



Lab 02

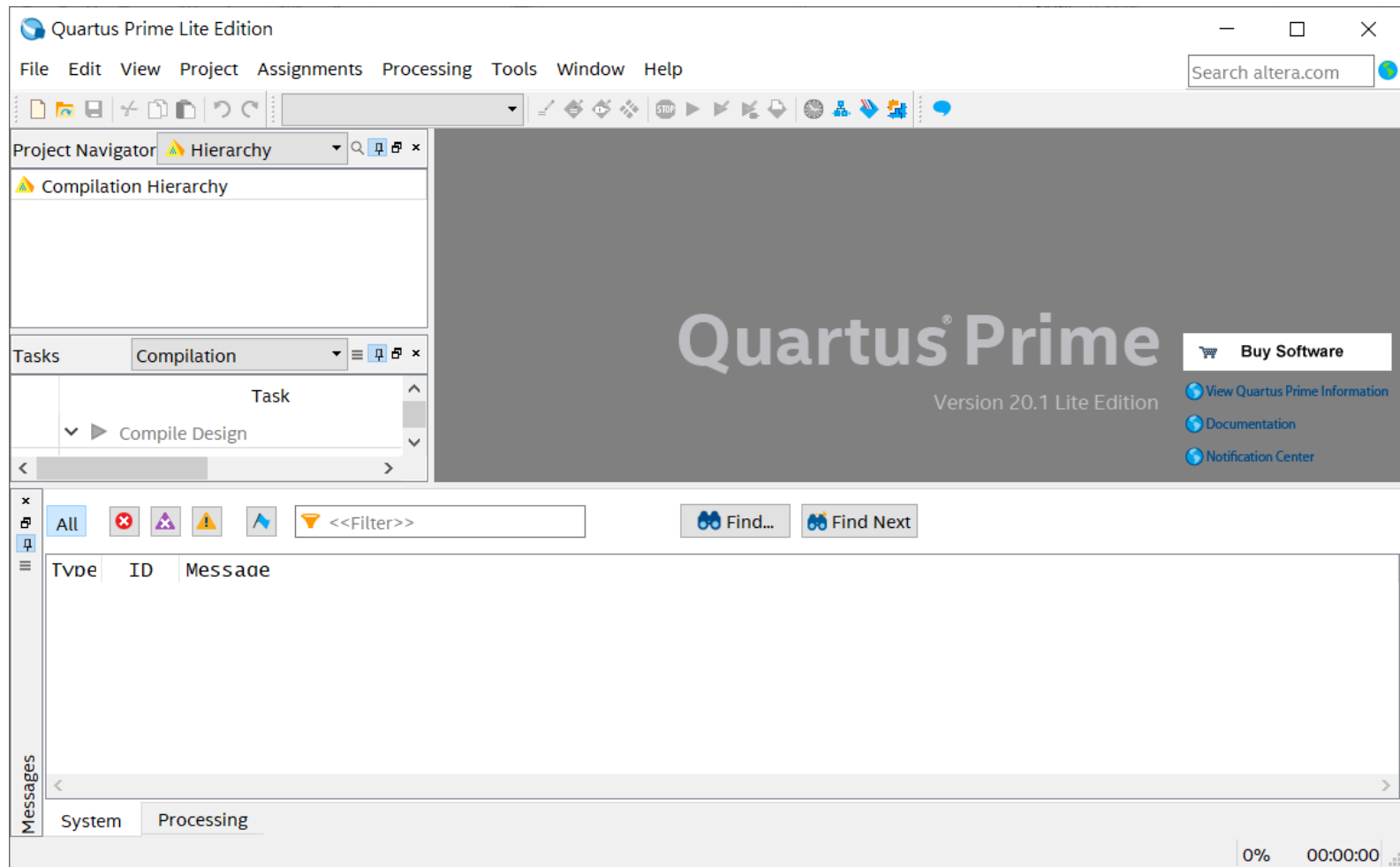
Full Adder 設計

Ren-Der Chen (陳仁德)
Department of Computer Science and
Information Engineering
National Changhua University of Education
E-mail: rdchen@cc.ncue.edu.tw
Spring, 2024

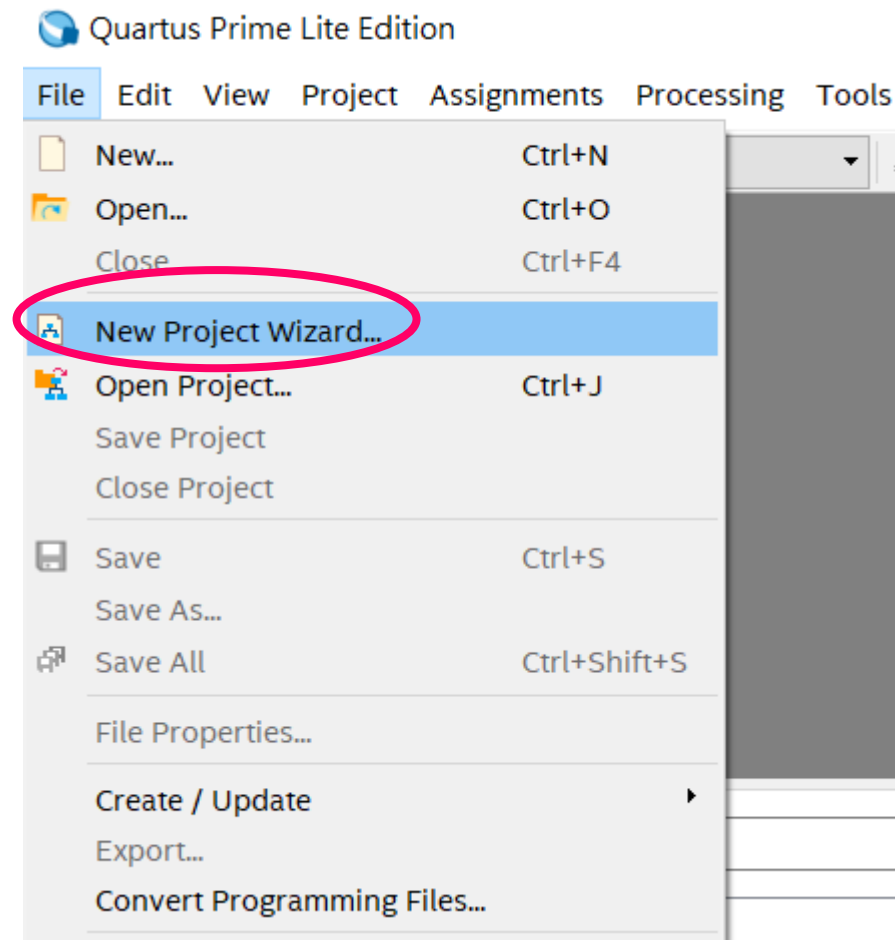
實驗目的

- 熟悉 Quartus 軟體之操作
- 學習如何以schematic方式設計half adder及full adder
- 學習如何對所設計之電路進行functional及timing simulation
- 學習如何將所設計之電路燒錄至FPGA晶片中，並利用實驗板週邊進行電路功能驗證

Get Started



Create a New Project



Create a New Project - add_half



Directory, Name, Top-Level Entity

What is the working directory for this project?

D:\csie_lab\lab02\add_half



What is the name of this project?

add_half



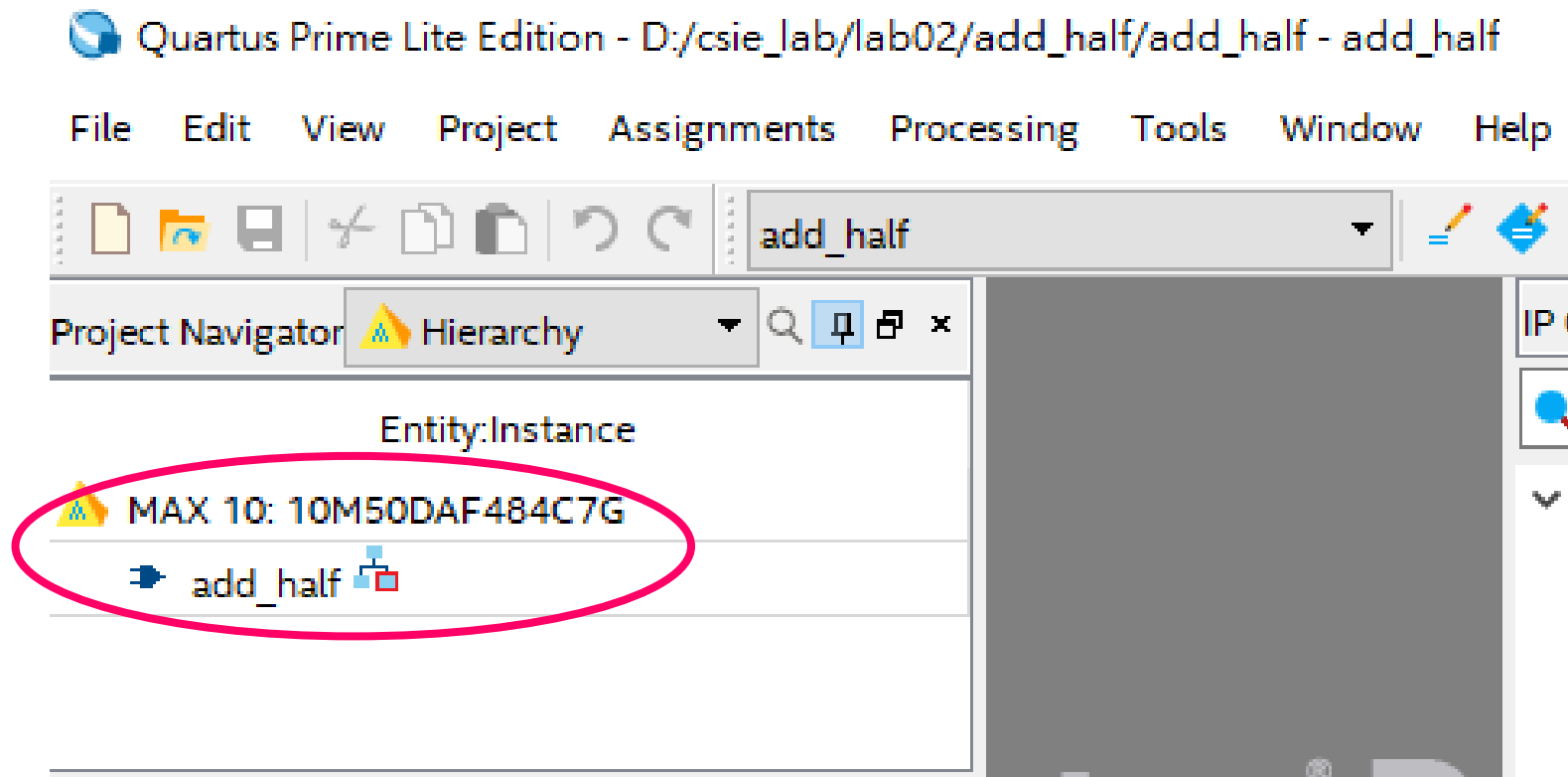
What is the name of the top-level design entity for this project?
The name must exactly match the entity name in the design file.

add_half



Use Existing Project Settings...

Select a Device



Design Example: Half Adder

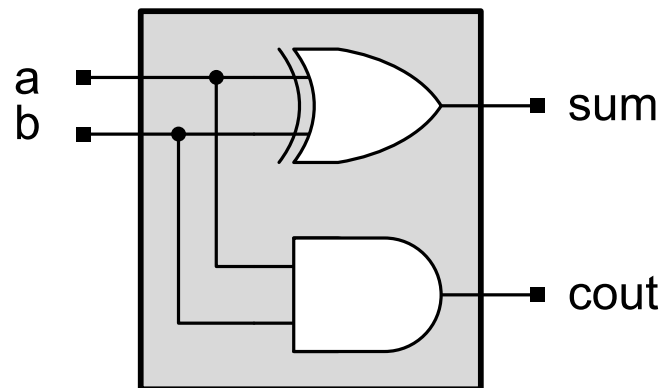


a\b	0	1
0	0	1
1	1	0

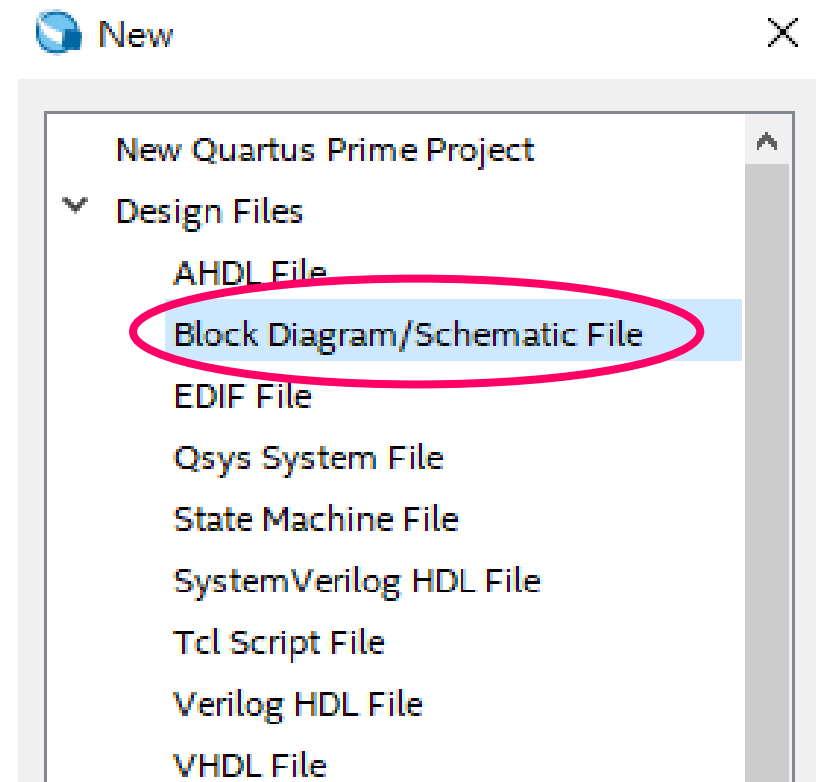
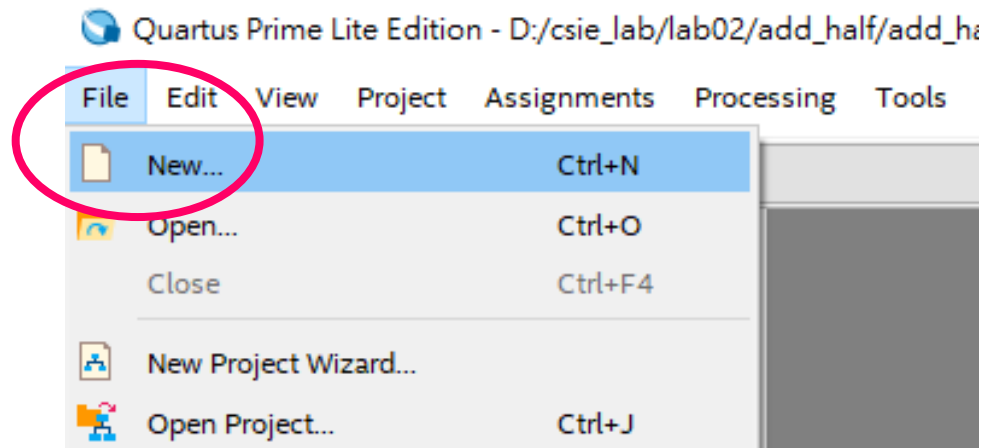
$$\text{sum} = a \oplus b$$

a\b	0	1
0	0	0
1	0	1

$$\text{cout} = a \cdot b$$



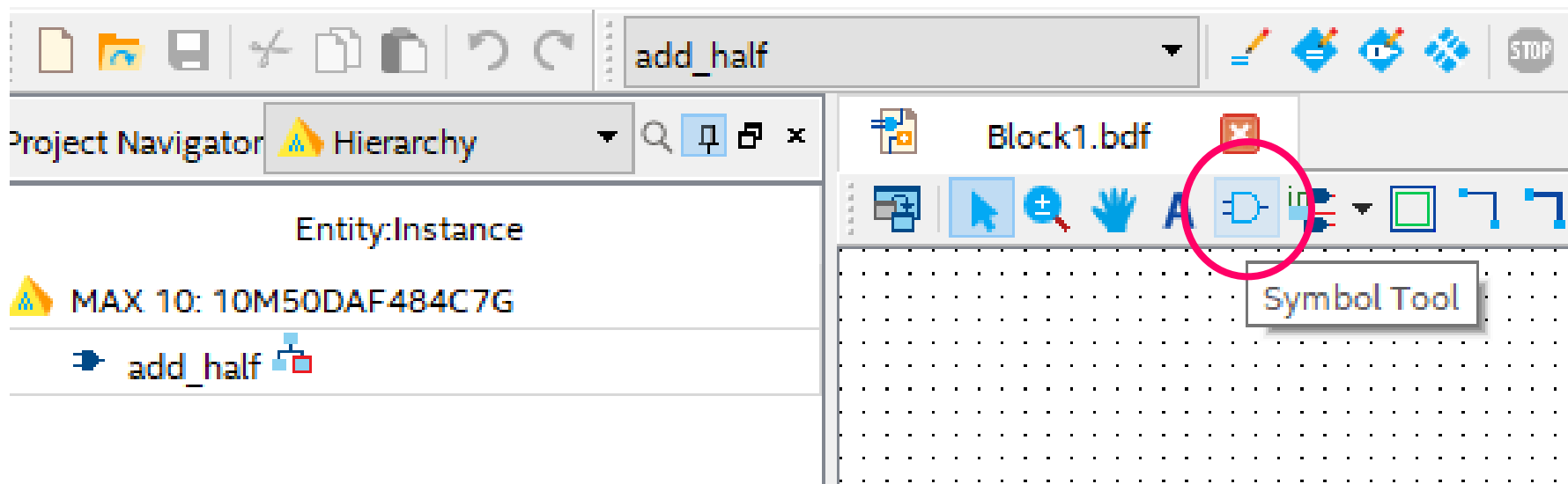
Use Block Editor for Schematic Design



Import Logic Gates

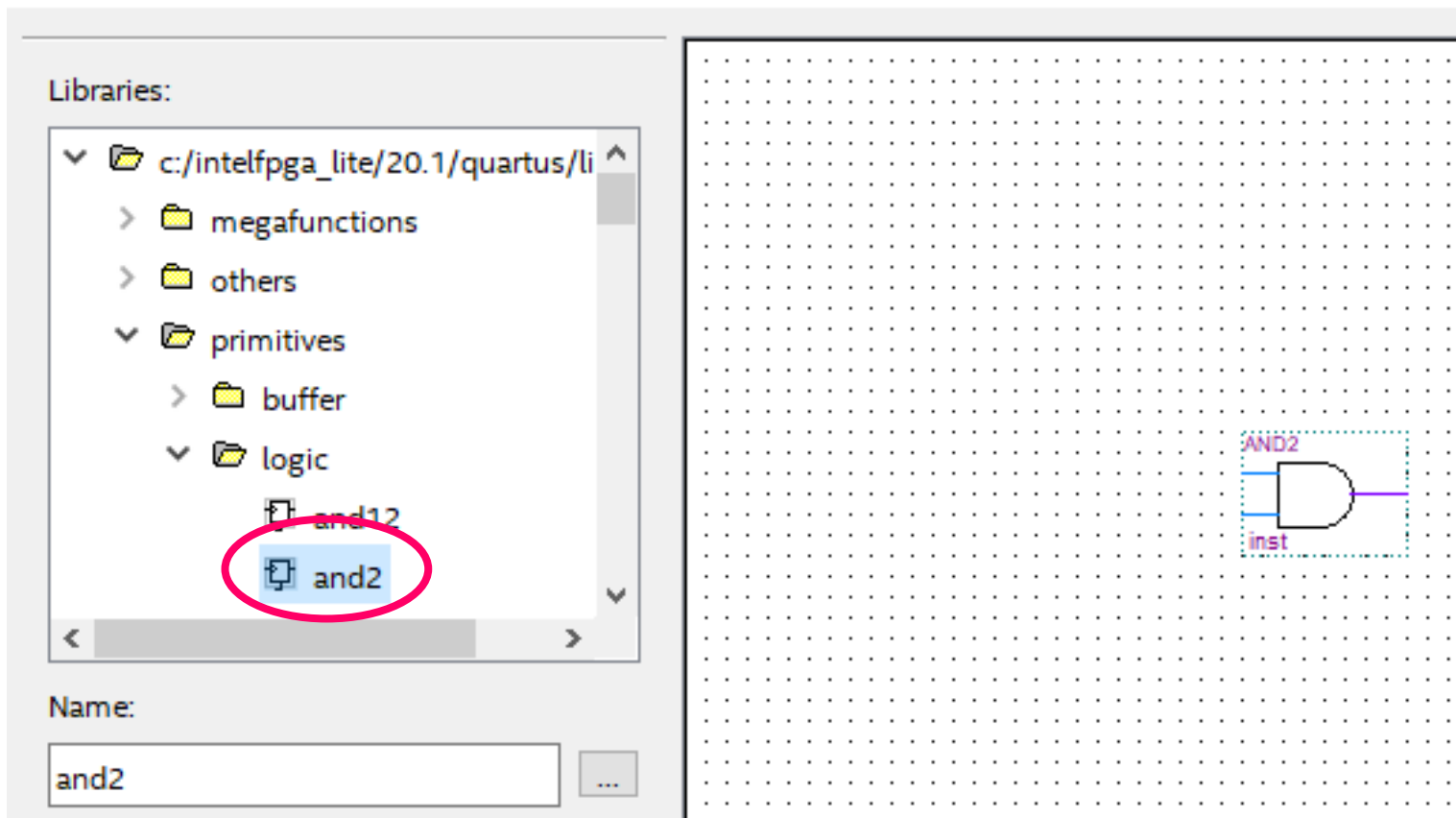
Quartus Prime Lite Edition - D:/csie_lab/lab02/add_half/add_half - add_half

File Edit View Project Assignments Processing Tools Window Help

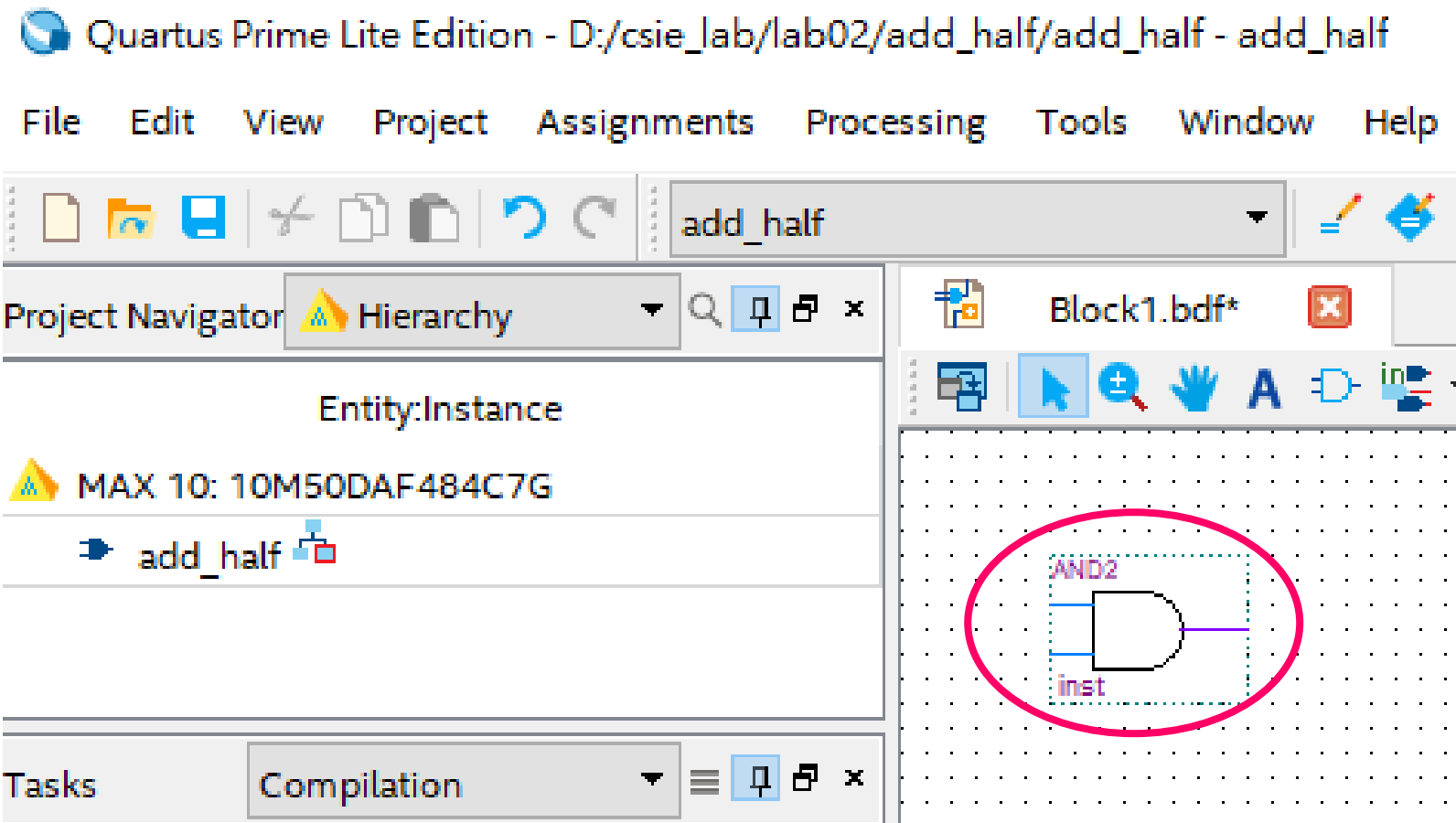


Select and Import an AND2 Gate

Symbol



An AND2 Gate is Imported



Select and Import an XOR Gate

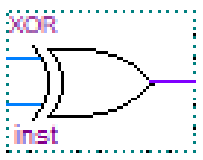
 Symbol

Libraries:

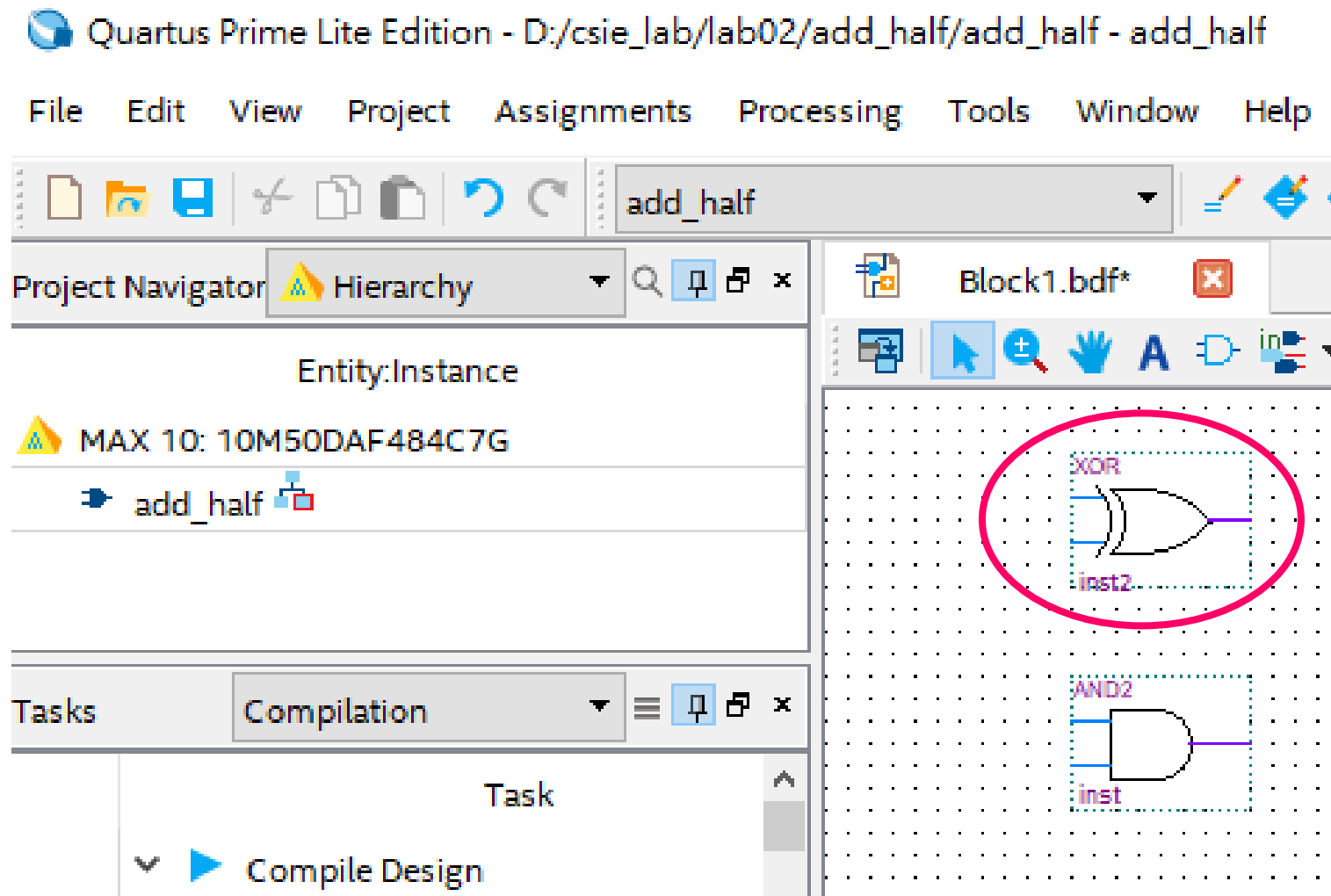
- or3
- or4
- or6
- or8
- xnor
- xor**
- other
- ...

Name:

xor



An XOR Gate is Imported

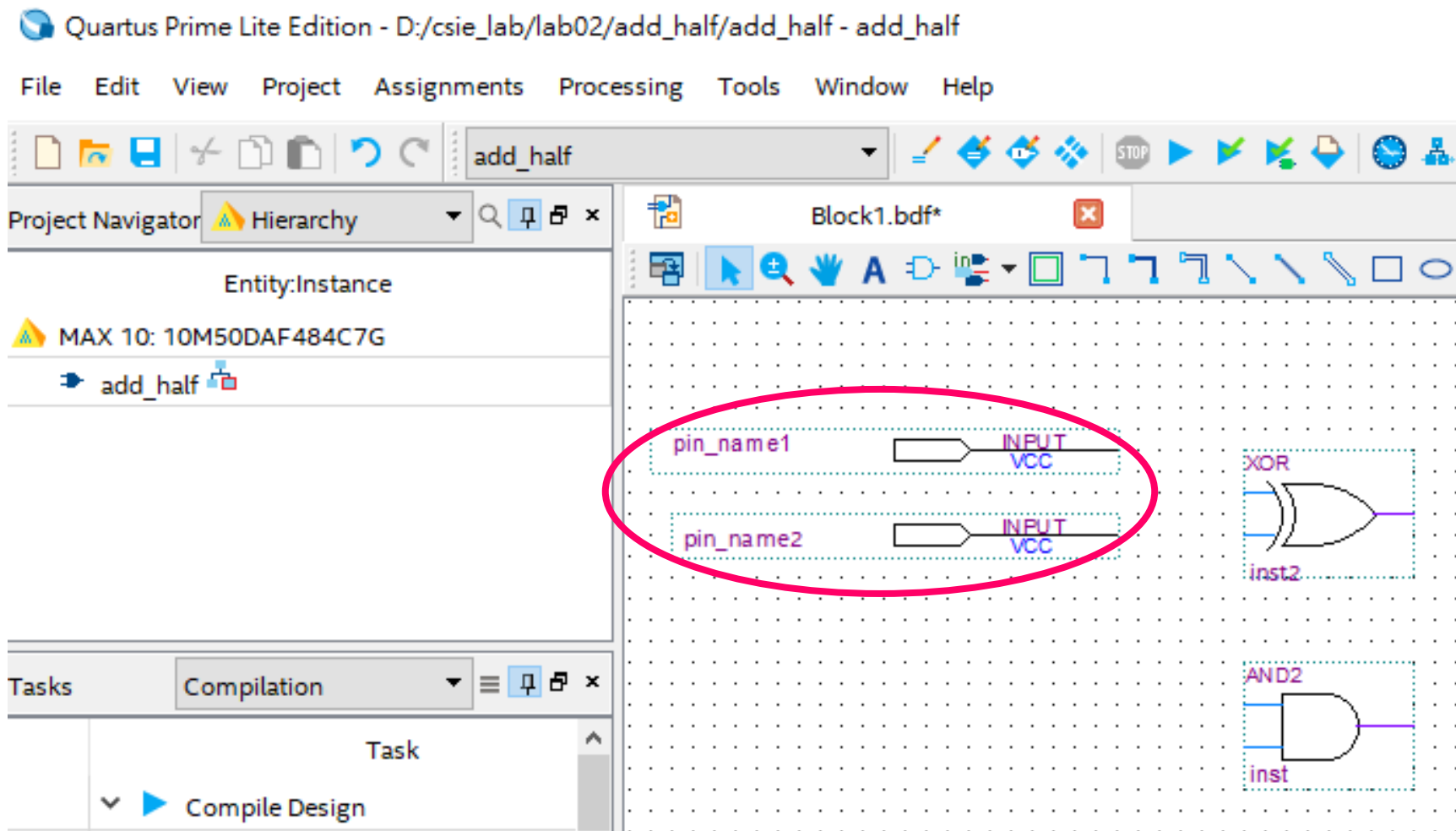


Select and Import an Input Pin

Symbol

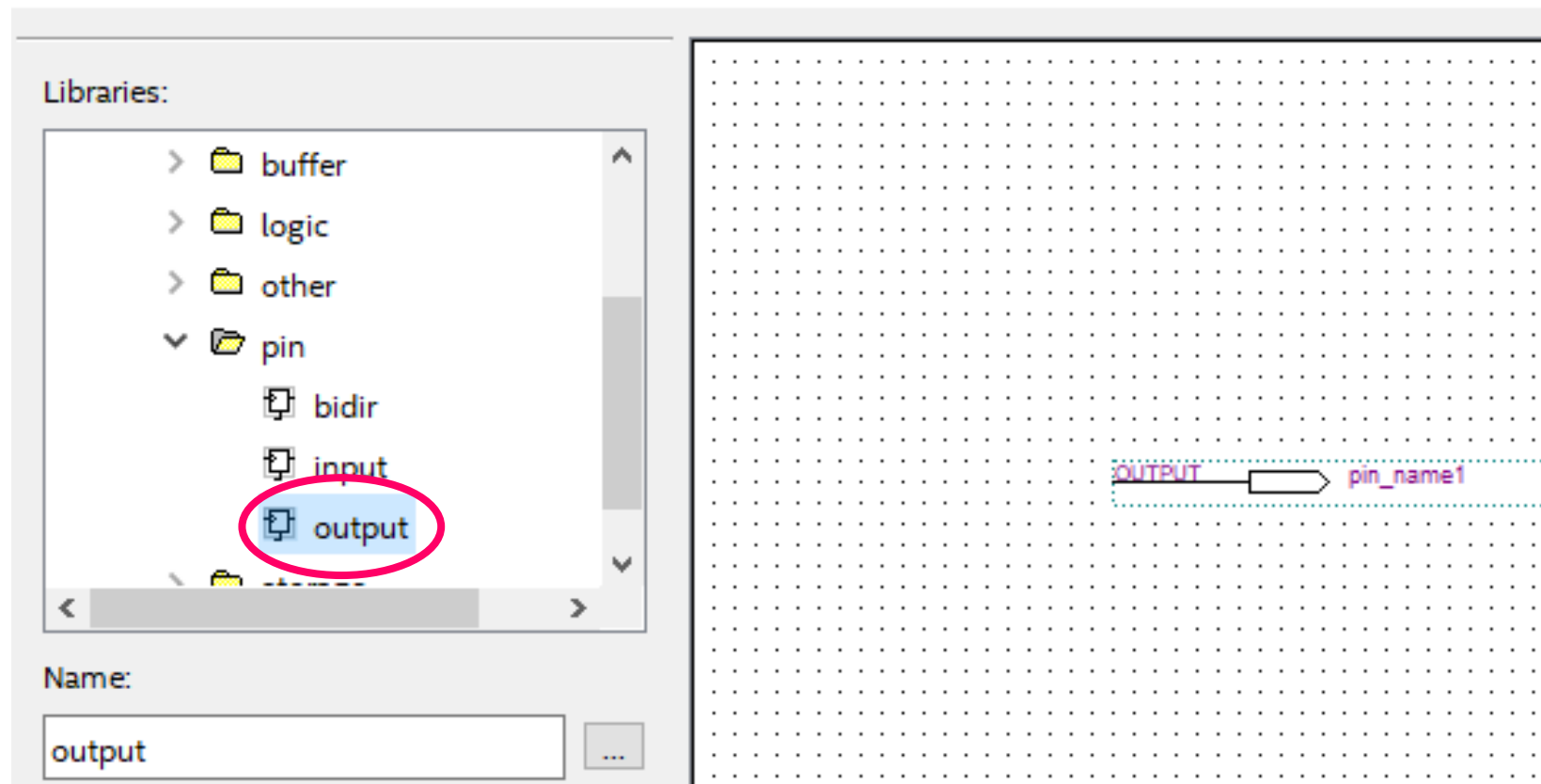


Two Input Pins are Imported

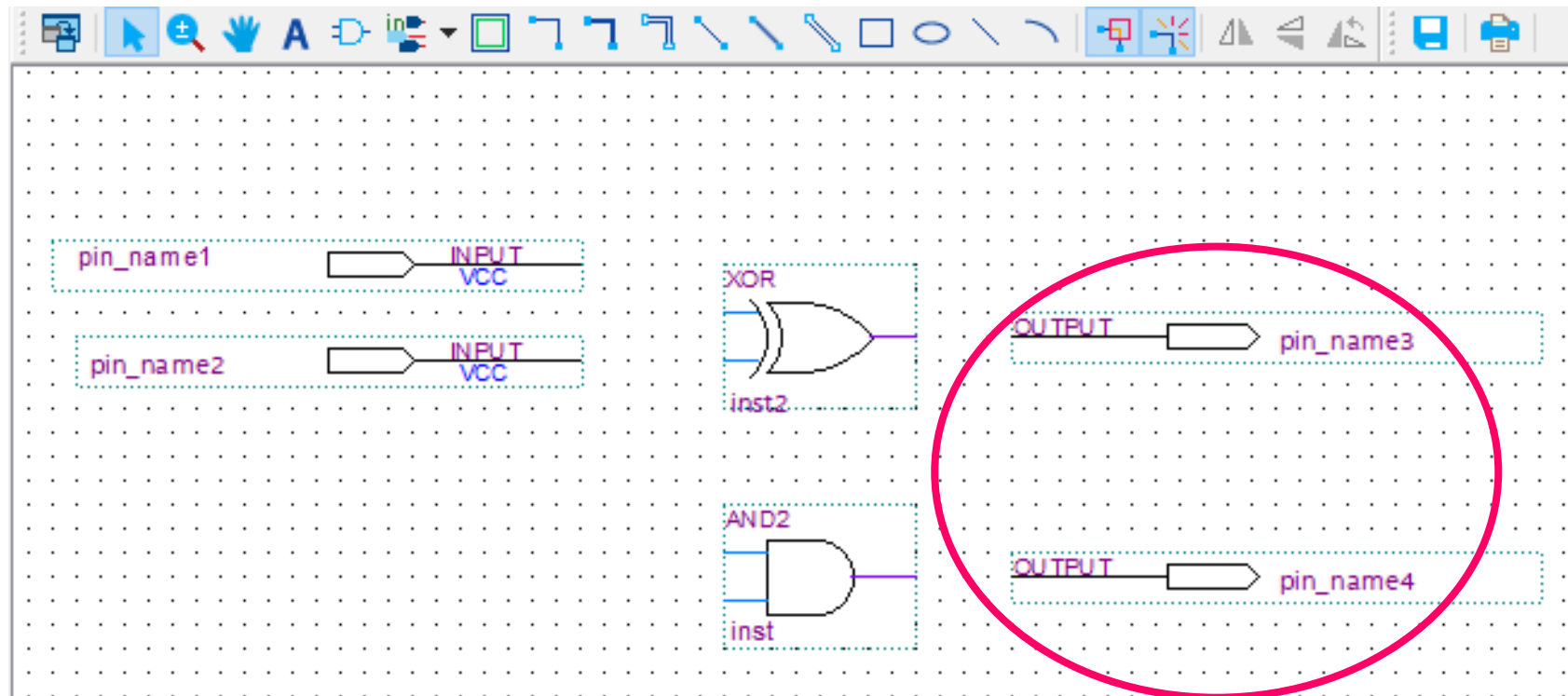


Select and Import an Output Pin

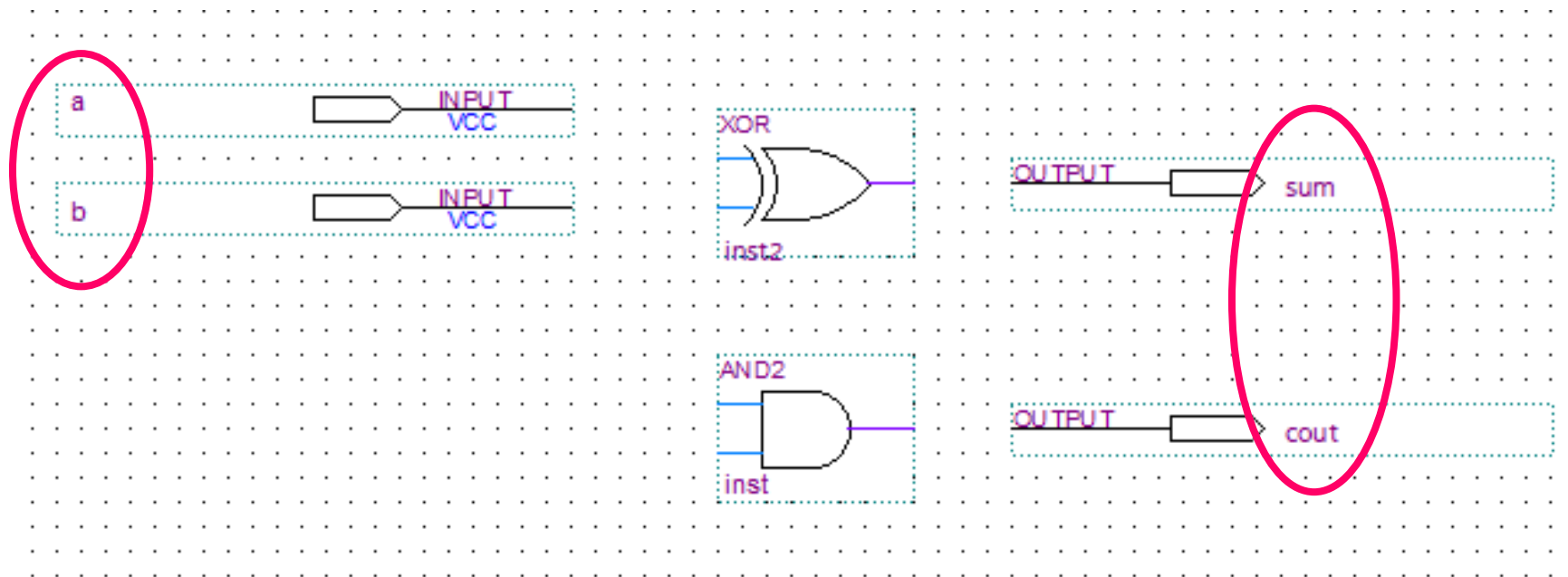
Symbol



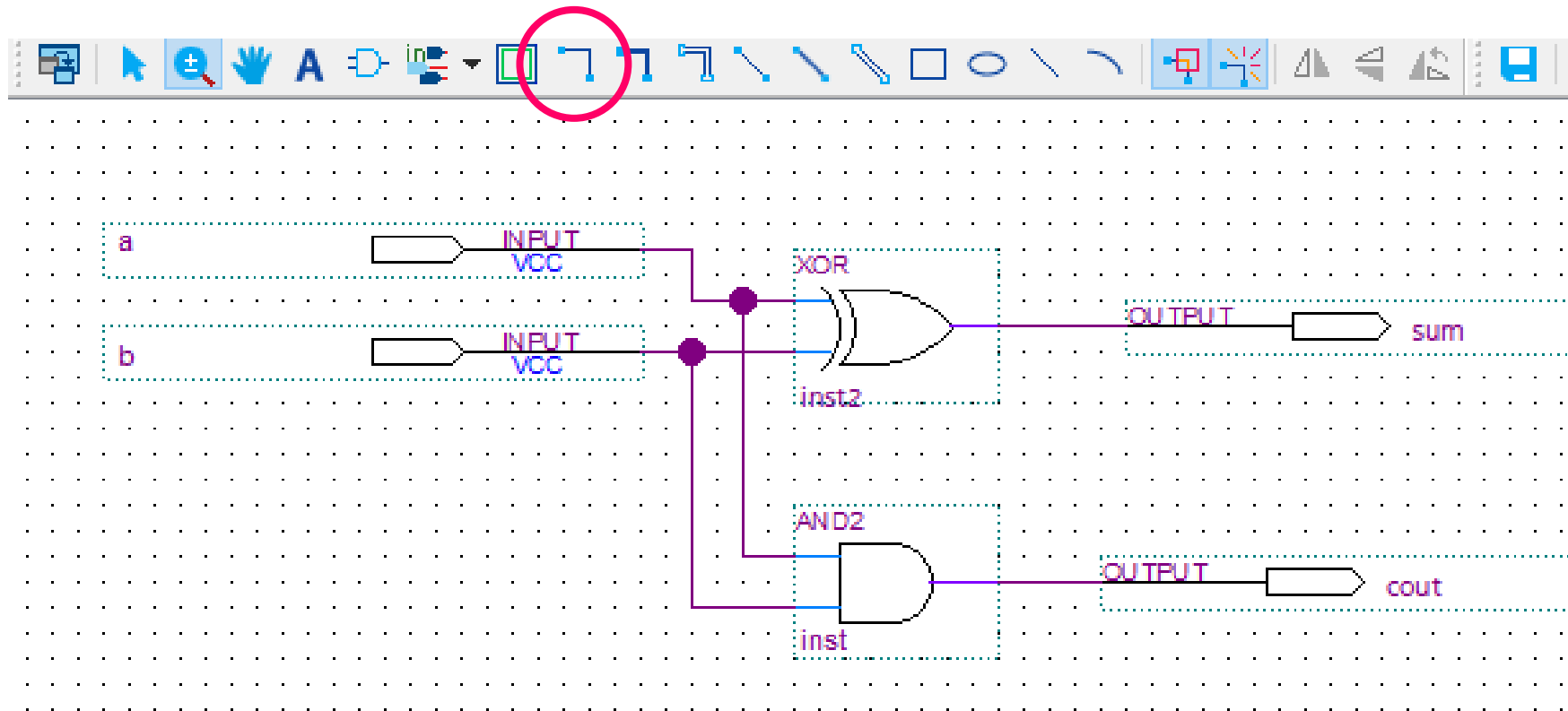
Two Output Pins are Imported



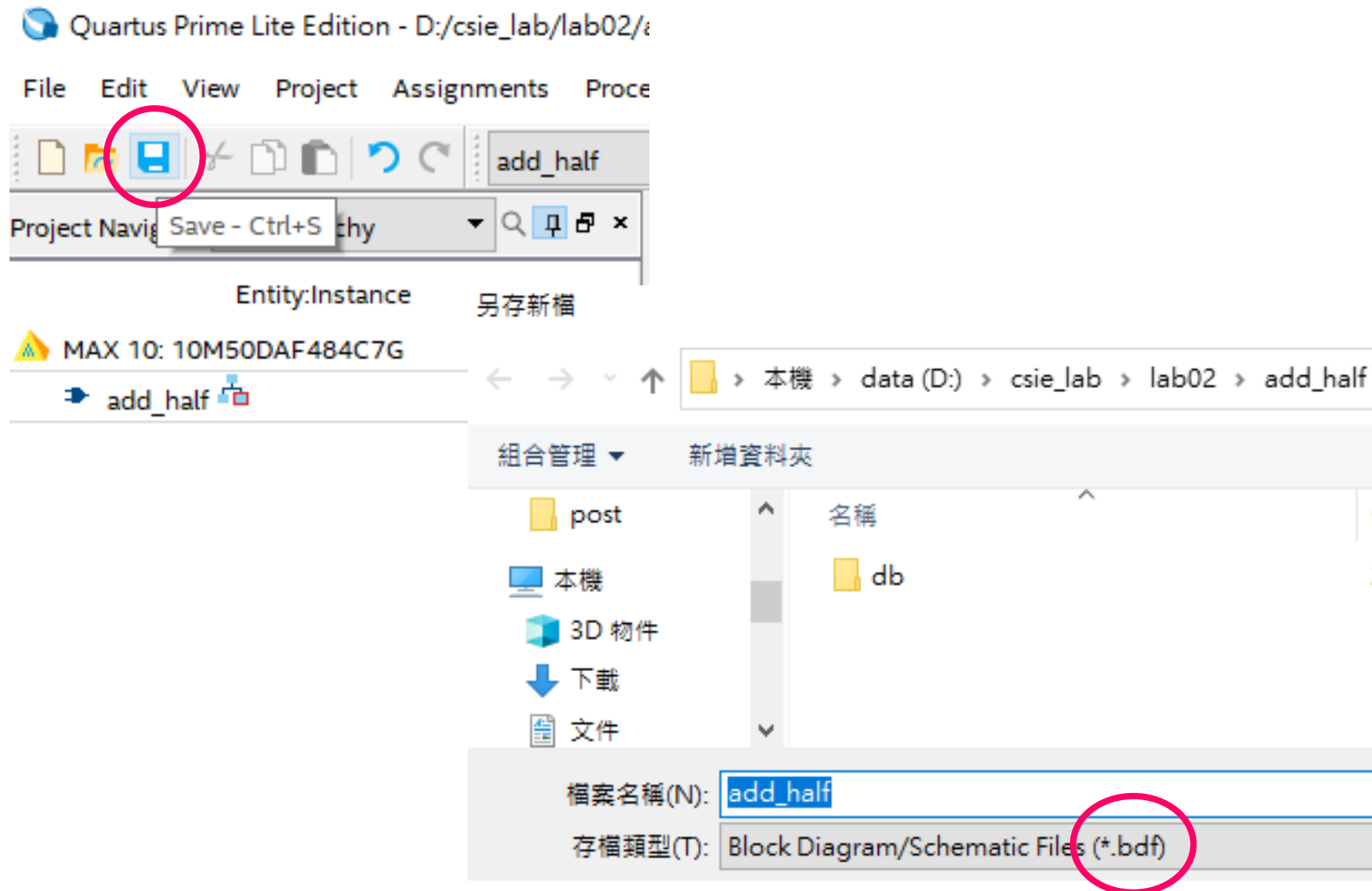
Assign Names to I/O Pins



Connect Nodes with Wires



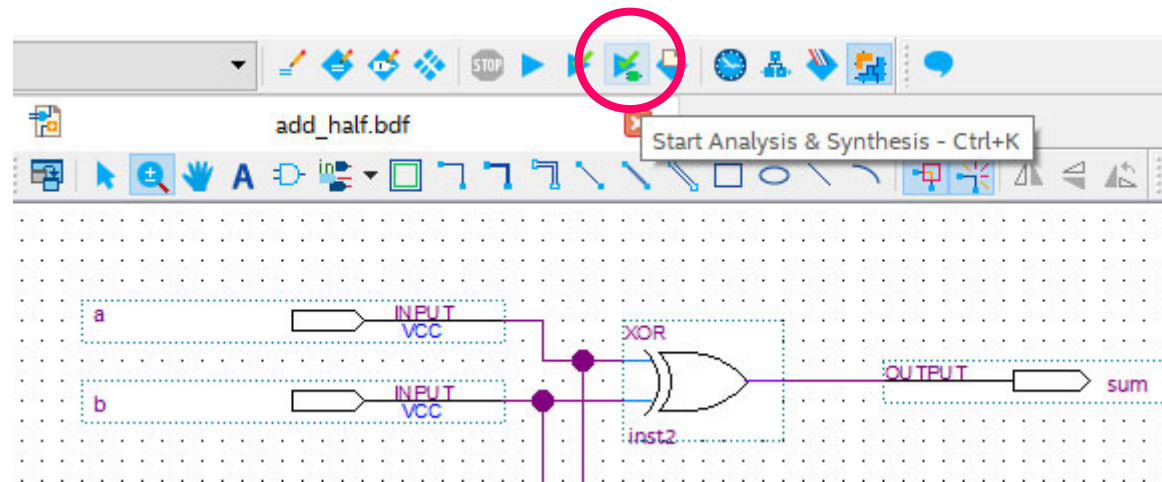
Save the Schematic File



Synthesize the Schematic Circuit

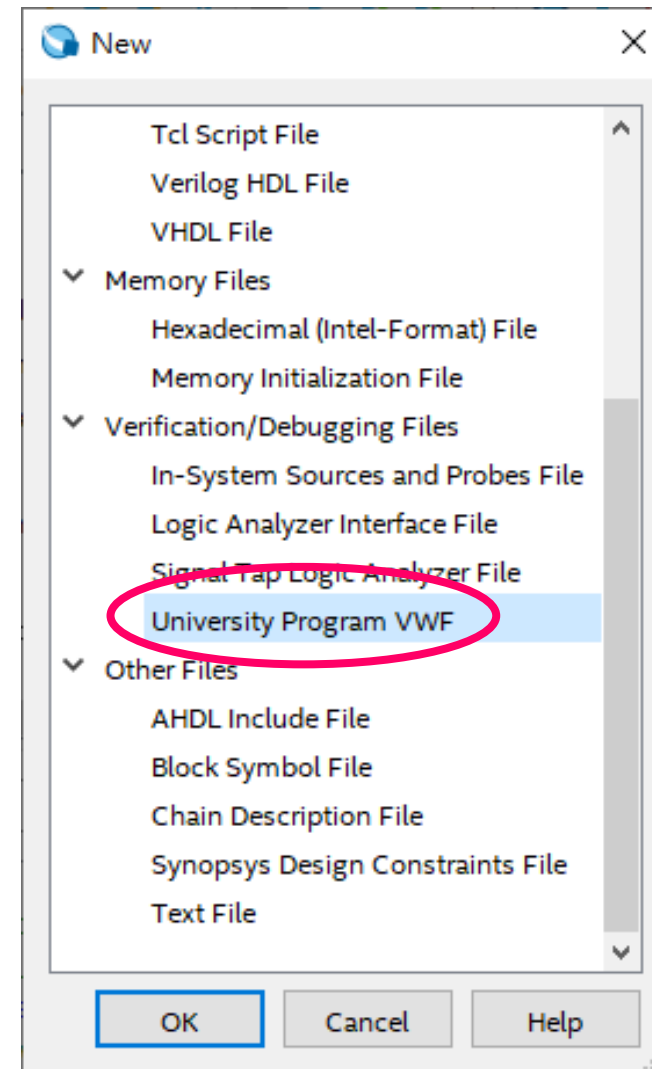
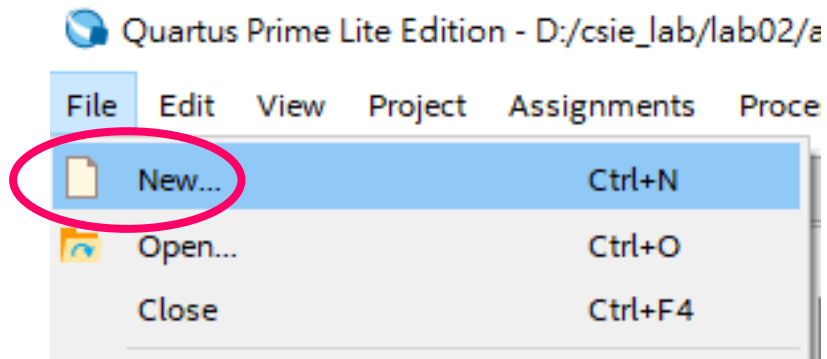
- The **synthesis** tool translates the schematic into logic expressions.
- Then the **technology mapping** step determines how each logic expression should be implemented in the **logic elements** (LEs) available in the target chip.
- The ***Analysis & Synthesis*** function performs the synthesis step.
- It produces a circuit of logic elements, where each element can be directly implemented in the target chip.

Run “Analysis & Synthesis” for Functional Simulation



Type	ID	Message
> i		Running Quartus Prime Analysis & Synthesis
> i		Command: quartus_map --read_settings_files=on --write_settings_files=off add_half -c add_half
> !	18236	Number of processors has not been specified which may cause overloading on shared machines. !
> i	20030	Parallel compilation is enabled and will use 8 of the 8 processors detected
> i	12021	Found 1 design units, including 1 entities, in source file add_half.bdf
> i	12127	Elaborating entity "add_half" for the top level hierarchy
> i	286030	Timing-Driven Synthesis is running
> i	16010	Generating hard_block partition "hard_block:auto_generated_inst"
> i	21057	Implemented 6 device resources after synthesis - the final resource count might be different
> i		Quartus Prime Analysis & Synthesis was successful. 0 errors, 1 warning

Create a Waveform File



Set End Time for Simulation

The image shows the Simulation Waveform Editor interface. The 'Edit' menu is open, and the 'Set End Time...' option is highlighted. A dialog box titled 'End Time' is displayed, showing 'End Time: 200' ns. The waveform editor below shows a time axis with a '200.0 ns' marker highlighted.

Simulation Waveform Editor - D:/csie_

File Edit View Simulation Help

Delete

Insert

Value

Grouping

Reverse Group or Bus Bit Order

Radix

Grid Size...

Set End Time...

Snap to Grid

Span to Transition

End Time

Set End Time

End Time: 200 ns

OK Cancel

Simulation Waveform Editor - D:/csie_lab/lab02/add_half/add_half - add_half - [Waveform.vwf]*

File Edit View Simulation Help

Search altera.com

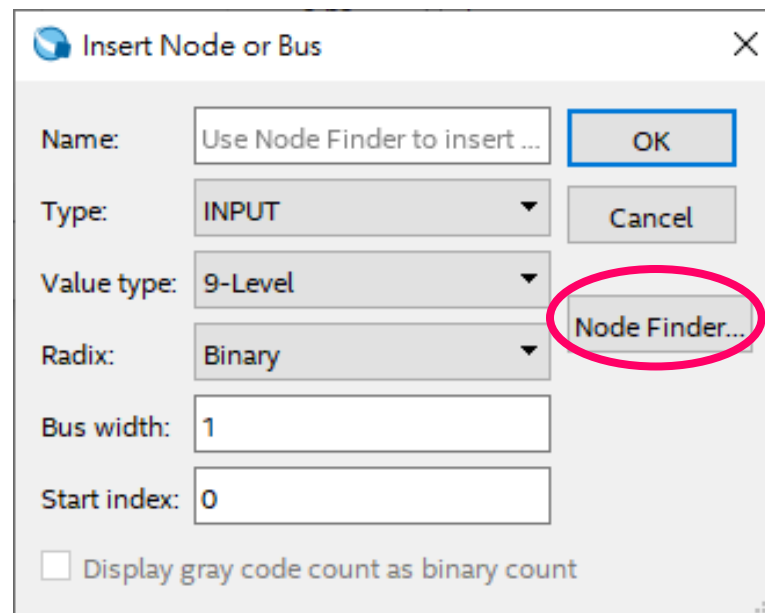
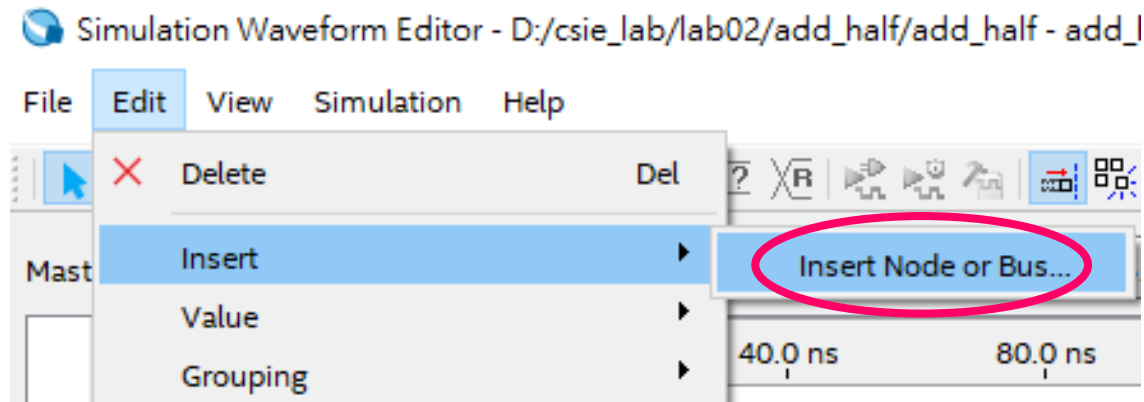
Master Time Bar: 0 ps Pointer: 55.18 ns Interval: 55.18 ns Start: End:

0 ps 40.0 ns 80.0 ns 120.0 ns 160.0 ns **200.0 ns**

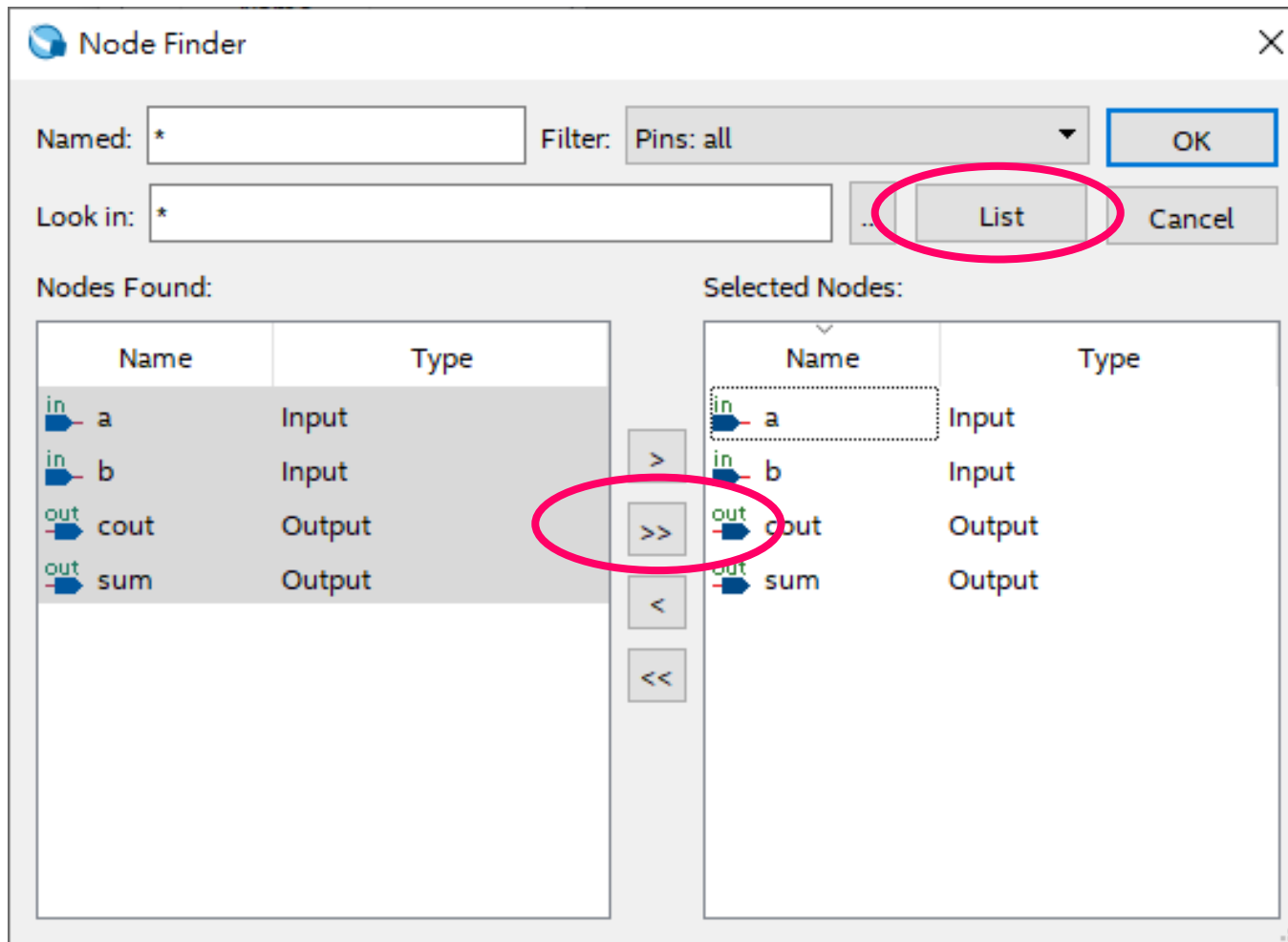
0 ps

0% 00:00:00

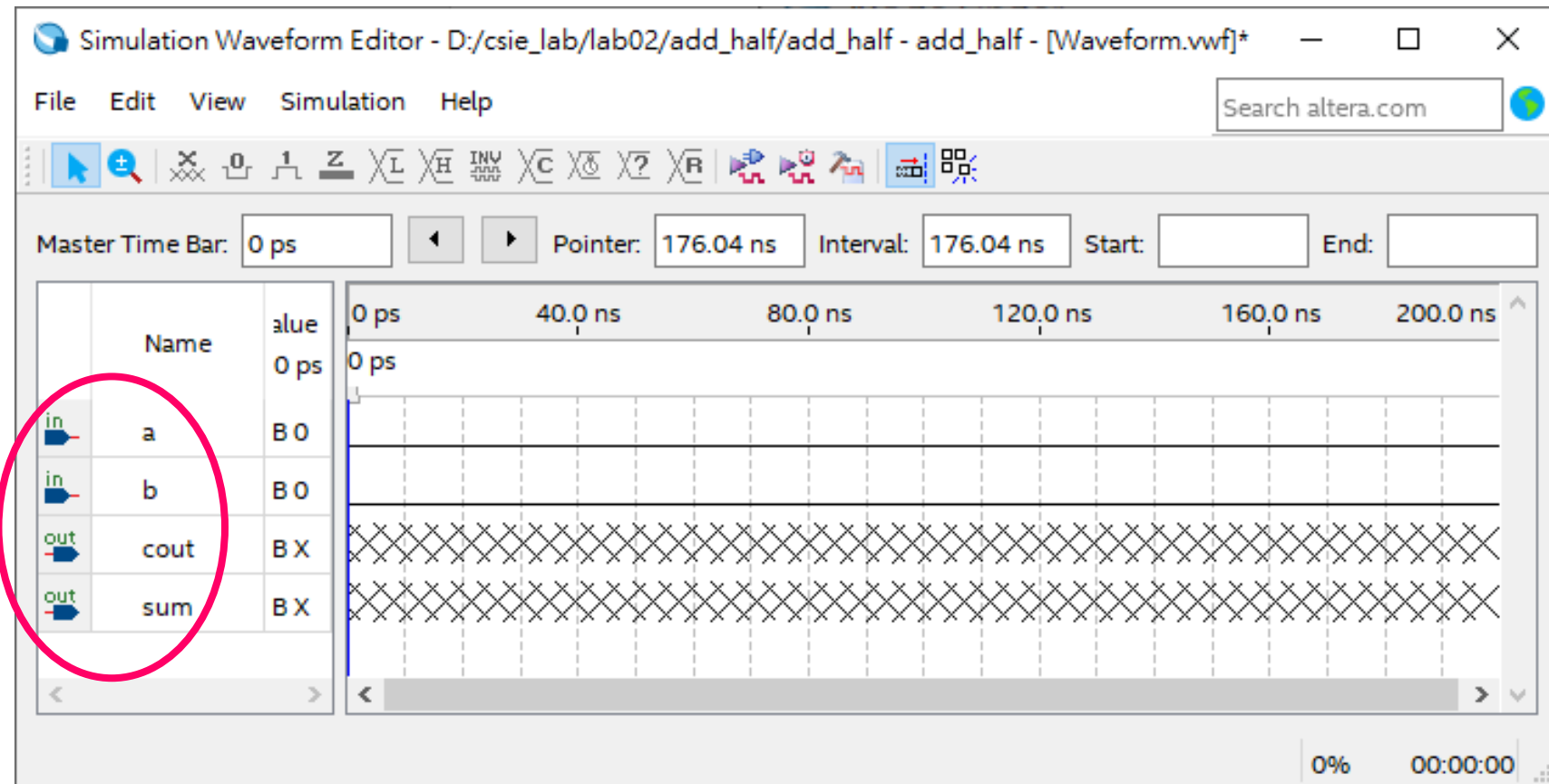
Insert Node for I/O Signals (1/3)



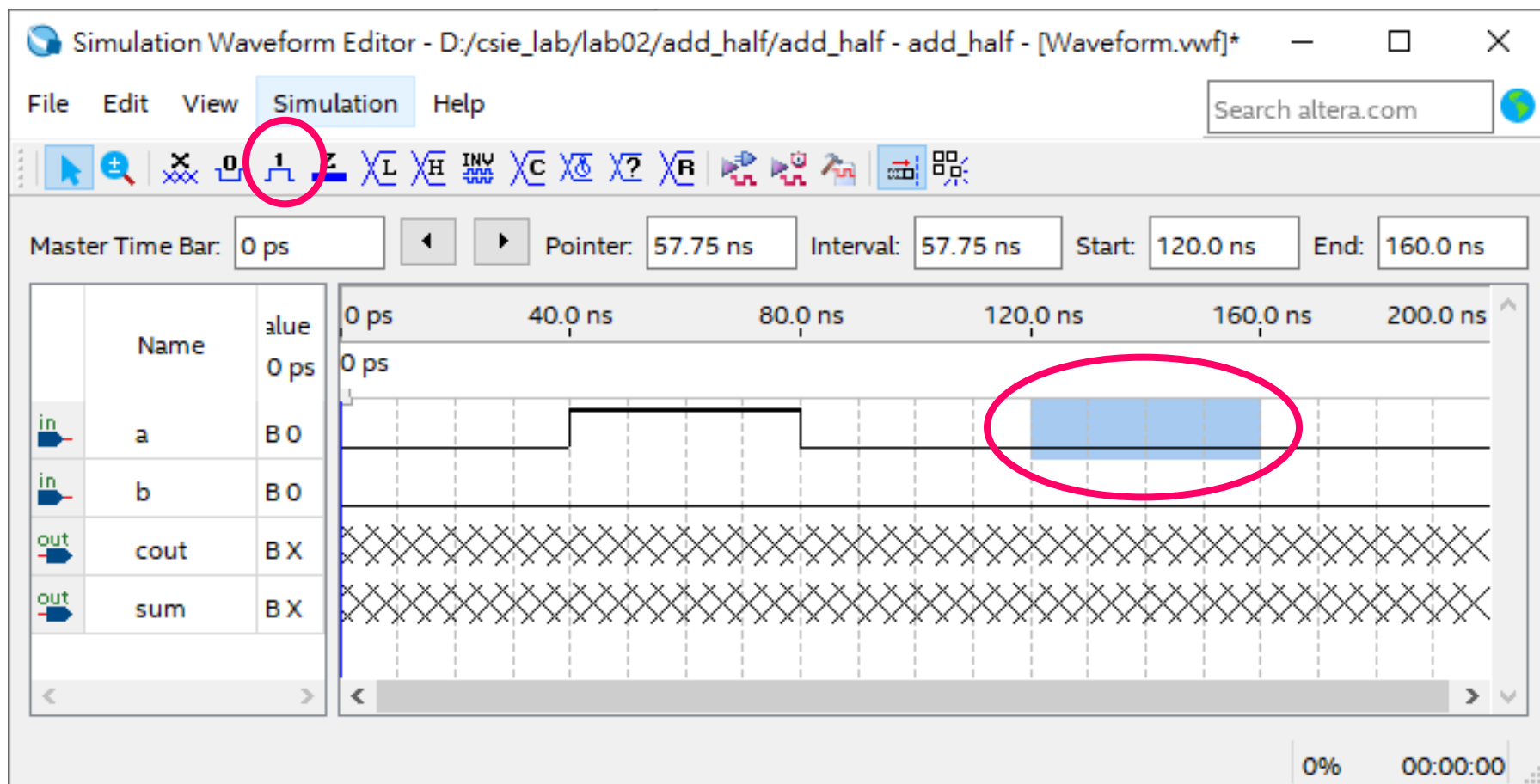
Insert Node for I/O Signals (2/3)



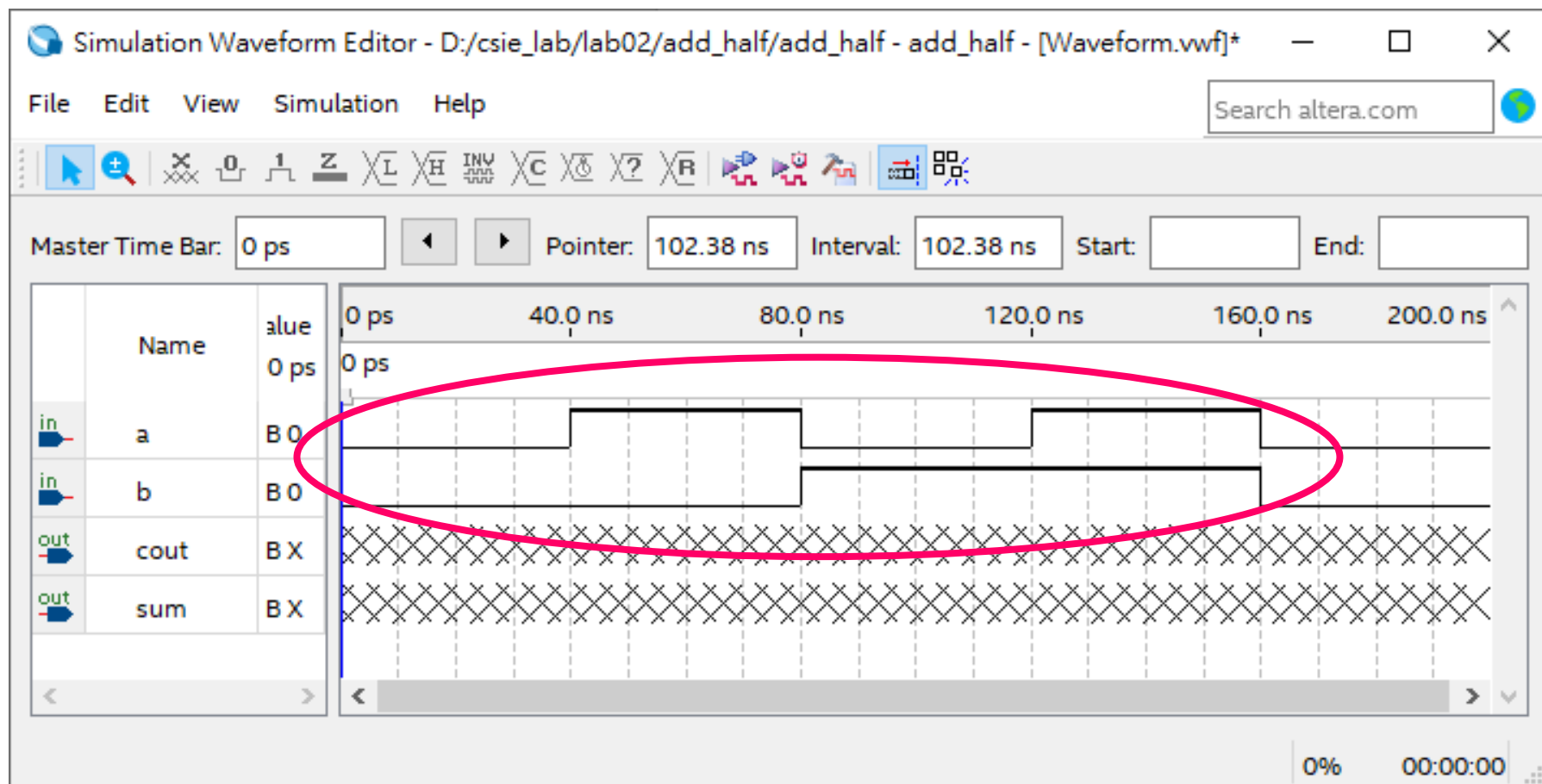
Insert Node for I/O Signals (3/3)



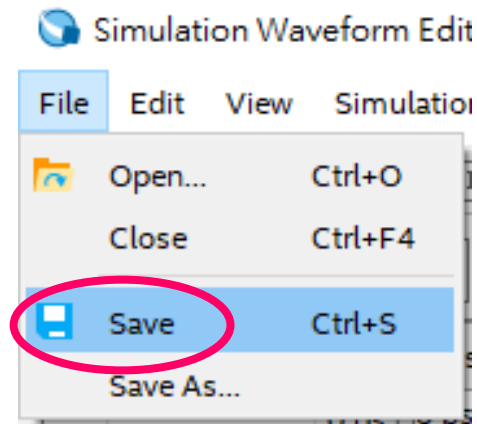
Set Values for Input Signals (1/2)



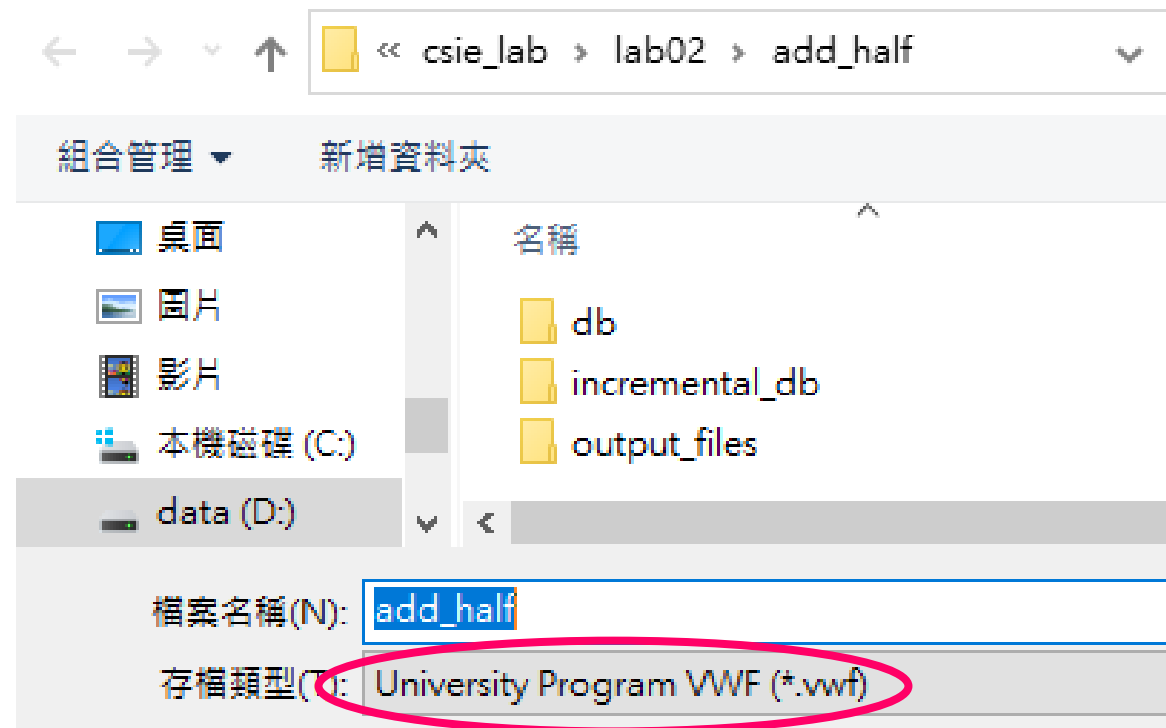
Set Values for Input Signals (2/2)



Save the Waveform File



Save Vector Waveform File



Simulation

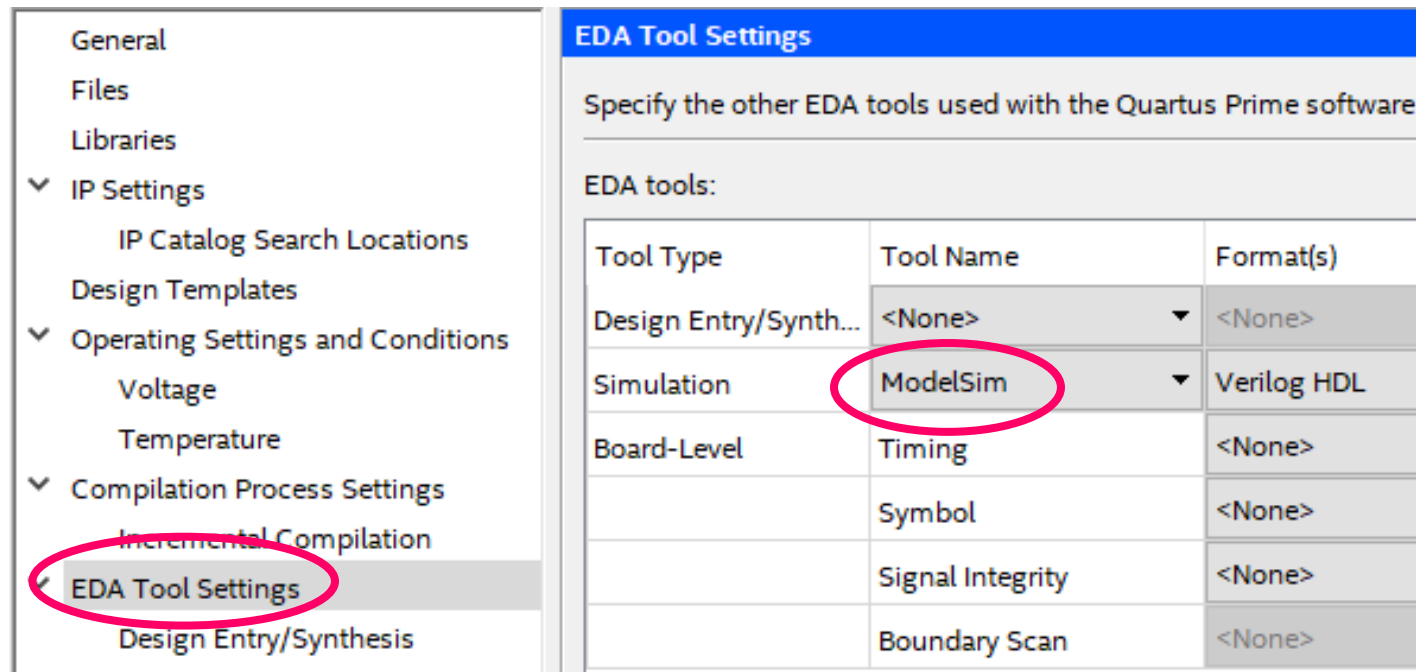
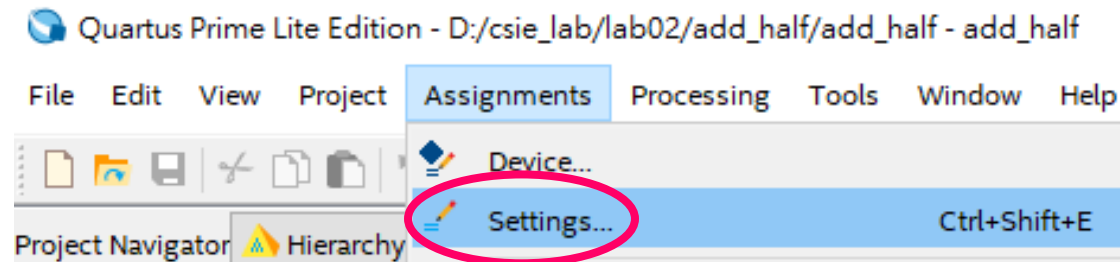
■ Functional simulation

- Logic elements and interconnection wires are perfect, causing no delay in propagation of signals through the circuit.
- Used to verify the functional correctness of a circuit.
- Takes much less time.

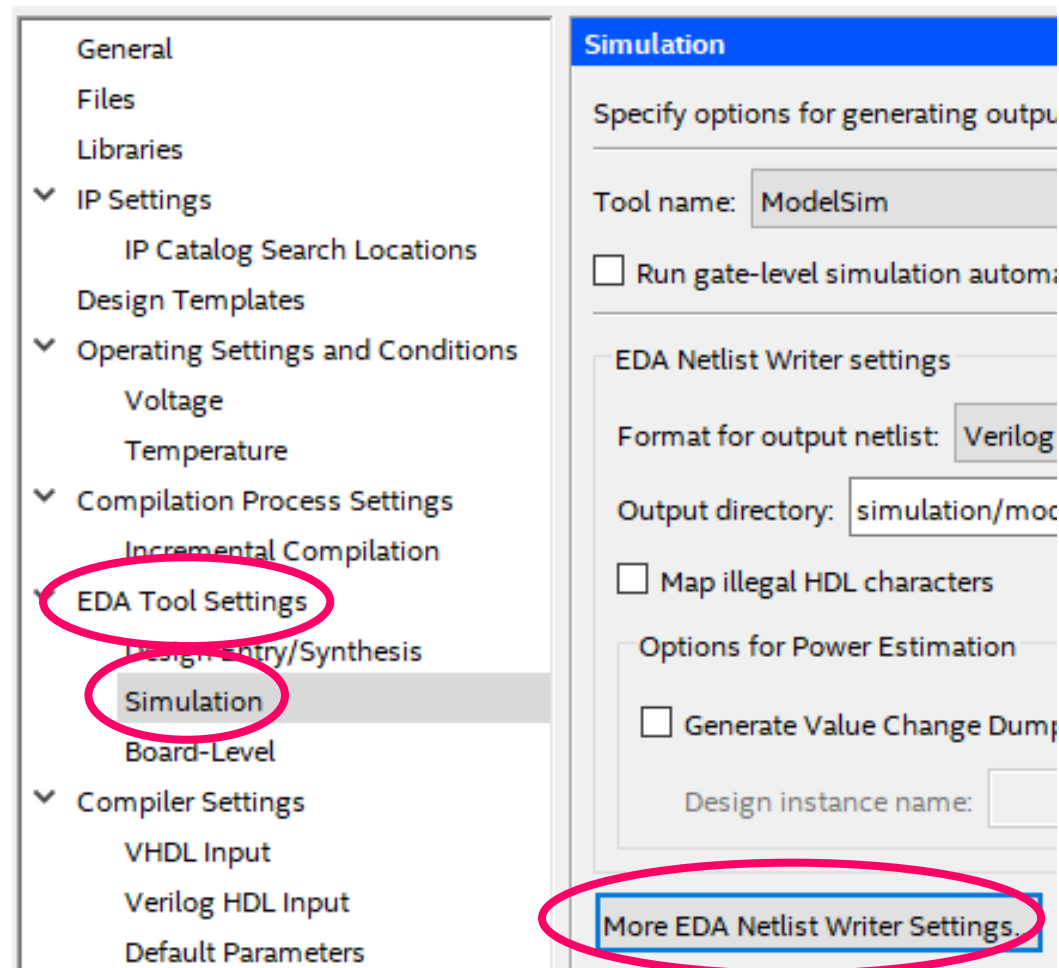
■ Timing simulation

- Takes all propagation delays into account.

Set Simulation Tool - ModelSim



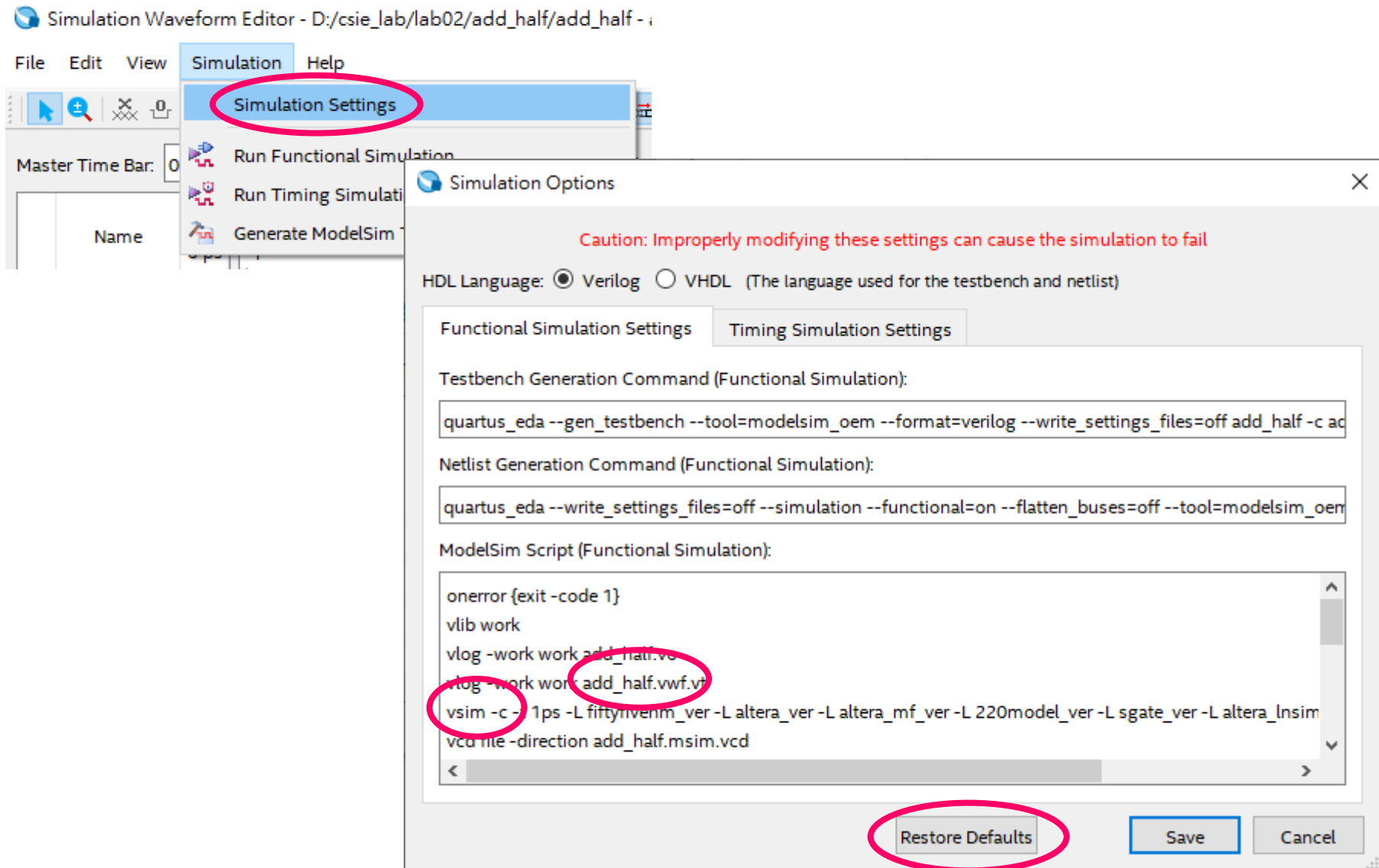
Check Functional Simulation Netlist (1/2)



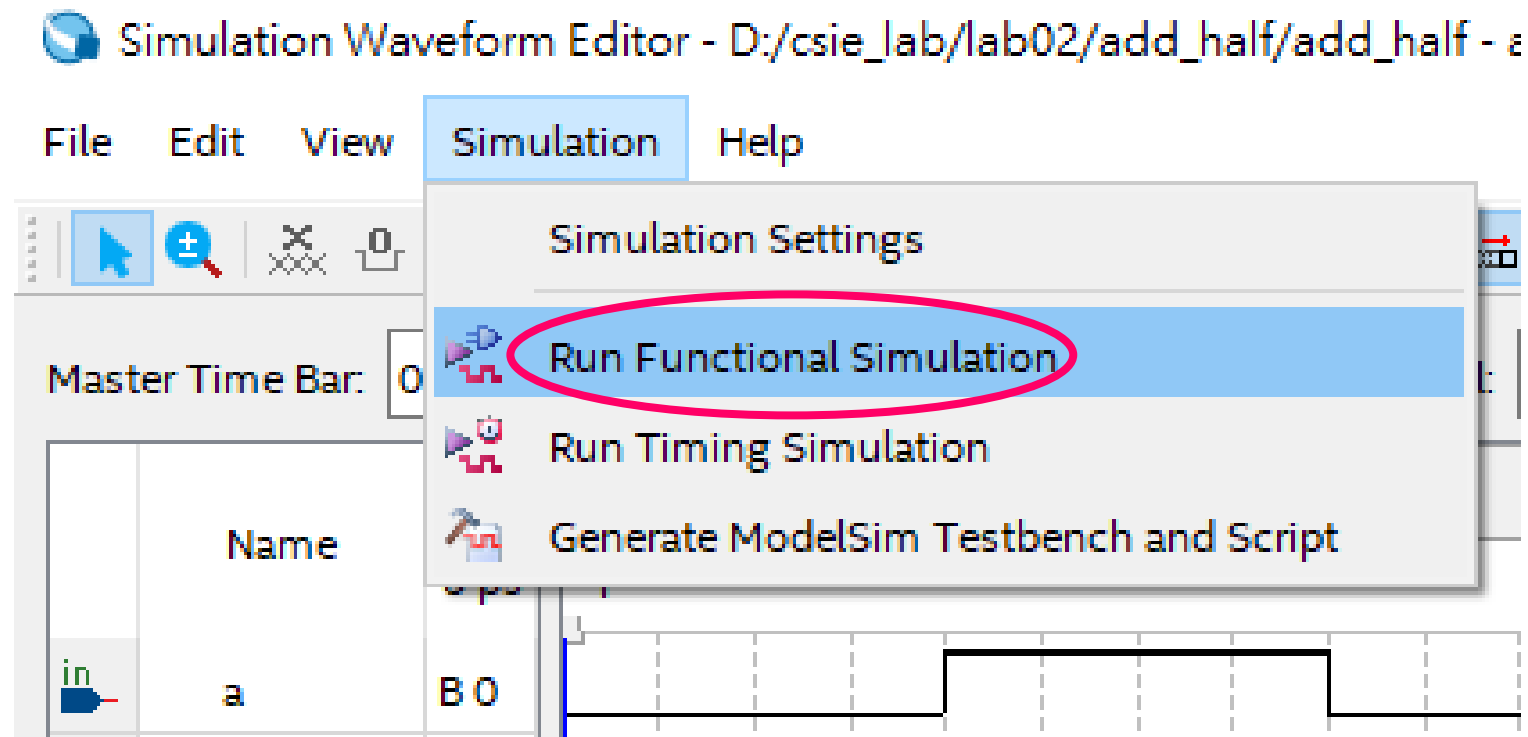
Check Functional Simulation Netlist (2/2)

Name:	Setting:
Architecture name in VHDL output netlist	structure
Bring out device-wide set/reset signals as ports	Off
Disable detection of setup and hold time violations in the input registers of bi-directional pins	Off
Do not write top level VHDL entity	Off
Flatten buses into individual nodes	Off
Generate functional simulation netlist	On
Generate third-party EDA tool command script for RTL functional simulation	Off
Generate third-party EDA tool command script for gate-level simulation	Off
Maintain hierarchy	Off
Truncate long hierarchy paths	Off

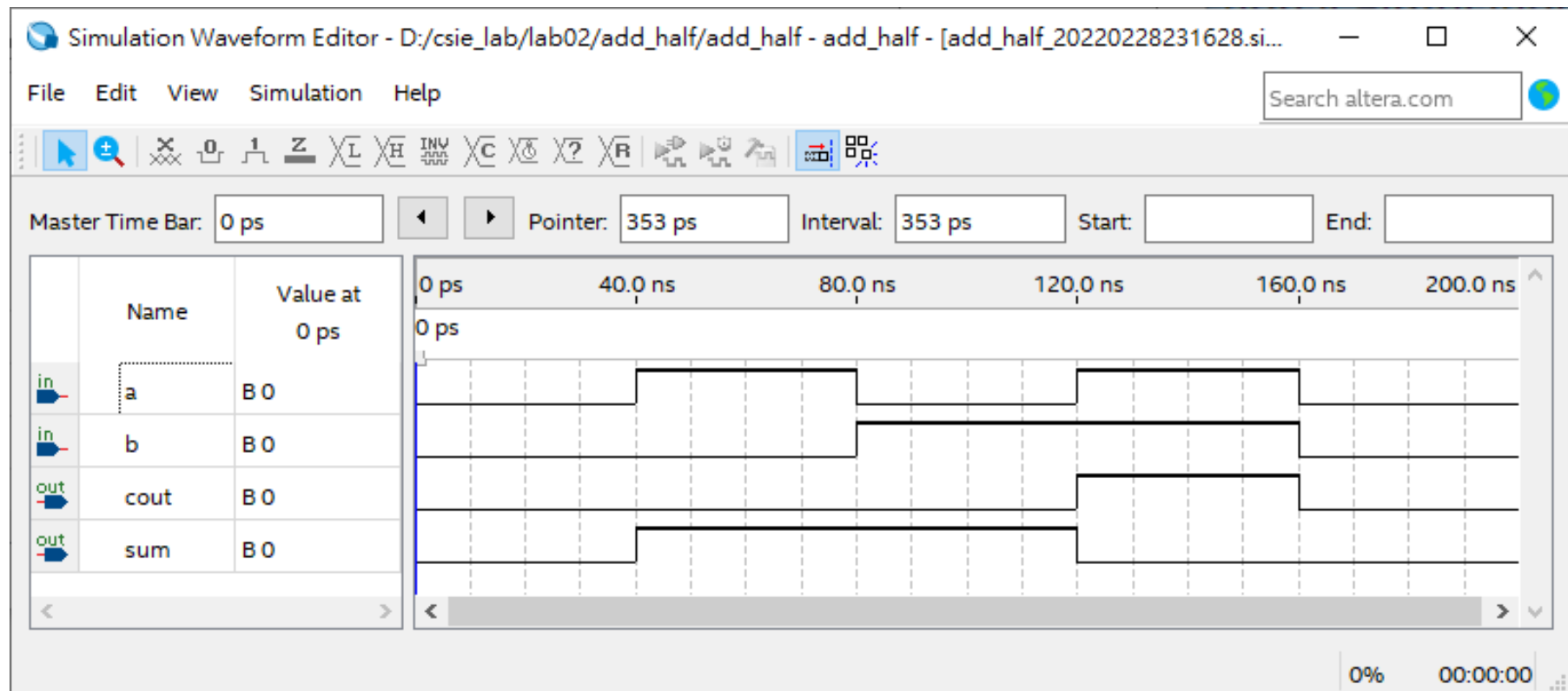
Modify ModelSim Script



Run Functional Simulation



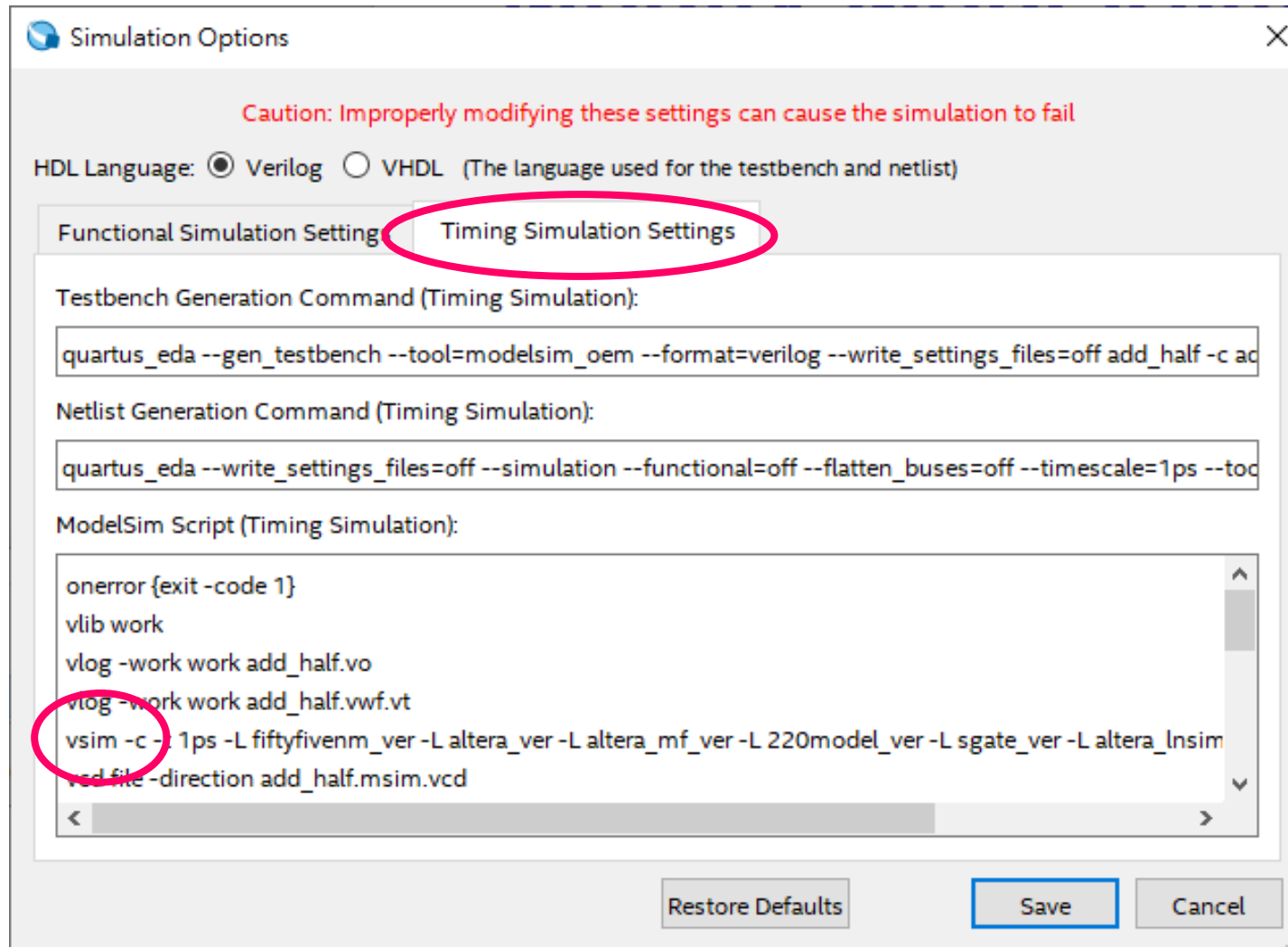
Functional Simulation Result



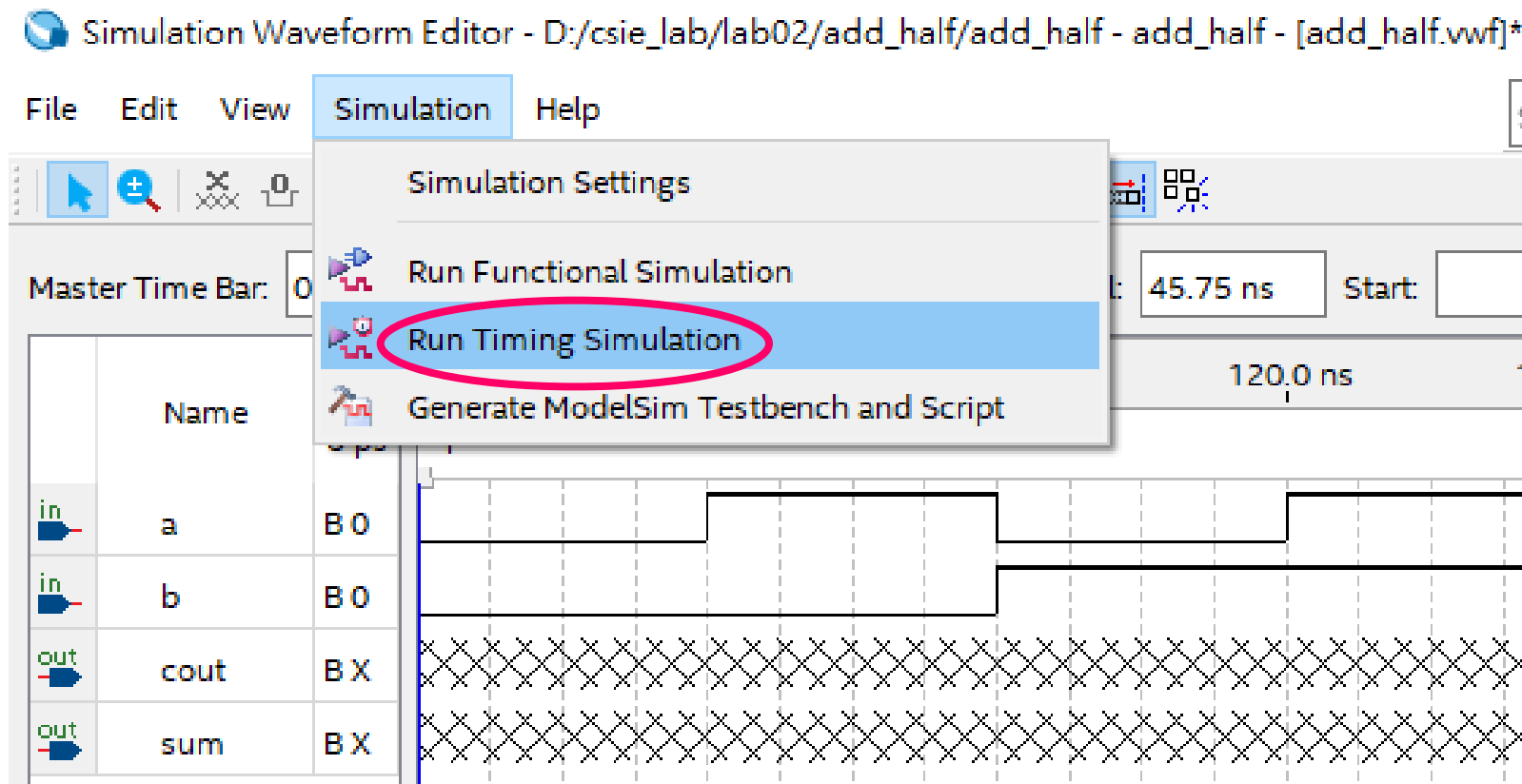
Perform Advanced Compilation for Timing Simulation

The screenshot shows the Quartus Prime IDE interface. The 'Flow Summary' window is open, displaying the compilation status as 'Successful - Mon Feb 28 23:19:2'. The 'Total logic elements' used is 3 / 49,760 (< 1 %), and the 'Total pins' used is 4 / 360 (1 %). The 'Messages' window shows the compilation progress, including the command 'quartus_eda --read_settings_files=off --write_settings_files:' and the final status 'Quartus Prime Full Compilation was successful. 0 errors, 14 warnings'.

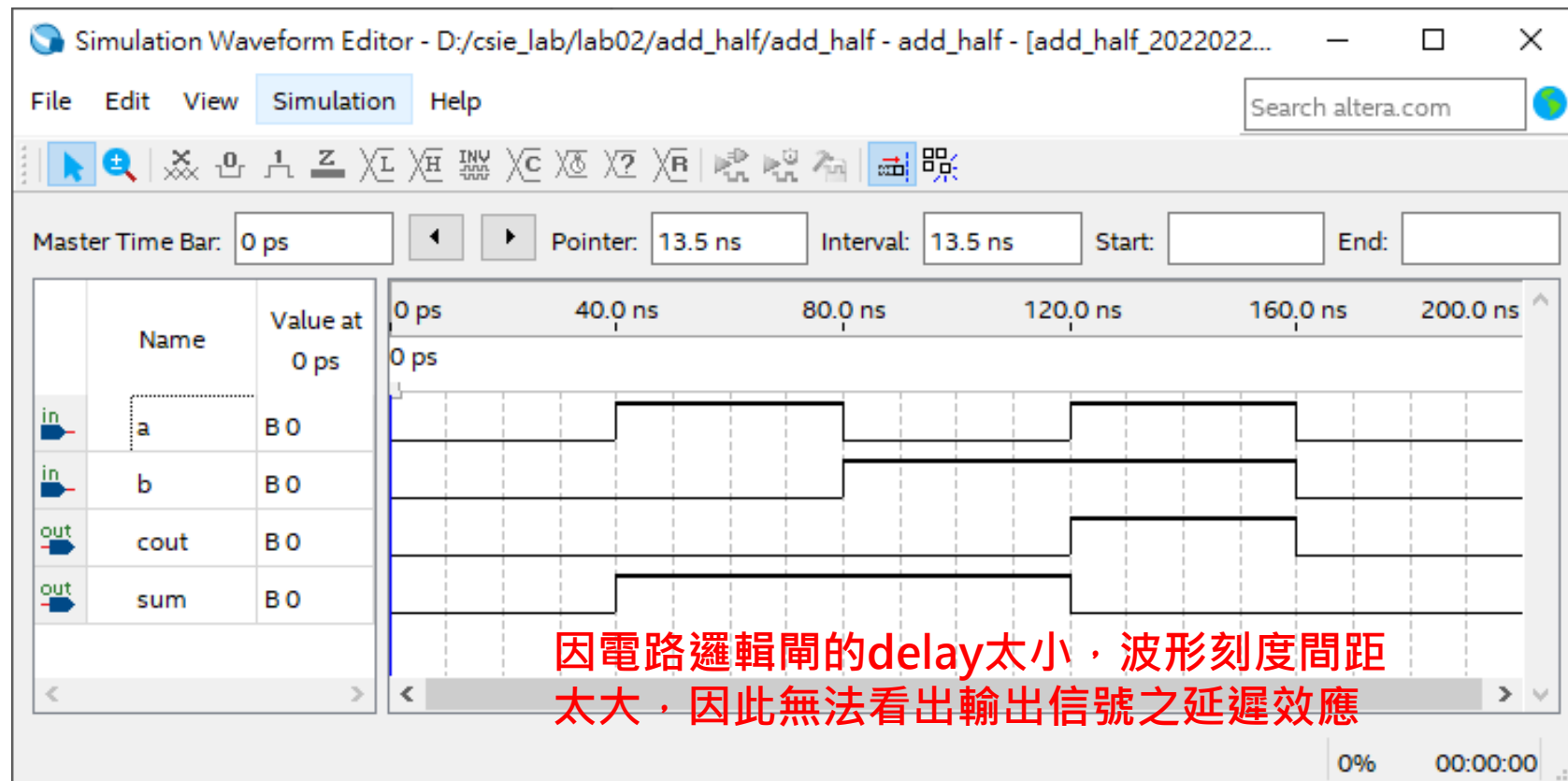
Modify ModelSim Script



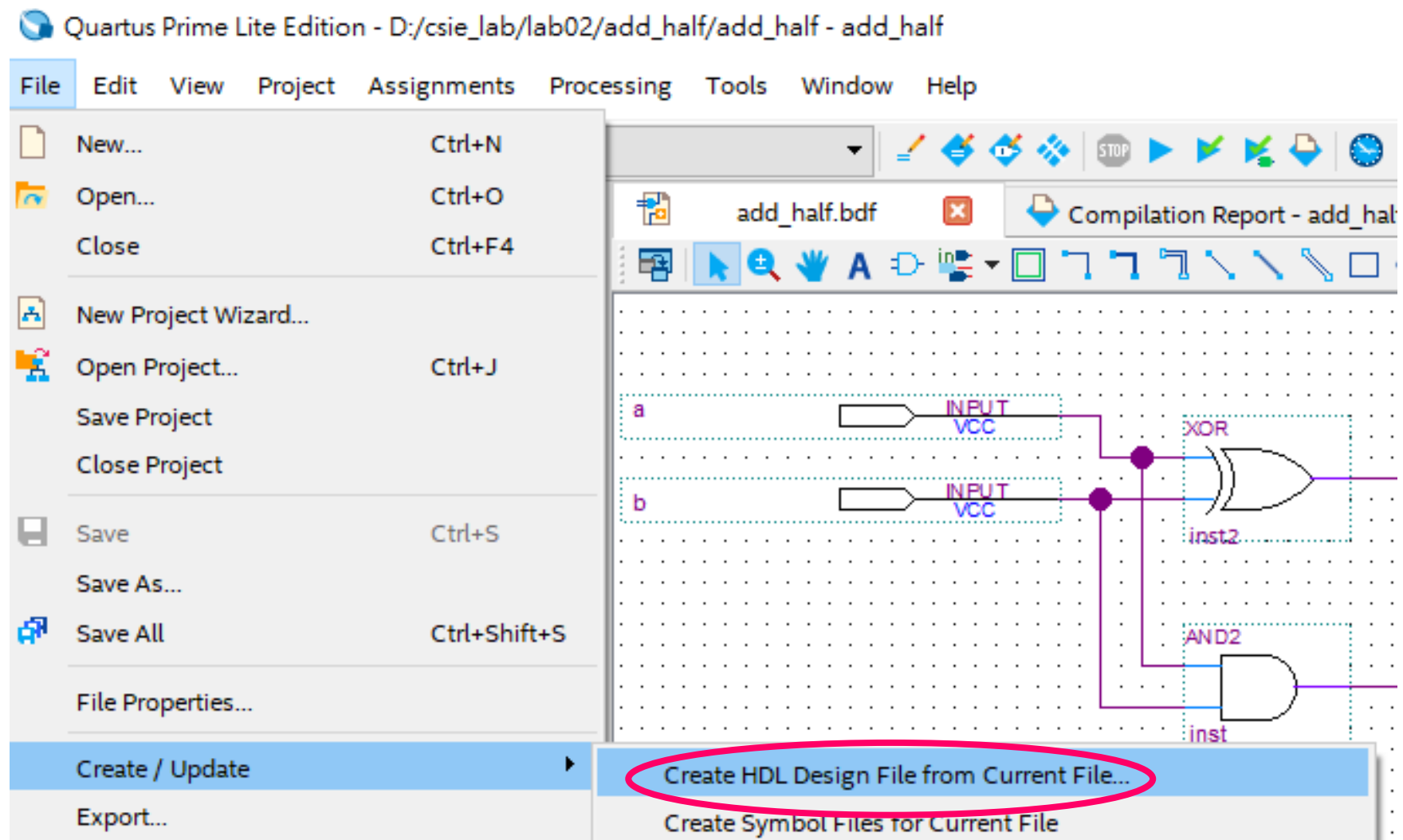
Run Timing Simulation



Timing Simulation Result



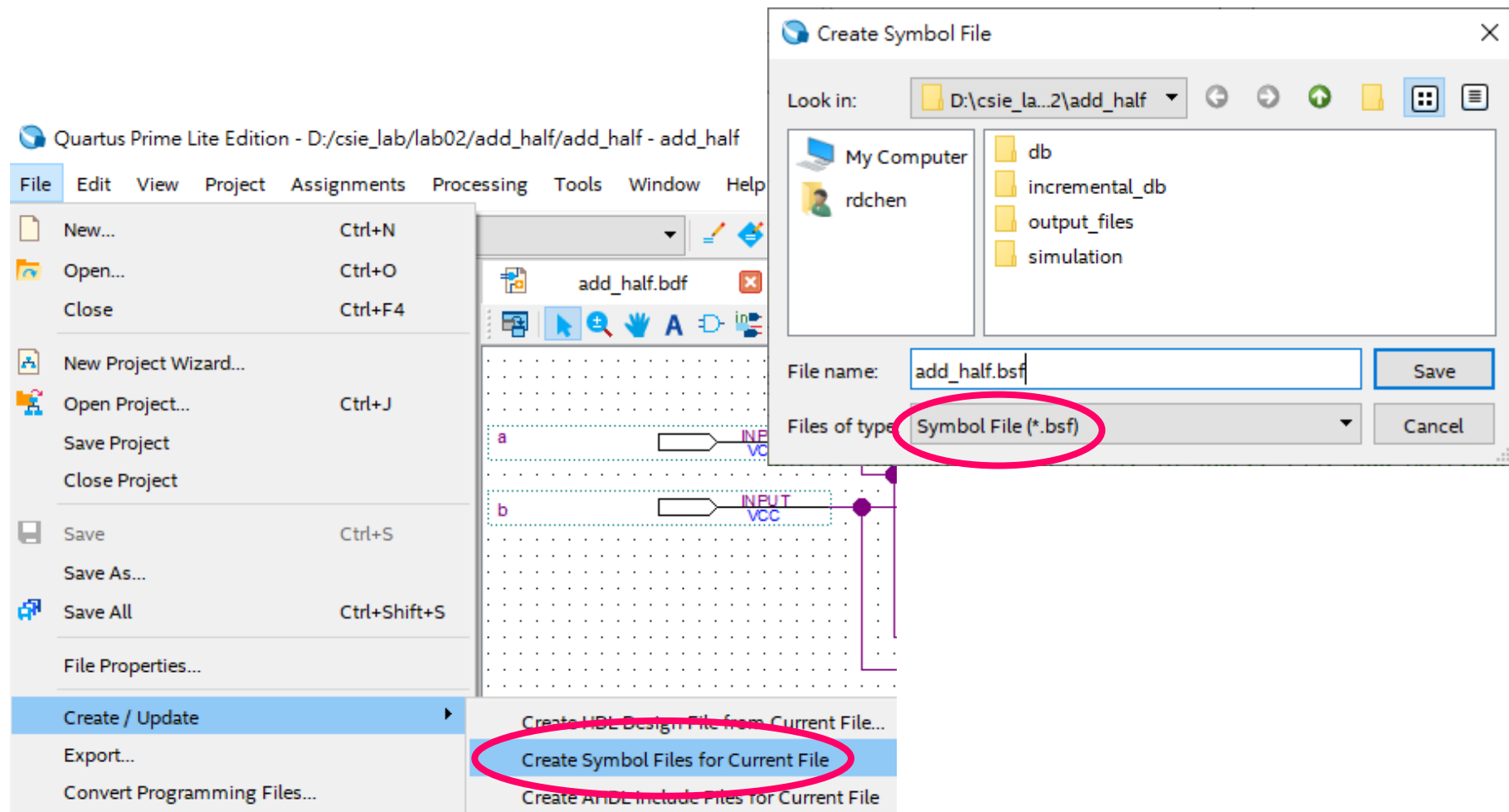
Create the HDL (Verilog) File



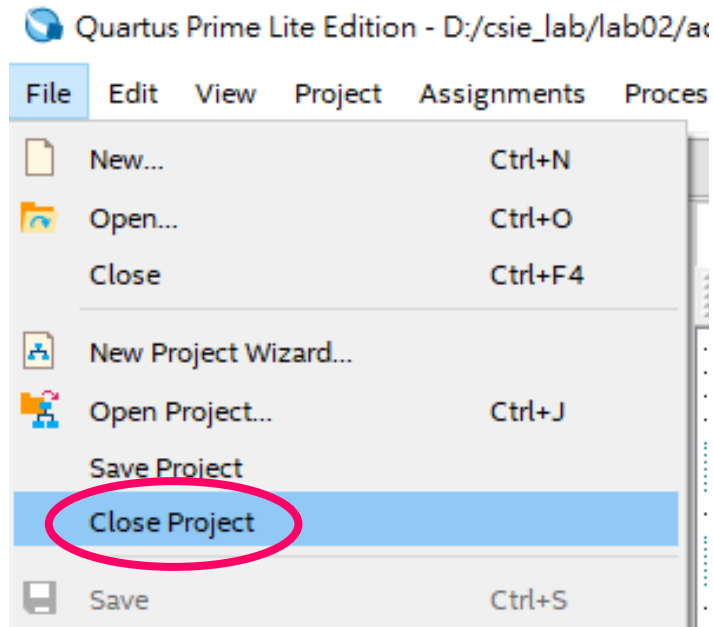
Content in add_half.v (Generated by Tool)

```
module add_half(  
    a,  
    b,  
    cout,  
    sum  
);  
  
input wire a;  
input wire b;  
output wire cout;  
output wire sum;  
  
assign cout = a & b;  
assign sum = a ^ b;  
  
endmodule
```

Create the Symbol File



Close the Current Project



All add_half files

data (D:) > csie_lab > lab02 > add_half

名稱	修改日期	類型
db	2022/2/28 下午 11:48	檔案資料夾
incremental_db	2022/2/28 下午 10:36	檔案資料夾
output_files	2022/2/28 下午 11:42	檔案資料夾
simulation	2022/2/28 下午 11:19	檔案資料夾
add_half	2022/2/28 下午 10:09	BDF 檔案
add_half	2022/2/28 下午 11:52	BSF 檔案
add_half	2022/2/28 下午 09:29	QPF 檔案
add_half.qsf	2022/2/28 下午 11:48	QSF 檔案
add_half.v	2022/2/28 下午 11:48	V 檔案
add_half.vwf	2022/2/28 下午 11:41	VWF 檔案

Design Example: Full Adder (1/2)



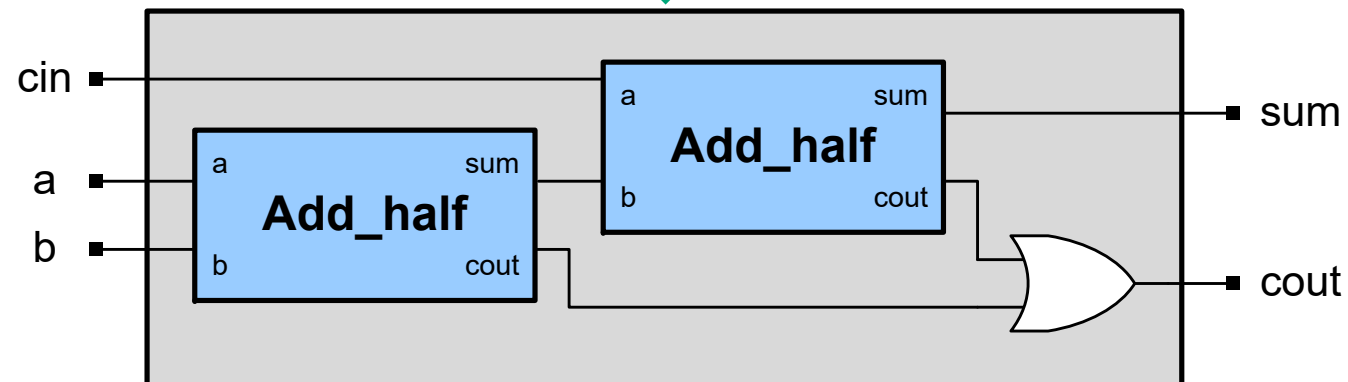
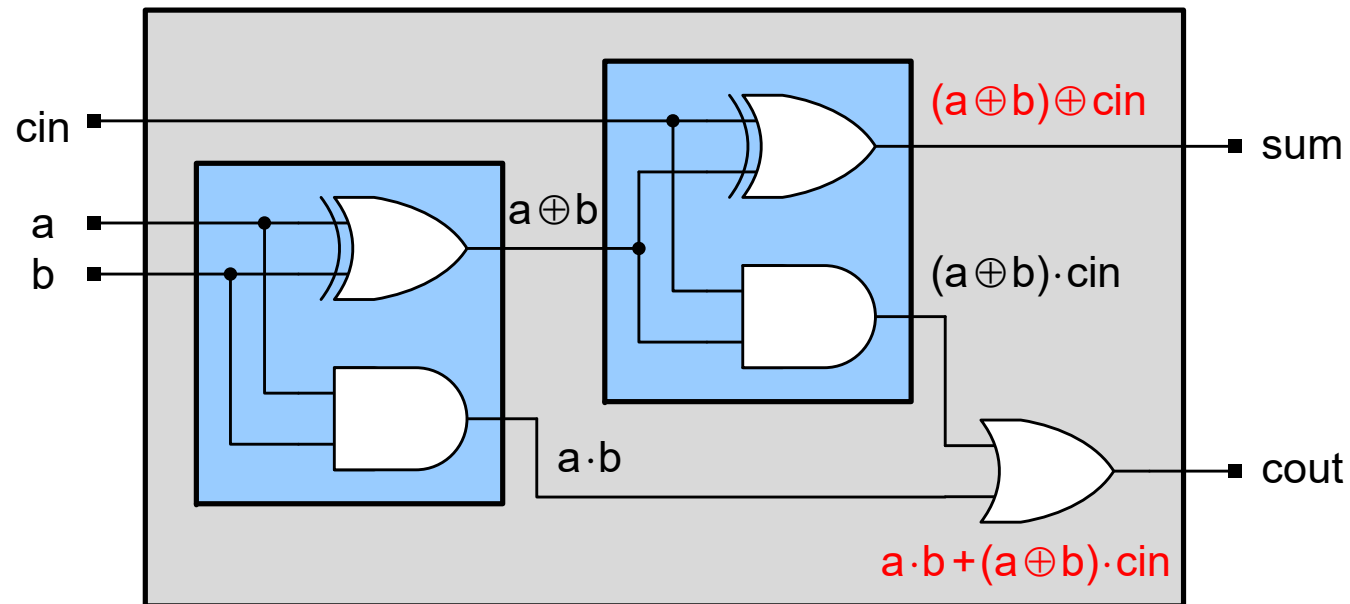
ab\cin	0	1
00	0	1
01	1	0
11	0	1
10	1	0

$$\text{sum} = (a \oplus b) \oplus \text{cin}$$

ab\cin	0	1
00	0	0
01	0	1
11	1	1
10	0	1

$$\text{cout} = a \cdot b + (a \oplus b) \cdot \text{cin}$$

Design Example: Full Adder (2/2)



Create a New Project - add_full



Directory, Name, Top-Level Entity

What is the working directory for this project?

D:/csie_lab/lab02/add_full



What is the name of this project?

add_full



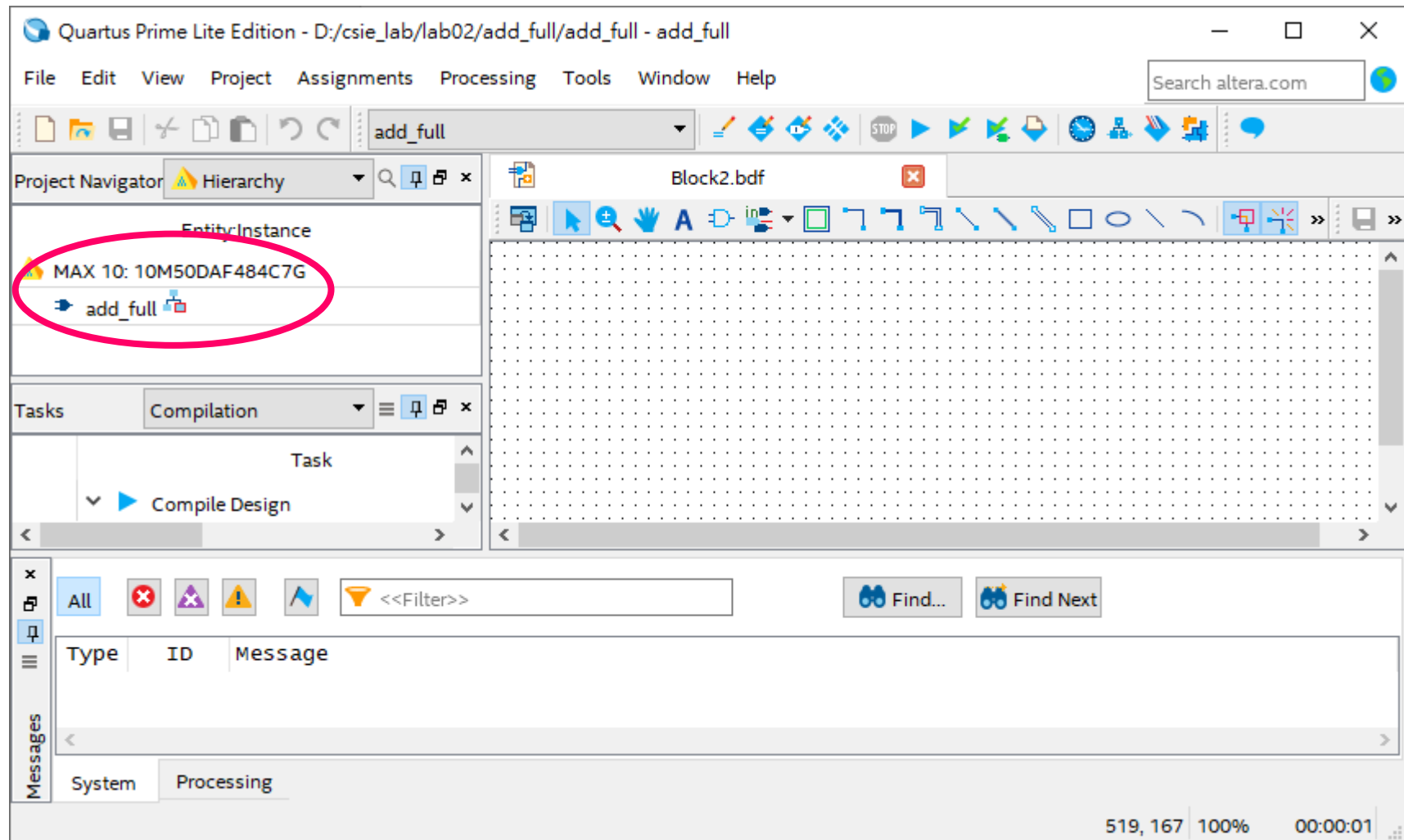
What is the name of the top-level design entity for this design file.

add_full

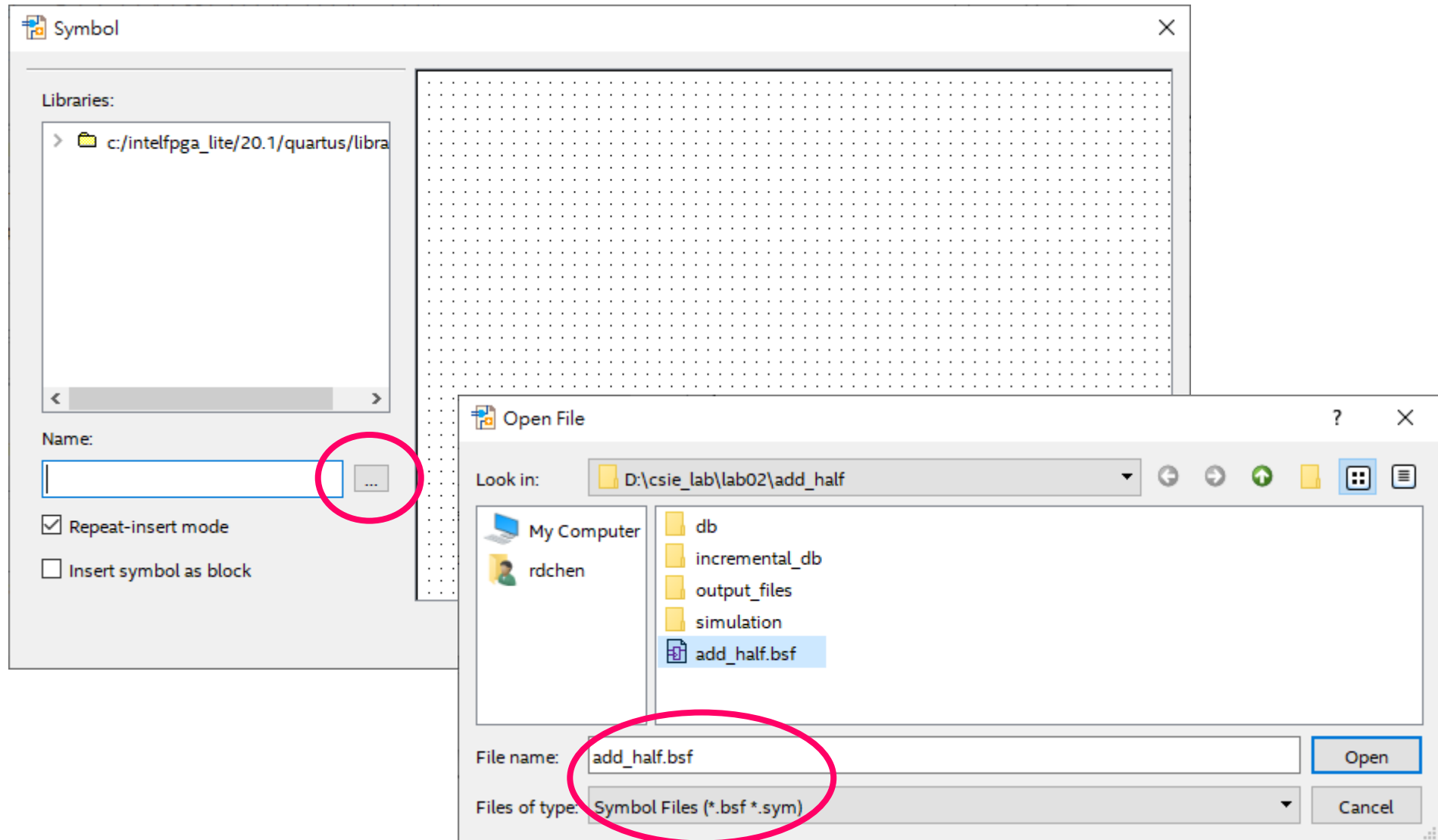


Use Existing Project Settings...

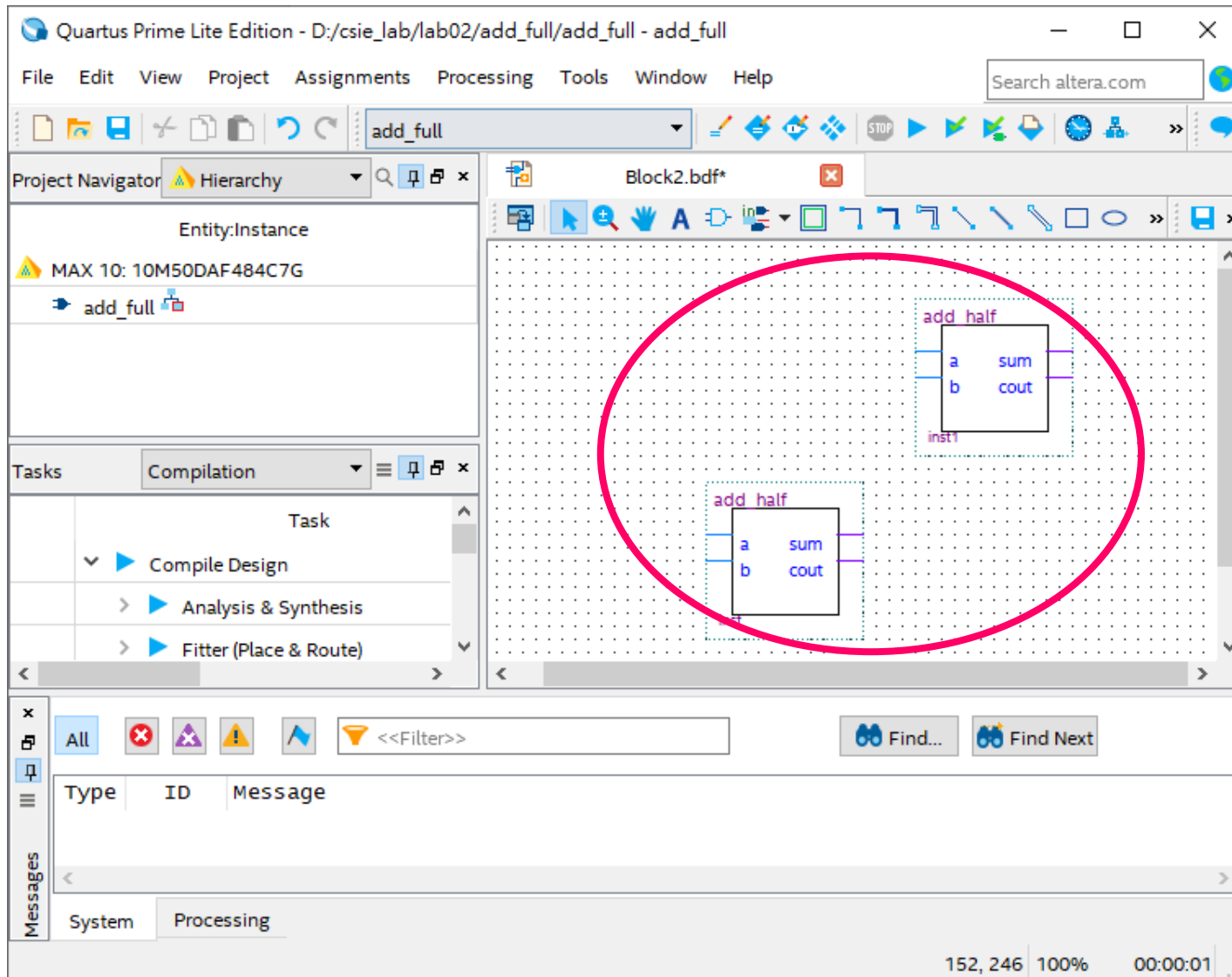
Create a Schematic Design for add_full



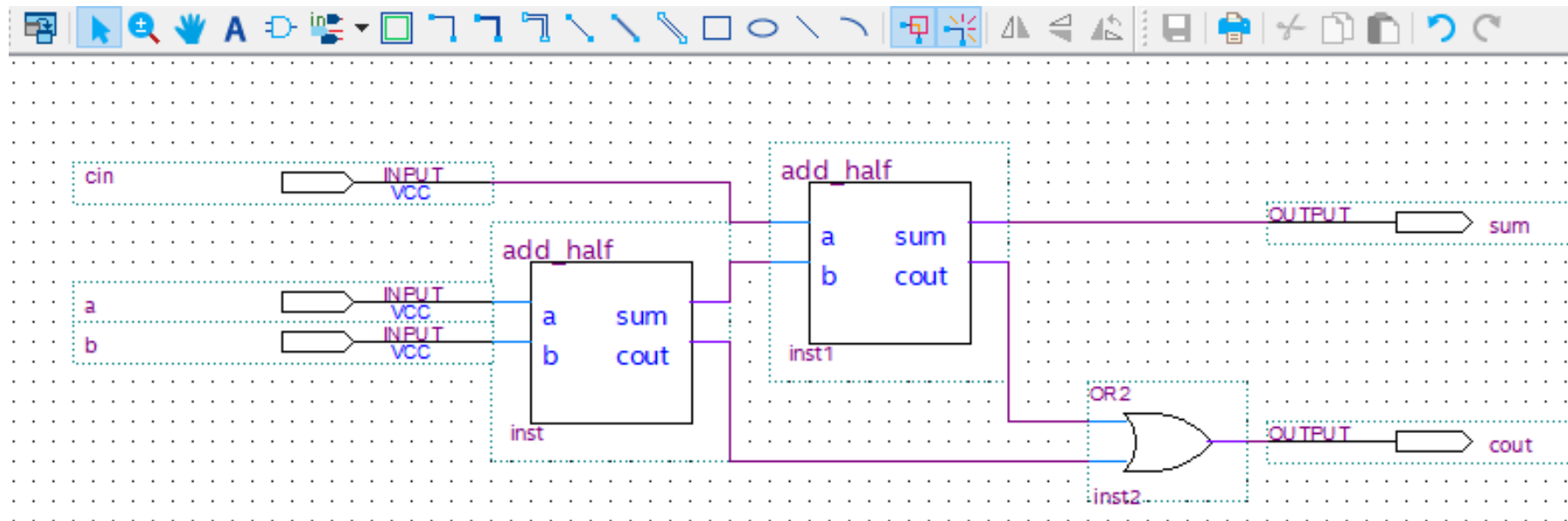
Select and Import a Half Adder



Two Half Adders are Imported



Import OR2 Gate and I/O Pins



Add the add_half Schematic File

Quartus Prime Lite Edition - D:/csie_lab/lab02/add_full/add_full - add_full

File Edit View Project Assignments Processing Tools Window

add_full

Project Navigator Files

Add/Remove Files in Project...

Category:

- General
- Files
- Libraries
- IP Settings
 - IP Catalog Search Locations
 - Design Templates
- Operating Settings and Conditions
 - Voltage

Files

Select the design files you want to include in the project. Click Add All to add all design files in the p directory to the project.

File name:

File Name	Type	Library	Design Entry/Synthesis To
../add_half/add_half.bdf	Block Diagram/Schematic File		<None>
add_full.bdf	Block Diagram/Schematic File		<None>

Quartus Prime Lite Edition - D:/csie_lab/lab02/add_full/add_full - add_full

File Edit View Project Assignments Processing Tools Window

add_full

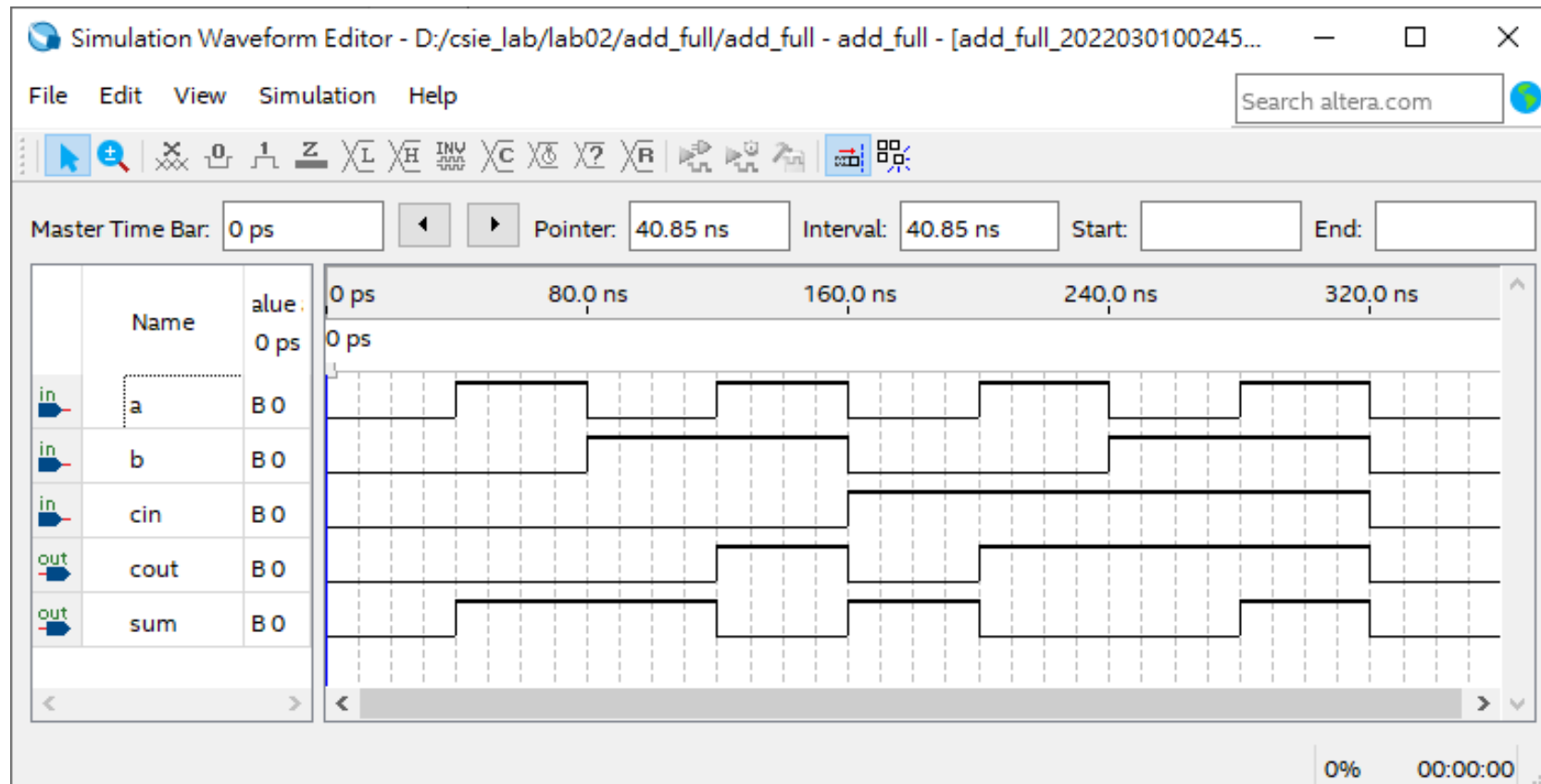
Project Navigator Files

Files

- ../add_half/add_half.bdf
- add_full.bdf

cin

Functional Simulation Result



Content in add_full.v (Generated by Tool)

```

    add_half    b2v_inst (
        .a(a) ,
        .b(b) ,
        .cout(SYNTHESIZED_WIRE_2) ,
        .sum(SYNTHESIZED_WIRE_0)) ;

module add_full(
    a,
    b,
    cin,
    cout,
    sum
);

    add_half    b2v_inst1 (
        .a(SYNTHESIZED_WIRE_0) ,
        .b(cin) ,
        .cout(SYNTHESIZED_WIRE_1) ,
        .sum(sum)) ;

input wire  a;
input wire  b;
input wire  cin;
output wire cout;
output wire sum;

    assign cout = SYNTHESIZED_WIRE_1 | SYNTHESIZED_WIRE_2;

endmodule

wire  SYNTHESIZED_WIRE_0;
wire  SYNTHESIZED_WIRE_1;
wire  SYNTHESIZED_WIRE_2;
```

Pin Assignments

Inputs

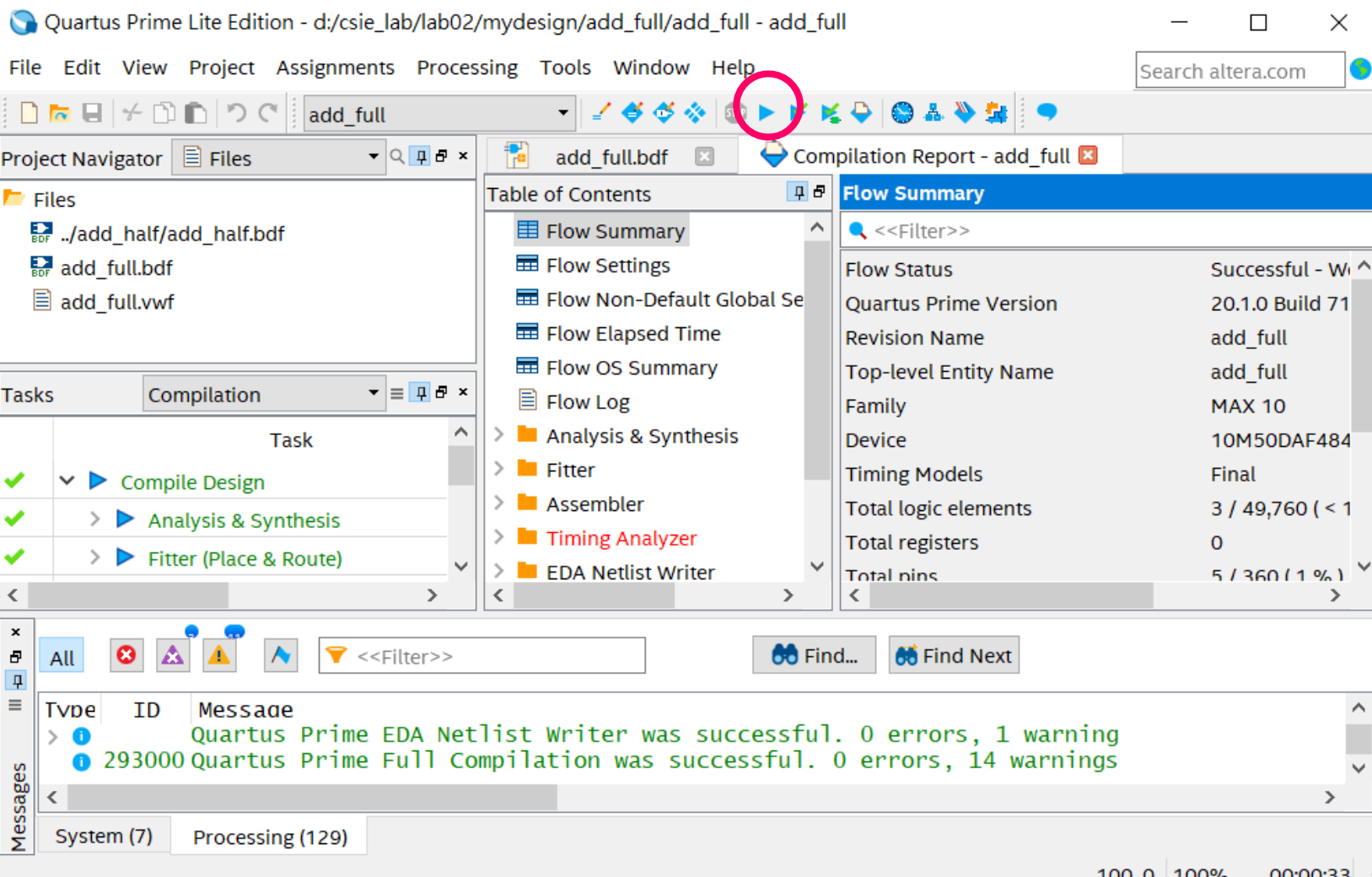


Outputs



$$\{\text{cout, sum}\} = a + b + \text{cin}$$

Start Compilation



Quartus Prime Lite Edition - d:/csie_lab/lab02/mydesign/add_full/add_full - add_full

File Edit View Project Assignments Processing Tools Window Help

add_full

Project Navigator

Files

- ../add_half/add_half.bdf
- add_full.bdf
- add_full.vwf

Tasks

Compilation

Task
Compile Design
Analysis & Synthesis
Fitter (Place & Route)

Table of Contents

- Flow Summary
- Flow Settings
- Flow Non-Default Global Se
- Flow Elapsed Time
- Flow OS Summary
- Flow Log
- Analysis & Synthesis
- Fitter
- Assembler
- Timing Analyzer
- EDA Netlist Writer

Flow Summary

<<Filter>>

Flow Status	Successful - W
Quartus Prime Version	20.1.0 Build 71
Revision Name	add_full
Top-level Entity Name	add_full
Family	MAX 10
Device	10M50DAF484
Timing Models	Final
Total logic elements	3 / 49,760 (< 1
Total registers	0
Total pins	5 / 360 (1 %)

Messages

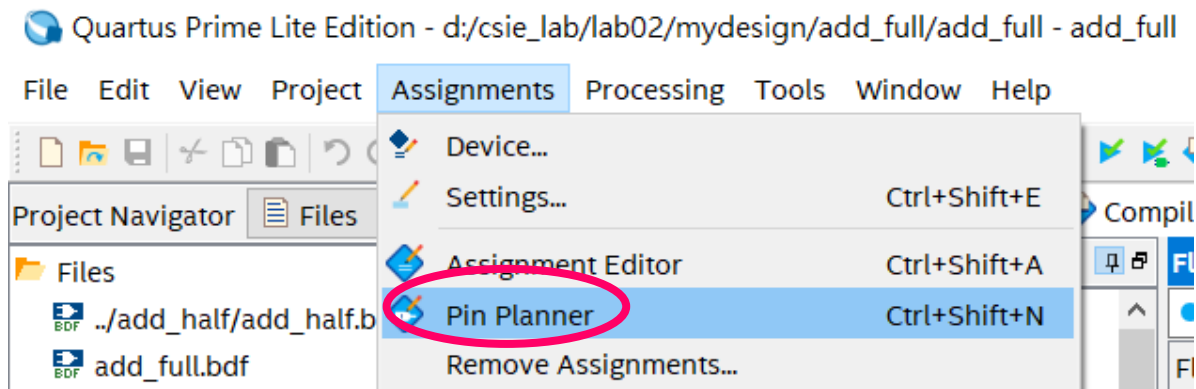
Type ID Message

- Quartus Prime EDA Netlist Writer was successful. 0 errors, 1 warning
- 293000 Quartus Prime Full Compilation was successful. 0 errors, 14 warnings

System (7) Processing (129)

100,0 100% 00:00:33

Make Pin Assignments



Node Name	Direction	Location	I/O Bank	VREF Group	Pin Location
in a	Input	PIN_D12	7	B7_N0	PIN_Y5
in b	Input	PIN_C11	7	B7_N0	PIN_AA2
in cin	Input	PIN_C10	7	B7_N0	PIN_AA1
out cout	Output	PIN_A9	7	B7_N0	PIN_W3
out sum	Output	PIN_A8	7	B7_N0	PIN_U6

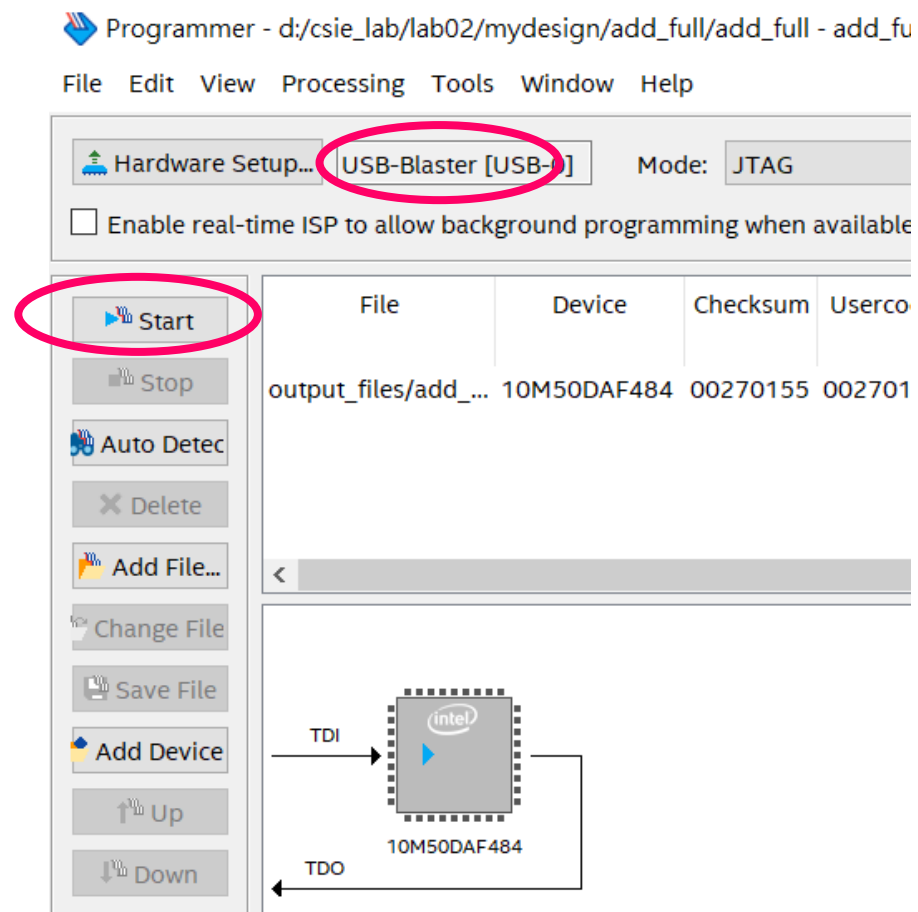
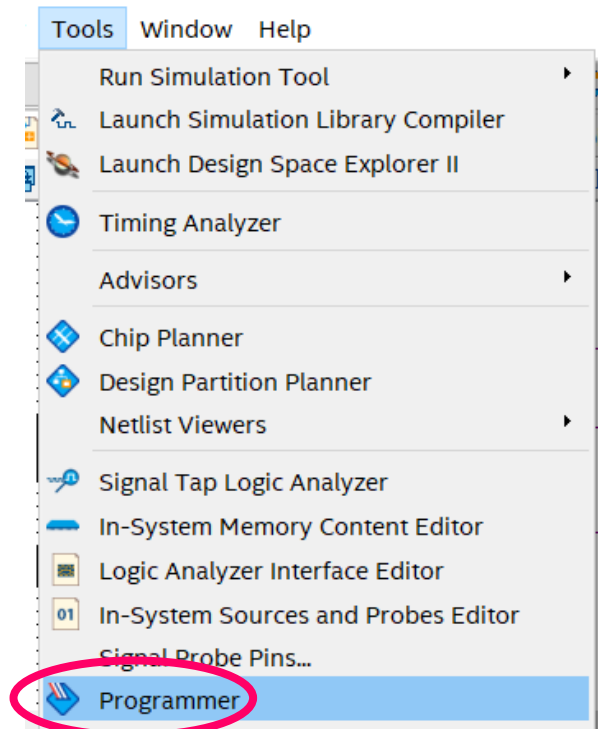
Start Compilation

The screenshot displays the Quartus Prime Lite Edition interface. The title bar indicates the project path: `d:/csie_lab/lab02/mydesign/add_full/add_full - add_full`. The menu bar includes File, Edit, View, Project, Assignments, Processing, Tools, Window, and Help. The toolbar features various icons, with the 'Run' icon (a green play button) circled in red. The Project Navigator on the left shows the file structure: `./add_half/add_half.bdf`, `add_full.bdf`, and `add_full.vwf`. The Tasks pane on the left lists the compilation steps: Compile Design, Analysis & Synthesis, Fitter (Place & Route), and Assembler (Generate program), all marked with green checkmarks. The main workspace shows a logic diagram with two `add_half` blocks, an OR2 gate, and output pins `PIN_A9` and `PIN_A8`, which are circled in red. The Messages pane at the bottom shows the following messages:

Type	ID	Message
Information	332102	Design is not fully constrained for hold requirements
Information		Quartus Prime Timing Analyzer was successful. 0 errors, 5 warnings

The status bar at the bottom indicates 278,481 cells, 100% completion, and a time of 00:00:35.

Program into Device



Verify Your Design

- sw2/sw1/sw0 : dn/dn/dn (0/0/0) => L1/L0: off/off (0/0)
- sw2/sw1/sw0 : dn/dn/up (0/0/1) => L1/L0: off/on (0/1)
- sw2/sw1/sw0 : dn/up/dn (0/1/0) => L1/L0: off/on (0/1)
- sw2/sw1/sw0 : dn/up/up (0/1/1) => L1/L0: on/off (1/0)
- sw2/sw1/sw0 : up/dn/dn (1/0/0) => L1/L0: off/on (0/1)
- sw2/sw1/sw0 : up/dn/up (1/0/1) => L1/L0: on/off (1/0)
- sw2/sw1/sw0 : up/up/dn (1/1/0) => L1/L0: on/off (1/0)
- sw2/sw1/sw0 : up/up/up (1/1/1) => L1/L0: on/on (1/1)

SW2 SW1 SW0



實驗報告

- 請老師or助教驗證 **add_full** 電路之模擬波形及實驗板之行為是否正確