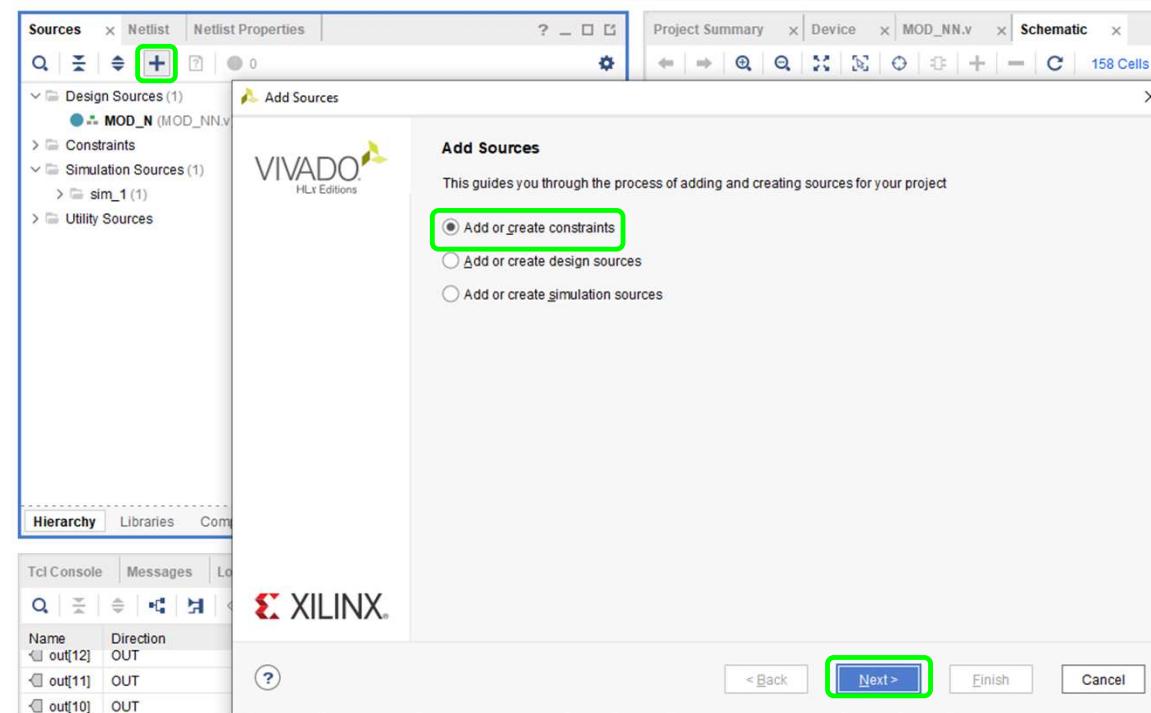
- IMPLEMENTED DESIGN - xc7a35tcpg236-1** (Netlist tab): Shows the project structure under MOD_N, including Nets (189) and Leaf Cells (160).
- Project Summary**: Device: MOD_NN.v, Schematic: constraints.xdc, 158 Cells, 22 I/O Ports, 189 Nets.
- Schematic View**: A green wireframe representation of the implemented design.
- Power Analysis**: Summary of power consumption (45% Logic, 0.470W Power).
- Implementation Report**: A detailed table of I/O pin assignments:

Name	Direction	Interface	Neg Diff Pair	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive Strength	Slew Type	Pull Type	Off-Chip Termination	IN_TERM	Partition Pin Location			
clk	IN			W5	<input checked="" type="checkbox"/>	34	LVC MOS18	-	1.800			NONE	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	N/A		
rstn	IN			R2	<input checked="" type="checkbox"/>	34	LVC MOS18	-	1.800			NONE	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	N/A		
N[3]	IN			W17	<input checked="" type="checkbox"/>	14	LVC MOS18	-	1.800			NONE	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	N/A		
N[2]	IN			W16	<input checked="" type="checkbox"/>	14	LVC MOS18	-	1.800			NONE	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	N/A		
N[1]	IN			V16	<input checked="" type="checkbox"/>	14	LVC MOS18	-	1.800			NONE	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	N/A		
N[0]	IN			V17	<input checked="" type="checkbox"/>	14	LVC MOS18	-	1.800			NONE	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	N/A		
out[15]	OUT			L1	<input checked="" type="checkbox"/>	35	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[14]	OUT			P1	<input checked="" type="checkbox"/>	35	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[13]	OUT			N3	<input checked="" type="checkbox"/>	35	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[12]	OUT			P3	<input checked="" type="checkbox"/>	35	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[11]	OUT			U3	<input checked="" type="checkbox"/>	34	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[10]	OUT			W3	<input checked="" type="checkbox"/>	34	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[9]	OUT			V3	<input checked="" type="checkbox"/>	34	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[8]	OUT			V13	<input checked="" type="checkbox"/>	14	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[7]	OUT			V14	<input checked="" type="checkbox"/>	14	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[6]	OUT			U14	<input checked="" type="checkbox"/>	14	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[5]	OUT			U15	<input checked="" type="checkbox"/>	14	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[4]	OUT			W18	<input checked="" type="checkbox"/>	14	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[3]	OUT			V19	<input checked="" type="checkbox"/>	14	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[2]	OUT			U19	<input checked="" type="checkbox"/>	14	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A
out[1]	OUT			F19	<input checked="" type="checkbox"/>	14	LVC MOS18	-	1.800	12	<	<	<input checked="" type="checkbox"/>	NONE	<input checked="" type="checkbox"/>	FP_VTT_50	<input checked="" type="checkbox"/>	N/A

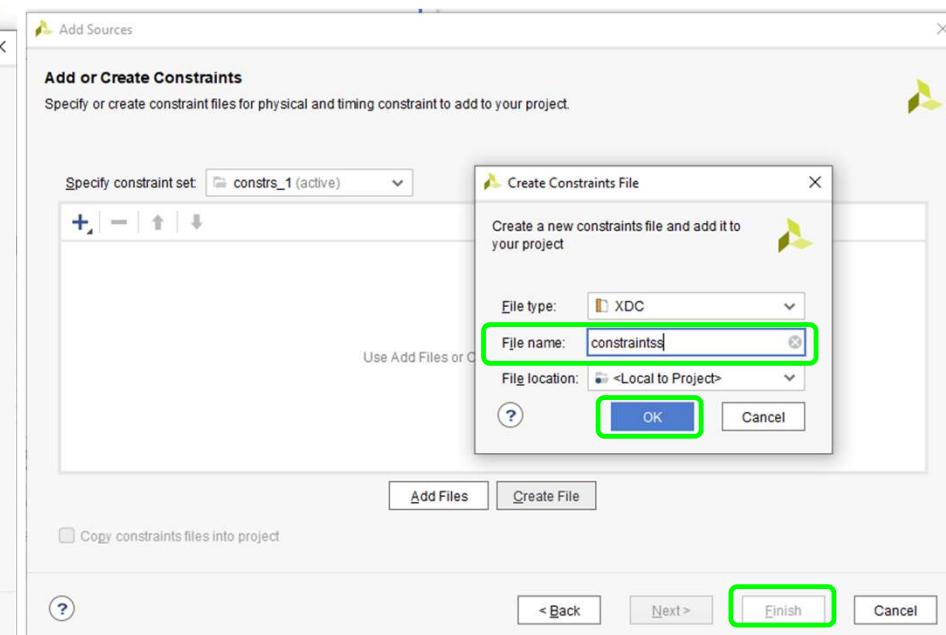
Generate Bitstream button is highlighted in the bottom-left corner of the implementation report panel.

STEP4: Sample of Pins Selected and click on 'Generate Bitstream'

I/O PLANNING - 2

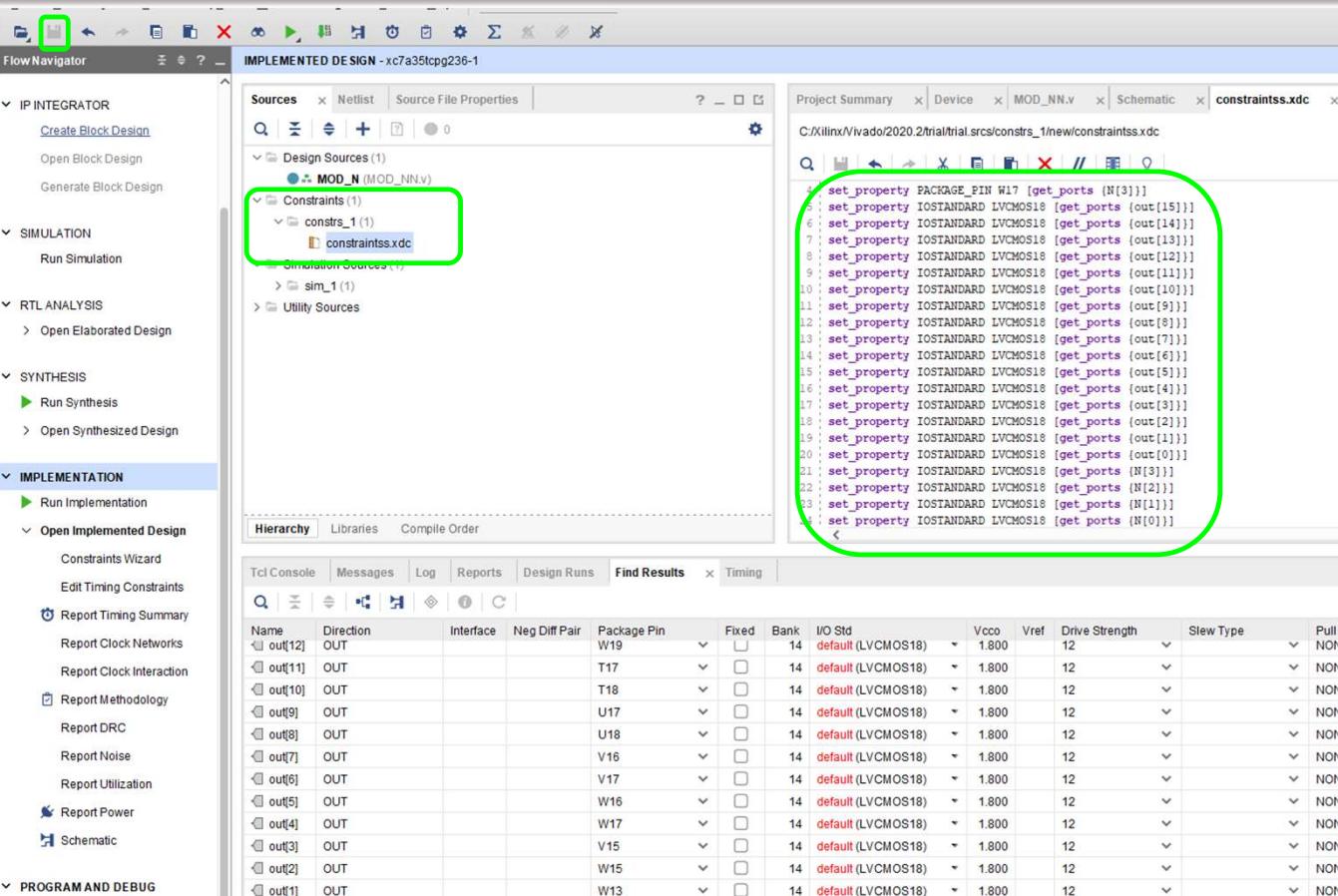


**STEP 1: If you already have a constraints file for your module with pins defined
You can directly add a constraint file by clicking on '(+) Add' Symbol
Click on 'Add or create constraints' and select 'Next'**



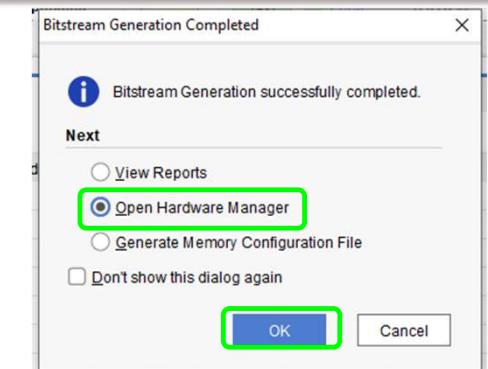
**STEP 2: Enter a 'Constraint File' name
Click on 'Add or create constraints' and
select 'Next'
And 'Finish' later**

I/O PLANNING - 2 and BITSTREAM



The screenshot shows the Xilinx Vivado IDE interface. The left sidebar has sections for IP INTEGRATOR, SIMULATION, RTL ANALYSIS, SYNTHESIS, IMPLEMENTATION, and PROGRAM AND DEBUG. The IMPLEMENTATION section is currently selected. The main workspace shows the 'IMPLEMENTED DESIGN - xc7a35tcpg236-1' project. In the Sources tab, under Design Sources, there is a 'Constraints' folder containing a file named 'constraints.xdc'. This file is highlighted with a green oval. Below the sources, there is a table titled 'Hierarchy' showing I/O pin assignments. The table includes columns for Name, Direction, Interface, Neg Diff Pair, Package Pin, Fixed, Bank, I/O Std, Vcco, Vref, Drive Strength, Slew Type, and Pull Typ. Most pins are assigned to Bank 14 and have 'default (LVCMS18)' as the I/O Std.

Name	Direction	Interface	Neg Diff Pair	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive Strength	Slew Type	Pull Typ
out[12]	OUT			W19		14	default (LVCMS18)	1.800		12		
out[11]	OUT					14	default (LVCMS18)	1.800		12		
out[10]	OUT					14	default (LVCMS18)	1.800		12		
out[9]	OUT					14	default (LVCMS18)	1.800		12		
out[8]	OUT					14	default (LVCMS18)	1.800		12		
out[7]	OUT					14	default (LVCMS18)	1.800		12		
out[6]	OUT					14	default (LVCMS18)	1.800		12		
out[5]	OUT					14	default (LVCMS18)	1.800		12		
out[4]	OUT					14	default (LVCMS18)	1.800		12		
out[3]	OUT					14	default (LVCMS18)	1.800		12		
out[2]	OUT					14	default (LVCMS18)	1.800		12		
out[1]	OUT					14	default (LVCMS18)	1.800		12		

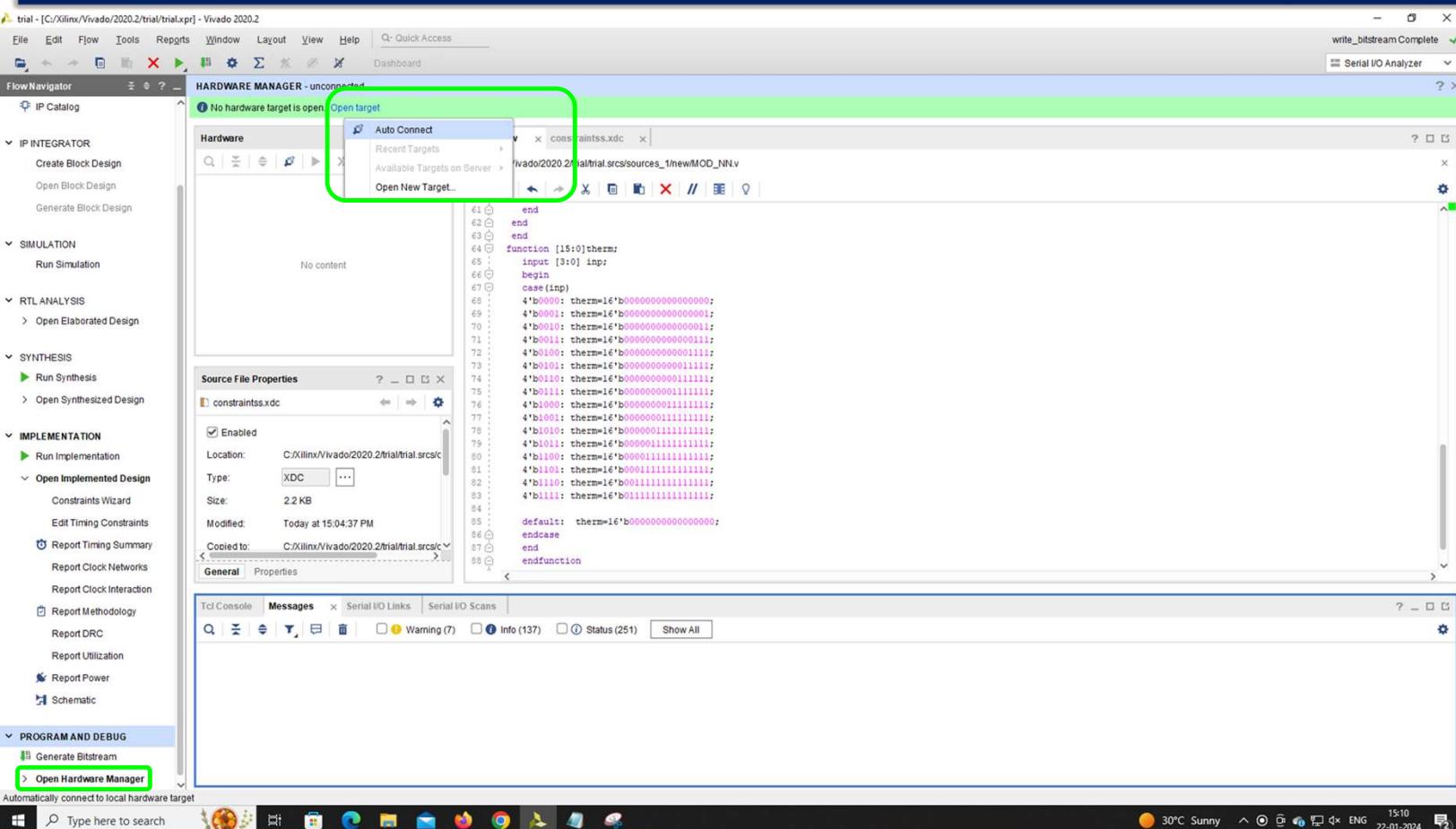


STEP 3: After the 'Constraint File' is added save and run the 'Generate Bitstream' as mentioned earlier

Once Bitstream Generation is completed a window is popped up as shown above Click on 'Open Hardware Manager'

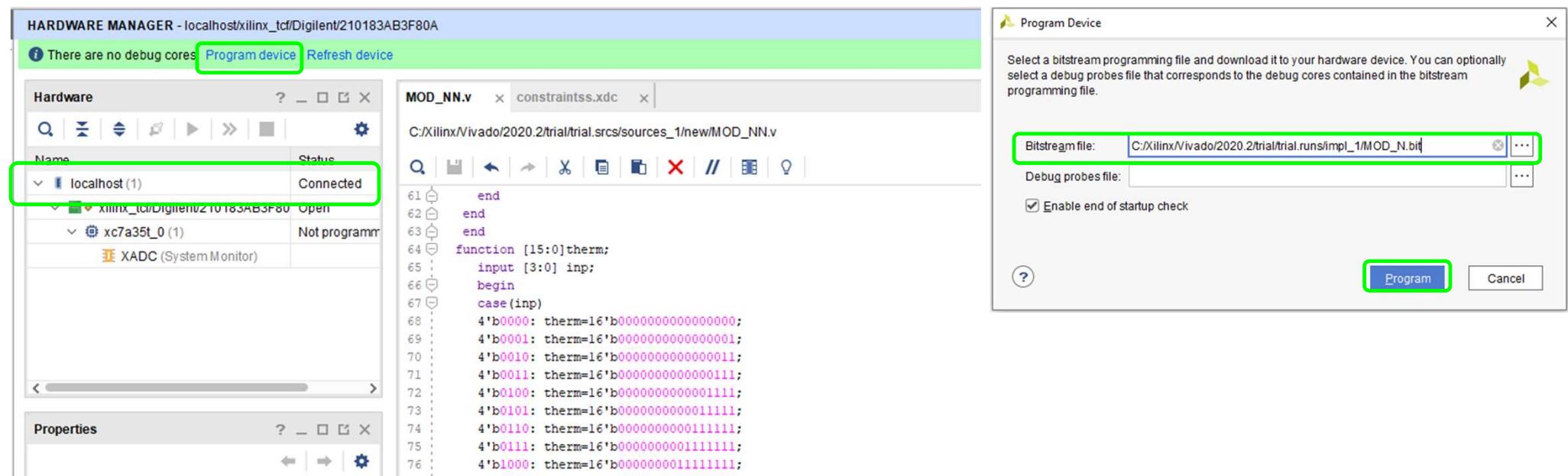


PROGRAMMING FPGA



**STEP 1: Once Bitstream is Generated
Open ‘Hardware Manager’
Select ‘Open target’
Select ‘Auto Connect’**

PROGRAMMING FPGA



**STEP 2: Once Target is Connected
Select 'Program Device'
Select '.bit File' and
Click on 'Program'**

Now your FPGA will be programmed and you can see your code working in real time



Artix-7 35T Features



- 33,280 logic cells in 5200 slices (each slice contains four 6-input LUTs and 8 flip-flops)
- 1,800 Kbits of fast block RAM
- Five clock management tiles, each with a phase-locked loop (PLL)
- 90 DSP slices
- Internal clock speeds exceeding 450MHz
- On-chip analog-to-digital converter (XADC)



Basys3 additional features



- 16 user switches
- 16 user LEDs
- 5 user pushbuttons
- 4-digit 7-segment display
- Three Pmod ports
- Pmod for XADC signals
- 12-bit VGA output
- USB-UART Bridge
- Serial Flash
- Digilent USB-JTAG port for FPGA programming and communication
- USB HID Host for mice, keyboards and memory sticks

Basys3 board

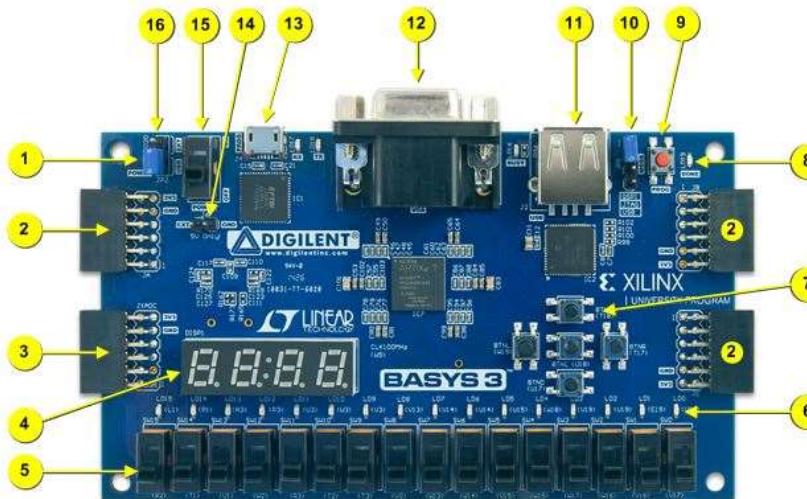


Figure 1. Basys3 board features

Callout	Component Description	Callout	Component Description
1	Power good LED	9	FPGA configuration reset button
2	Pmod connector(s)	10	Programming mode jumper
3	Analog signal Pmod connector (XADC)	11	USB host connector
4	Four digit 7-segment display	12	VGA connector
5	Slide switches (16)	13	Shared UART/JTAG USB port
6	LEDs (16)	14	External power connector
7	Pushbuttons (5)	15	Power Switch
8	FPGA programming done LED	16	Power Select Jumper

<https://digilent.com/reference/programmable-logic/basys-3/reference-manual>

Thank You

