

Laboratory 1 (100 Points)

Assignment 1 (40 Points)

Familiarization and Implementation of AND Gate by using schematic design entry and gate level model using Verilog.

Rule: If both the inputs are 1, then output will be 1, otherwise output will be 0. (If any input is 0 then output will be 0, otherwise output is 1)

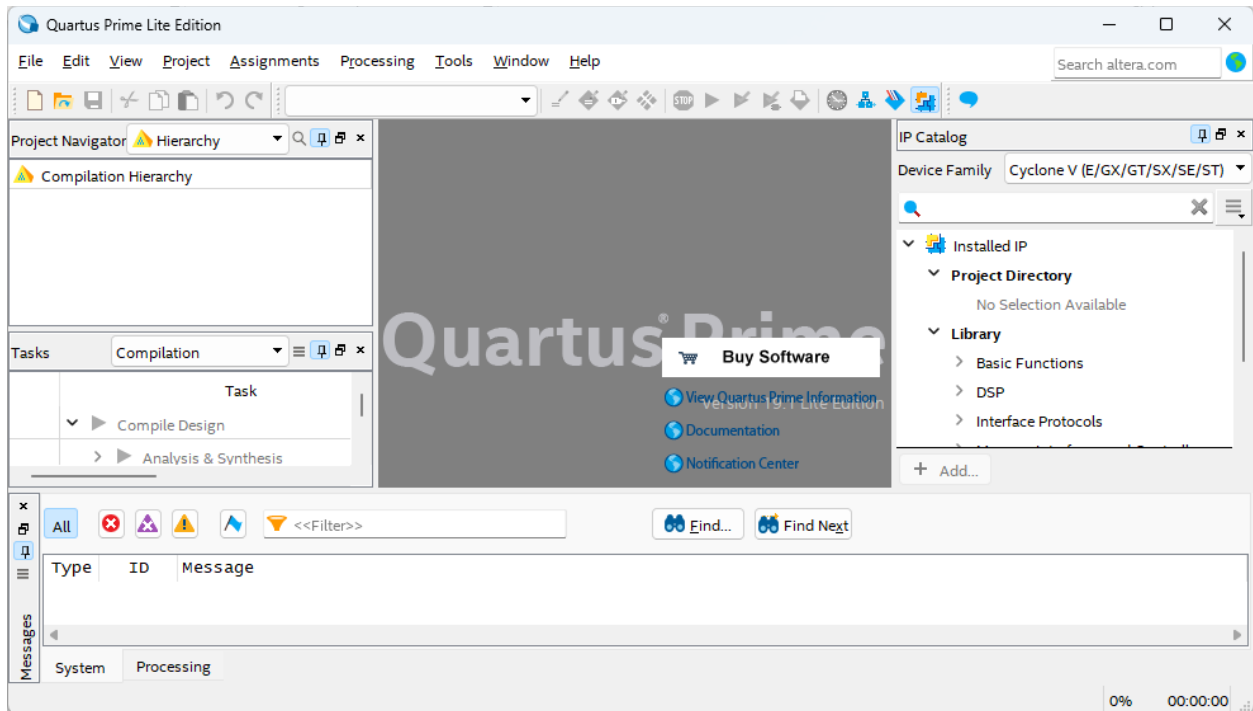
1. Boolean/Logic Expression: $Z = X \cdot Y$

2. Truth Table:

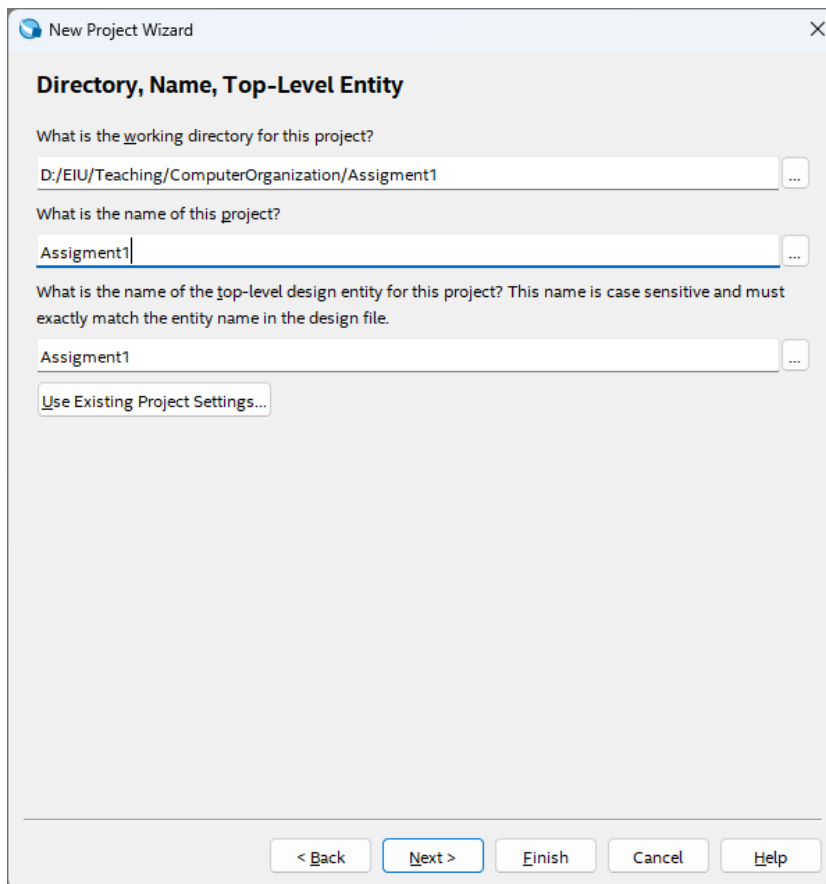
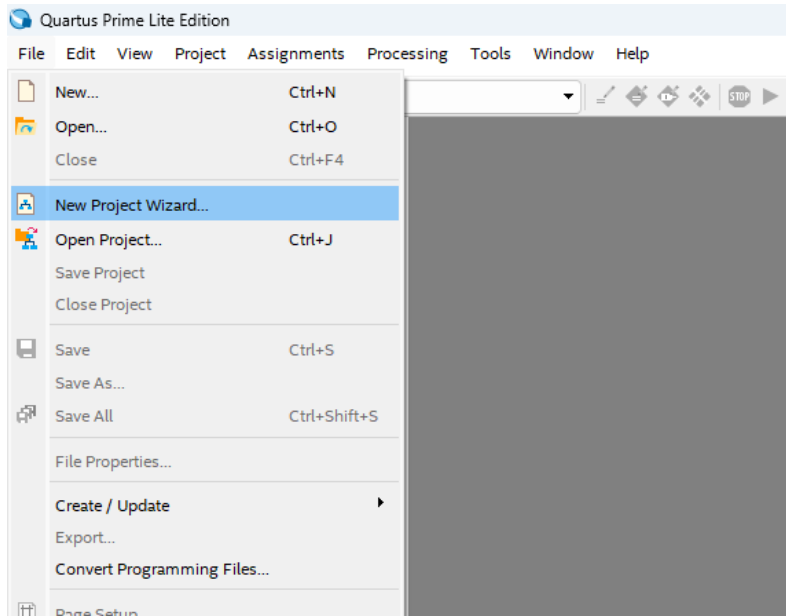
X	Y	Z = X and Y
0	0	0
0	1	0
1	0	0
1	1	1

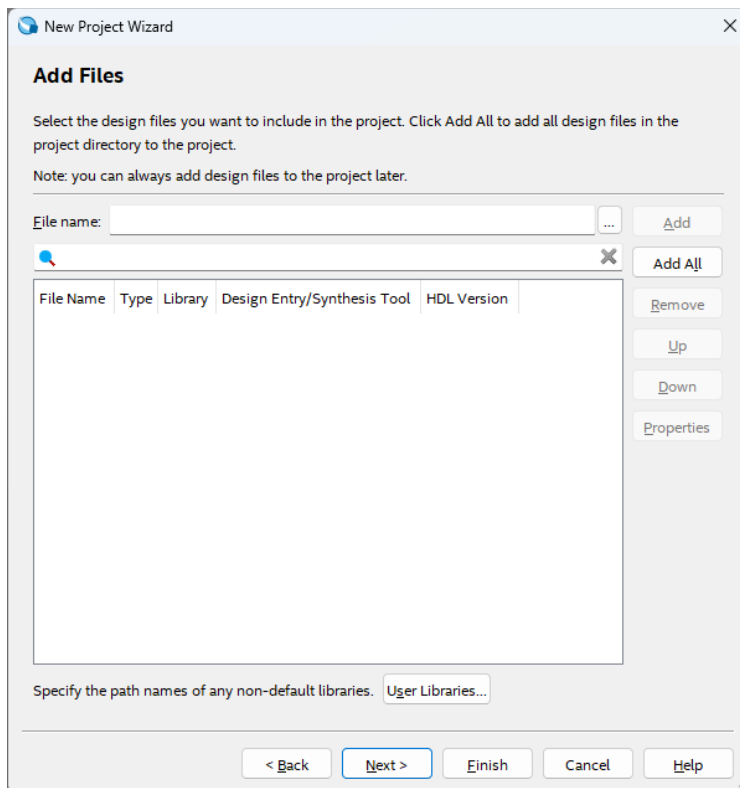
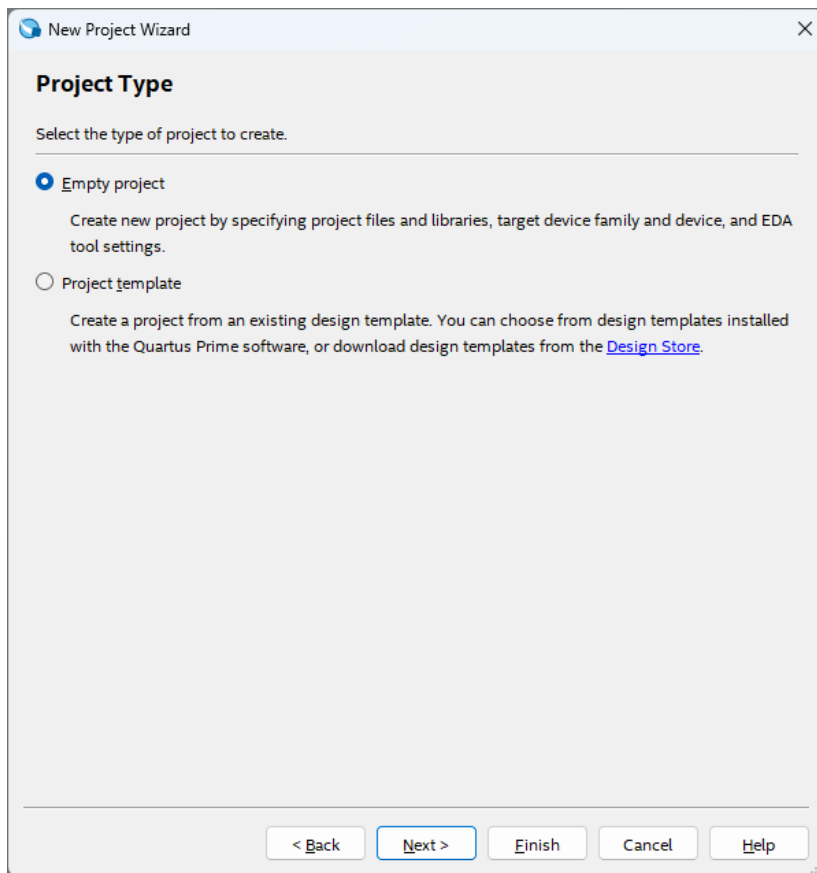
3. Using schematic design entry:

a. Run Quartus Prime Lite program



b. Create a new project named *Assignment1*:





New Project Wizard

Family, Device & Board Settings

Device

Board

Select the family and device you want to target for compilation.

You can install additional device support with the Install Devices command on the Tools menu.

To determine the version of the Quartus Prime software in which your target device is supported, refer to the [Device Support List](#) webpage.

Device family

Family: MAX 10 (DA/DF/DC/SA/SC)

Device: All

Target device

☐ Auto device selected by the Fitter

☒ Specific device selected in 'Available devices' list

☐ Other: n/a

Show in 'Available devices' list

Package: Any

Pin count: Any

Core speed grade: Any

Name filter:

☒ Show advanced devices

Available devices:

Name	Core Voltage	LEs	Total I/Os	GPIOs	Memory Bits	Embedded multiplier
10M02DCU324I7G	1.2V	2304	160	160	110592	32
10M02DCV36C7G	1.2V	2304	27	27	110592	32
10M02DCV36C8G	1.2V	2304	27	27	110592	32
10M02DCV36I7G	1.2V	2304	27	27	110592	32

< Back

Next >

Finish

Cancel

Help

New Project Wizard

EDA Tool Settings

Specify the other EDA tools used with the Quartus Prime software to develop your project.

EDA tools:

Tool Type	Tool Name	Format(s)	Run Tool Automatically
Design Entry/Synthesis	<None>	<None>	<input type="checkbox"/> Run this tool automatically to synthesize the current design
Simulation	<None>	<None>	<input type="checkbox"/> Run gate-level simulation automatically after compilation
Board-Level	Timing	<None>	
	Symbol	<None>	
	Signal Integrity	<None>	
	Boundary Scan	<None>	

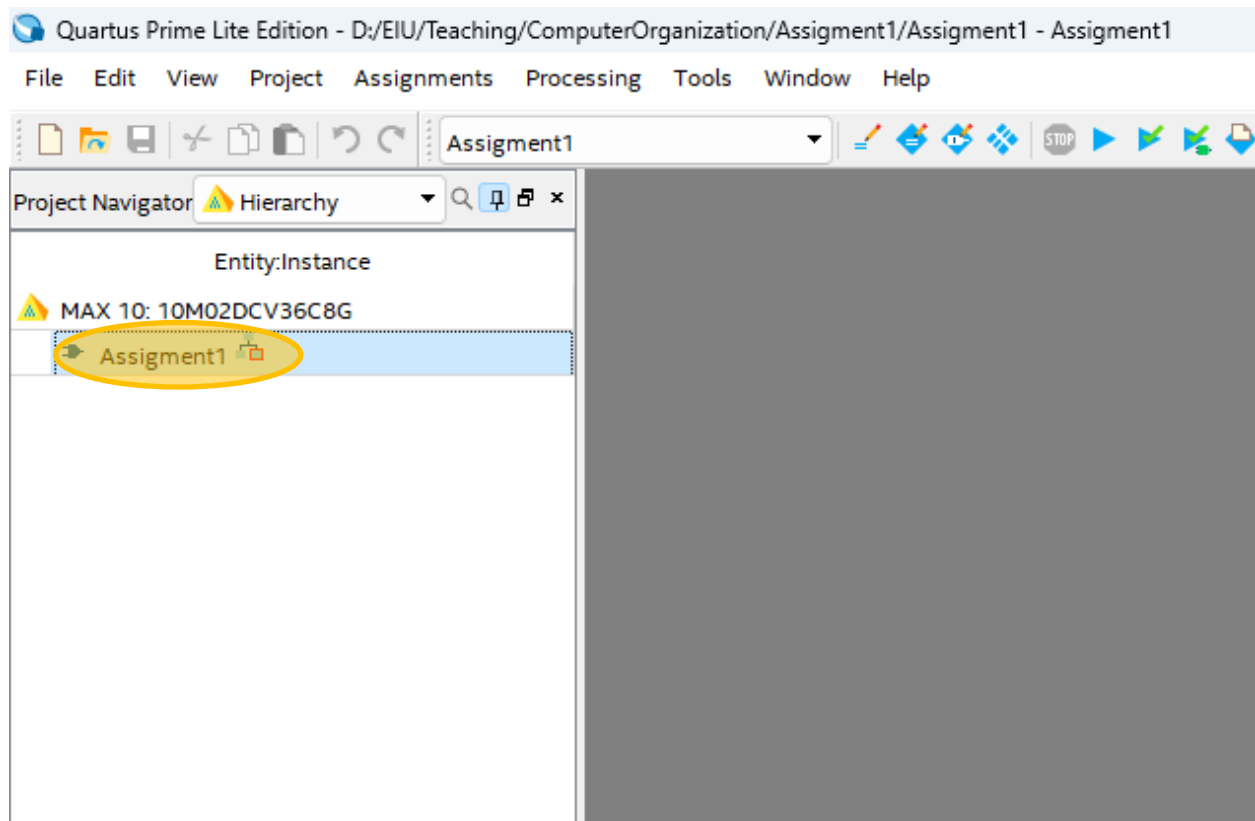
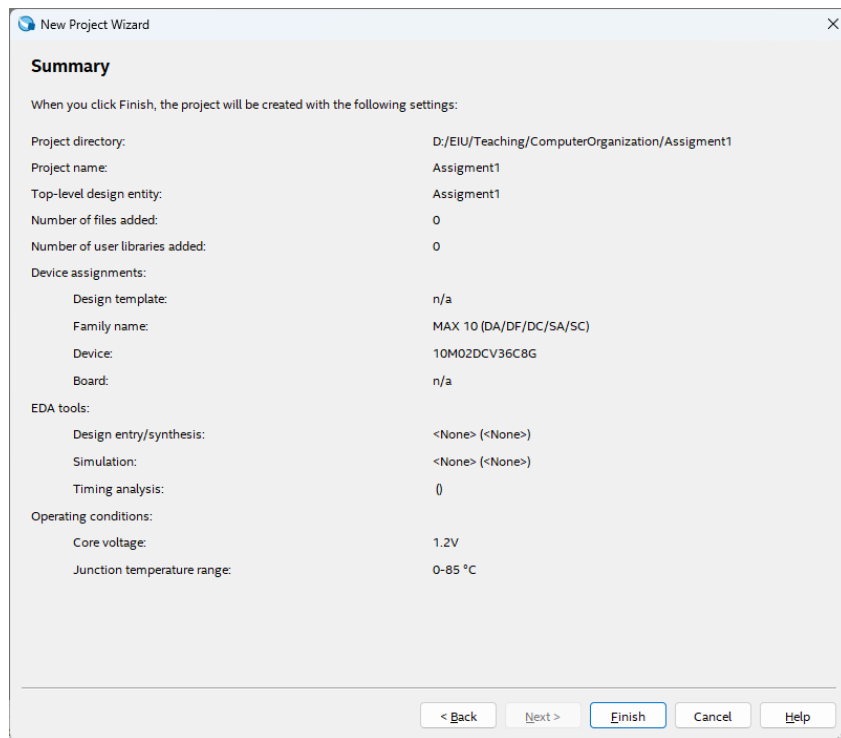
< Back

Next >

Finish

Cancel

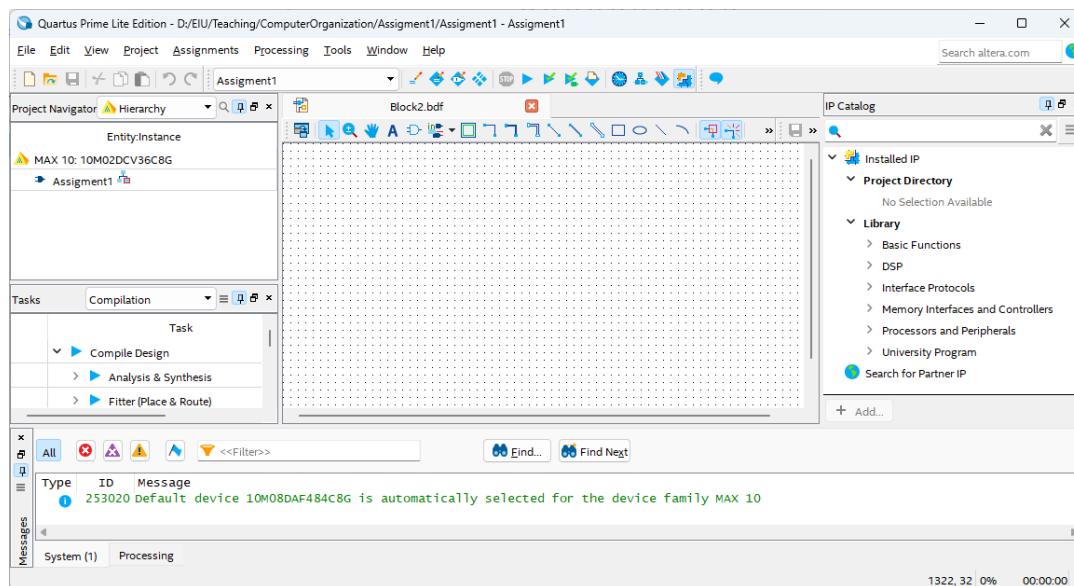
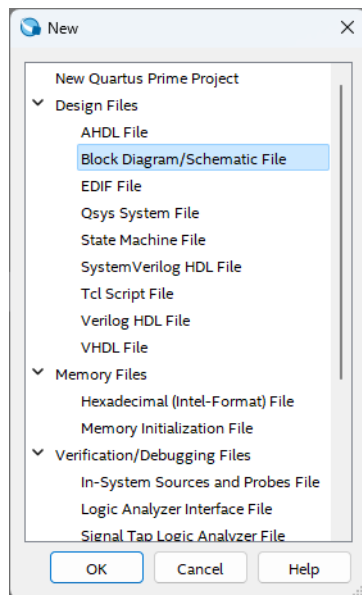
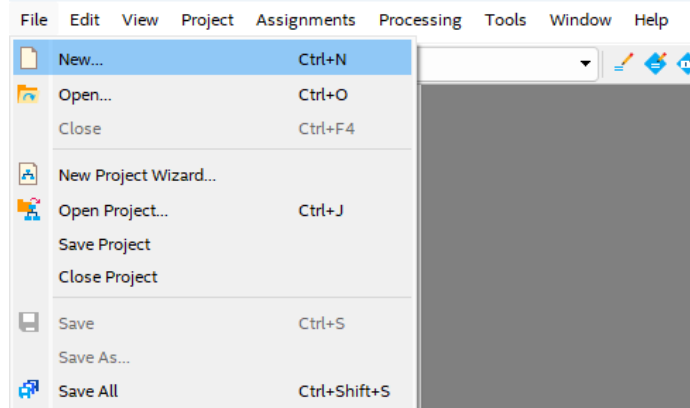
Help



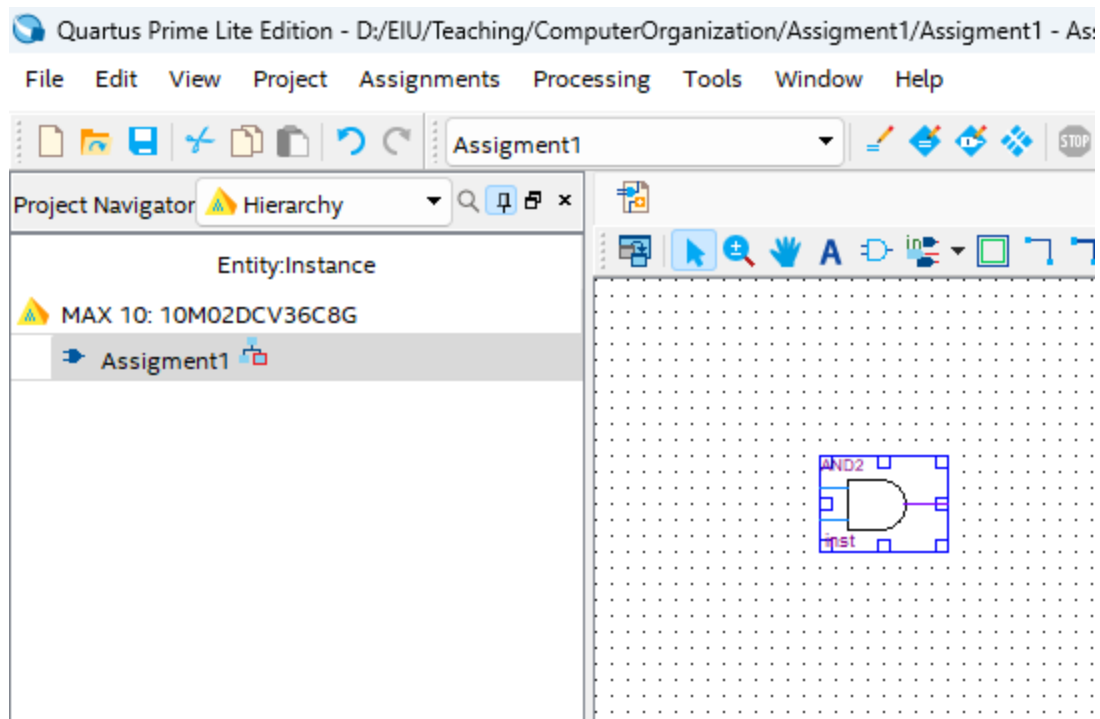
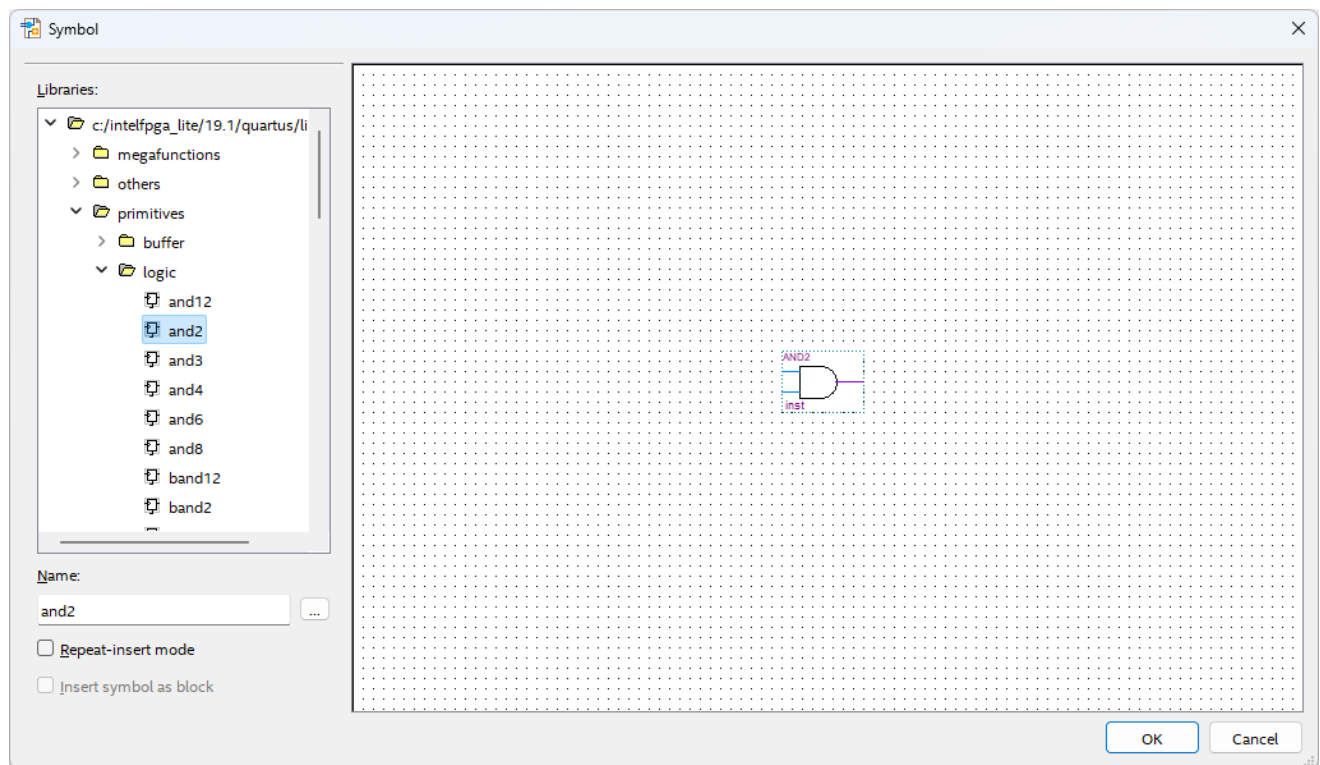
c. Design entry using Graphic editor:

- Create a Block Diagram/Schematic file: *Assignment1.bdf*

Quartus Prime Lite Edition - D:/EIU/Teaching/ComputerOrganization/Assignment1/Assign

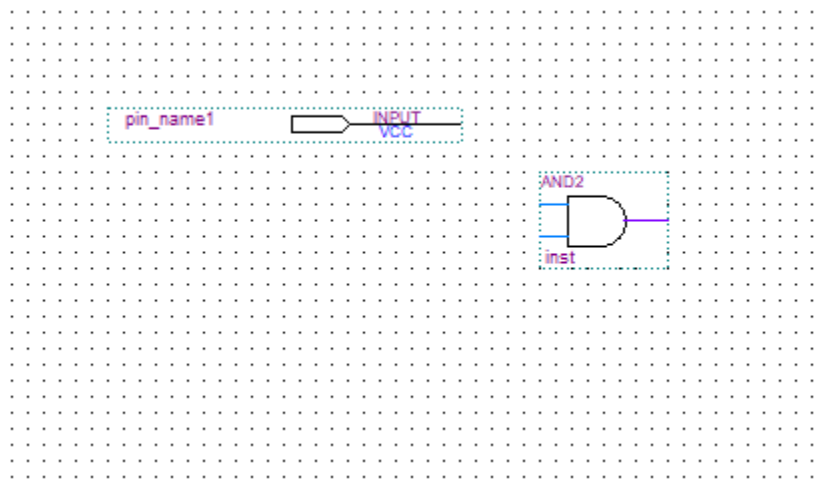
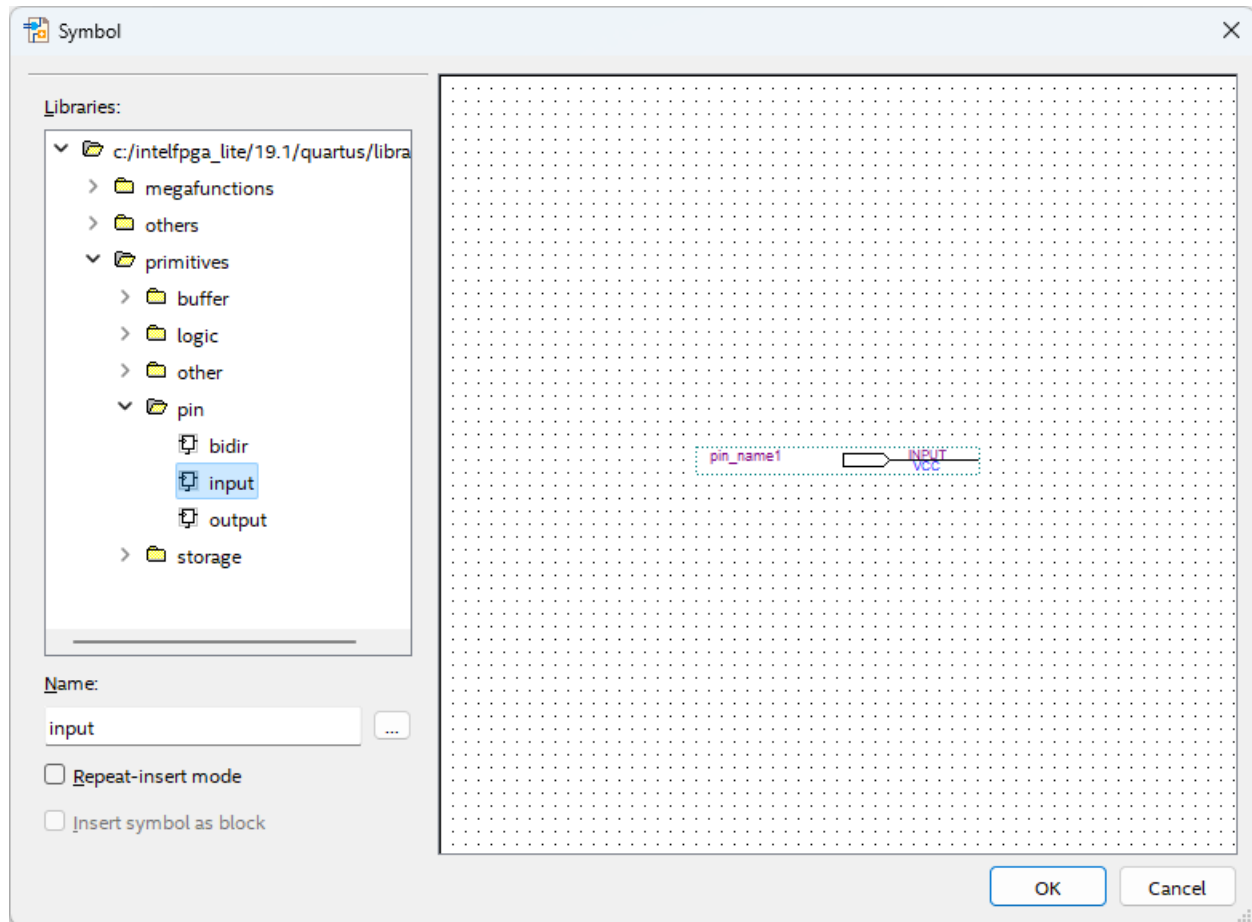


- Insert a logic gate symbol: Double click on the blank space in the Graphic Editor window

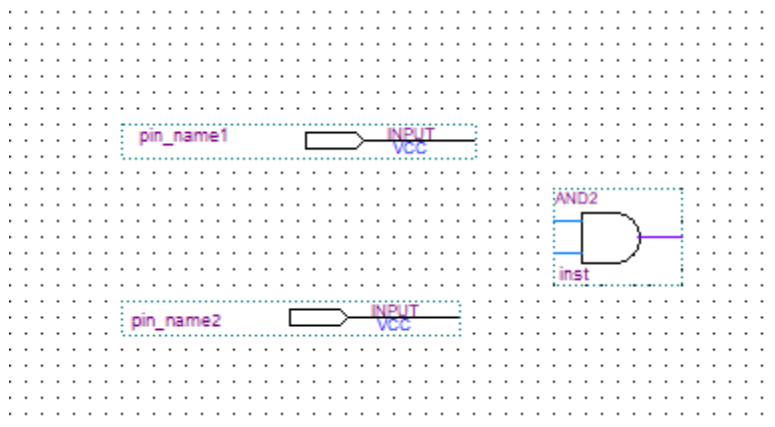
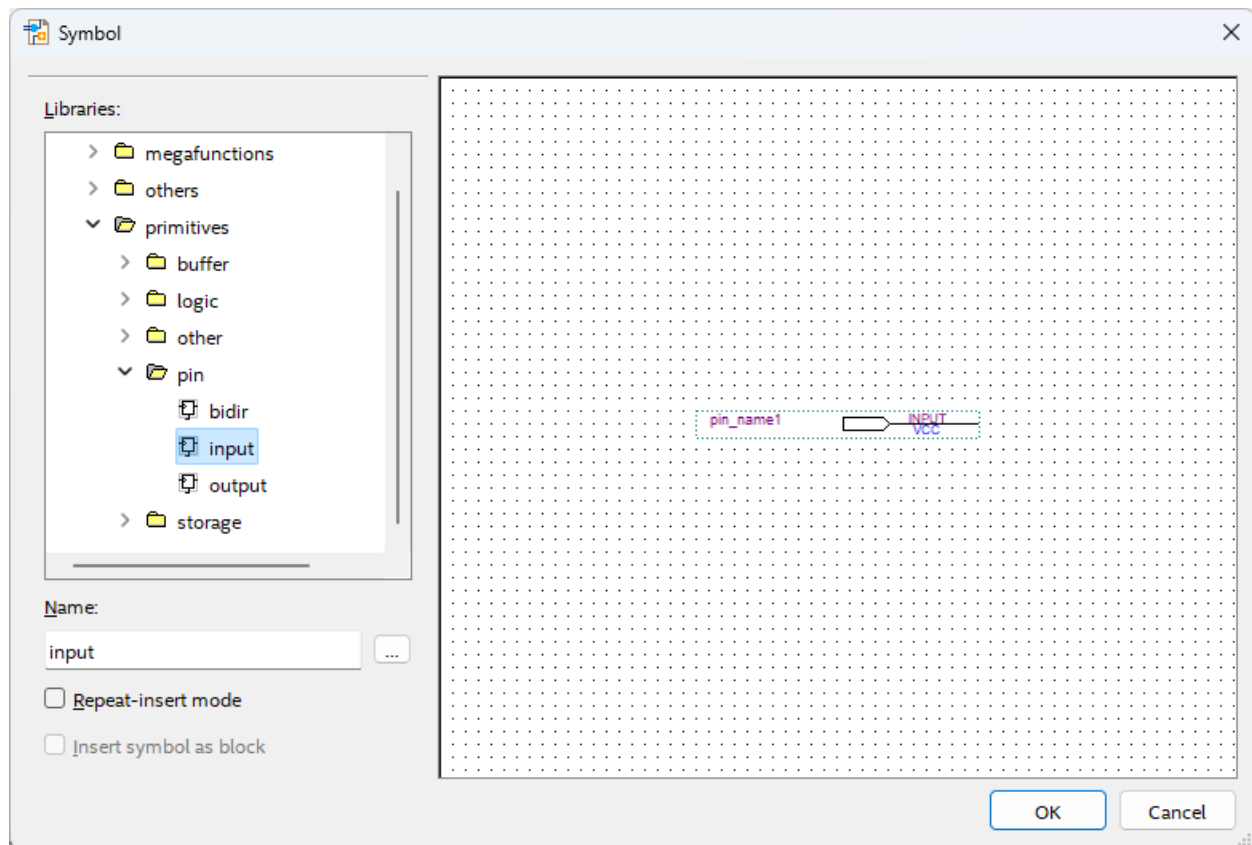


- Insert 2 input symbols

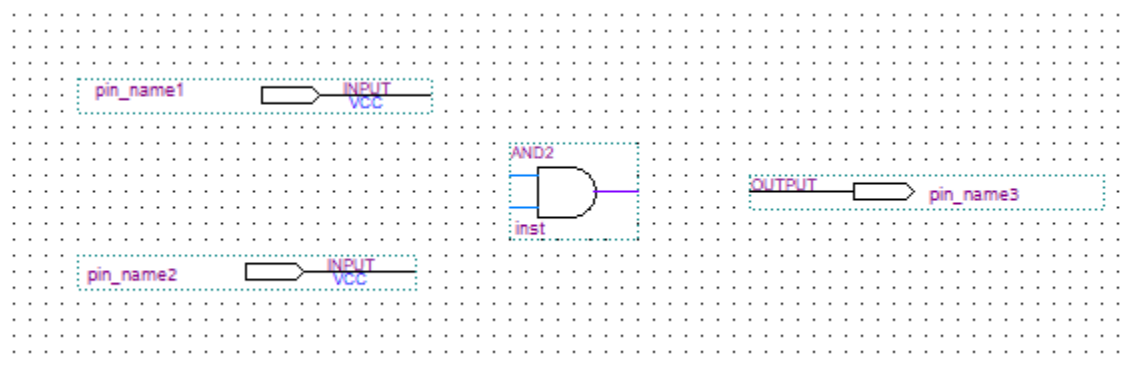
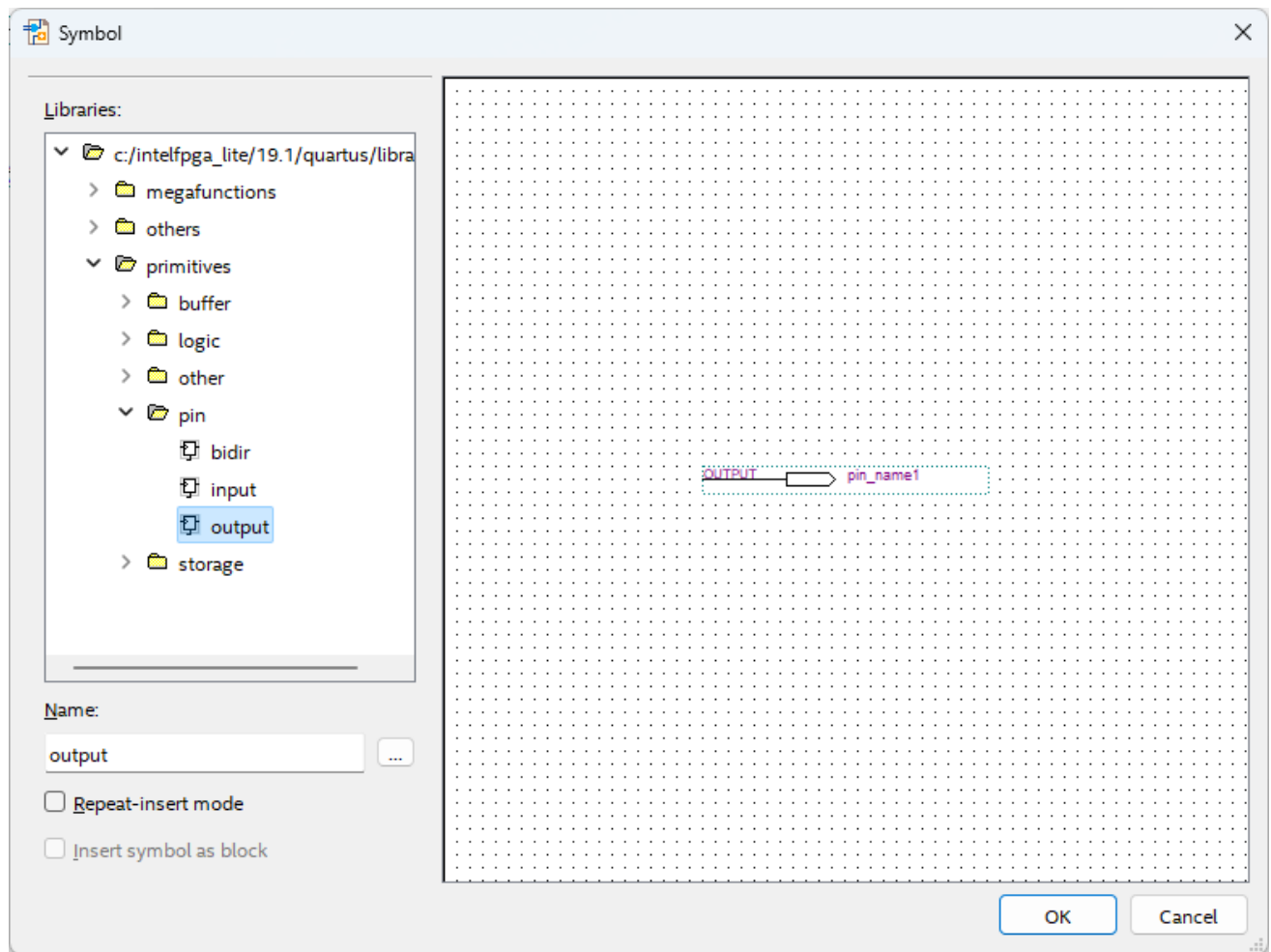
Double click on the blank space in the Graphic Editor window



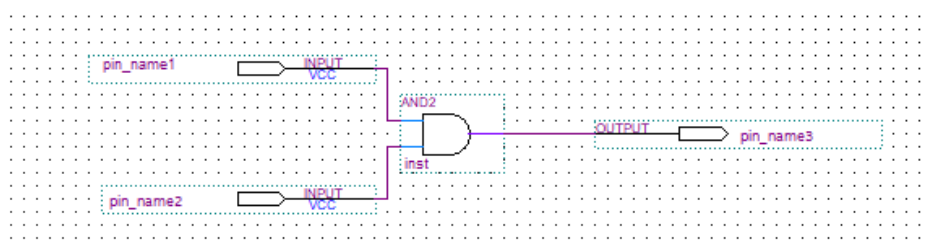
Insert the second input symbol:



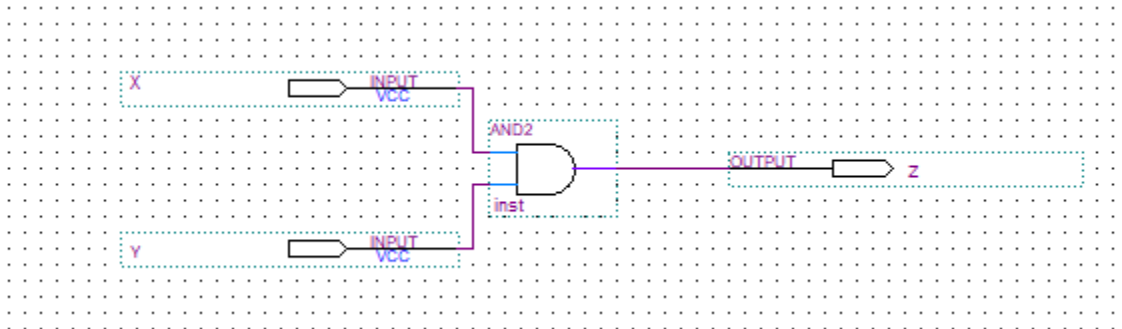
- Insert an output symbol:



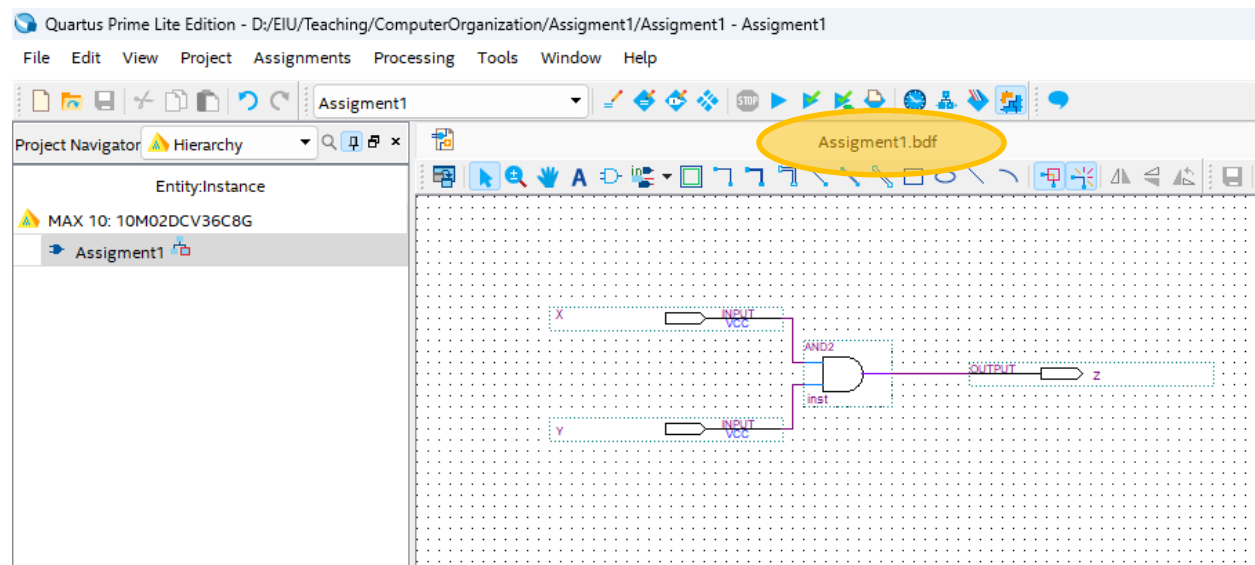
- Connecting nodes with wires:



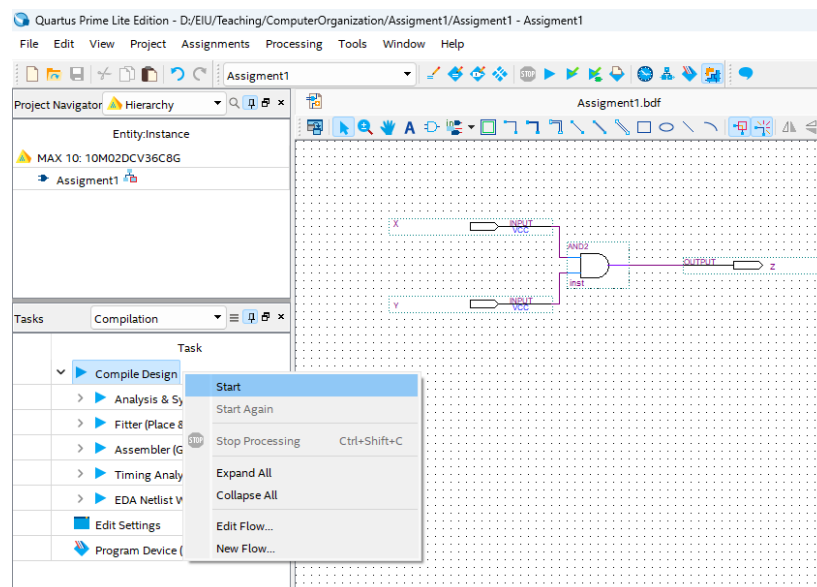
- Assign names to the input and output symbols:

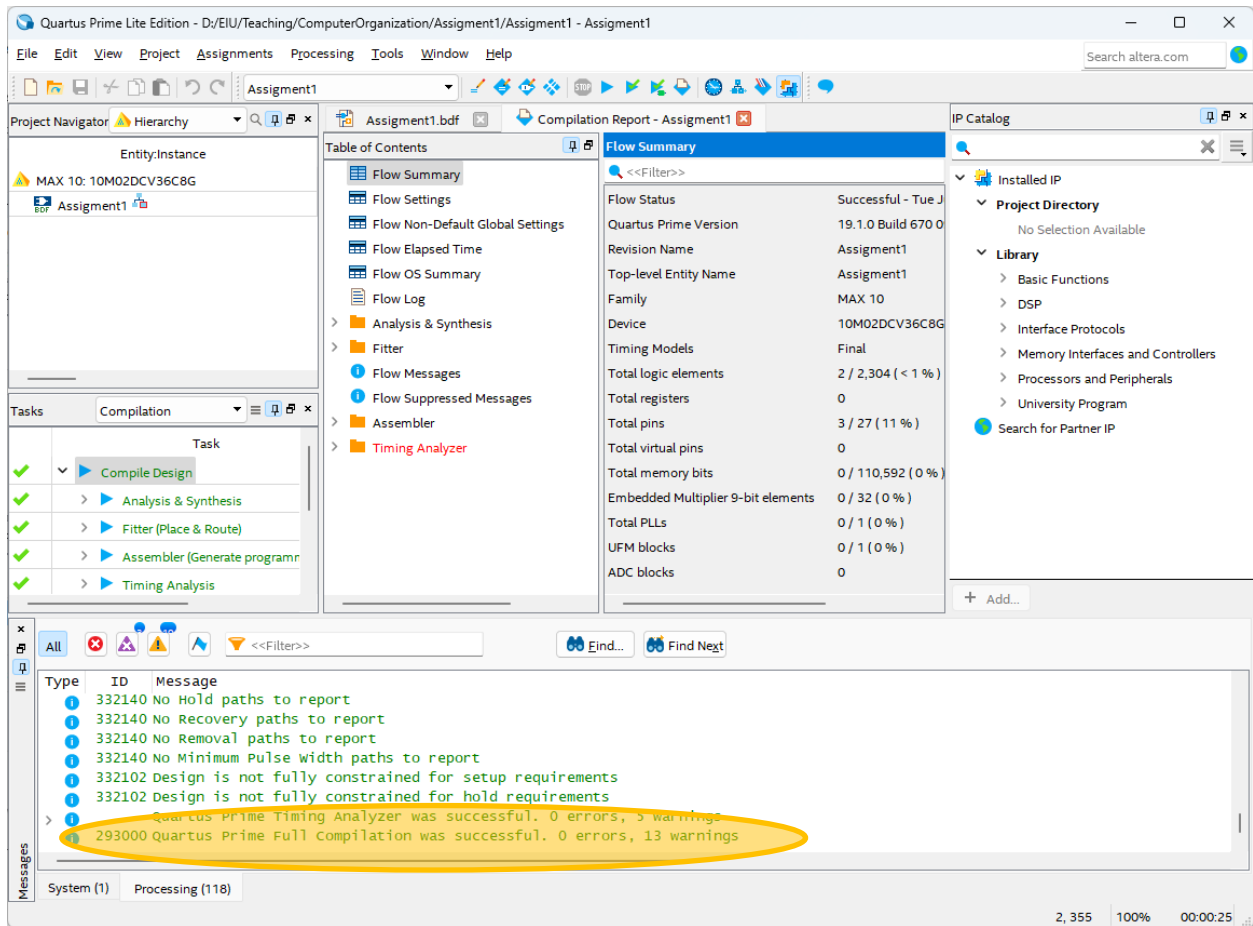


- Save the file named *Assignment1.bdf*



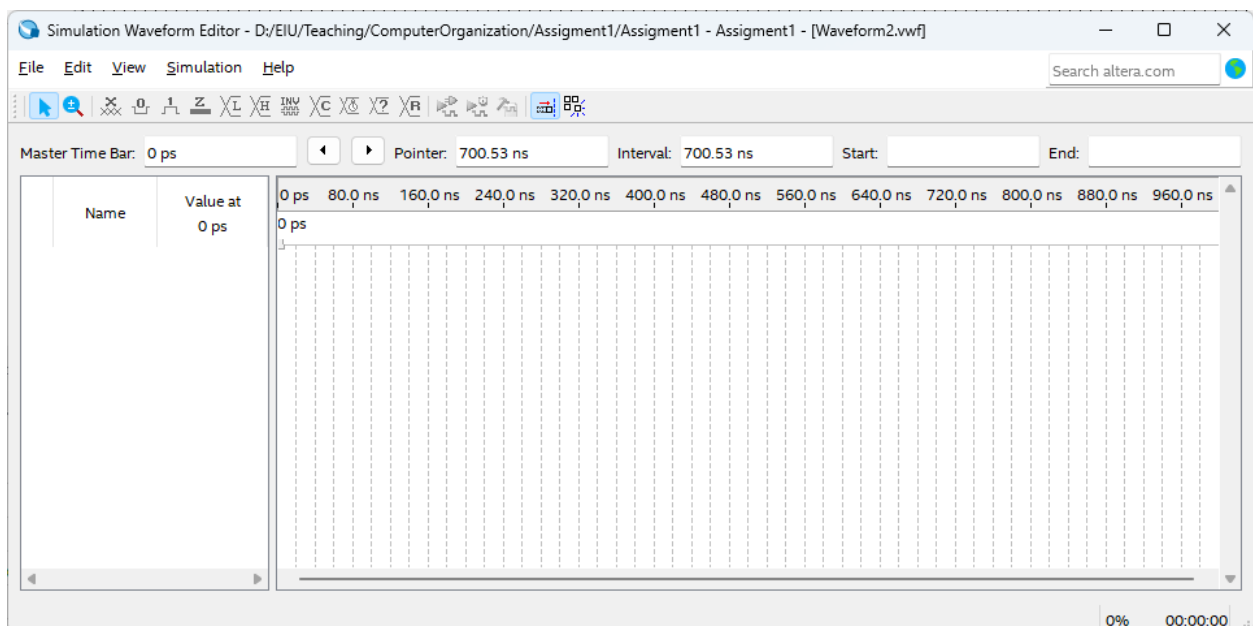
d. Compiling the Designed Circuit



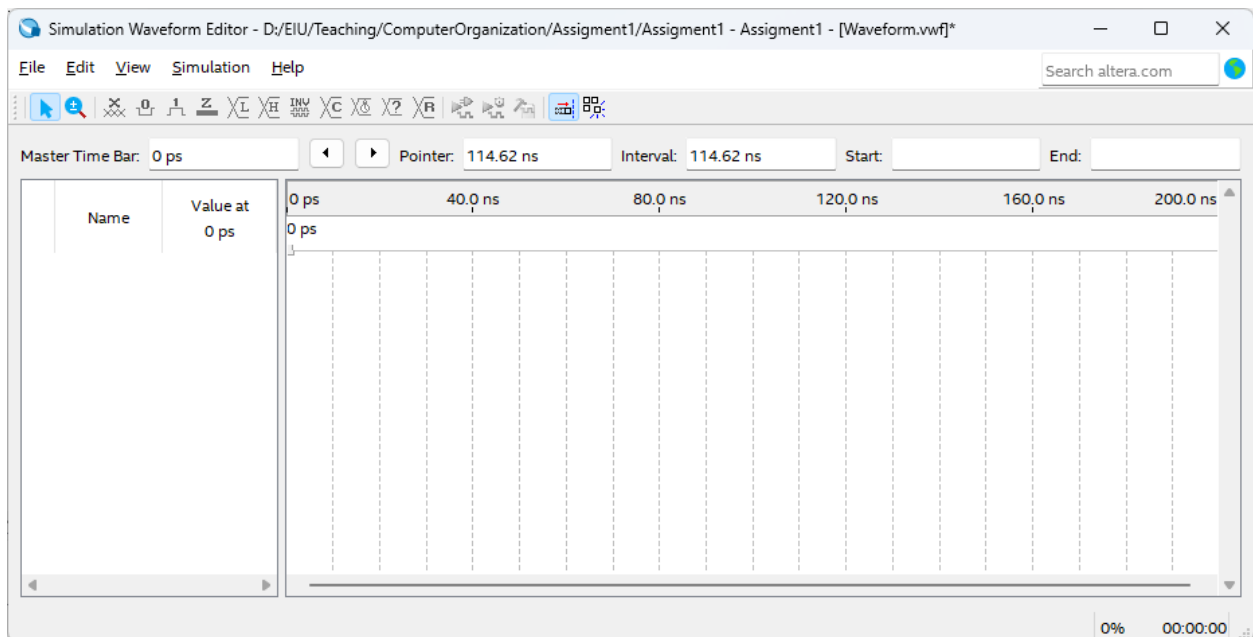
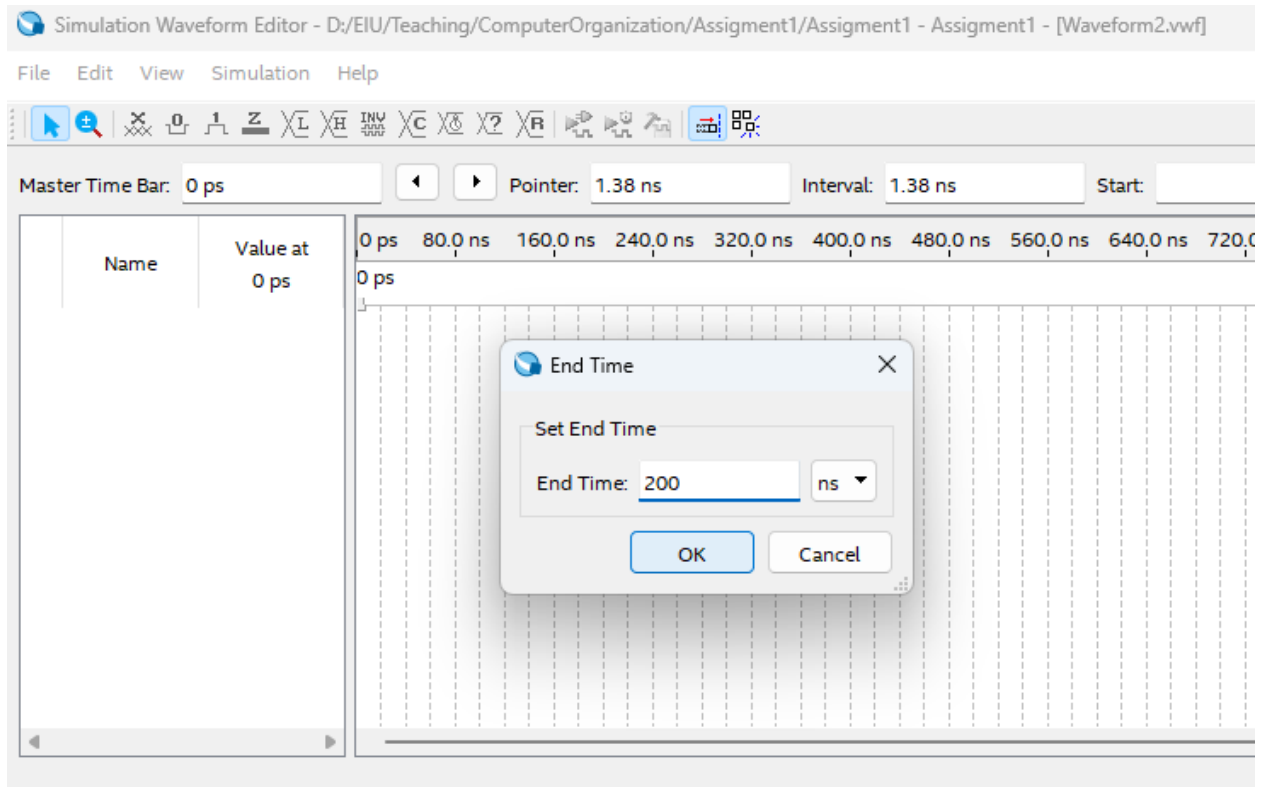


e. Simulating the desired circuit using Simulation Waveform Editor tool:

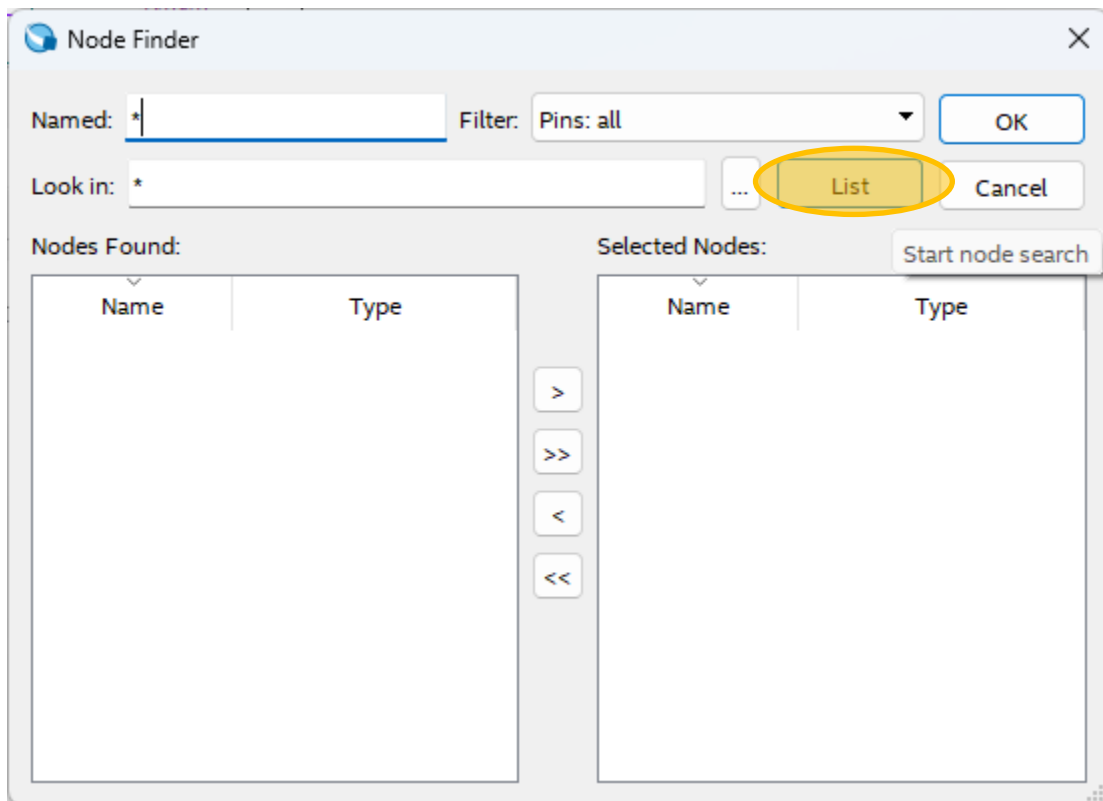
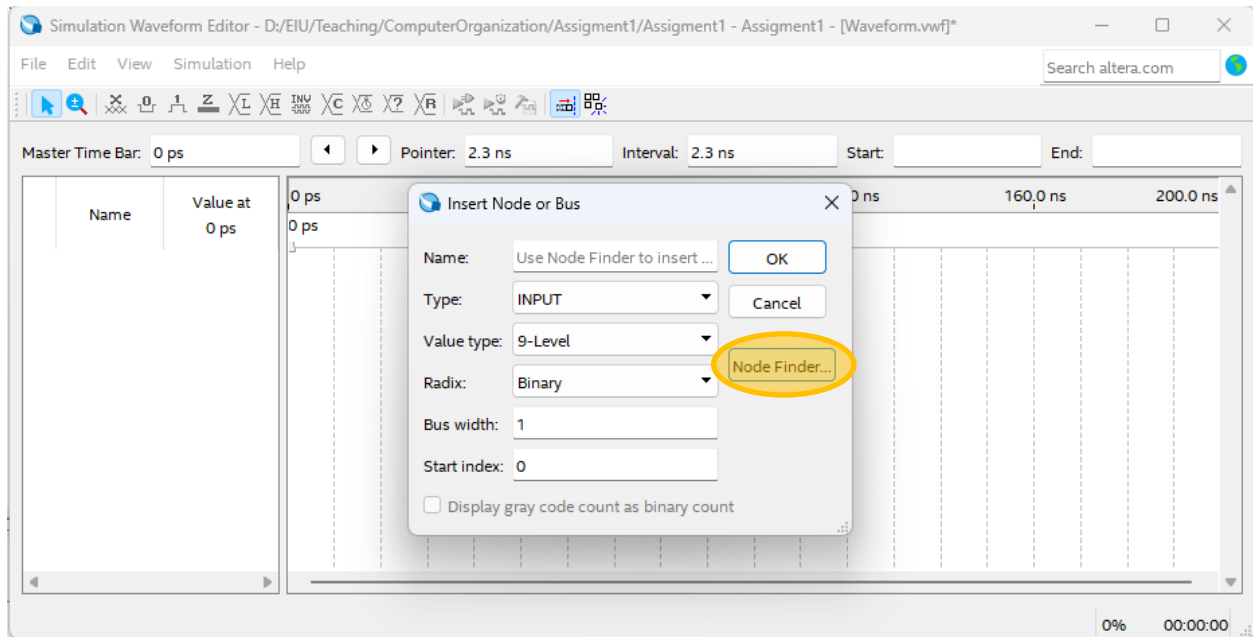
- In the main Quartus Prime window, select **File > New > Verification/Debugging Files > University Program VWF** to open the Simulation Waveform Editor.

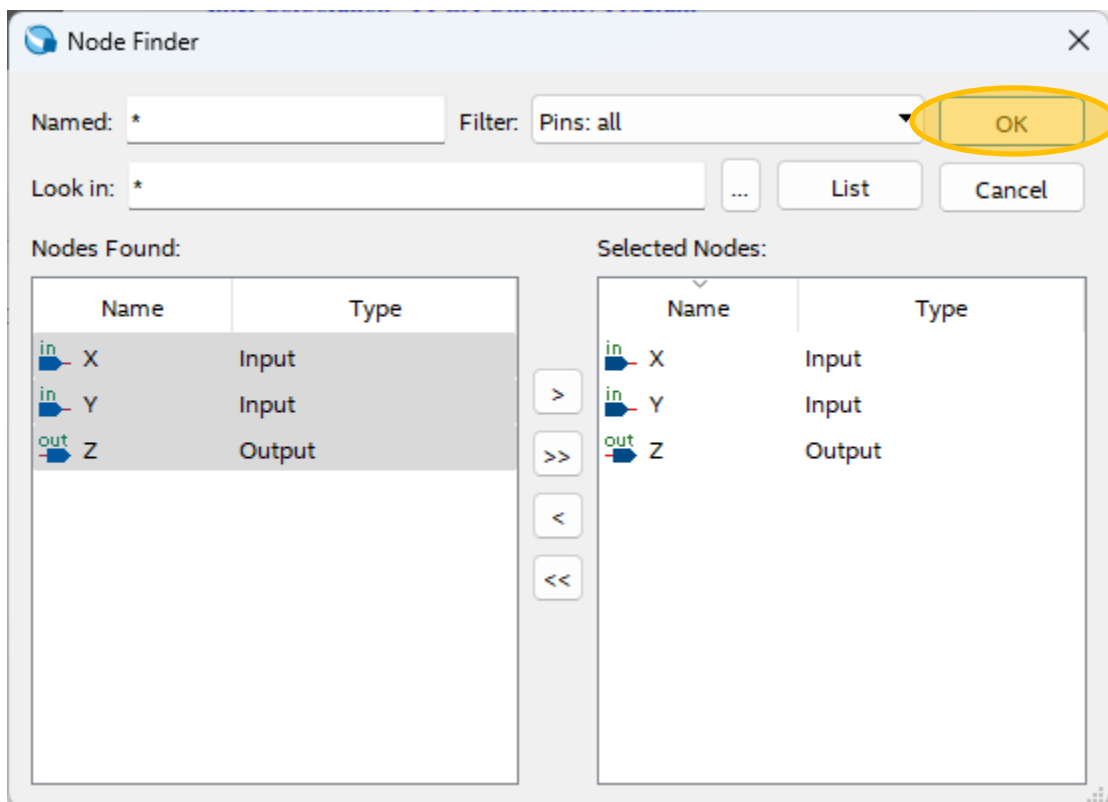
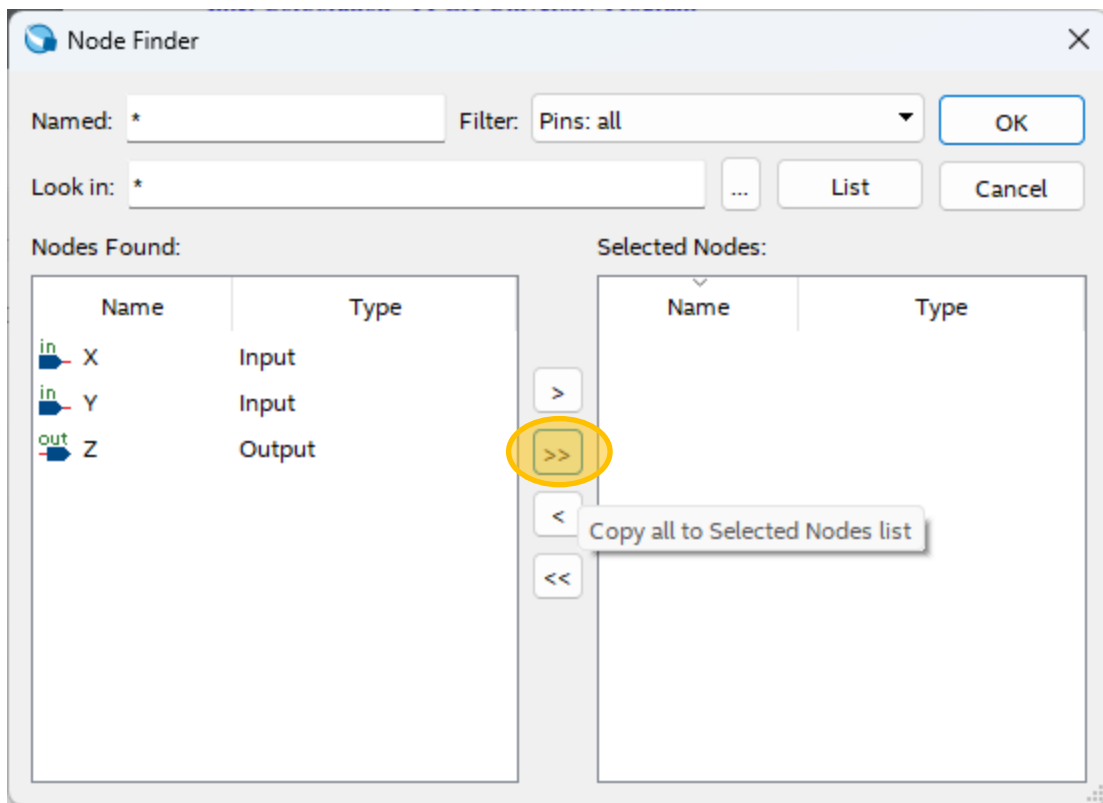


- Set the desired simulation to run from 0 to 200 ns by selecting **Edit > Set End Time** and entering 200 ns in the dialog box that pops up. Selecting **View > Fit** in Window displays the entire simulation range of 0 to 200 ns in the window.



- Include the input and output nodes of the circuit to be simulated: **Click Edit > Insert > Insert Node or Bus** to open the *Insert Node or Bus* window





Insert Node or Bus

Name: ****Multiple Items**** **OK**

Type: ****Multiple Items**** **Cancel**

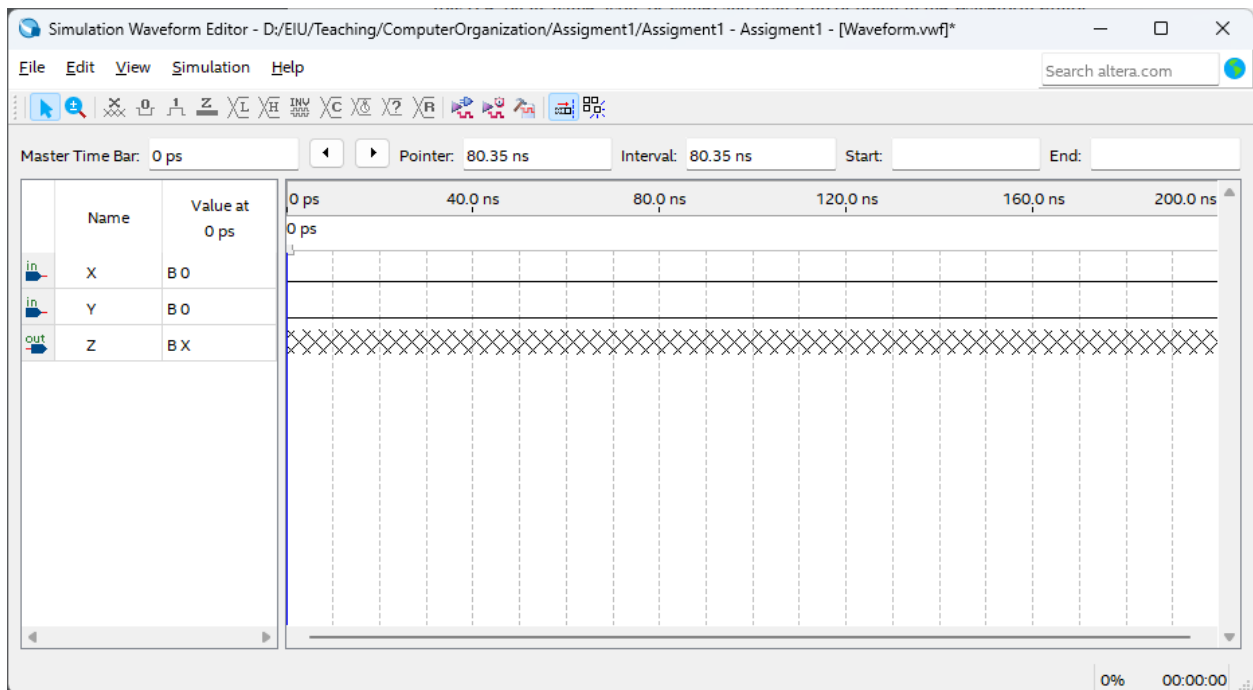
Value type: **9-Level** **Node Finder...**

Radix: **Binary**

Bus width: **1**

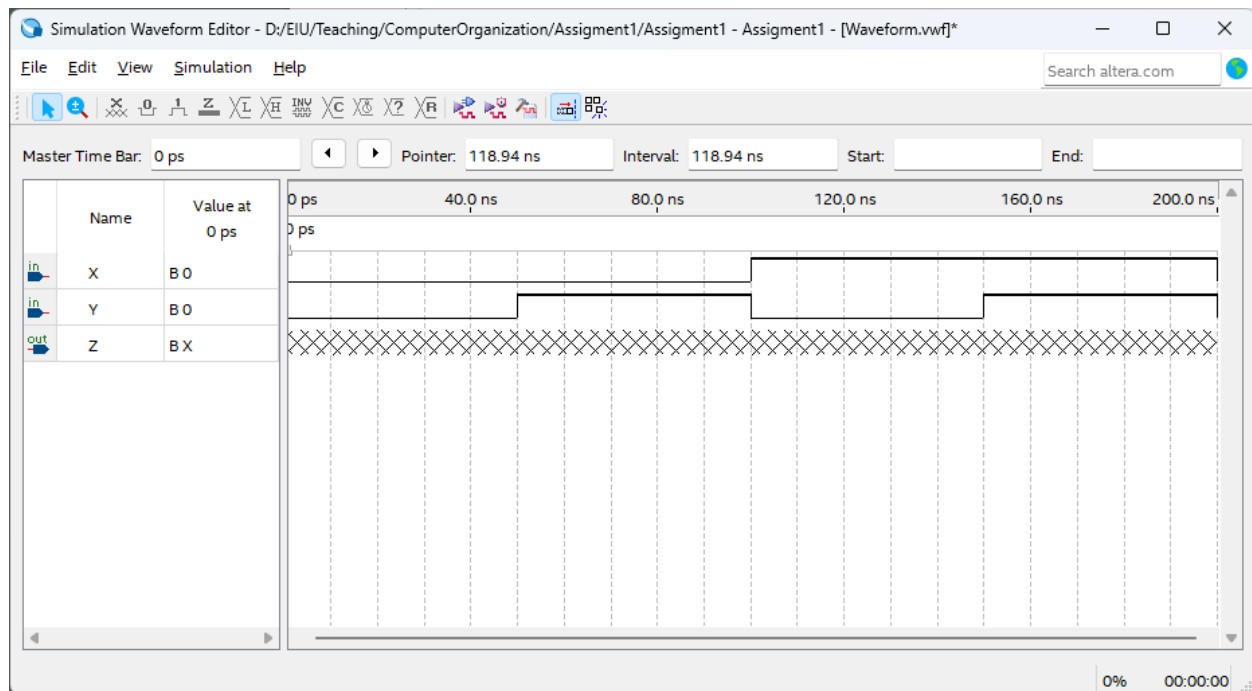
Start index: **0**

☐ Display gray code count as binary count



- Waveform:

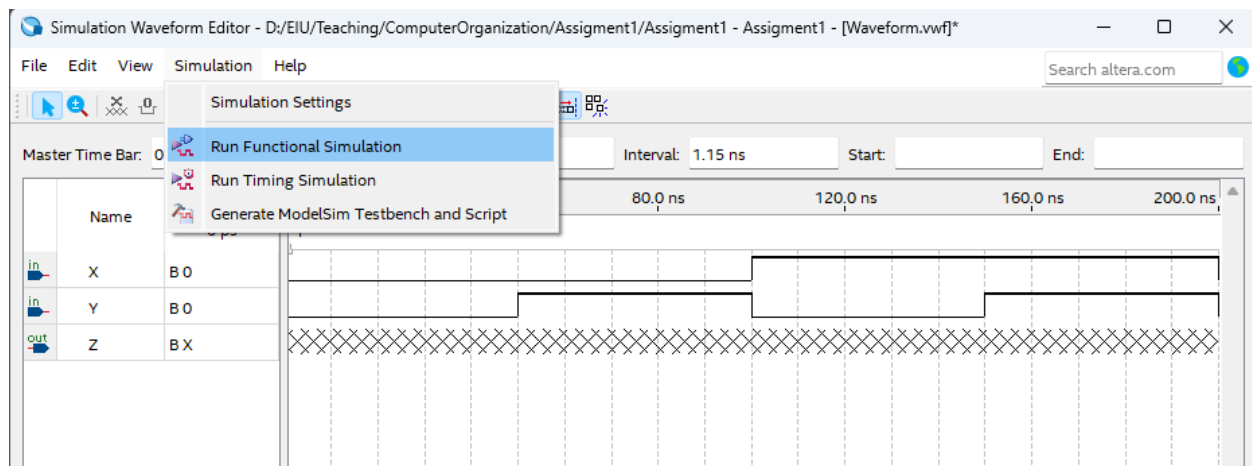
Specify the logic values to be used for the input signals X and Y during simulation and the logic values at the output Z will be generated automatically by the simulator.

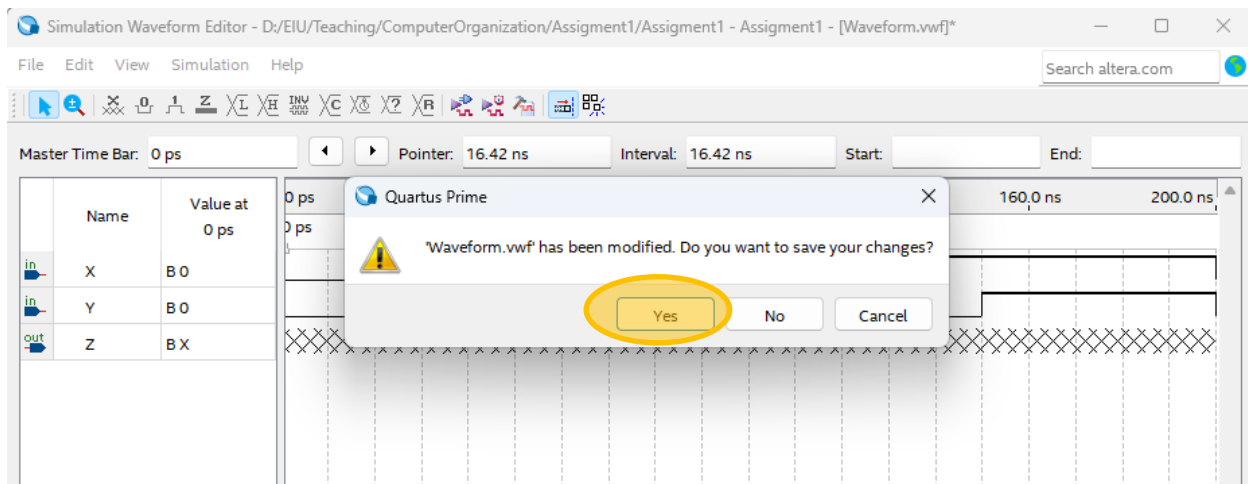


- Performing the Simulation:

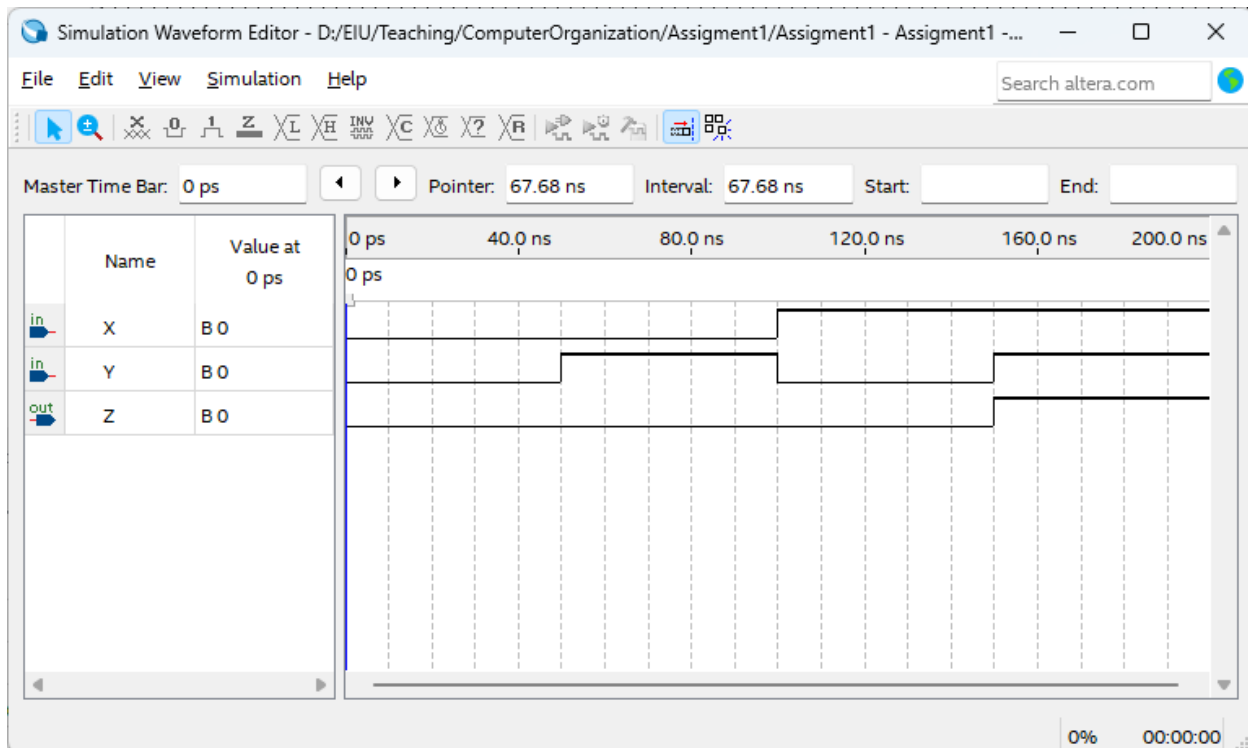
- + Functional simulation: Causing no delay in propagation of signals through the circuit
- + Timing simulation: Take all propagation delays into account

Typically, functional simulation is used to verify the functional correctness of a circuit as it is being designed.





- At the end of the simulation, a second Waveform Editor window will open the results of the simulation as illustrated in the figure below.



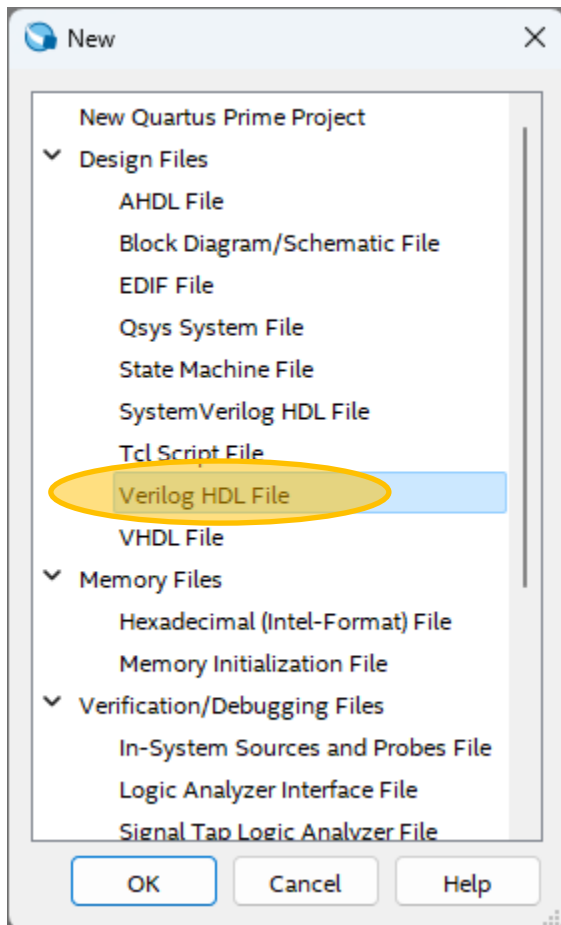
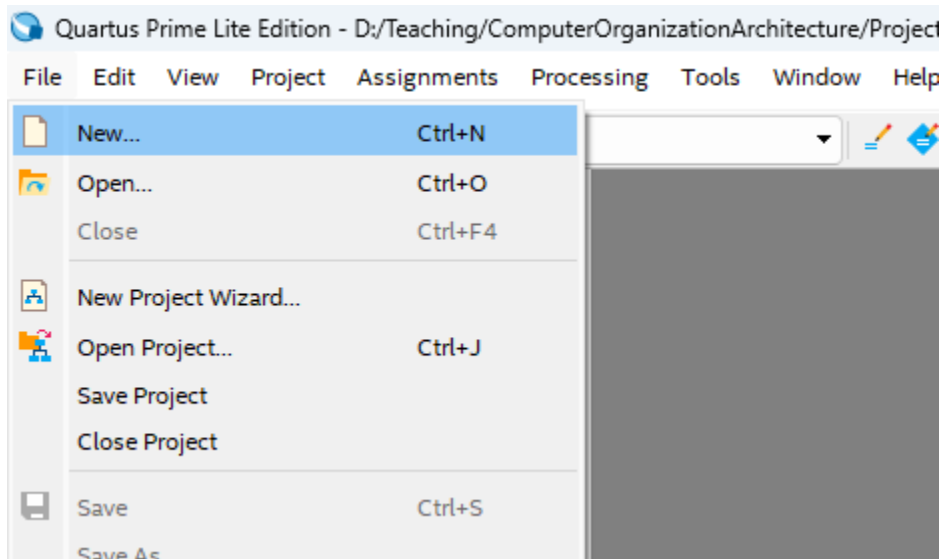
- Observe that the output Z is as specified in the truth table.

4. Verilog code:

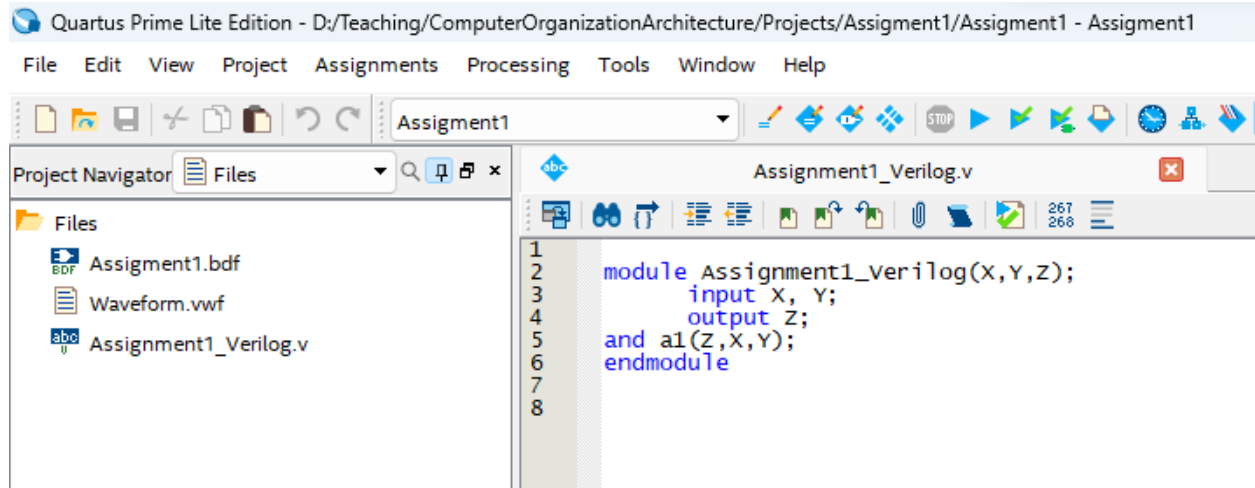
```
module Assignment1_Verilog(X,Y,Z);
    input X, Y;
    output Z;

    and a1(Z,X,Y);
endmodule
```

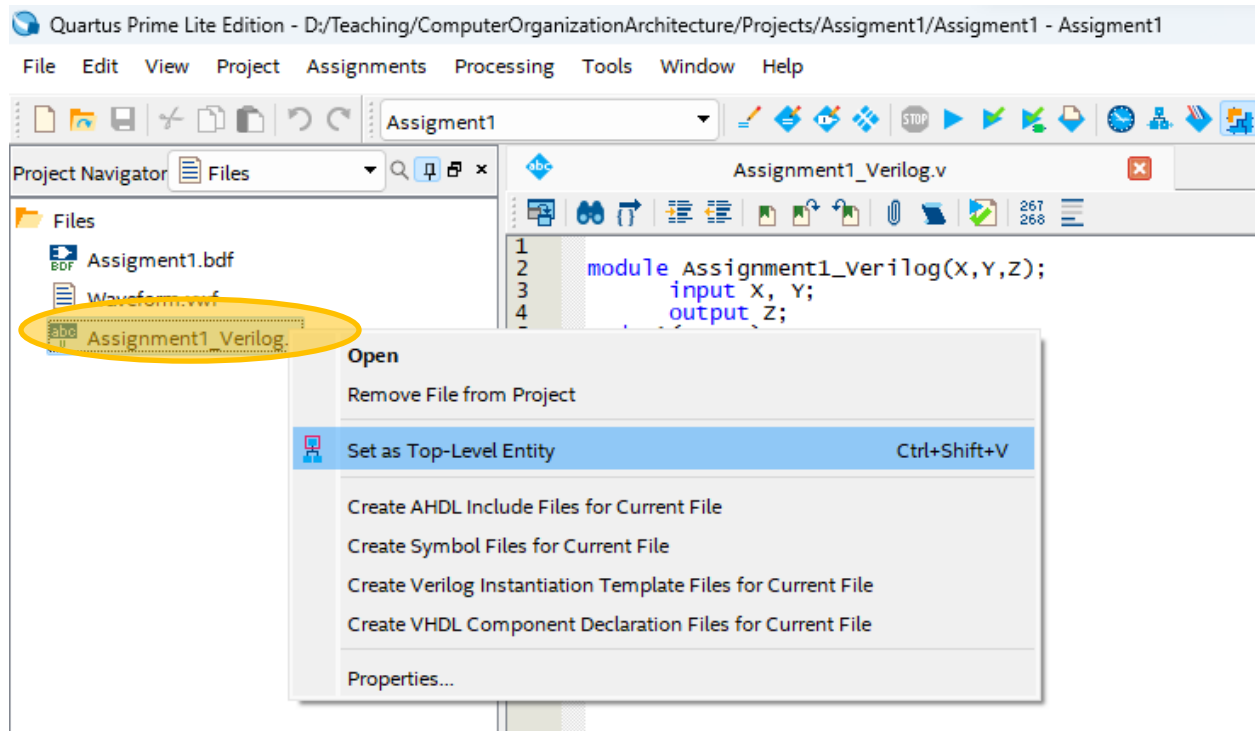
- Insert a new Verilog file into the project:



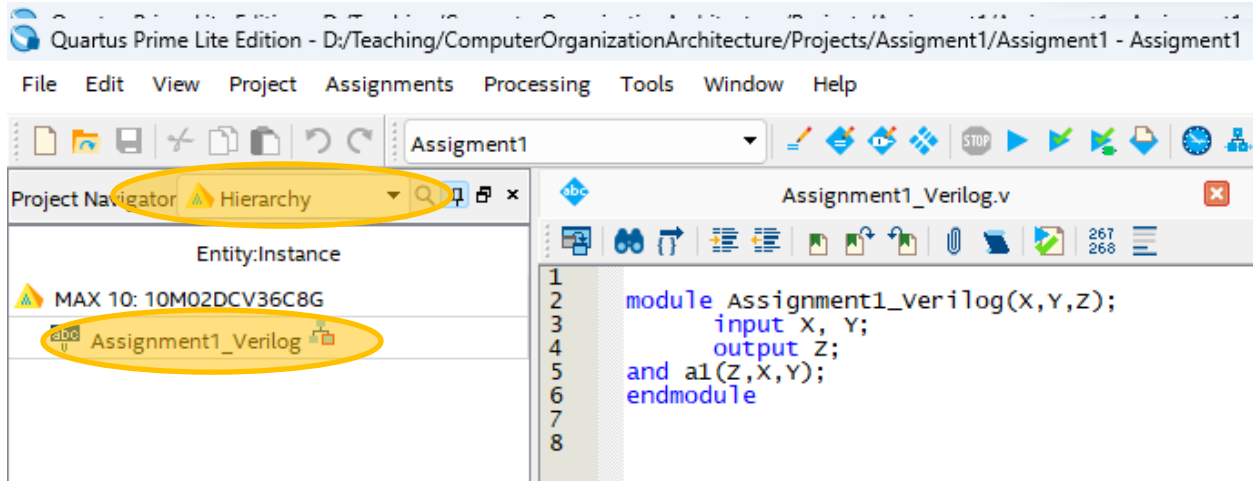
- Save the file *Assigment1_Verilog.v*



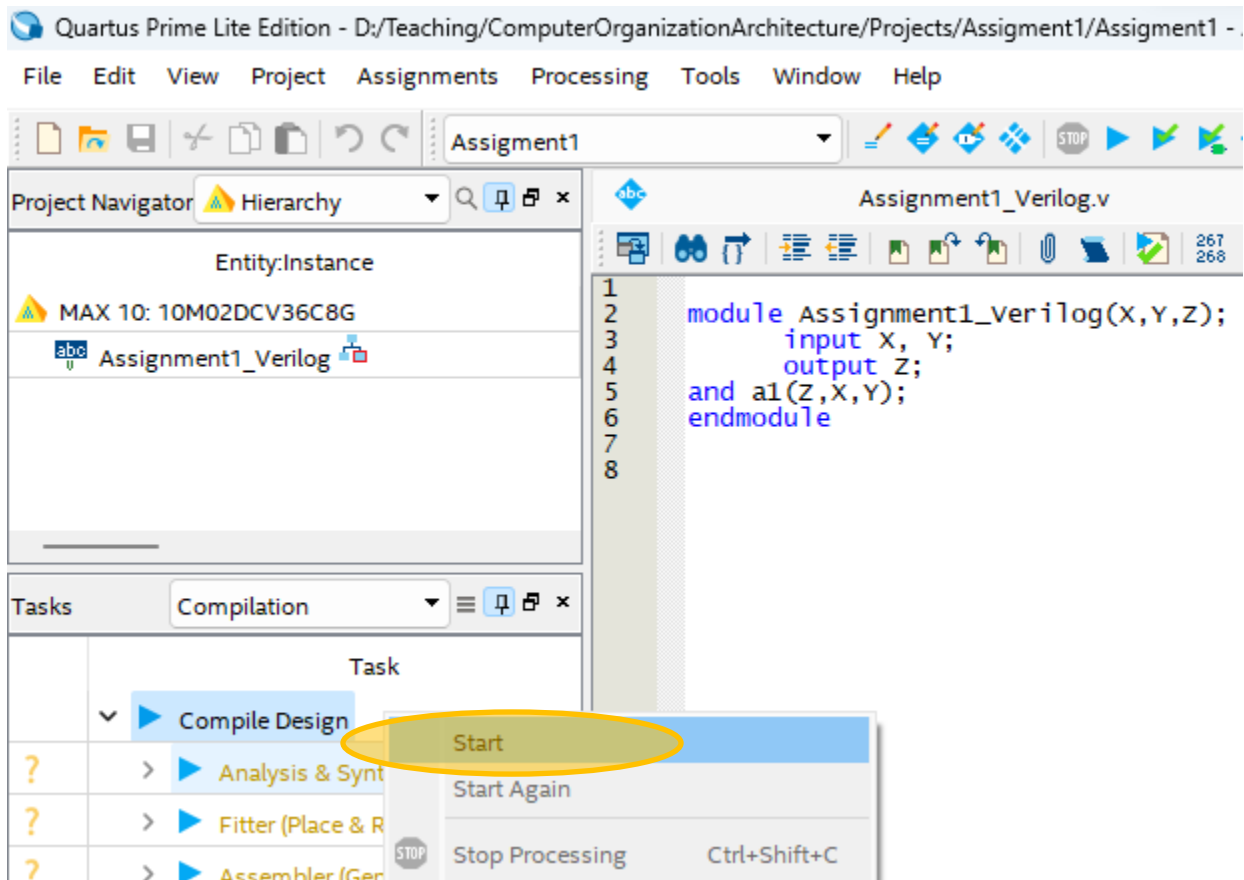
- Select the Verilog file as *Top Level Entry*:

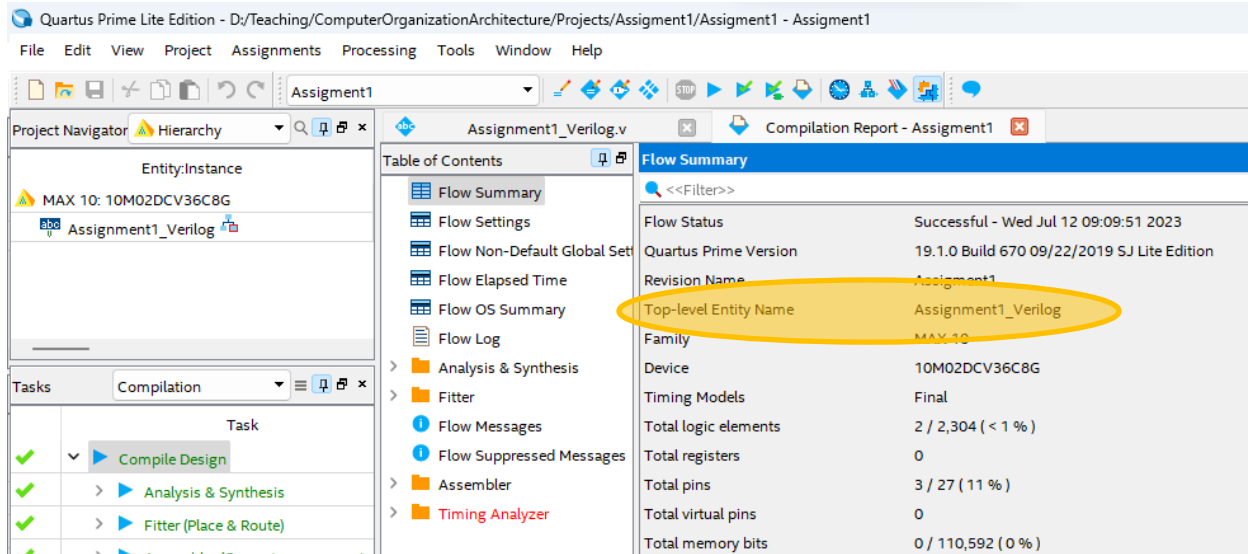


- Check whether the Verilog file is selected as *Top Level Entry* or not:



- Re-compiling the design:





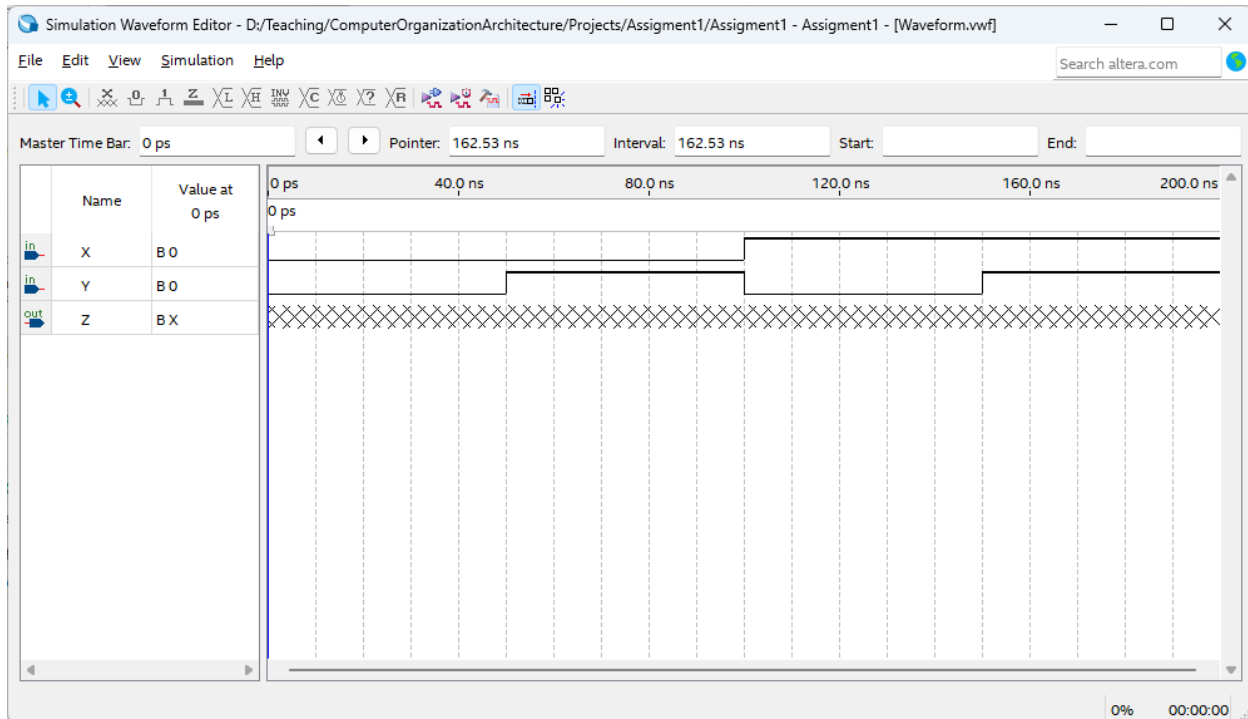
- Performing the simulation for the Verilog file:

+ The Verilog is selected as Top Level Entry of the project.

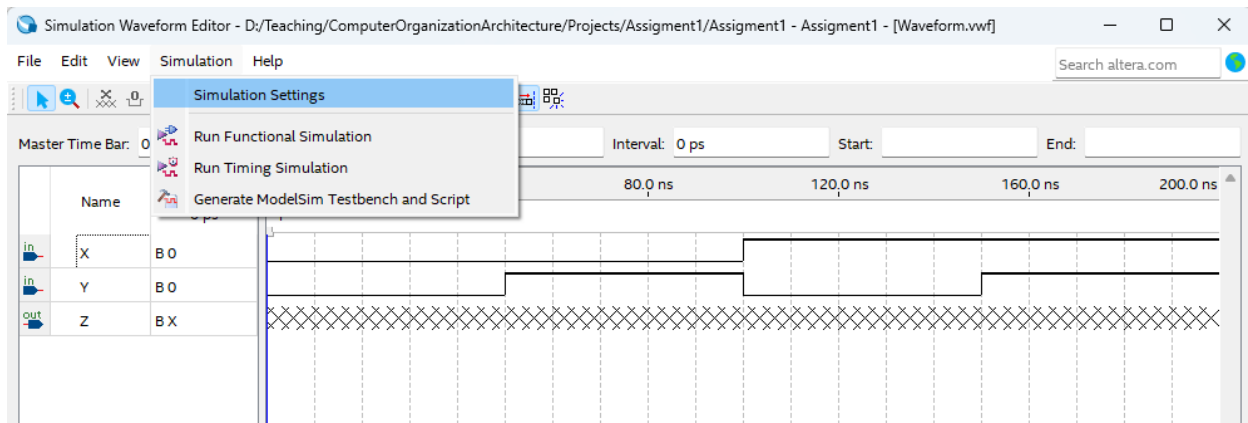
+ The name of the top level module (Verilog) is the same as the Verilog file' name.

+ The name of inputs and outputs are the same as the name of inputs and outputs' pins in the block design file.

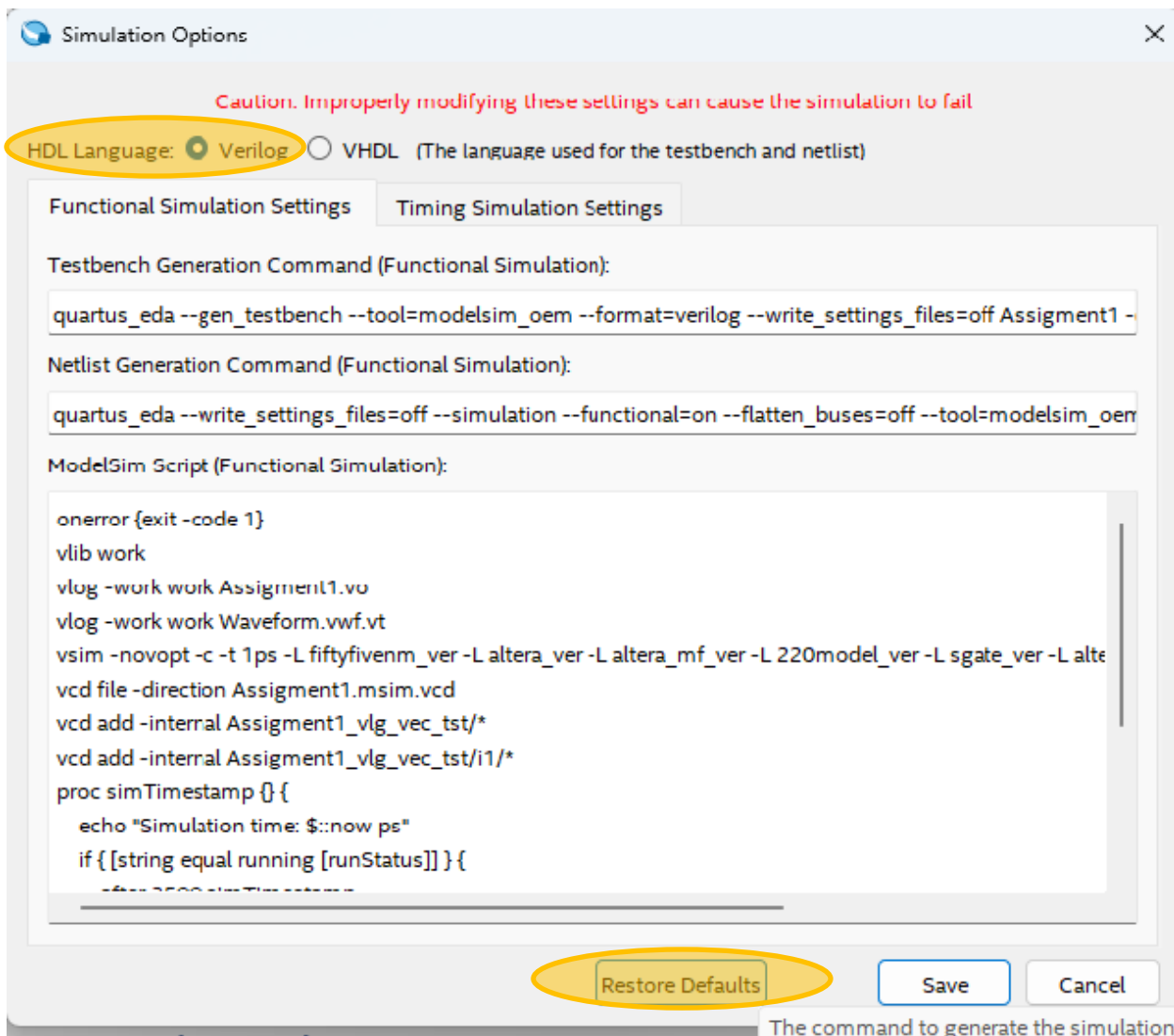
+ Open the *Waveform*. Vwf file in the Simulation Waveform Editor:

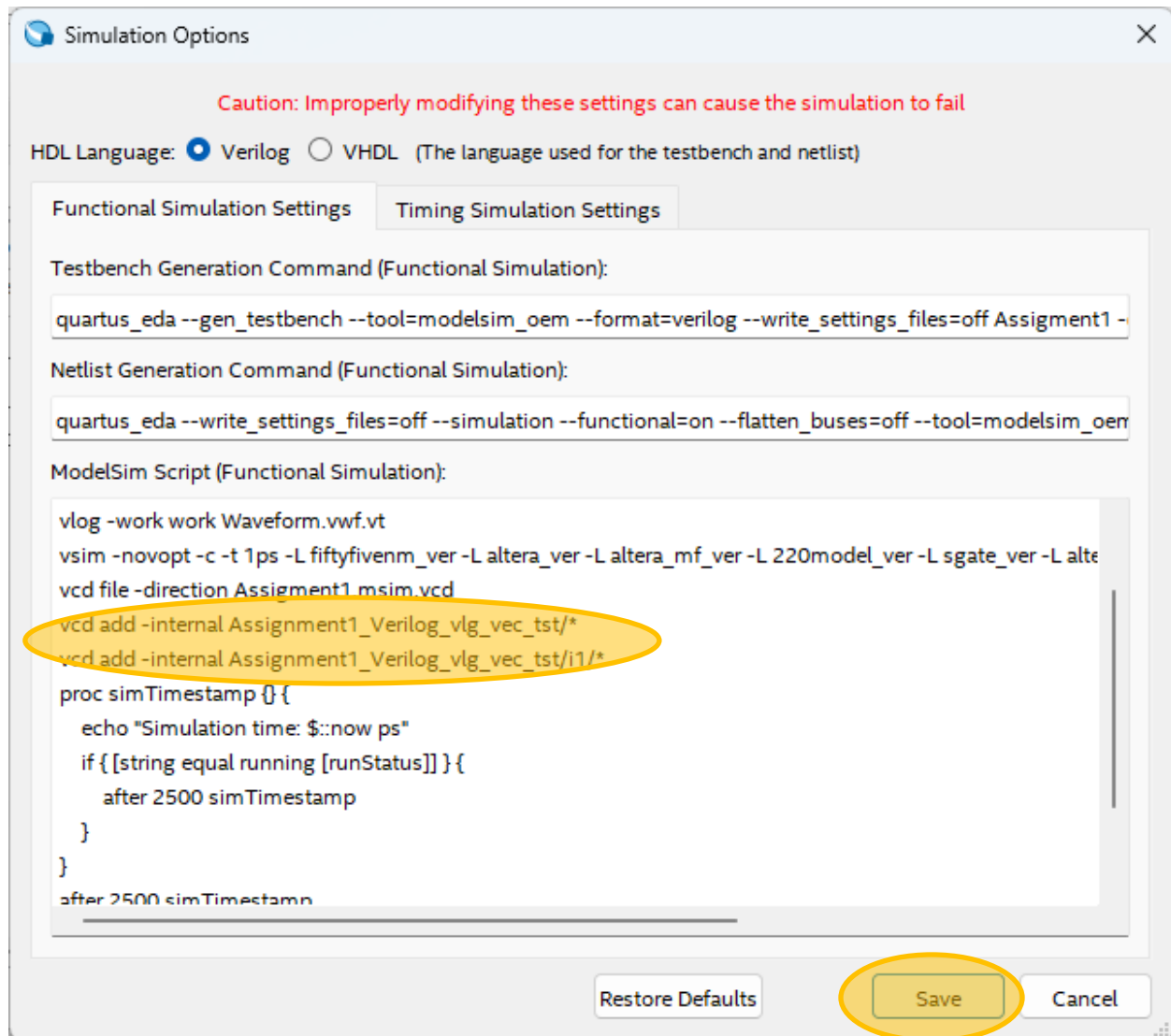


+ Open the *Simulation Options* window:

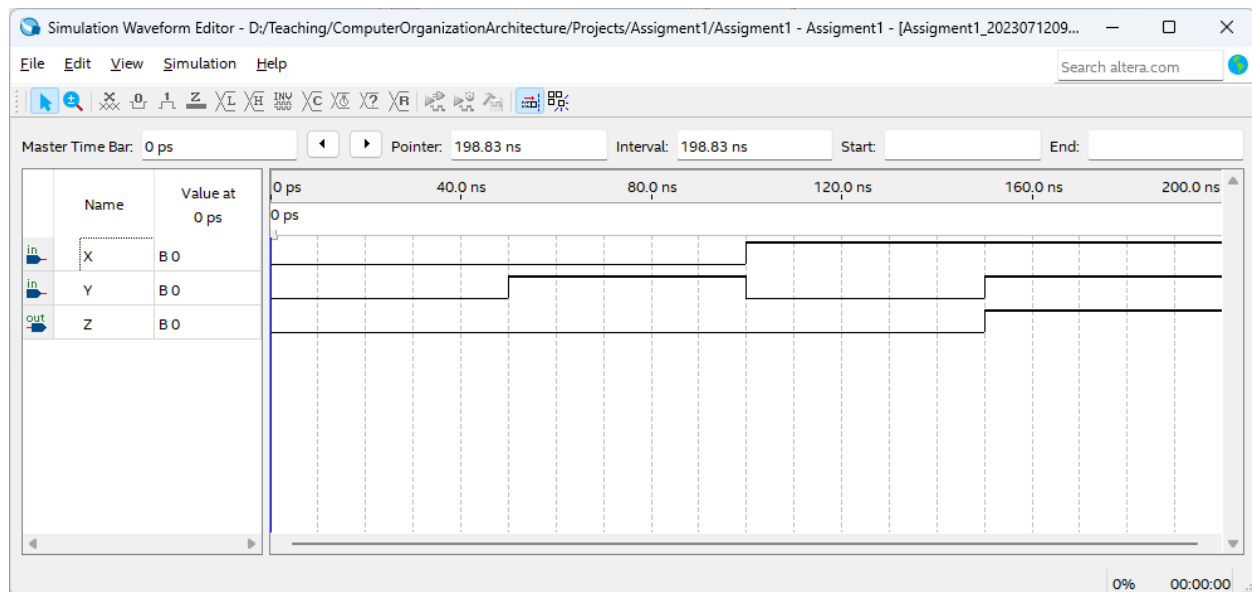


+ Restore the default settings





+ Run the functional simulation (or timing simulation) and check the output with the truth table.



Assignment 2 (10 points)

Familiarization and Implementation of OR Gate by using schematic design entry and gate level model using Verilog.

Assignment 3 (10 points)

Familiarization and Implementation of NAND Gate by using schematic design entry and gate level model using Verilog.

Assignment 4 (10 points)

Familiarization and Implementation of NOR Gate by using schematic design entry and gate level model using Verilog.

Assignment 5 (10 points)

Familiarization and Implementation of NOT Gate by using schematic design entry and gate level model using Verilog.

Assignment 6 (10 points)

Familiarization and Implementation of XOR Gate by using schematic design entry and gate level model using Verilog.

Assignment 7 (10 points)

Familiarization and Implementation of XNOR Gate by using schematic design entry and gate level model using Verilog.