

Running Verilog-HDL
file – tutorial
using Vivado 2017.3

LSI Design Contest 2018

The 21st

LSI 2018

Design Contest
In Okinawa



Adding zybo board package

- Download zybo board files from the link below:

- <https://reference.digilentinc.com/reference/software/vivado/board-files>

- Unarchive the file

- Copy all the board files from the folder

- ¥vivado-boards-master¥new¥board_files

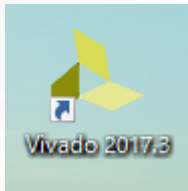
- Paste the board files to your installed vivado path directory

- C:¥Xilinx¥Vivado¥2017.3¥data¥boards¥board_files

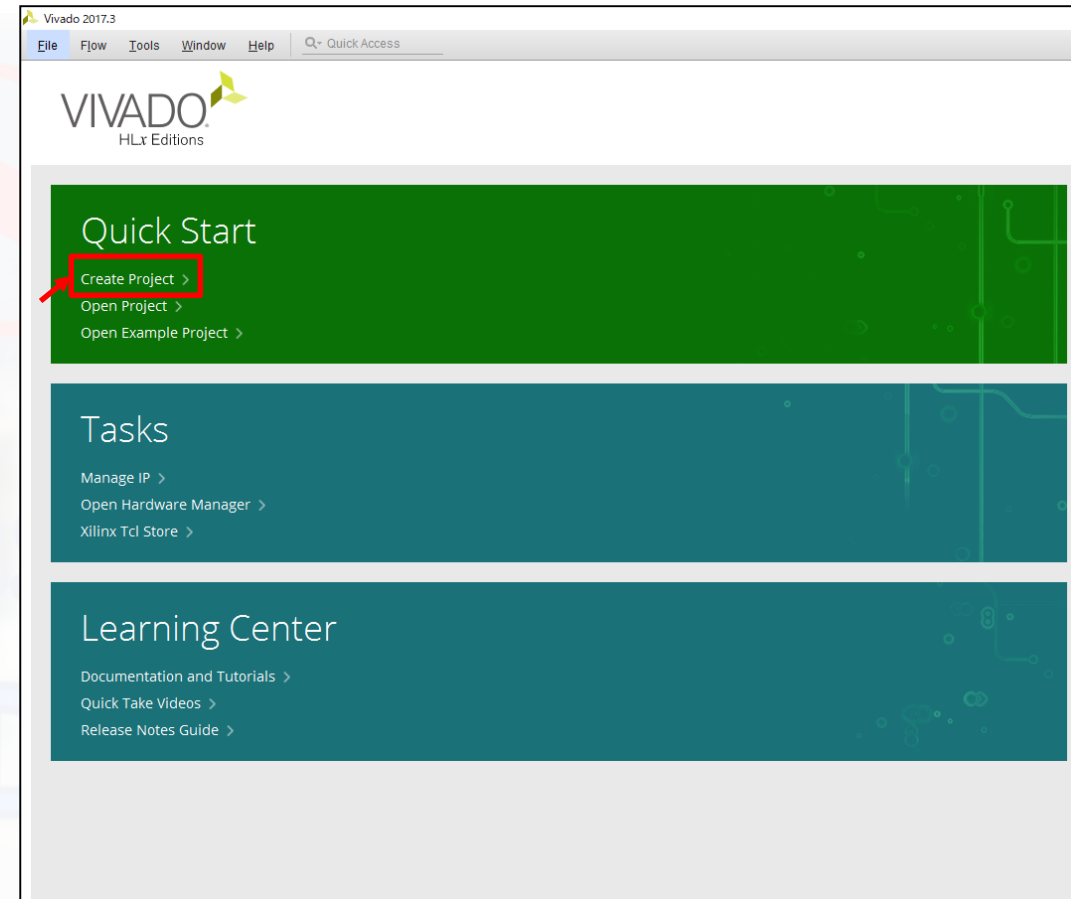
Opening Vivado

■ Open Vivado

□ Note that the version used here is 2017.3

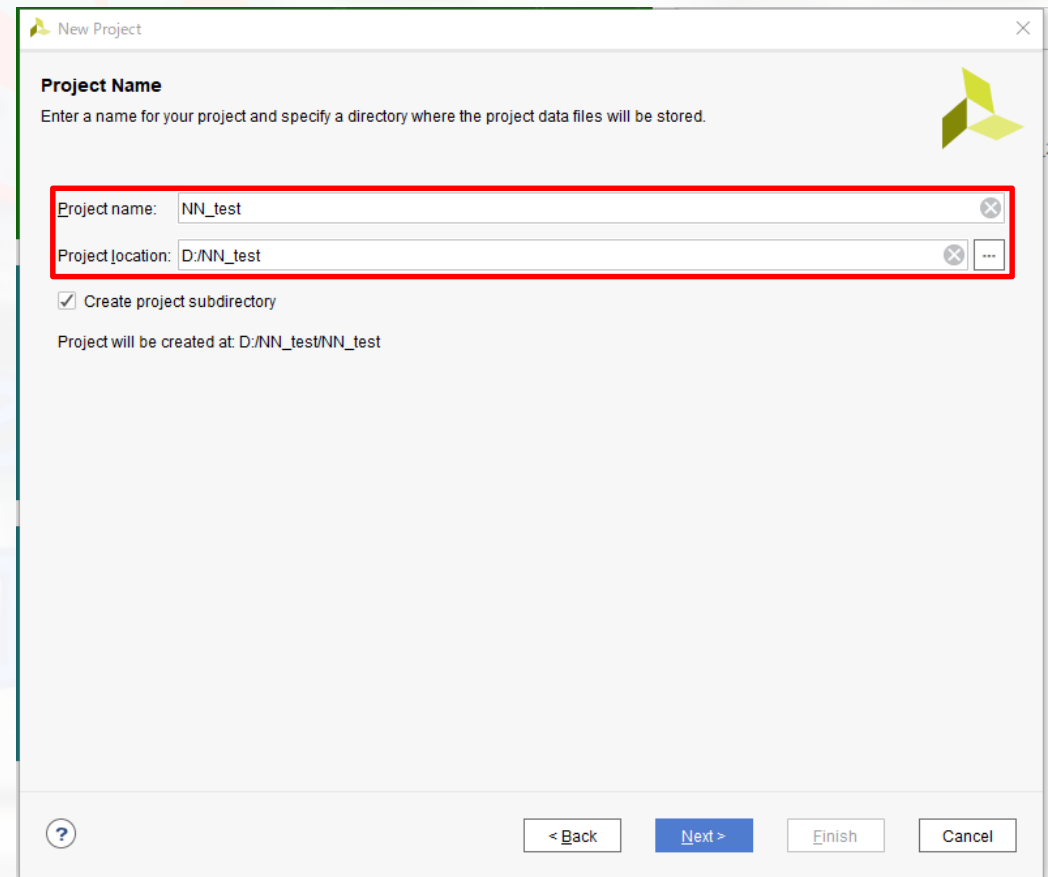


□ Next, click the 「Create Project...」 as shown in the figure.



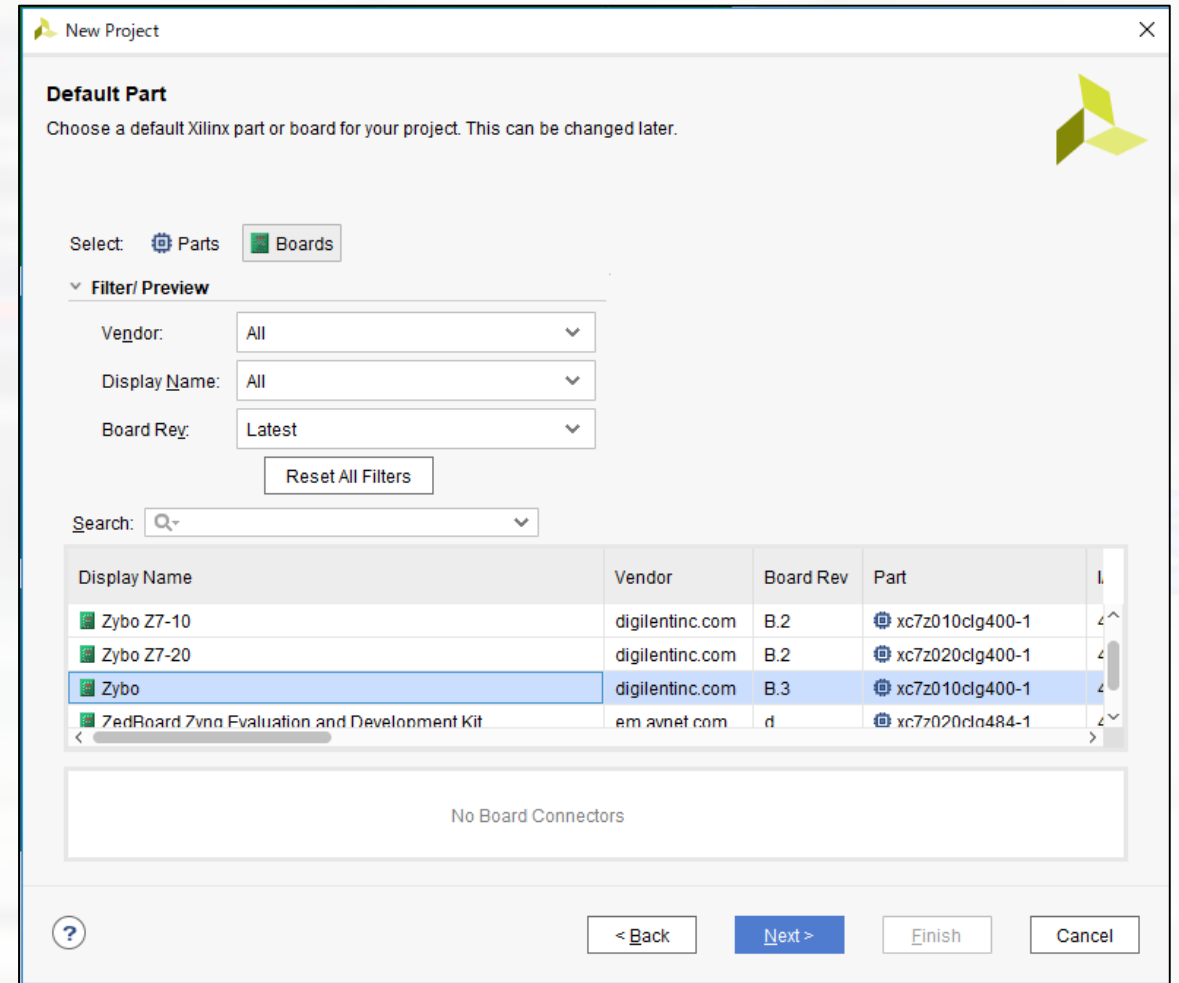
Locating project file

- Create a New Vivado project will come out, click next.
- Assign the location to your selected folder
- Enter the name
 - For example : NN_test
- Click, next
- Project type terminal will come out
 - click next
- Add Sources terminal will come out
 - click next
- Add Constraints terminal will come out
 - click next
- Default part terminal will come out
 - select your specified boards
 - and finish.



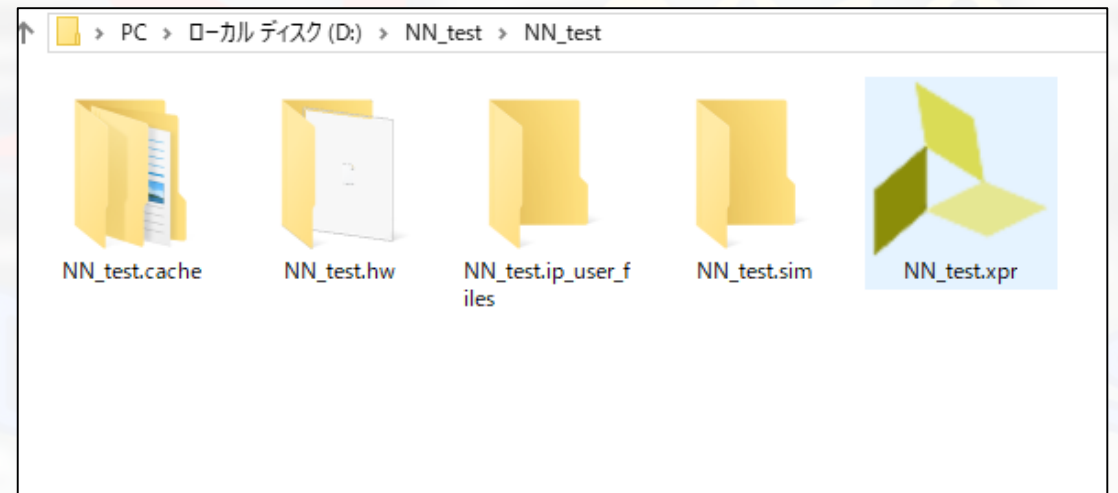
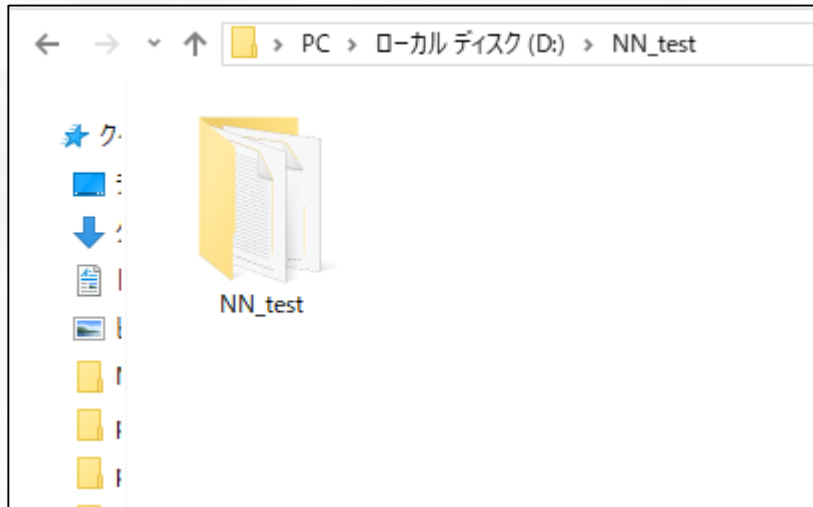
Specifying board

- Default part terminal will come out
 - select your specified boards
 - click next
- Click finish.



Locating project file

- A new folder will be created in your selected location



Download Verilog file

- Download the zip file from LSI design contest HP.

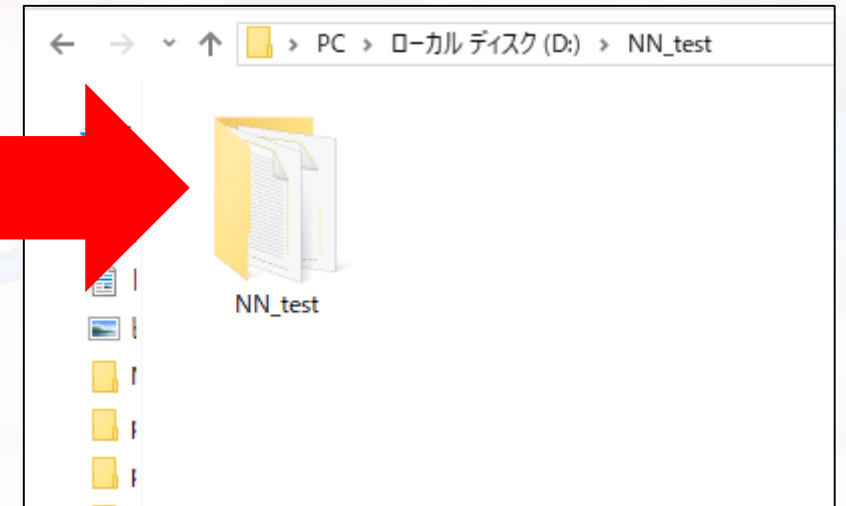
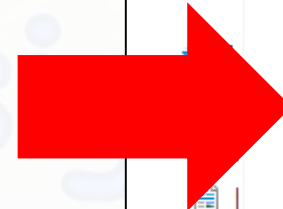
- Link : http://www.lsi-contest.com/shiyou_4e.html

- (Verilog file for simulation)

- Extract the file

- There will be around 45 Verilog file

- Move all the Verilog file into NN_test folder



Add source file

■ Click Add source

The image shows the Vivado 2017.3 interface. On the left, the 'PROJECT MANAGER - NN_test' window is open, showing a tree view of project components. The 'Add Sources' button is circled in red. A red circle with the number '1' and the text 'Click this button' points to it. On the right, the 'Add Sources' dialog box is open, showing three radio button options: 'Add or create constraints', 'Add or create design sources' (which is selected), and 'Add or create simulation sources'. At the bottom of the dialog, the 'Next >' button is circled in red. A red circle with the number '2' and the text 'Click Next >' points to it.

① Click this button

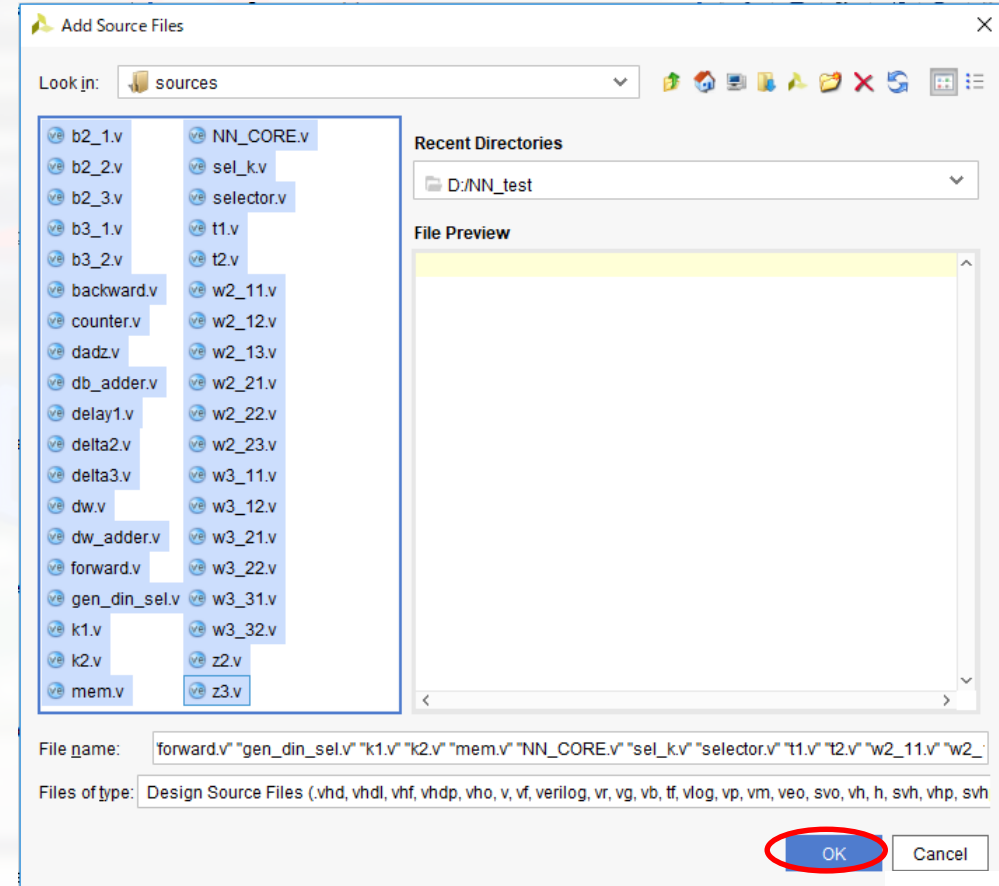
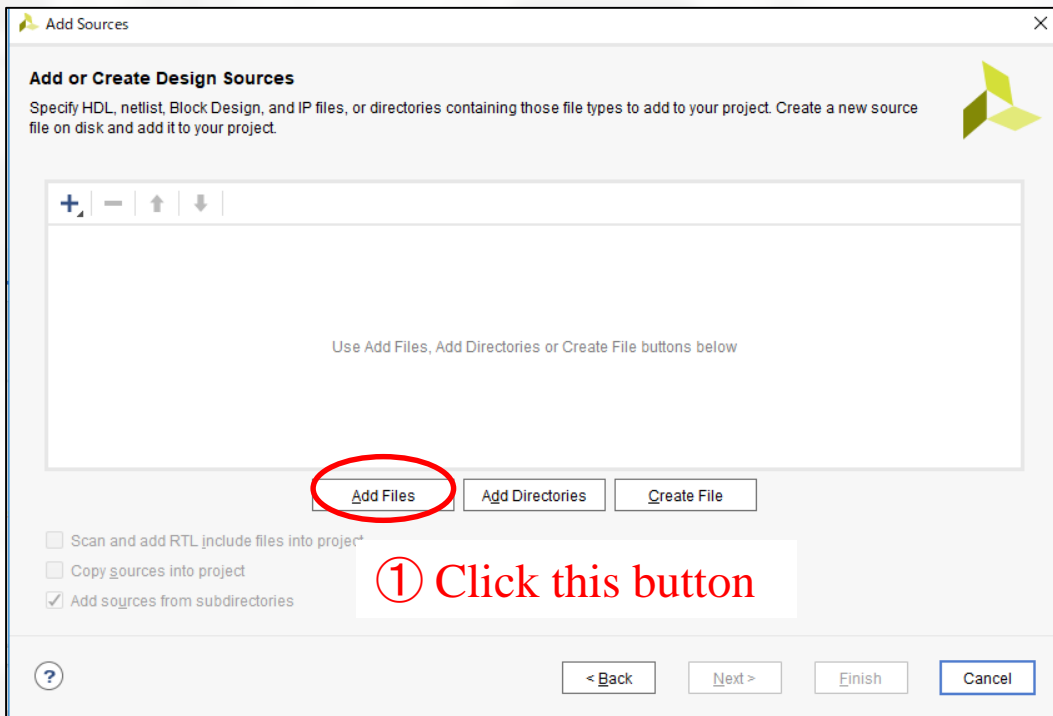
② Click Next >

Add source file

■ Click Add file

□ Go to folder 'sources' directory

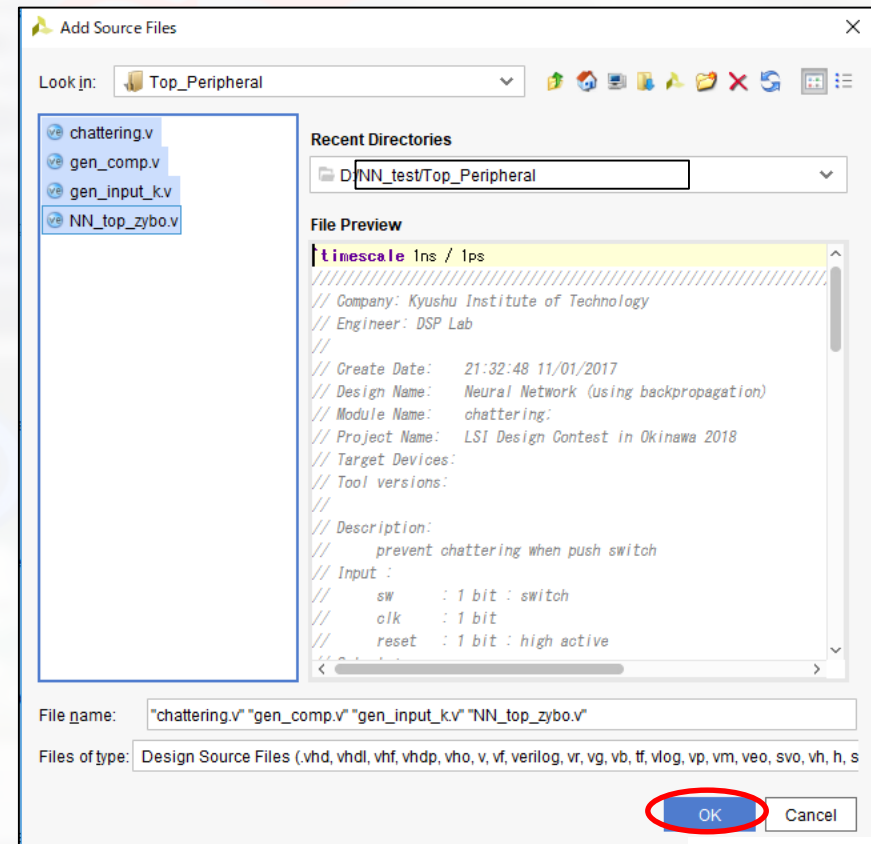
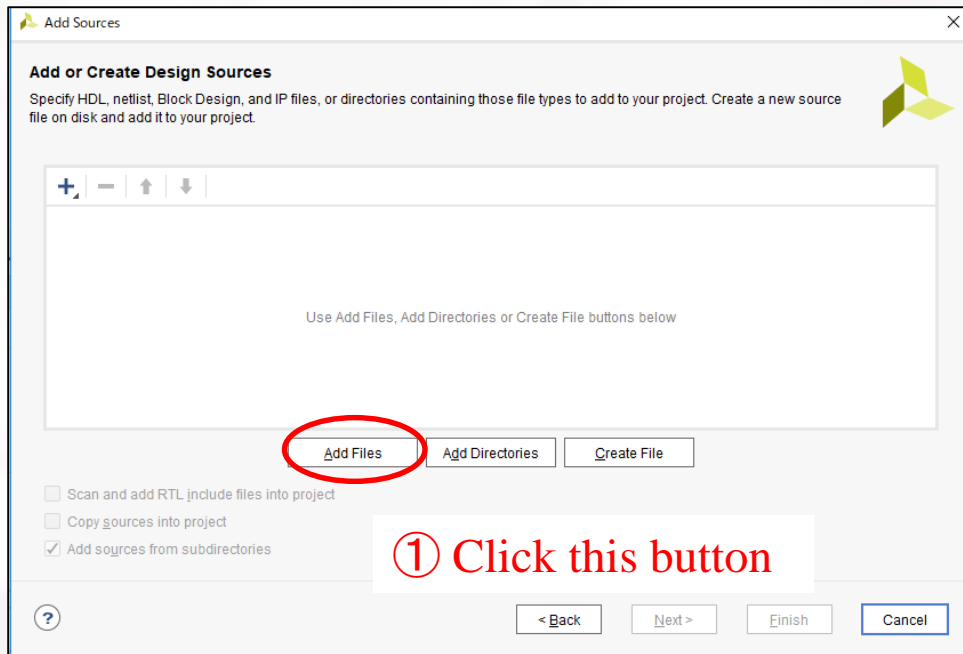
□ Select all the files



Add source file

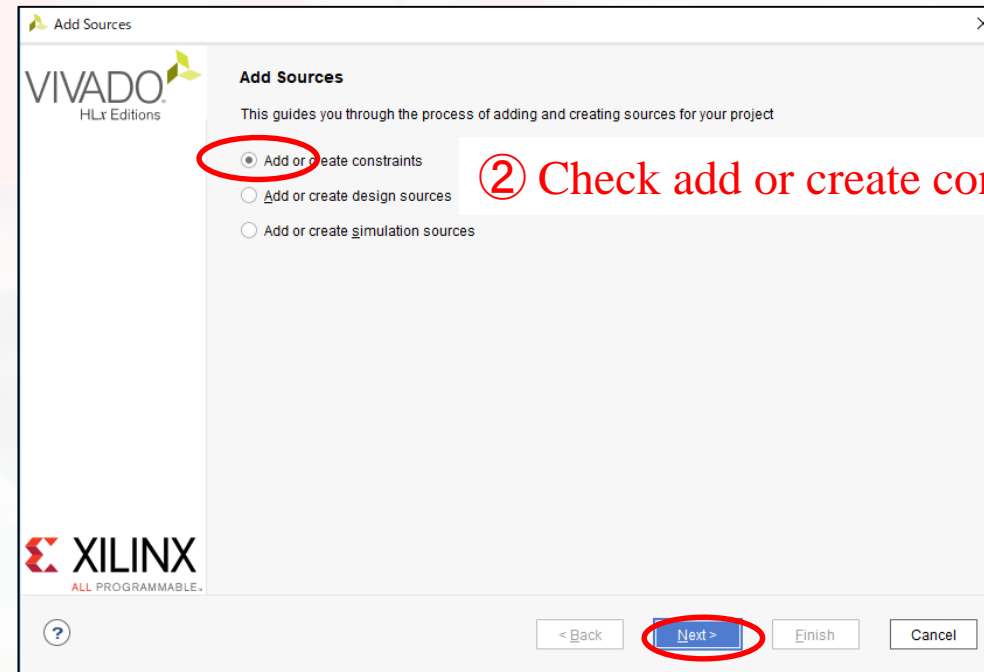
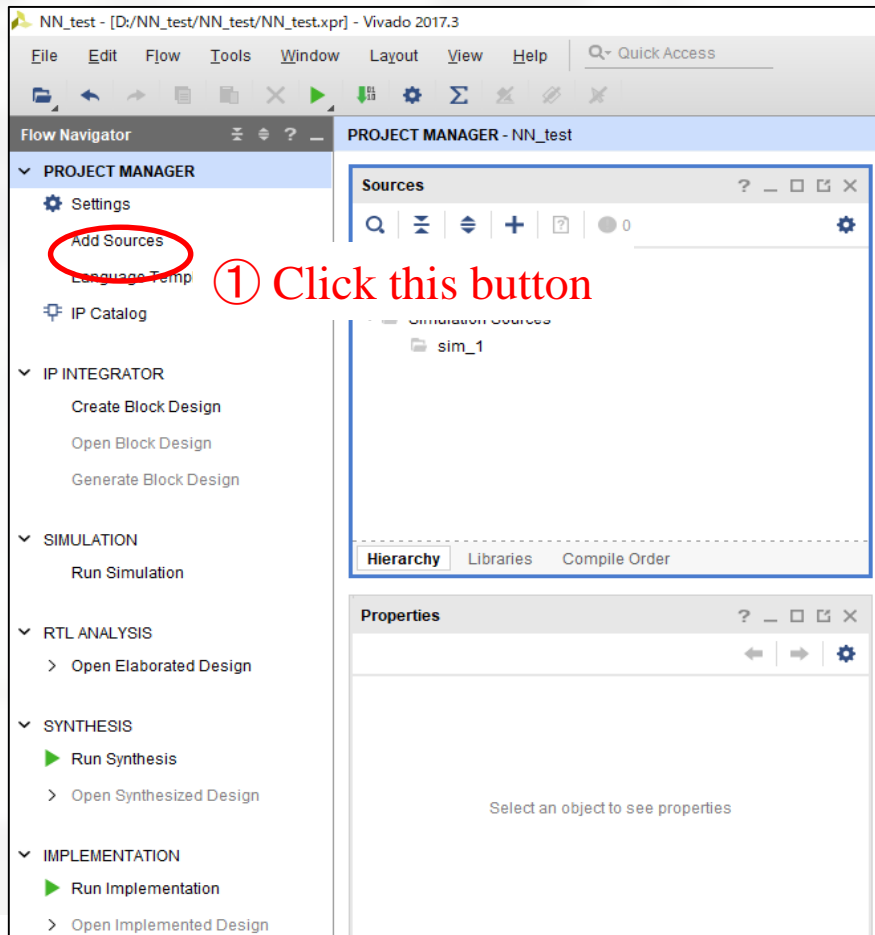
■ Click Add file

- Go to folder 'Top_Peripheral' directory
- Select all the files



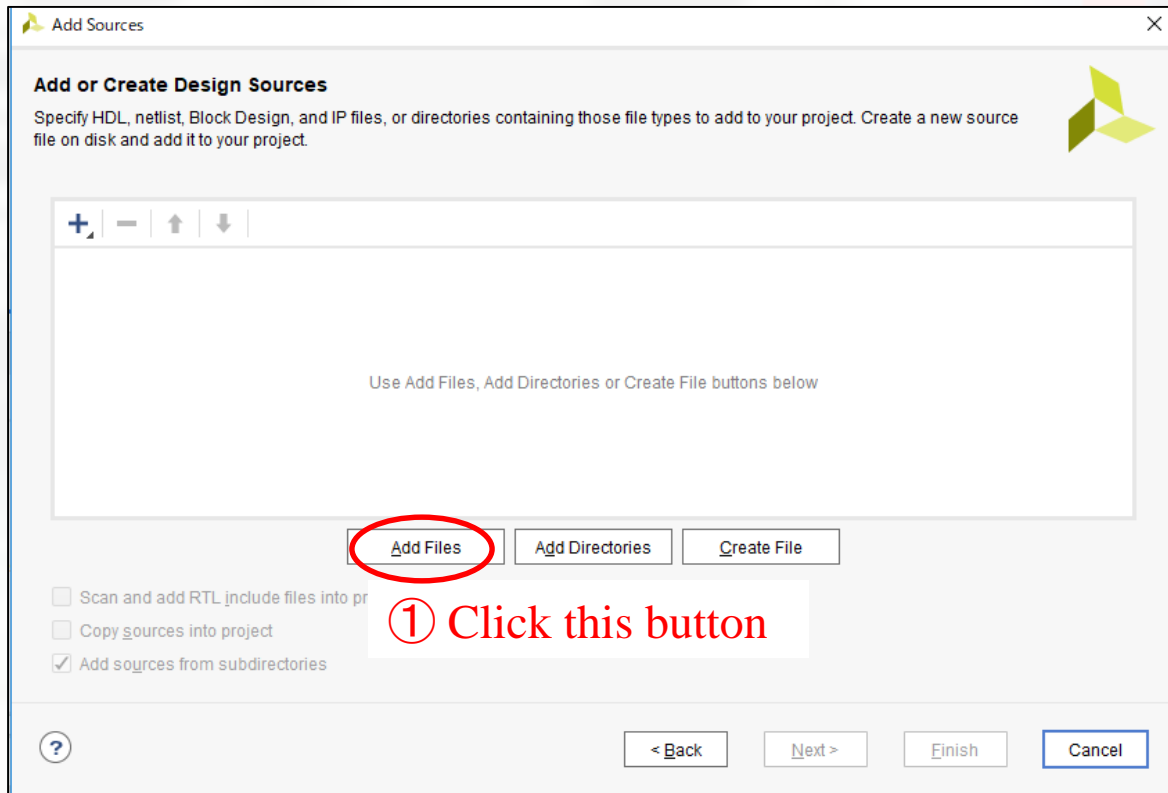
Add constraints file

■ Click Add source



Add source file

■ Click Add files



- ② Select design Constraint files (NN_top_zybo.xdc)
- ③ Click OK, then Finish

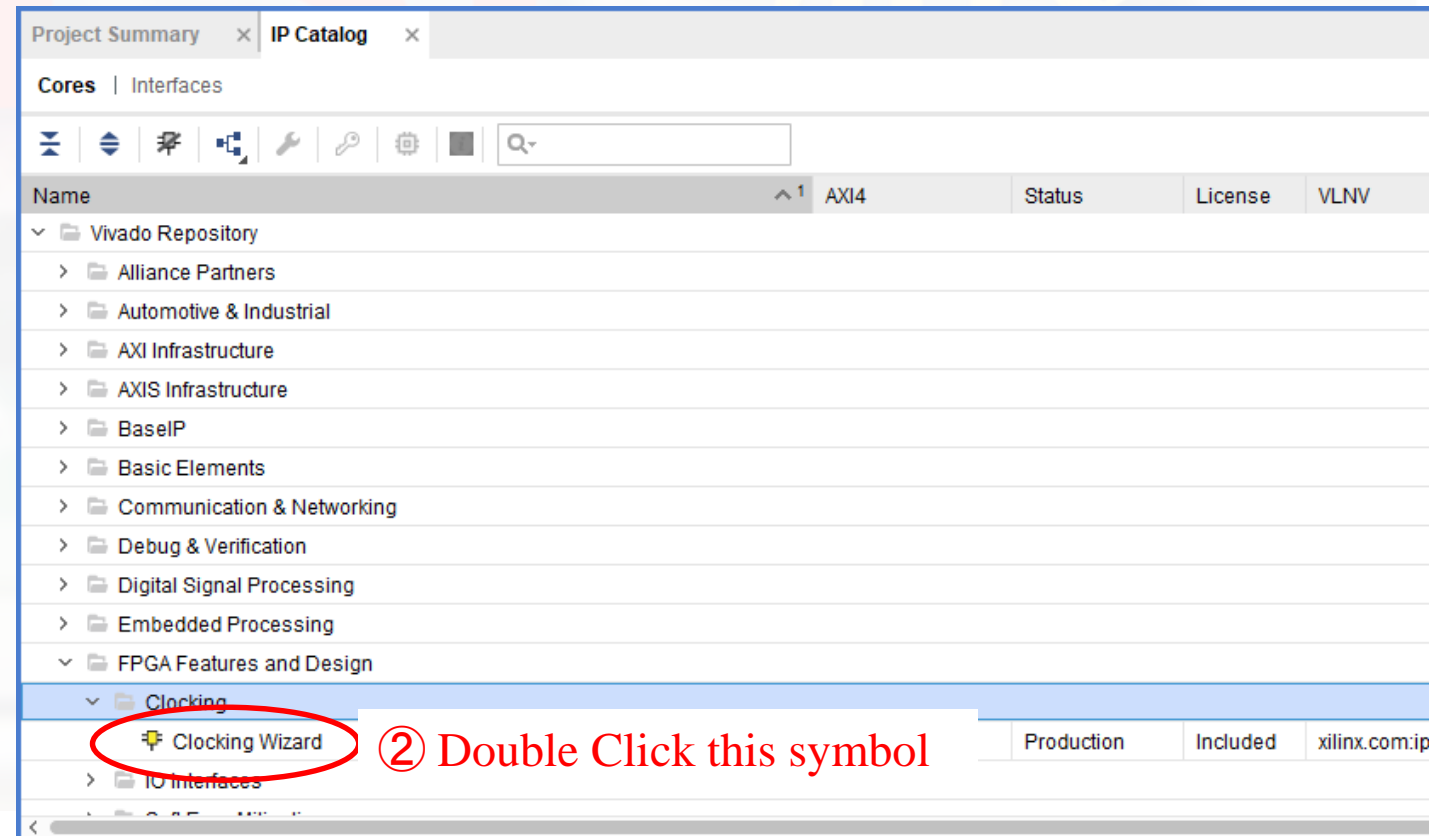
① Click this button

Add clocking wizard

- Click the IP Catalog below PROJECT MANAGER
- In the IP Catalog window, expand FPGA Features and Design
 - Then expand Clocking
 - Double click the clocking wizard



① Click this button



② Double Click this symbol

Add clocking wizard

- Customize IP window will open
- Open the Clocking Options tab
 - At the Input Clock Information, for Primary Input Clock, change the Input Frequency to 125 MHz

Clocking Wizard (5.4)

Documentation IP Location Switch to Defaults

Component Name: clk_wiz_0

Board: Clocking Options Output Clocks Port Renaming MMCM Settings Summary

Frequency Synthesis Minimize Power Balanced

Phase Alignment Spread Spectrum Minimize Output Jitter

Dynamic Reconfig Dynamic Phase Shift Maximize Input Jitter filtering

Safe Clock Startup

Dynamic Reconfig Interface Options

AXI4Lite DRP Phase Duty Cycle Config Write DRP registers

Input Clock Information

Input Clock	Port Name	Input Frequency (MHz)	Jitter Optio
<input checked="" type="checkbox"/> Primary	clk_in1	125	UI
<input type="checkbox"/> Secondary	clk_in2	100.000	82 192 - 164 384

① Change to 125

Add clocking wizard

- Open the Output Clocks tab
 - At the Output Clock, change the Requested Output Freq to 25 MHz
- Click OK

Clocking Wizard (5.4)

Documentation IP Location Switch to Defaults

IP Symbol Resource

Show disabled ports

Component Name clk_wiz_0

Board Clocking Options **Output Clocks** Port Renaming MMCM Settings Summary

The phase is calculated relative to the active input clock.

Output Clock	Port Name	Output Freq (MHz)		Phase (degrees)		Duty Cycle (%)	
		Requested	Actual	Requested	Actual	Requested	Actual
<input checked="" type="checkbox"/> clk_out1	clk_out1	25	25.000	0.000	0.000	50.000	50.0
<input type="checkbox"/> clk_out2	clk_out2	100.000				50.000	N/A
<input type="checkbox"/> clk_out3	clk_out3	100.000				50.000	N/A
<input type="checkbox"/> clk_out4	clk_out4	100.000	N/A	0.000	N/A	50.000	N/A
<input type="checkbox"/> clk_out5	clk_out5	100.000	N/A	0.000	N/A	50.000	N/A
<input type="checkbox"/> clk_out6	clk_out6	100.000	N/A	0.000	N/A	50.000	N/A
<input type="checkbox"/> clk_out7	clk_out7	100.000	N/A	0.000	N/A	50.000	N/A

USE CLOCK SEQUENCING

Clocking Feedback

Output Clock	Sequence Number
clk_out1	1
clk_out2	1
clk_out3	1
clk_out4	1
clk_out5	1

Source

Automatic Control On-Chip
 Automatic Control Off-Chip
 User-Controlled On-Chip
 User-Controlled Off-Chip

Signaling

Single-ended
 Differential

OK Cancel

Generate Bitstream

■ In Flow Navigator, select **PROGRAM AND DEBUG**

□ Click Generate Bitstream

■ When you program device, you should turn on board.

Flow Navigator

- PROJECT MANAGER
 - Settings
 - Add Sources
 - 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
- PROGRAM AND DEBUG**
 - Generate Bitstream**
 - Open Hardware Manager

PROJECT MANAGER - NN_test

Sources

- Design Sources (1)
 - NN_top (NN_top_zybo.v) (5)
 - CLK_WIZ_0 : clk_wiz_0 (clk_wiz_0.xci)
 - GEN_INPUT_K : gen_input_k (gen_input_k.v)
 - CHATTERING : chattering (chattering.v)
 - DUT_NN_CORE : NN_CORE (NN_CORE.v) (25)
 - GEN_COMP : gen_comp (gen_comp.v)
- Constraints (1)
- Simulation Sources (1)

Source File Properties

clk_wiz_0.xci

- Enabled
- Location: d:/project_VIVADO/NN_test/NN_test.srcs/source:
- Type: IP
- Part: xc7z010clg400-1
- Size: 86.6 KB
- Modified: Today at 10:49:45 AM
- Copied to: d:/project_VIVADO/NN_test/NN_test.srcs/source:
- Read-only: No

Launch Runs

Launch the selected synthesis or implementation runs.

Launch directory: <Default Launch Directory>

Options

- Launch runs on local host: Number of jobs: 2
- Generate scripts only
- Don't show this dialog again

OK Cancel

② Click OK

Bitstream Generation Completed

Bitstream Generation successfully completed.

Next

- Open Implemented Design
- View Reports
- Generate Memory Configuration File
- Don't show this dialog again

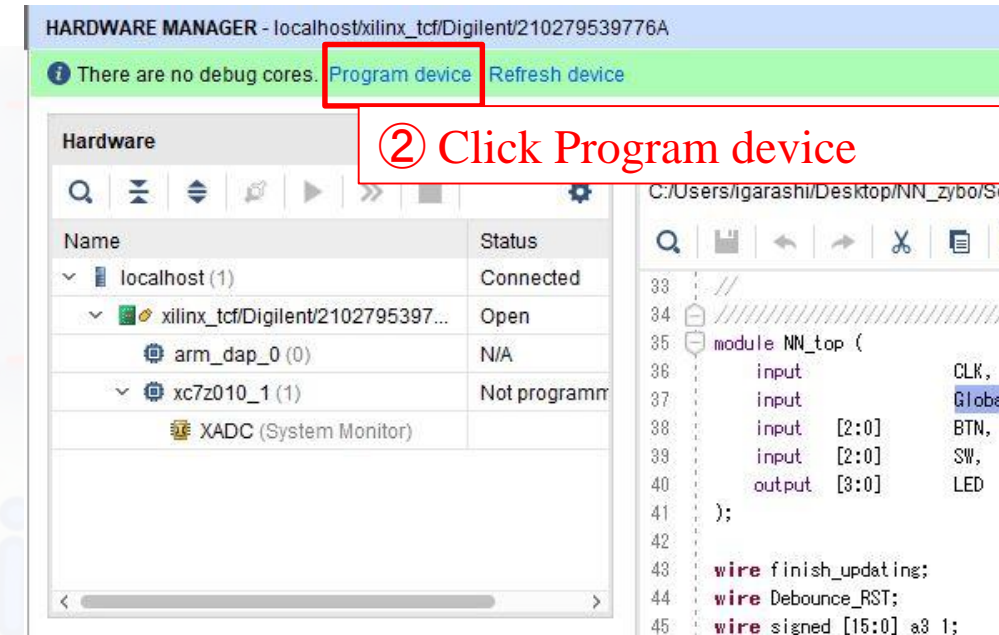
OK Cancel

③ Click Cancel

① Click run Generate Bitstream

Program the device

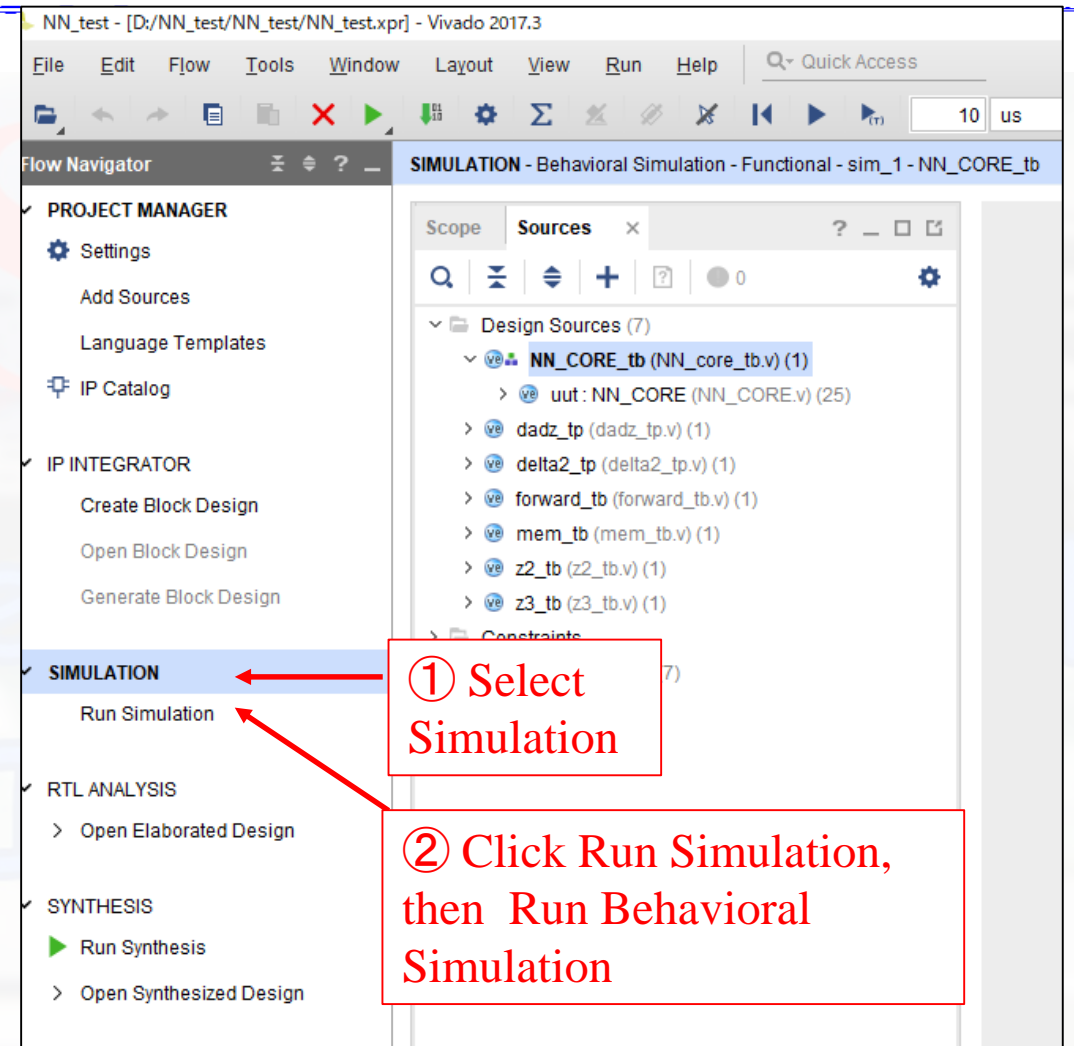
- Connect your board to your PC and turn on your board
- Expand the Open Hardware Manager below the Generate Bitstream
- Click Open Target and click auto connect



① Click Auto Connect

Simulate the project

- Select Simulation
- Click Run Simulation
 - Click Run Behavioral Simulation





Vivado Simulator

- Vivado simulator will open
- Choose forward block in Scope
 - Choose signal b2_1 until w3_32
 - Drag the signal into simulator
- Choose forward > selector block in instance and process name
 - Choose signal enable update
 - Drag the signal into simulator
- See the figure in the next slide...

Vivado Simulator

The screenshot displays the Vivado Simulator interface for a behavioral simulation. The main window is titled "SIMULATION - Behavioral Simulation - Functional - sim_1 - NN_CORE_tb".

Scope Panel: Shows a tree view of the design hierarchy. The "FORWARD" block is highlighted with a red circle and a red arrow pointing to it. A red callout box with the text "① open forward block" is positioned over this area.

Objects Panel: A table listing various objects in the design. The "Value" column shows the current value of each object. A red circle highlights the "FORWARD" object in this panel, with a red arrow pointing to it.

Variable Declaration Table: A table with two columns: "Name" and "Value".

Name	Value
clk	1
res	0
update_coeff	1
input_k_[15:0]	2000
input_k_[15:0]	2000
a3_1[15:0]	0191
a3_2[15:0]	0173
finish_...dating	0
STEP[31:0]	00000064

Waveform Viewer: Shows a timing diagram with a time axis from 999,995 ps to 1,000,000 ps. A red arrow points from the "STEP[31:0]" signal in the Objects panel to a signal in the waveform viewer. A red callout box with the text "② drag this signal into simulator" is positioned over this area.

Vivado Simulator

■ Click the restart button

The screenshot shows the Vivado Simulator interface. The top menu bar includes 'Layout', 'View', 'Run', 'Help', and 'Quick Access'. The simulation controls bar shows a red circle around the restart button (a square with a left-pointing arrow). A red box with the text '1 click restart button' is overlaid on the interface. The 'Scope' window on the left lists various components and their values. The 'Sources' window on the right shows a list of signals and their values. The 'Untitled 15*' window on the right shows a waveform viewer with a table of signal values and a corresponding waveform plot.

Name	Value	Data T...
> w3_11[15:0]	XXXX	Array
> w3_12[15:0]	XXXX	Array
> w3_21[15:0]	XXXX	Array
> w3_22[15:0]	XXXX	Array
> w3_31[15:0]	XXXX	Array
> w3_32[15:0]	XXXX	Array
> a2_1[15:0]	XXXX	Array
> a2_2[15:0]	XXXX	Array
> a2_3[15:0]	XXXX	Array
> a3_1[15:0]	XXXX	Array
> a3_2[15:0]	XXXX	Array
> k1[15:0]	XXXX	Array
> k2[15:0]	XXXX	Array
> t1[15:0]	XXXX	Array
> t2[15:0]	XXXX	Array
finish_upd...	X	Logic
select_upd...	X	Logic
> out[3:0]	X	Array
> z2_1[7:0]	XX	Array
> z2_2[7:0]	XX	Array
> z2_3[7:0]	XX	Array
> z3_1[7:0]	XX	Array
> z3_2[7:0]	XX	Array
> b2_1[15:0]	XXXX	Array
> b2_2[15:0]	XXXX	Array
> b2_3[15:0]	XXXX	Array
> b3_1[15:0]	XXXX	Array
> b3_2[15:0]	XXXX	Array
> b3_3[15:0]	XXXX	Array

Name	Value
> b3_1[15:0]	XXXX
> b3_2[15:0]	XXXX
> w2_11[15:0]	XXXX
> w2_12[15:0]	XXXX
> w2_13[15:0]	XXXX
> w2_21[15:0]	XXXX
> w2_22[15:0]	XXXX
> w2_23[15:0]	XXXX
> w3_11[15:0]	XXXX
> w3_12[15:0]	XXXX
> w3_21[15:0]	XXXX
> w3_22[15:0]	XXXX
> w3_31[15:0]	XXXX
> w3_32[15:0]	XXXX
clk	X
res	X
update_coeff	X
> input_k_[15:0]	XXXX
> input_k_[15:0]	XXXX
> a3_1[15:0]	XXXX
> a3_2[15:0]	XXXX
finish_...daring	X
> STEP[31:0]	00000064

Vivado Simulator

■ Click the Run All button

The screenshot shows the Vivado Simulator interface. The top toolbar contains a red circle around the Run All button (a play icon). A red box with the text "1 click Run All button" is overlaid on the Run All button. The main window displays a behavioral simulation for a design named "NN_CORE_tb". The left pane shows a hierarchical tree of components, including "NN_CORE_tb", "uut", "GEN_DIN...", and "FORWARD". The middle pane shows a table of signals and their values. The right pane shows a waveform viewer with a time axis from 0 ms to 12 ms and various signal traces.

Name	Value	Data T...
> a3_1[15:0]	0016	Array
> a3_2[15:0]	027d	Array
↳ finish_upd...	1	Logic

Name	Value
> b3_1[15:0]	f044
> b3_2[15:0]	fded
> w2_11[15:0]	0584
> w2_12[15:0]	0223
> w2_13[15:0]	03b7
> w2_21[15:0]	0584
> w2_22[15:0]	02f0
> w2_23[15:0]	01a9
> w3_11[15:0]	197b
> w3_12[15:0]	e467
> w3_21[15:0]	fed0
> w3_22[15:0]	06ec
> w3_31[15:0]	039b
> w3_32[15:0]	0913
↳ clk	0
↳ res	0
↳ update_coeff	1
> input_k_[15:0]	2000
> input_k_[15:0]	2000
> a3_1[15:0]	0016
> a3_2[15:0]	027d
↳ finish_...datinc	1
> STEP[31:0]	00000064



Conclusion

- New weight and bias will be updated every time the enable update signal is active.