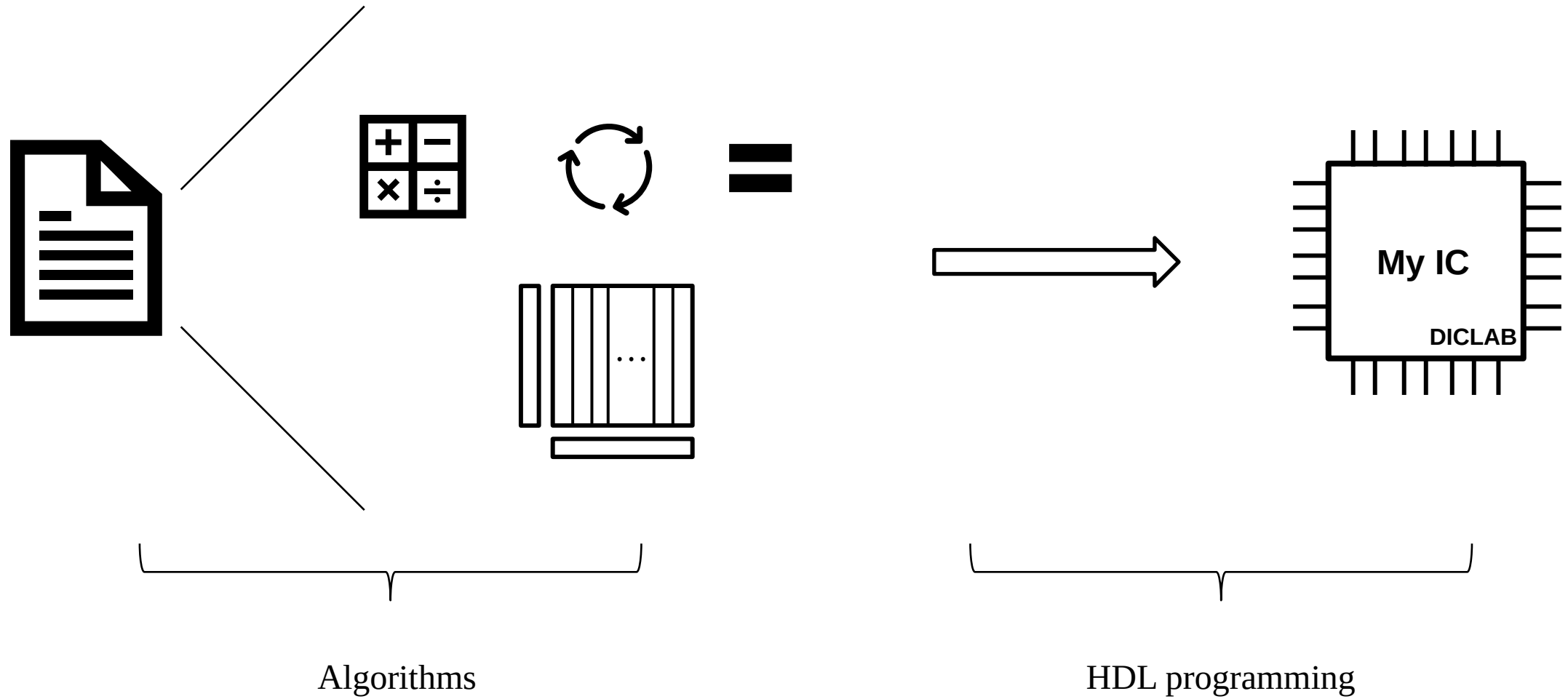




LAB - 05

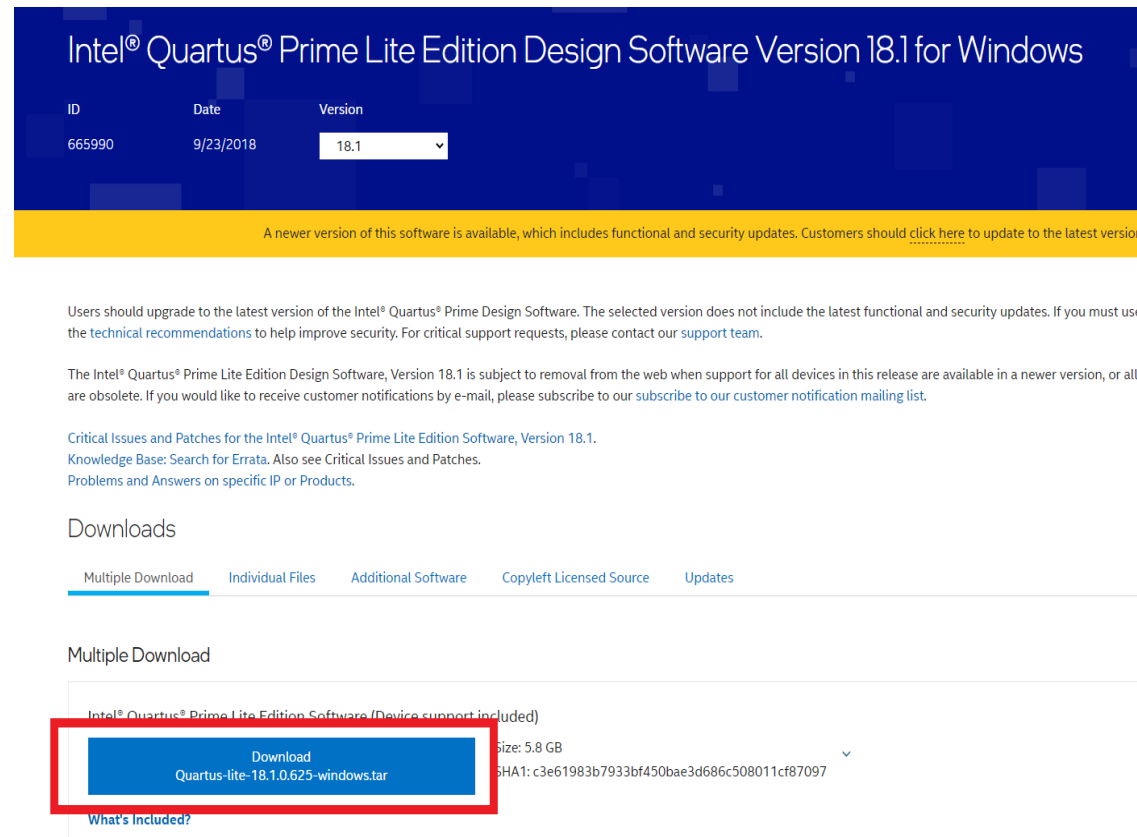
陳培殷老師
國立成功大學 資訊工程系

Introduction



Install Quartus II

- 下載網址 - <https://reurl.cc/eOQK7Q>



Intel® Quartus® Prime Lite Edition Design Software Version 18.1 for Windows

ID	Date	Version
665990	9/23/2018	18.1

A newer version of this software is available, which includes functional and security updates. Customers should [click here](#) to update to the latest version.

Users should upgrade to the latest version of the Intel® Quartus® Prime Design Software. The selected version does not include the latest functional and security updates. If you must use the [technical recommendations](#) to help improve security. For critical support requests, please contact our [support team](#).

The Intel® Quartus® Prime Lite Edition Design Software, Version 18.1 is subject to removal from the web when support for all devices in this release are available in a newer version, or all are obsolete. If you would like to receive customer notifications by e-mail, please subscribe to our [subscribe to our customer notification mailing list](#).

[Critical Issues and Patches for the Intel® Quartus® Prime Lite Edition Software, Version 18.1.](#)
[Knowledge Base: Search for Errata.](#) Also see [Critical Issues and Patches](#).
[Problems and Answers on specific IP or Products](#).

Downloads

- [Multiple Download](#)
- [Individual Files](#)
- [Additional Software](#)
- [Copyleft Licensed Source](#)
- [Updates](#)

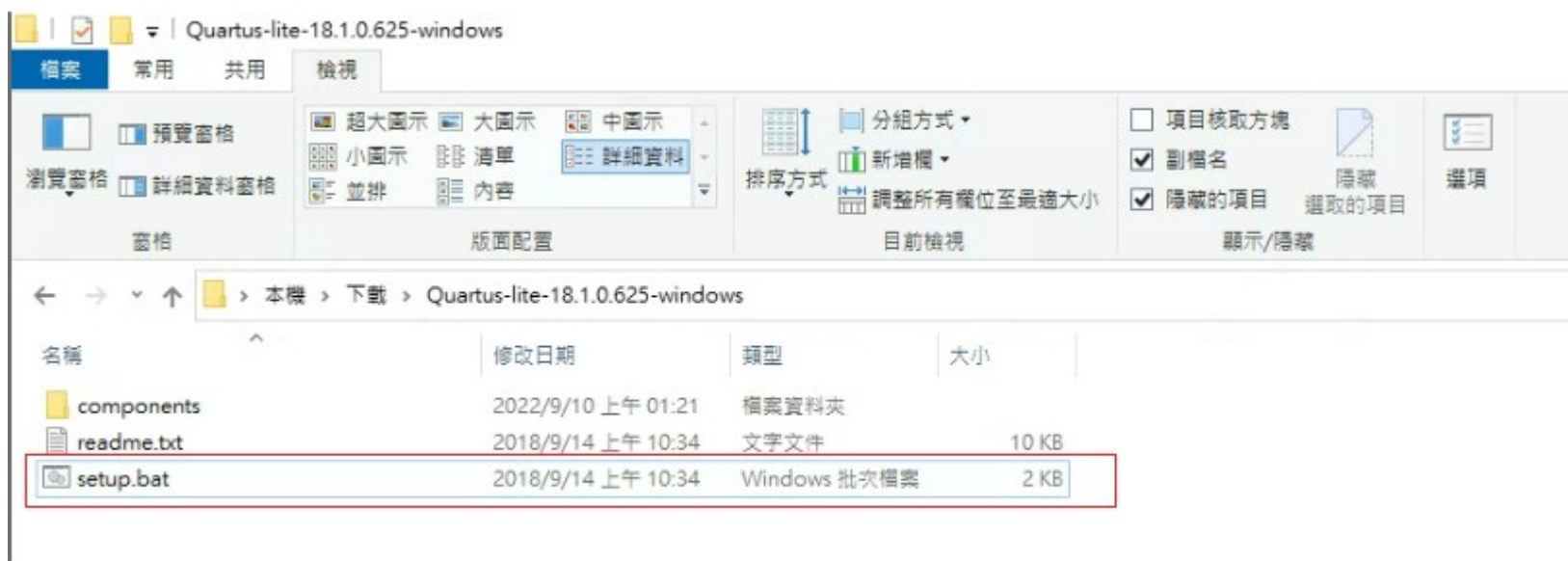
Multiple Download

Download	Size	SHA1
Download Quartus-lite-18.1.0.623-windows.tar	5.8 GB	c3e61983b7933bf450bae3d686c508011cf87097

[What's Included?](#)

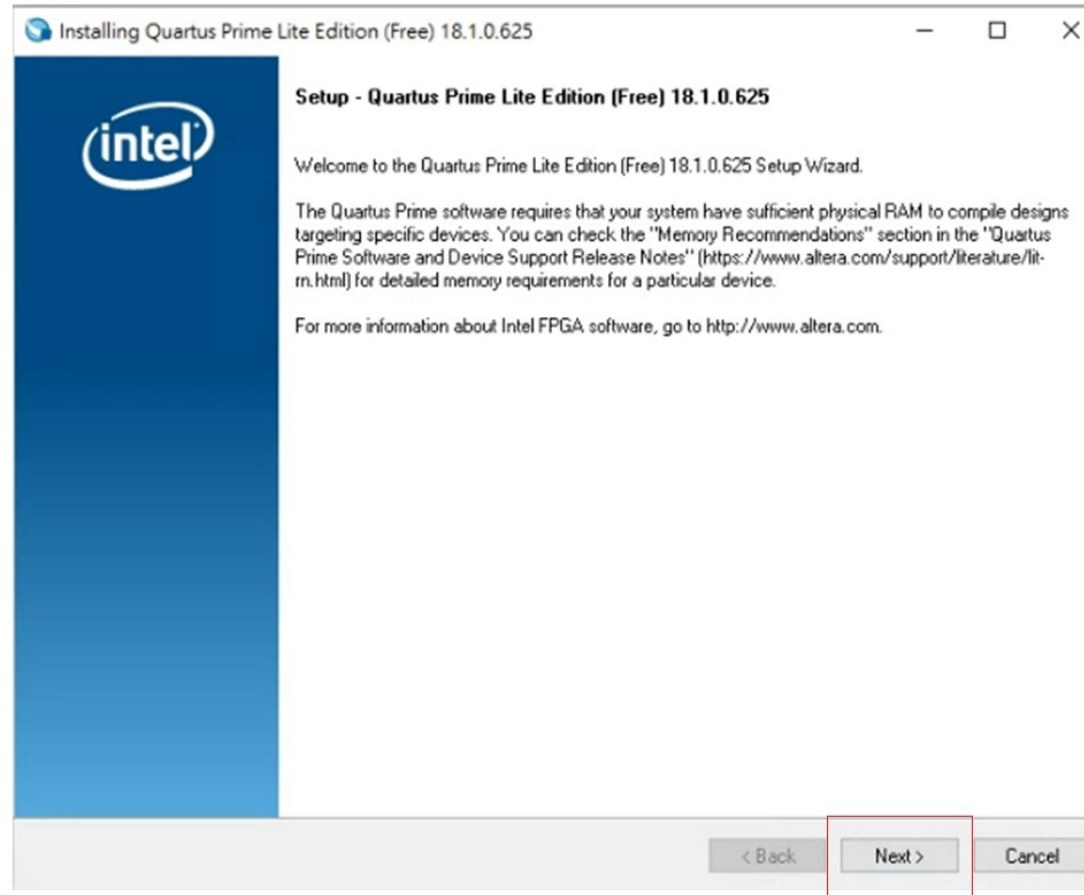
Install Quartus II

- 解壓縮後打開 setup.bat



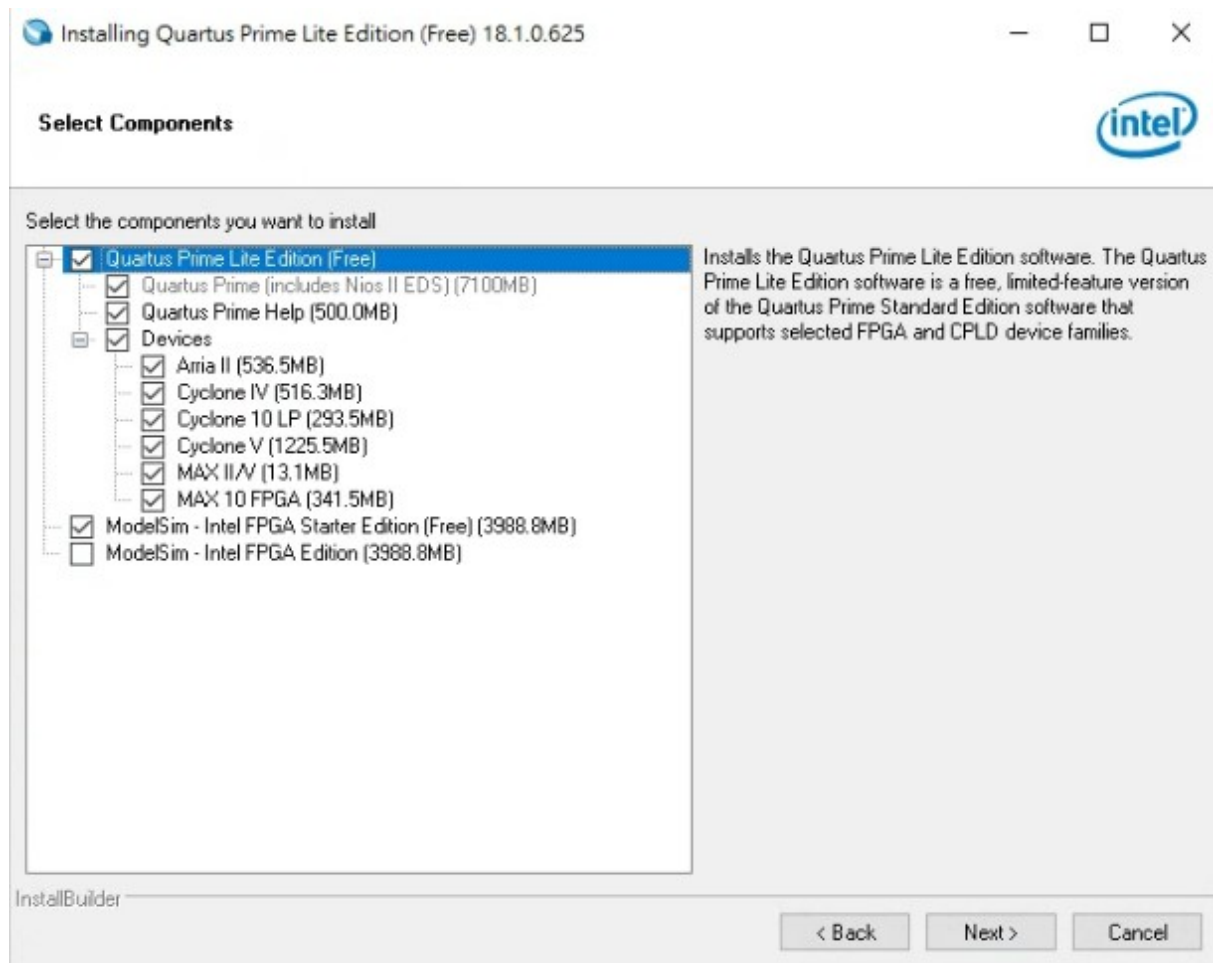
Install Quartus II

- Next



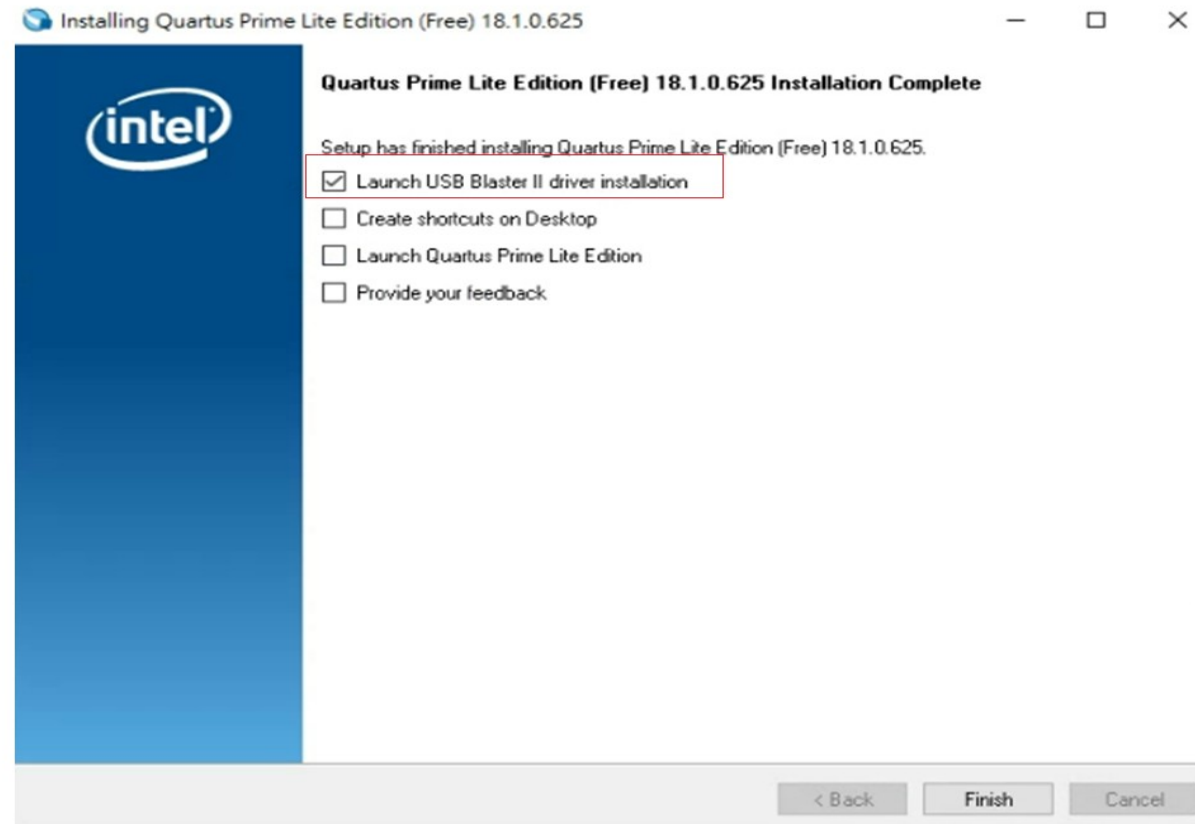
Install Quartus II

■ 勾選選項



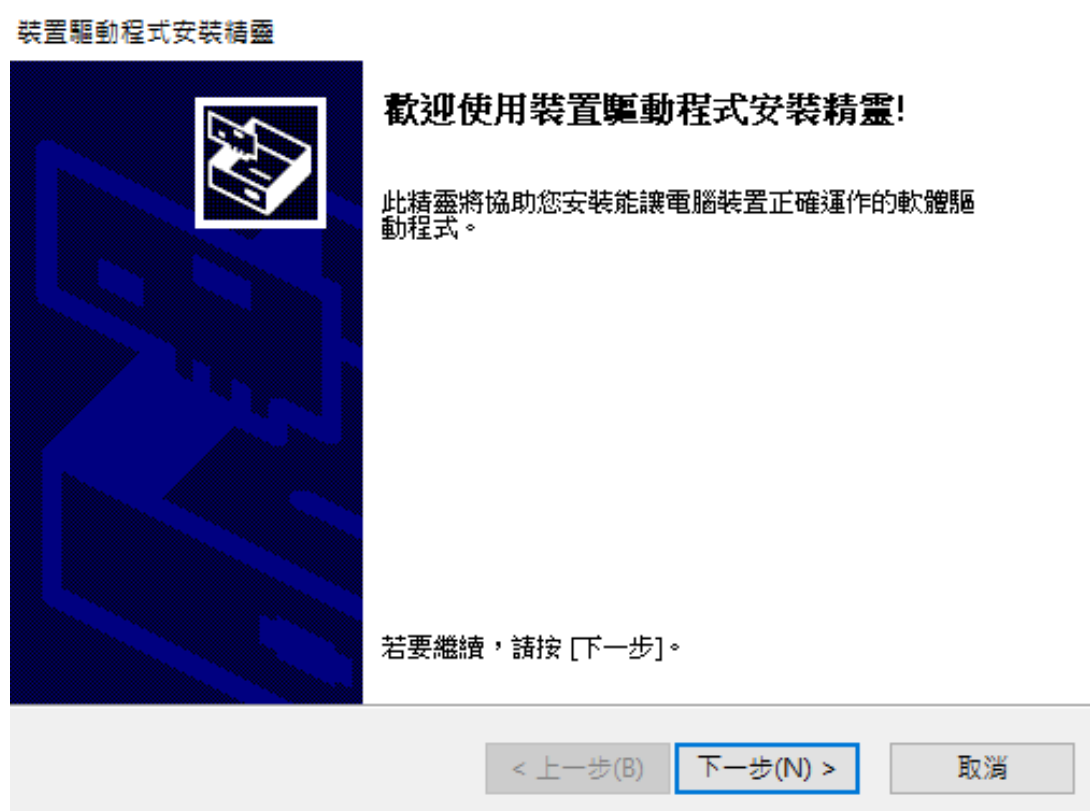
Install Quartus II

- 完成後勾選 Launch USB Blaster II driver installation



Install Quartus II

- 跳出裝置驅動程式安裝精靈後一直按下一步



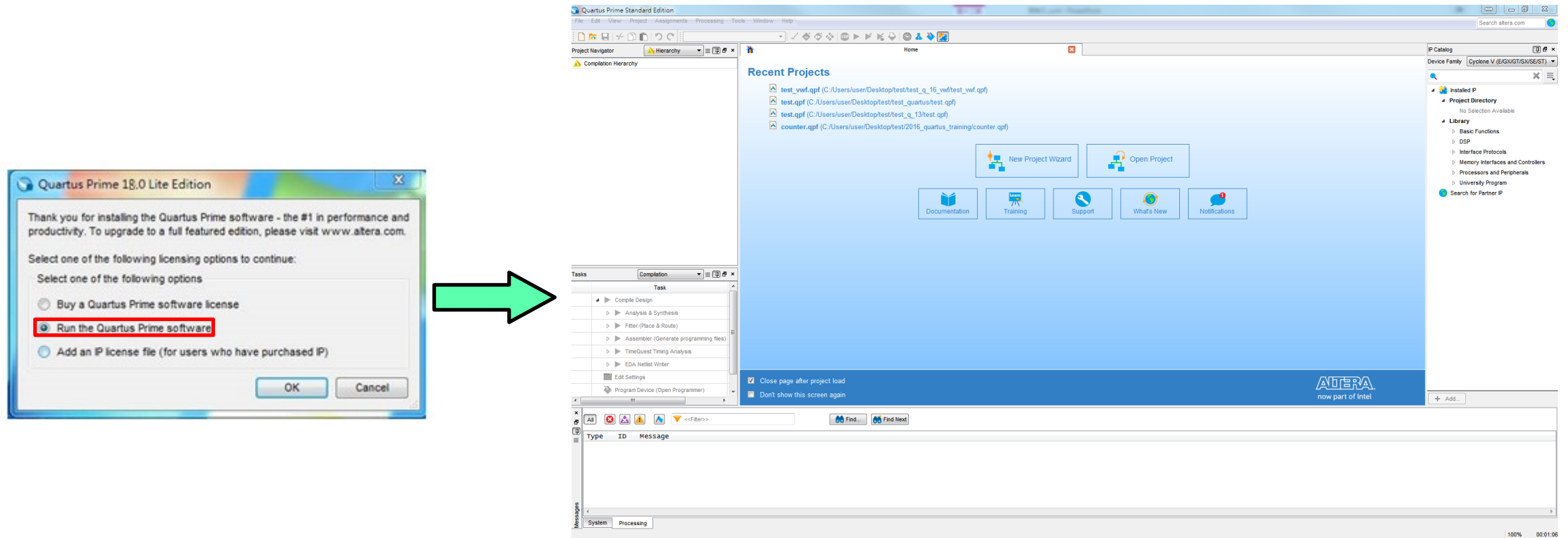
Install Quartus II

- 出現“可以使用”就完成安裝了



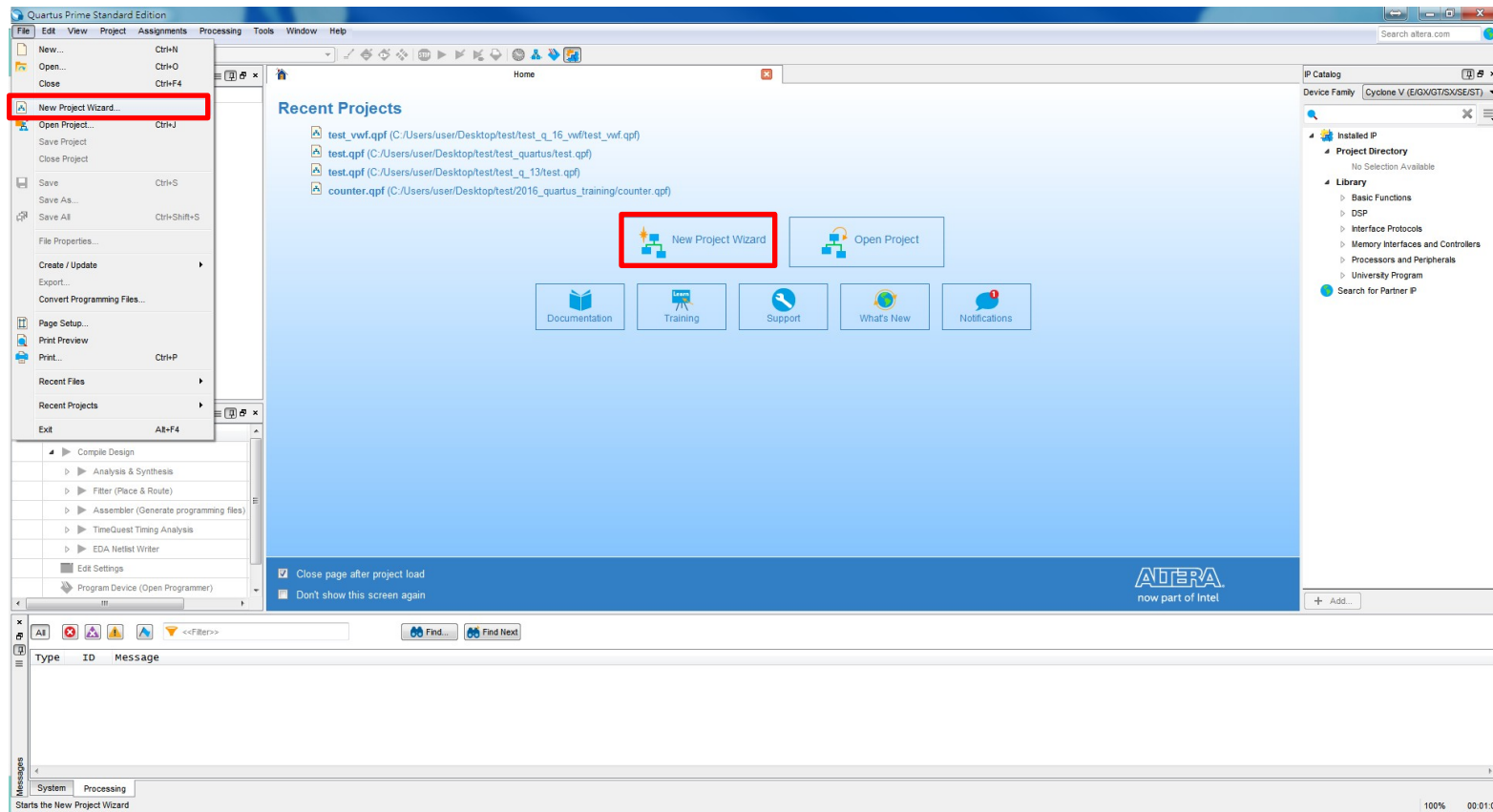
Quartus II Tutorial (1/19)

- Getting Started –
 - Start the Quartus II software



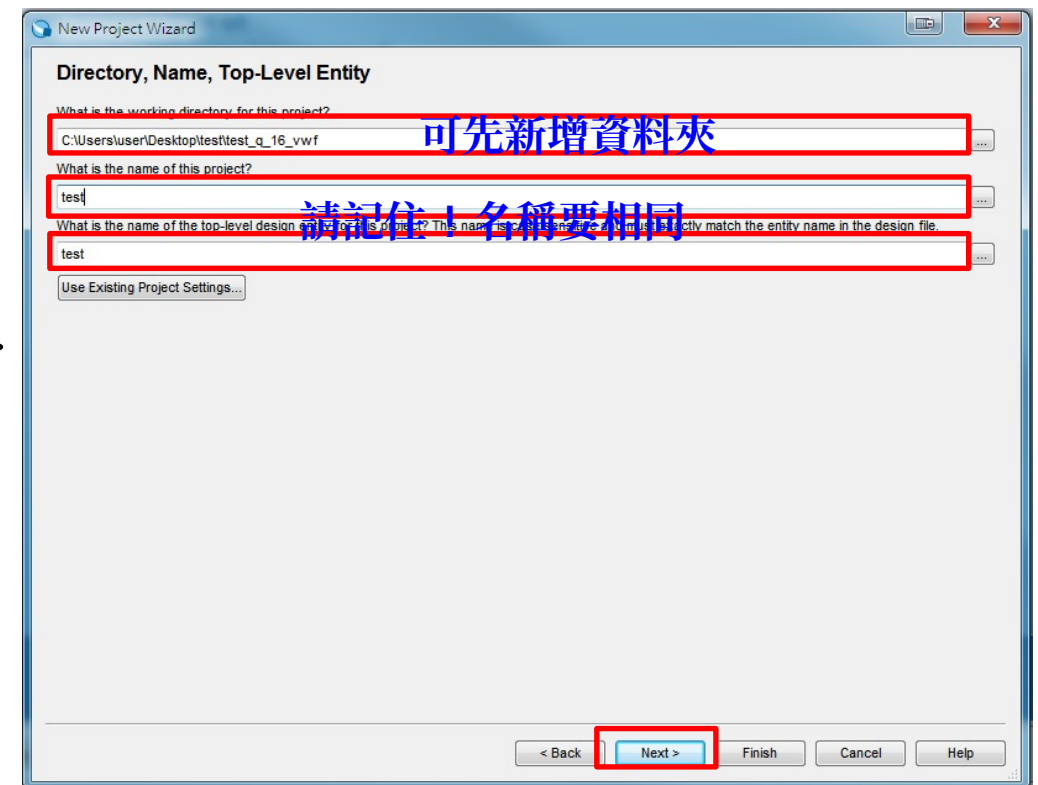
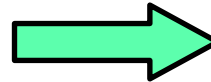
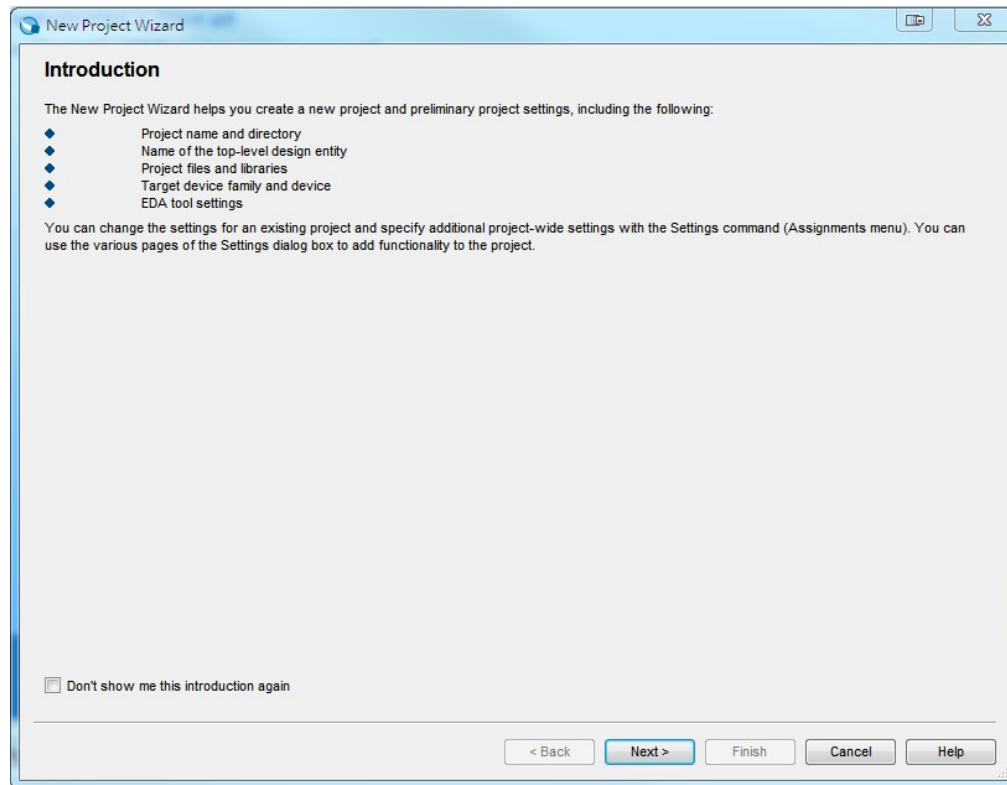
Quartus II Tutorial (2/19)

- Create a New Project –
 - Open New Project Wizard (File → New Project Wizard...)



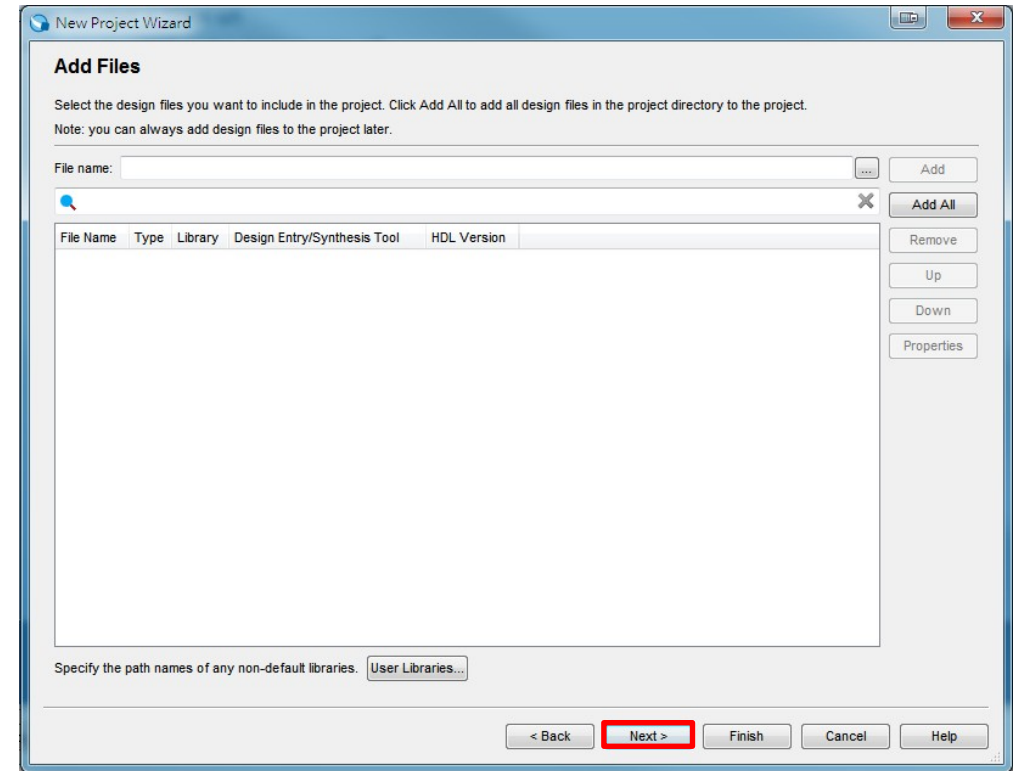
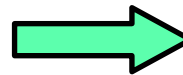
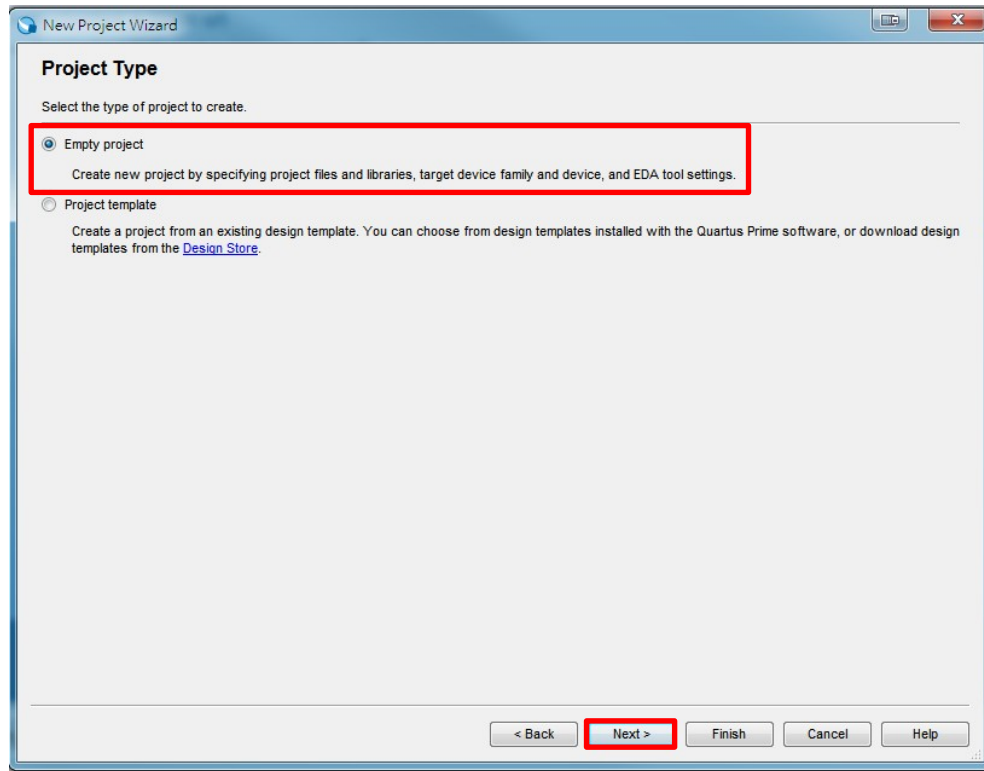
Quartus II Tutorial (3/19)

- Specify the working directory and the name of the project



Quartus II Tutorial (4/19)

- Select “Empty project”. Then, click “Next”.
- Select design files. Or click “Next” to skip this step.



Quartus II Tutorial (5/19)

- Specify device settings - (DE10-Lite Device family are used). Click “Next.”

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

10M50DAF484C7G

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

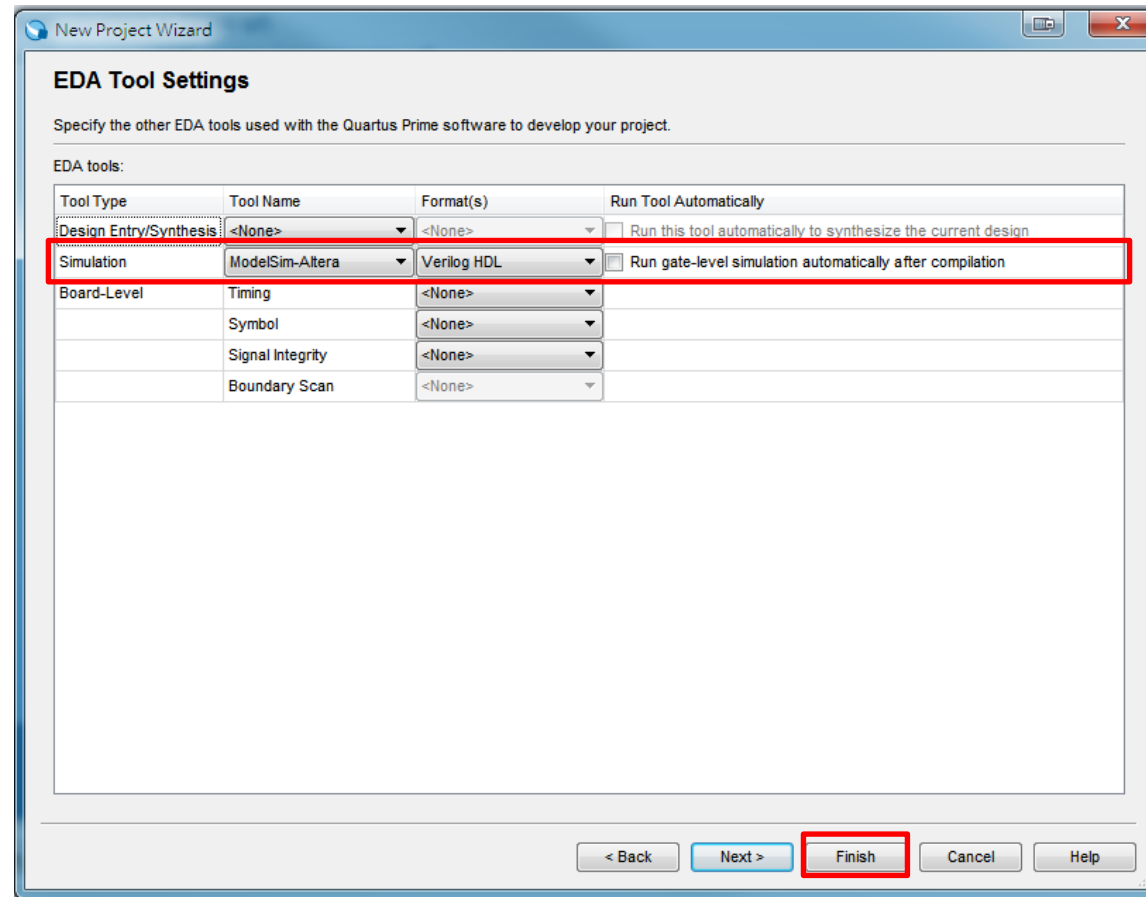
Available devices:

Name	Core Voltage	LEs	Total I/Os	GPIOs	Memory Bits	Embedded multiplier 9-bit el
10M50DAF256I7G	1.2V	49760	178	178	1677312	288
10M50DAF484C6GES	1.2V	49760	360	360	1677312	288
10M50DAF484C7G	1.2V	49760	360	360	1677312	288
10M50DAF484C8G	1.2V	49760	360	360	1677312	288
10M50DAF484C8GES	1.2V	49760	360	360	1677312	288

< Back Next > Finish Cancel Help

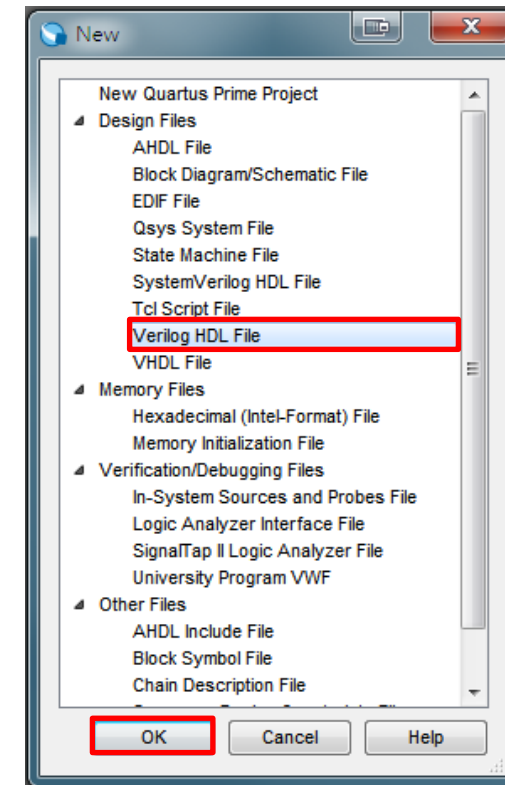
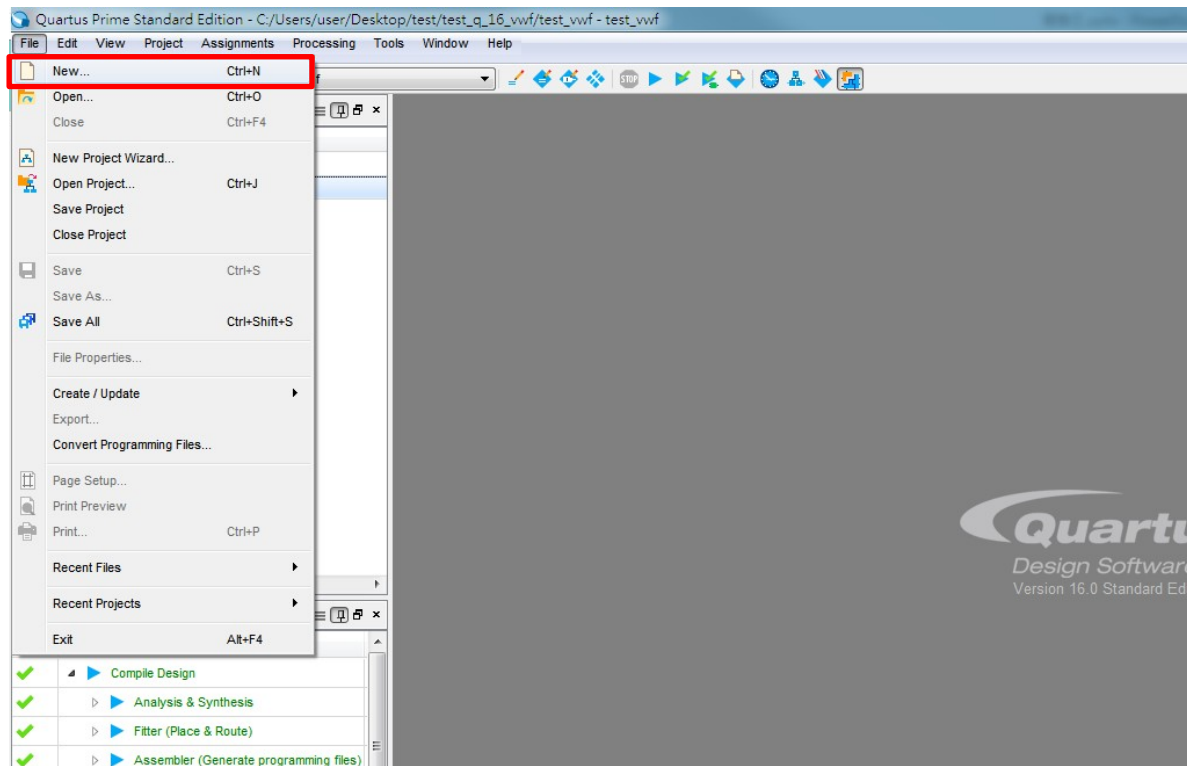
Quartus II Tutorial (6/19)

- Specify EDA Tool – (**Modelsim-Altera** is selected for simulation). Click “Finish.”



Quartus II Tutorial (7/19)

- Edit a new file by opening a Verilog HDL file
 - (File → New → **Verilog HDL File** → OK)



Quartus II Tutorial (8/19)

■ Write Verilog code

Top module name 一定要跟 Project name 相同 !!

輸入 (input)		輸出 (output)	
被加數 (a)	加數 (b)	和 (sum)	進位 (carry)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



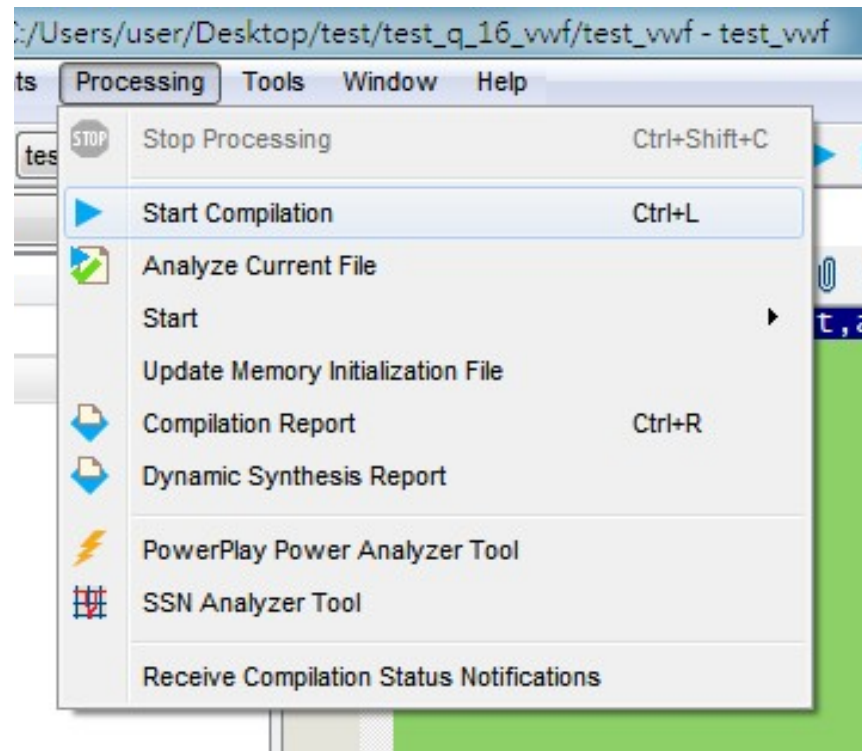
```
module test (  
    input a,  
    input b,  
    output sum,  
    output carry  
);
```

```
/* Write your code here. */
```

```
endmodule
```

Quartus II Tutorial (9/19)

- **Compiling the Designed Circuit (synthesis 合成)**
 - (Processing → Start Compilation)



Quartus II Tutorial (10/19)

■ Successful compilation

The screenshot displays the Quartus Prime Standard Edition interface. The main window shows the 'Flow Summary' for the project 'test_vwf.v'. The summary indicates a successful compilation on Tuesday, October 25, 2016, at 10:54:02. The device used is Cyclone V: SCEBA7F23C7. The summary includes the following statistics:

Category	Value
Flow Status	Successful - Tue Oct 25 10:54:02 2016
Quartus Prime Version	16.0.1 Build 218 06/01/2016 SJ Standard Edition
Revision Name	test_vwf
Top-level Entity Name	test_vwf
Family	Cyclone V
Device	SCEBA7F23C7
Timing Models	Final
Logic utilization (in ALMs)	2 / 56,480 (< 1 %)
Total registers	0
Total pins	6 / 240 (3 %)
Total virtual pins	0
Total block memory bits	0 / 7,024,640 (0 %)
Total DSP Blocks	0 / 156 (0 %)
Total HSSI RX PCSs	0
Total HSSI PMA RX Deserializers	0
Total HSSI TX PCSs	0
Total HSSI PMA TX Serializers	0
Total PLLs	0 / 7 (0 %)
Total DLLs	0 / 4 (0 %)

The 'Tasks' window shows a list of tasks that were completed successfully:

- Compile Design
- Analysis & Synthesis
- Fitter (Place & Route)
- Assembler (Generate programming files)
- TimeQuest Timing Analysis
- EDA Netlist Writer
- Edit Settings
- Program Device (Open Programmer)

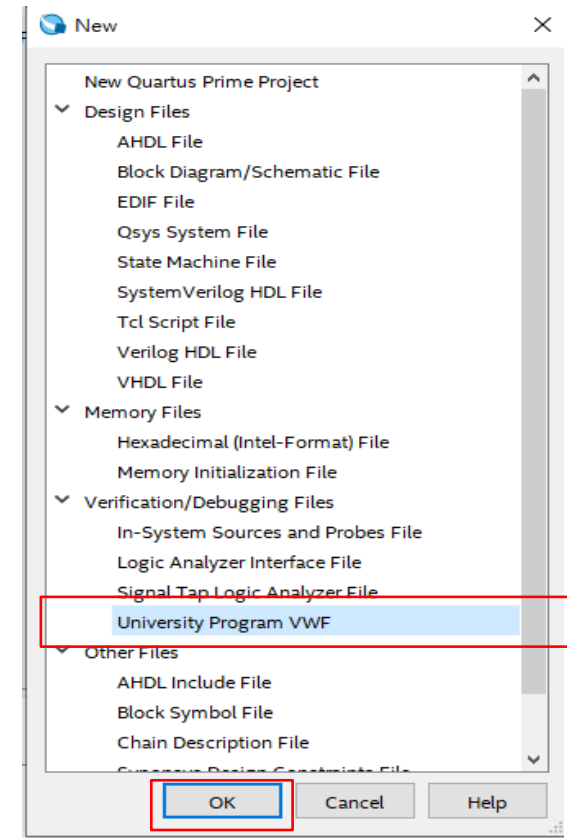
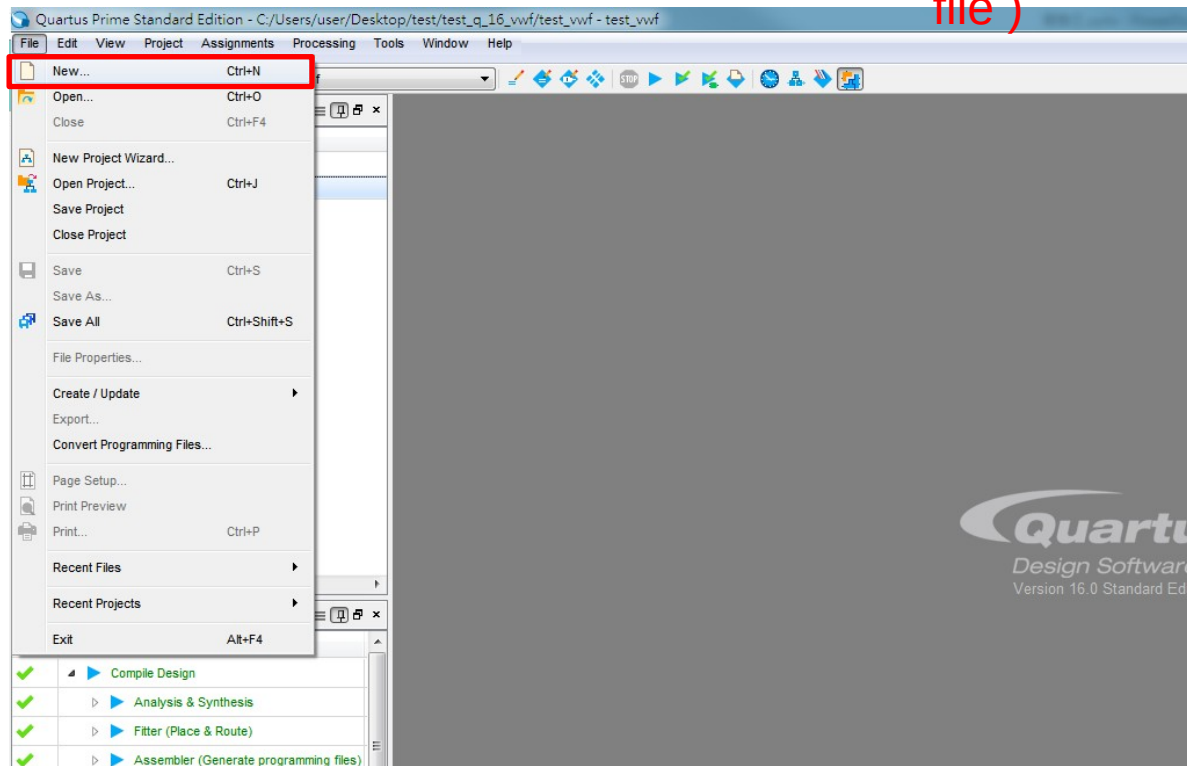
The 'Messages' window at the bottom shows the following messages:

```
Running Quartus Prime EDA Netlist Writer
Command: quartus_eda --read_settings_files=off --write_settings_files=off test_vwf -c test_vwf
18236 Number of processors has not been specified which may cause overloading on shared machines. Set the global assignment NUM_PARALLEL_PROCESSORS in your QSF to an appropriate value for best performance.
10905 Generated the EDA functional simulation netlist because it is the only supported netlist type for this device.
204019 Generated file test_vwf.vo in folder "C:/users/user/desktop/test/test_q16_vwf/simulation/modelsim/" for EDA simulation tool
Quartus Prime EDA Netlist writer was successful. 0 errors, 2 warnings
293000 Quartus Prime Full compilation was successful. 0 errors, 18 warnings
```

Quartus II Tutorial (11/19)

■ Edit a new file by opening a Verilog HDL file

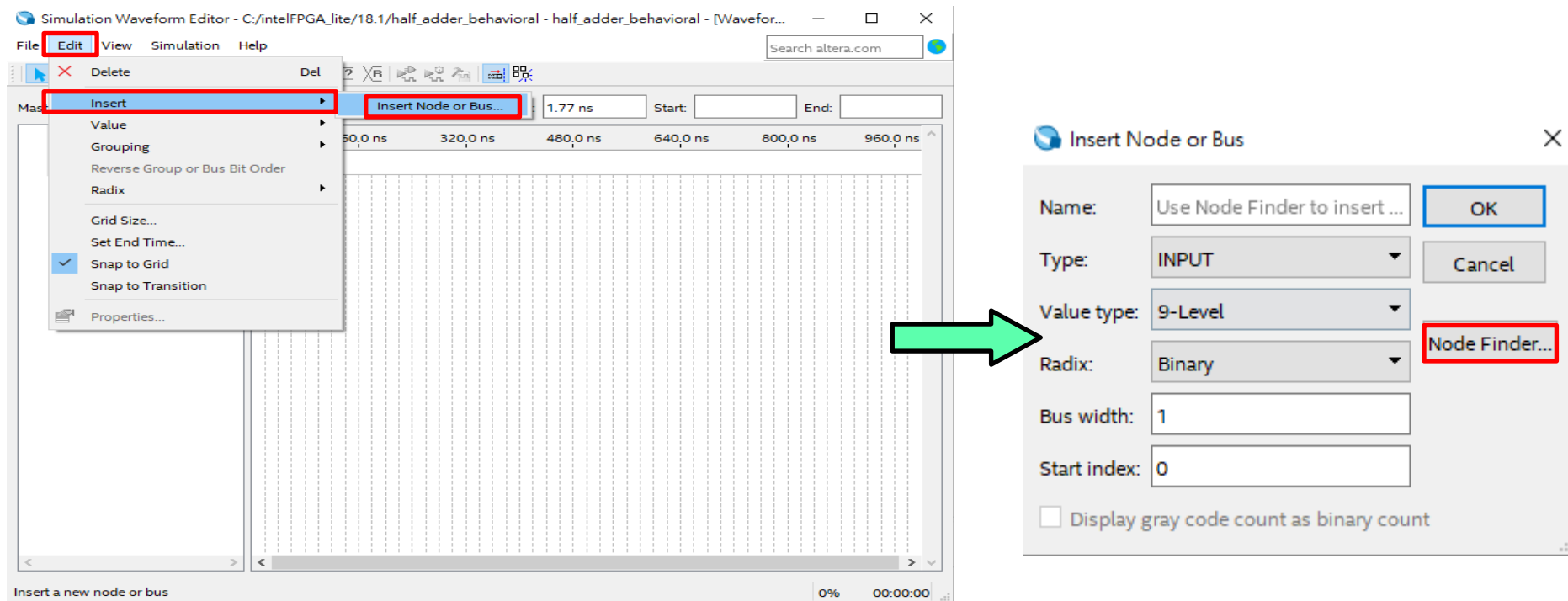
- (File → New → **University Program VWF** → OK)
(vector waveform file)



Quartus II Simulation (12/19)

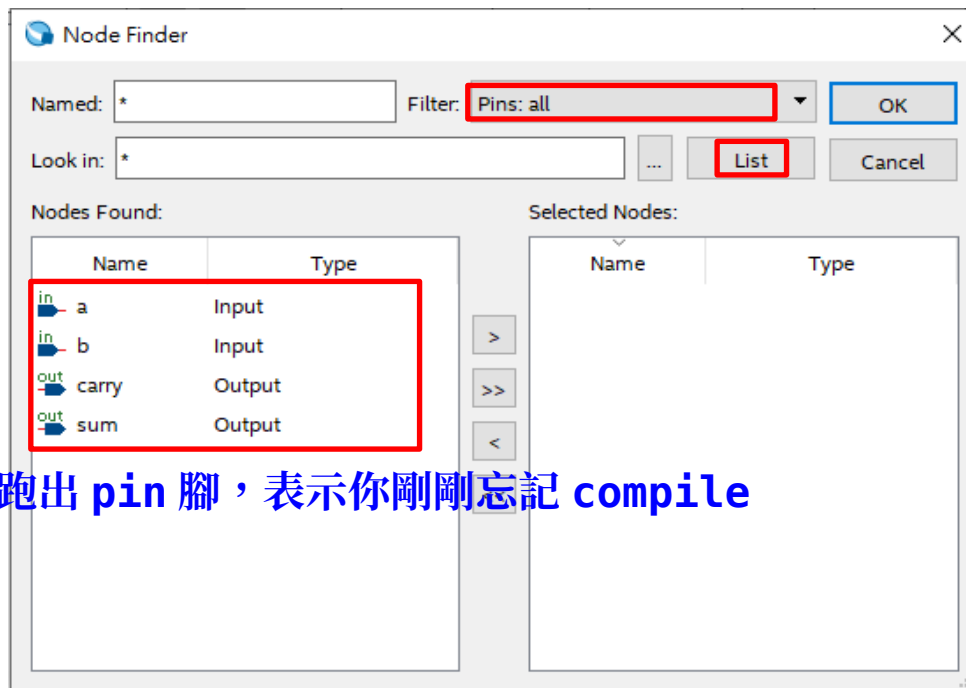
■ Simulating the Designed Circuit

- Use node finder to find all the I/O pins (Edit → Insert → Insert Node or Bus...)

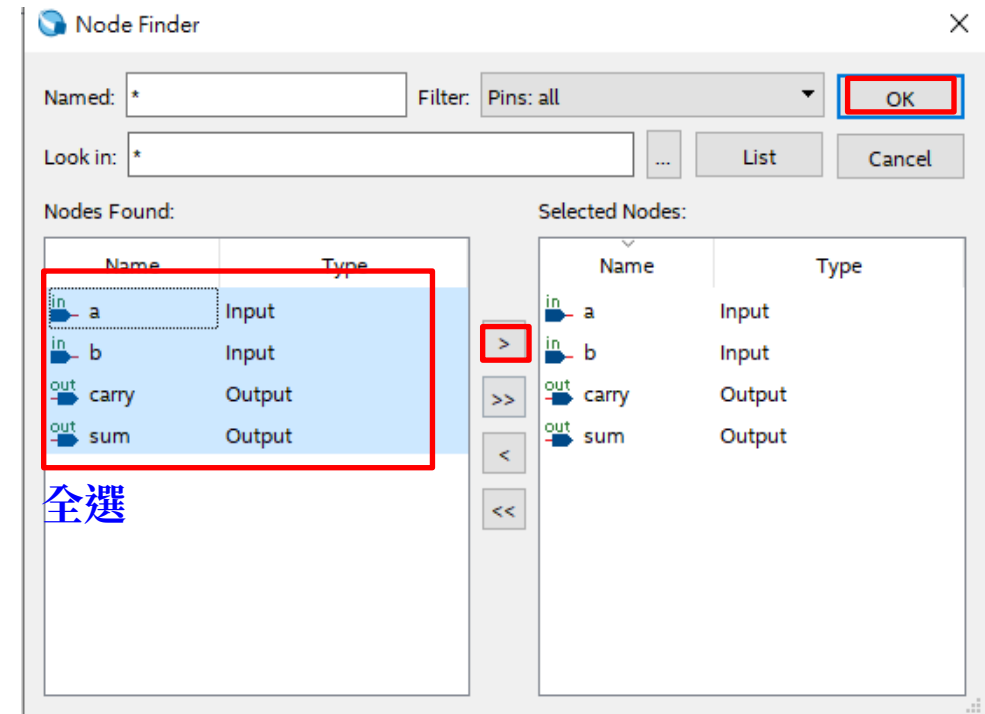
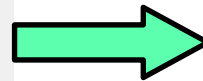


Quartus II Simulation (13/19)

- **Simulating the Designed Circuit**
 - **Selecting nodes to insert into the Waveform Editor**



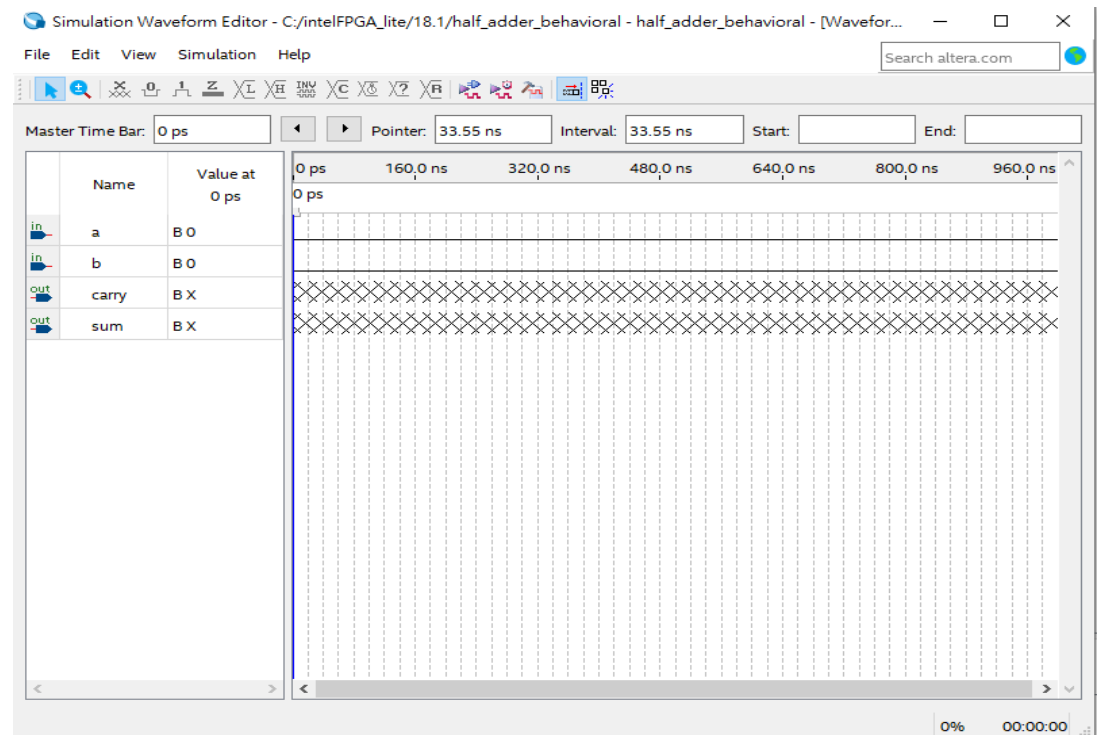
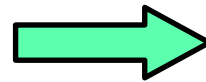
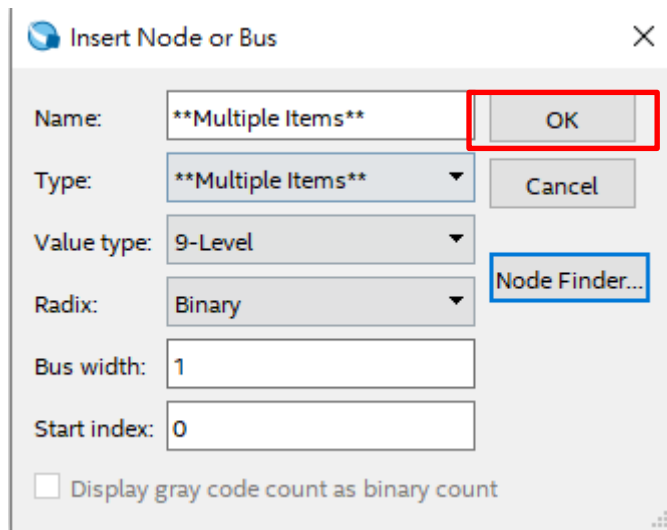
沒跑出 pin 腳，表示你剛剛忘記 compile



全選

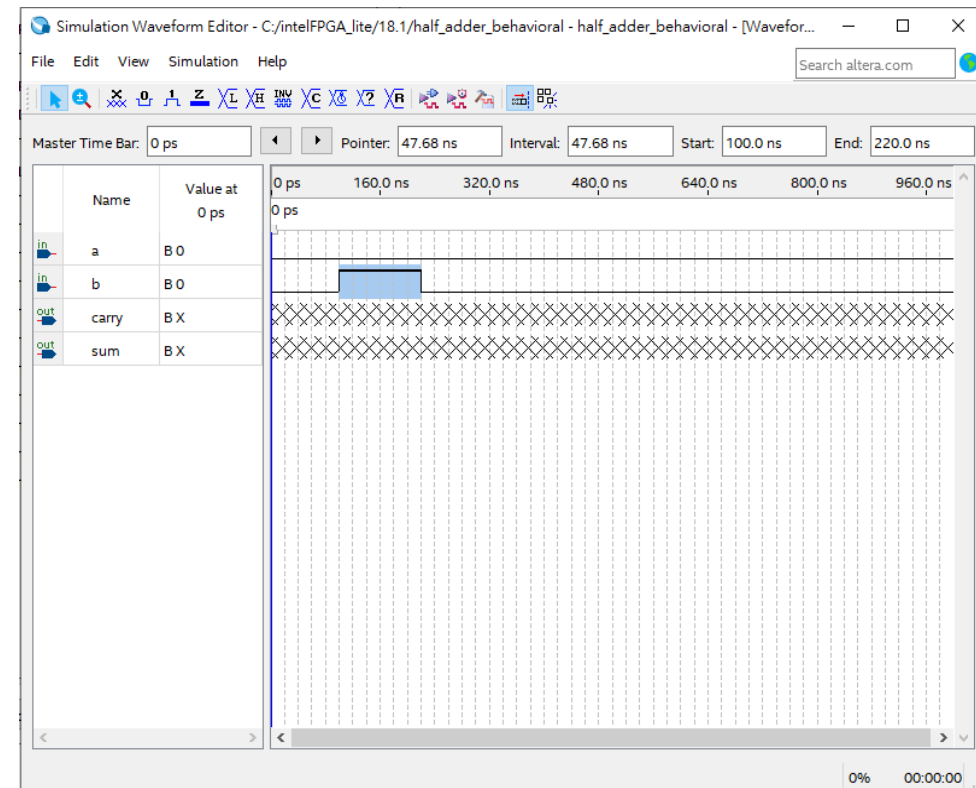
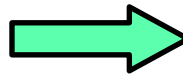
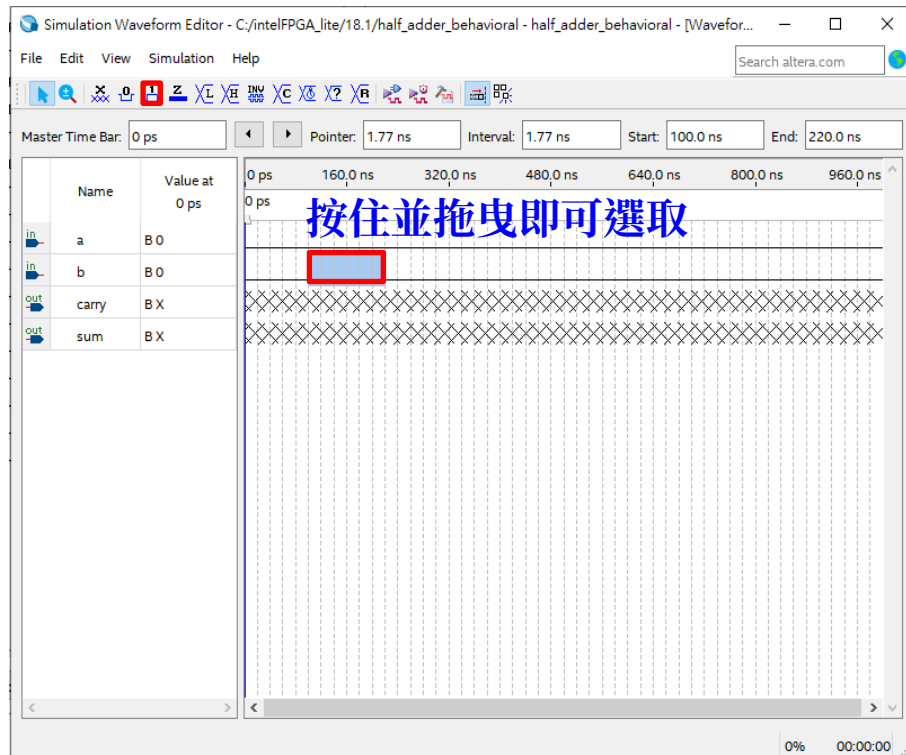
Quartus II Simulation (14/19)

- **Simulating the Designed Circuit**
 - **Selecting nodes to insert into the Waveform Editor**



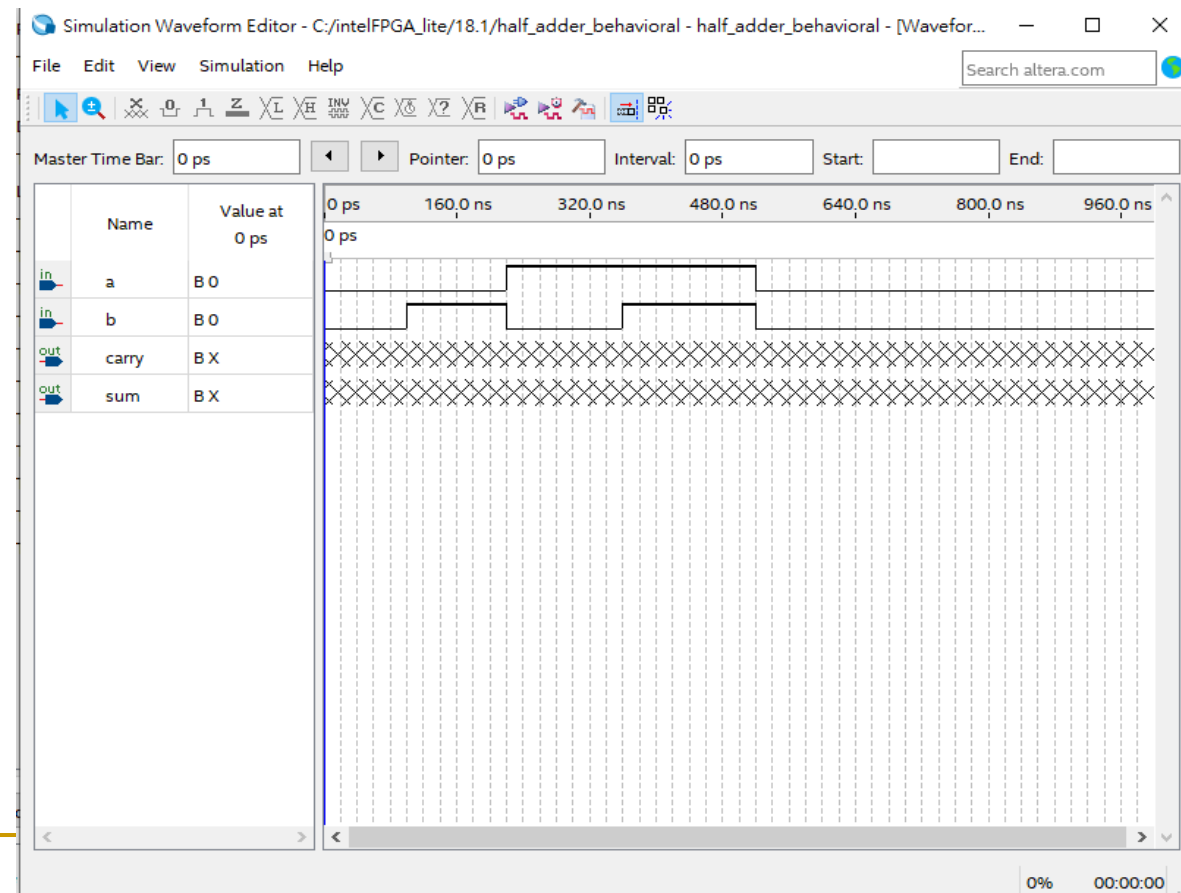
Quartus II Simulation (15/19)

- **Simulating the Designed Circuit**
 - **Select and edit waveform**



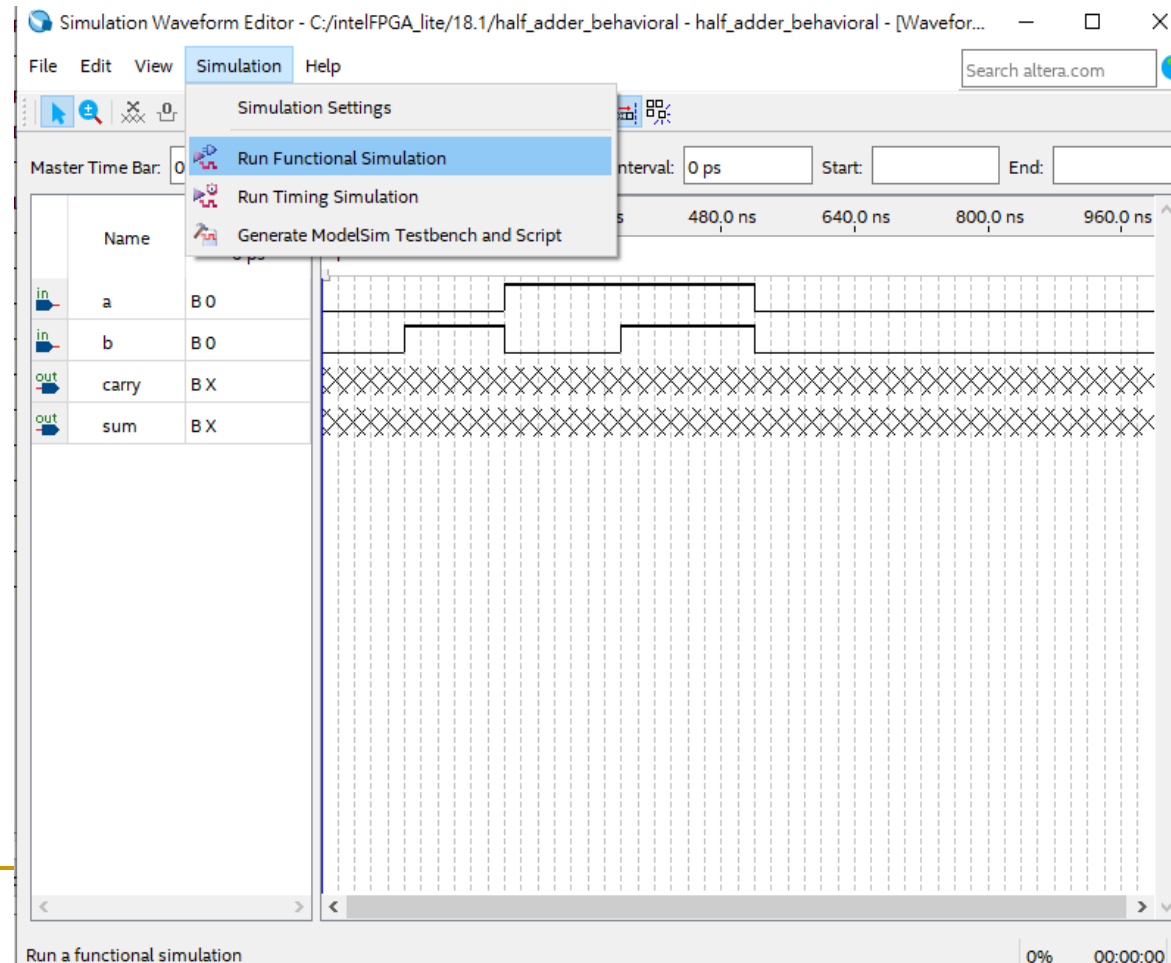
Quartus II Simulation (16/19)

- **Simulating the Designed Circuit**
 - **Setting of test values**



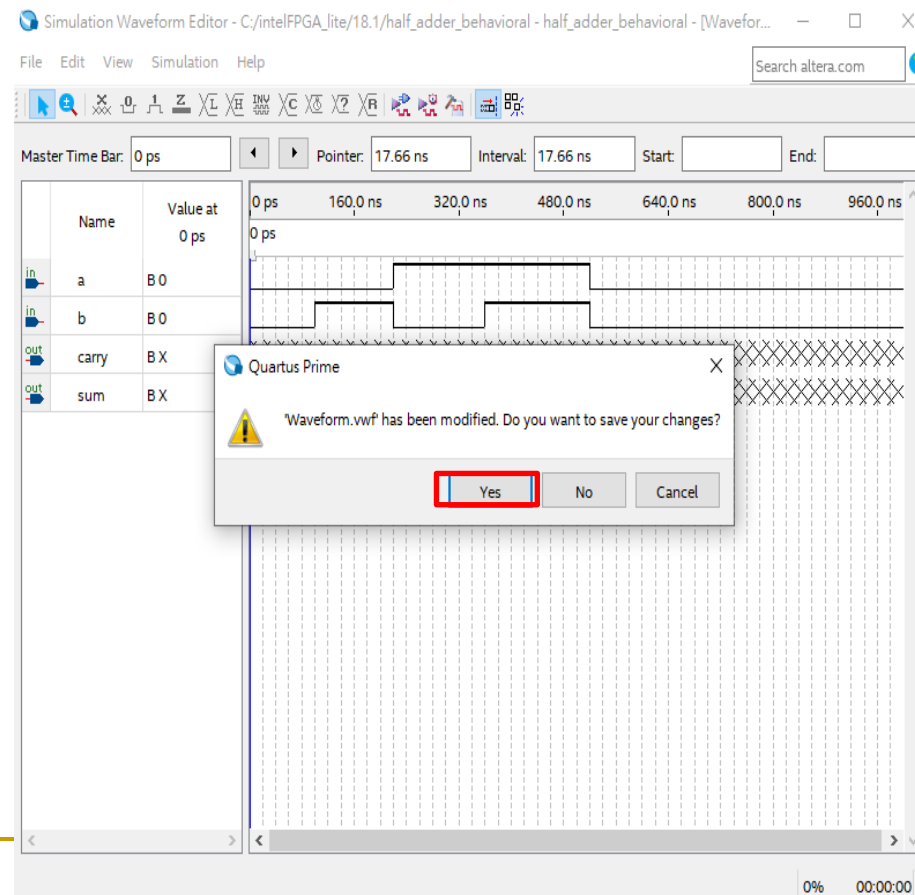
Quartus II Simulation (17/19)

- **Simulating the Designed Circuit**
 - **Run Functional Simulation**



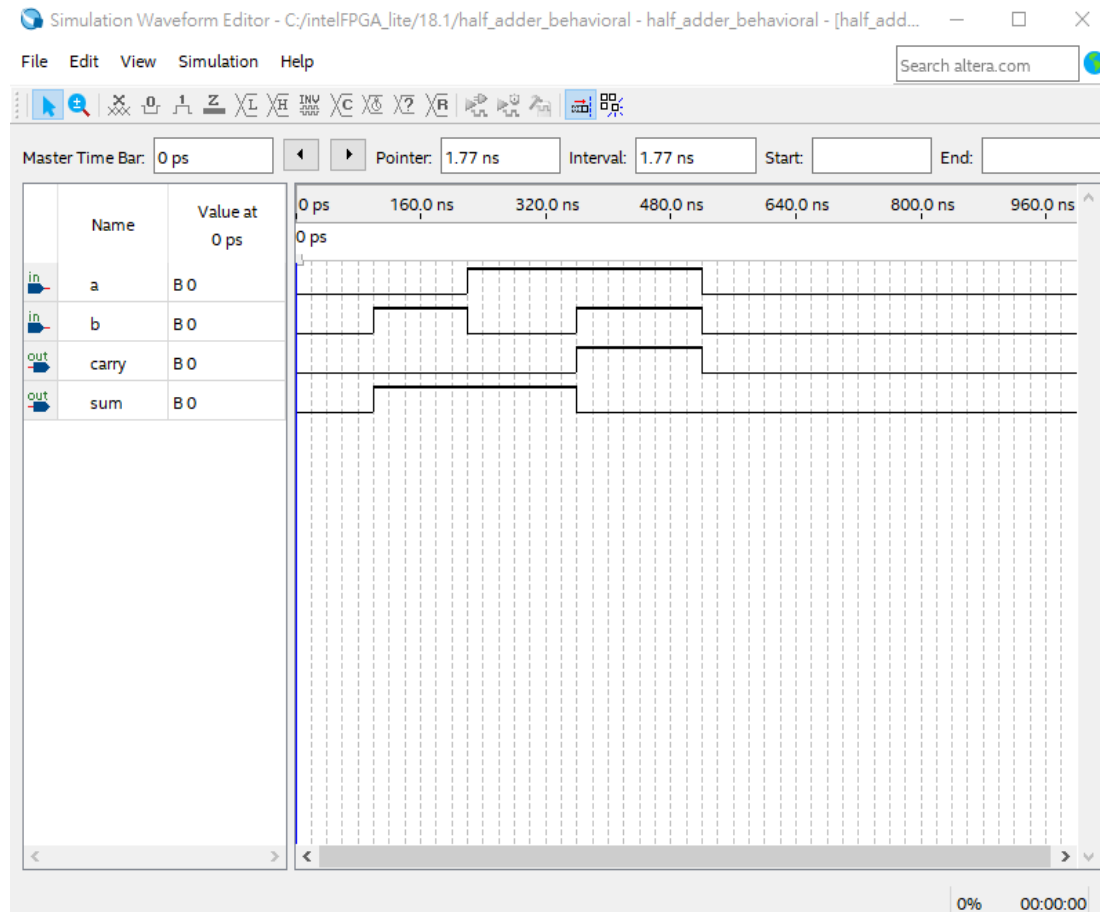
Quartus II Simulation (18/19)

- **Simulating the Designed Circuit**
 - **Save .vwf**



Quartus II Simulation (19/19)

■ The result of functional simulation

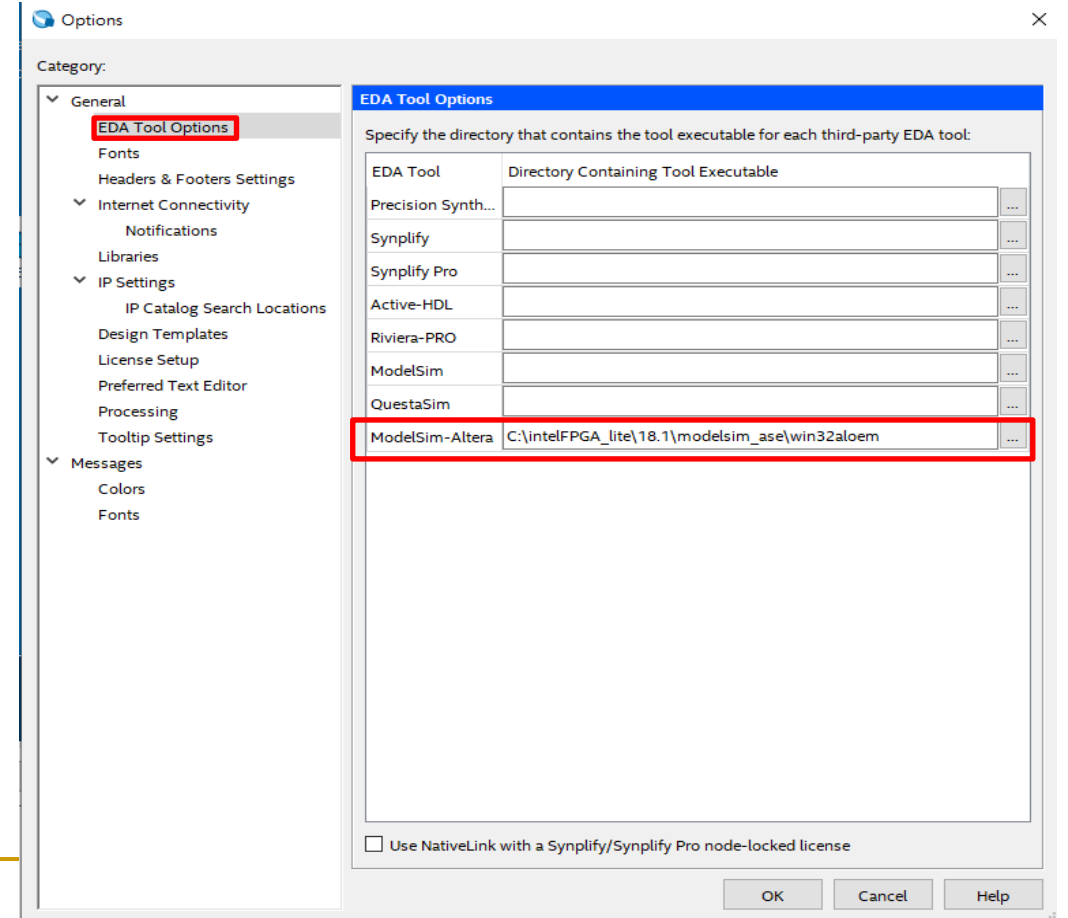
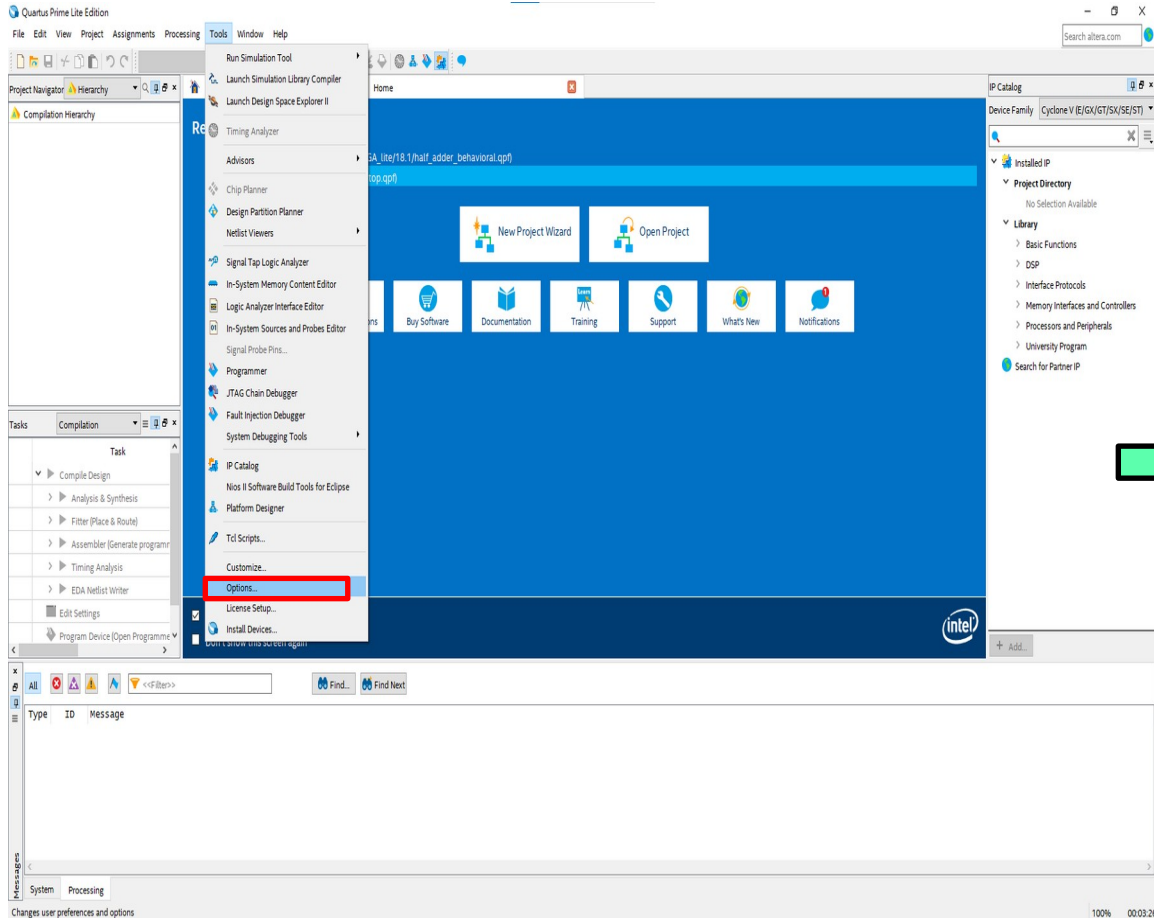


輸入 (input)		輸出 (output)	
被加數 (a)	加數 (b)	和 (sum)	進位 (carry)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Quartus II Simulation Notice

■ 如果 simulation 時出現 error ，修改 ModelSim-Altera 路徑

- 個人電腦 : (Tools → Options → EDA Tool Options → ModelSim-Altera → C:\intelFPGA_lite\18.1\modelsim_ase\win32aloem)
- 實驗室電腦 : (Tools → Options → EDA Tool Options → ModelSim-Altera → C:\intelFPGA\16.1\modelsim_ase\win32aloem)



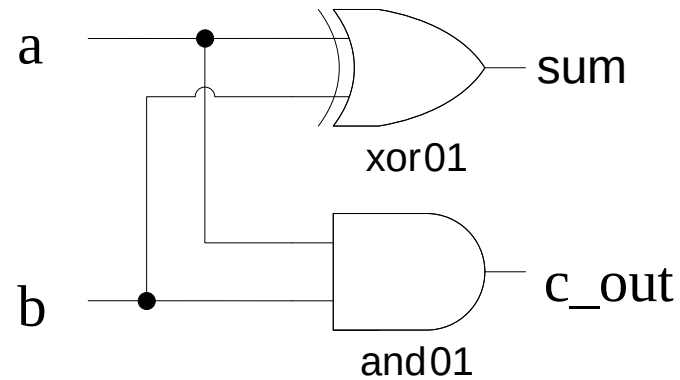
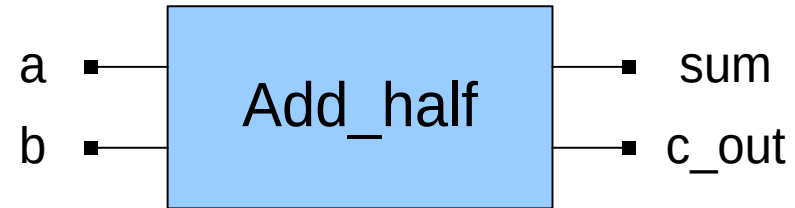
Lab I - Half Adder

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

$$\text{c_out} = ab$$



Lab I - Half Adder

- **(a) Using Verilog to implement a 1-bit Half Adder**

Please implement with 3 methods

- 1) Structural description**
- 2) Data flow description**
- 3) Behavioral description**

- **(b) Simulation with waveform**

Lab I - Hint

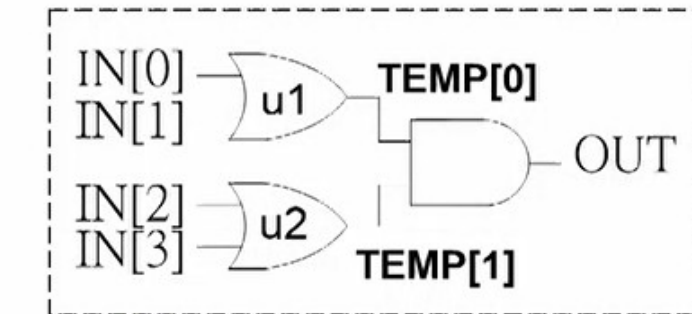
Structural Description

Verilog allows three kinds of descriptions for circuits:

(1) Structural description (2) Data flow description (3) Behavioral description

Structural description:

```
1. module OR_AND_STRUCTURAL(IN,OUT);  
  
2.   input      [3:0]    IN;  
3.   output  
4.   wire       [1:0]    TEMP;  
  
5.   or u1(TEMP[0], IN[0], IN[1]);  
6.   or u2(TEMP[1], IN[2], IN[3]);  
7.   and (OUT, TEMP[0], TEMP[1]);  
8. endmodule
```



***Synthesized (synthesis) +
optimized by tools***

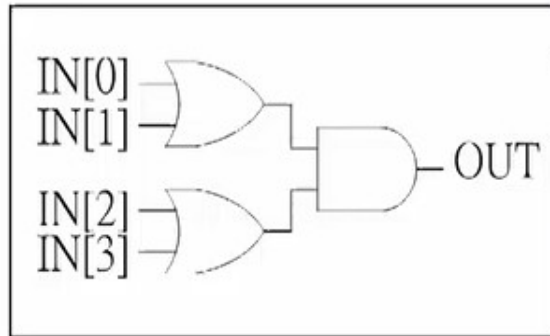
Lab I - Hint

Data Flow Description

Data flow description

```
1. module OR_AND_DATA_FLOW(IN, OUT);  
2. input    [3:0]    IN;  
3. output                      OUT;  
  
4. assign OUT = (IN[0] | IN[1]) & (IN[2] | IN[3]);  
  
endmodule
```

***Synthesized and
optimized by tools***

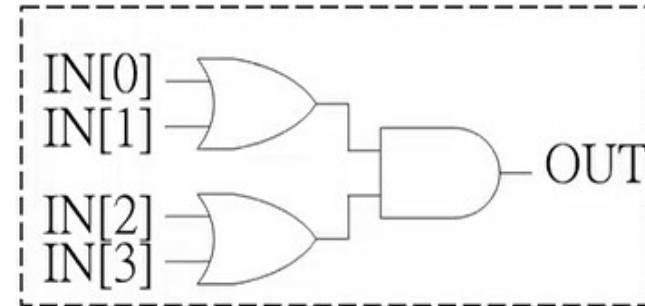


Lab I - Hint

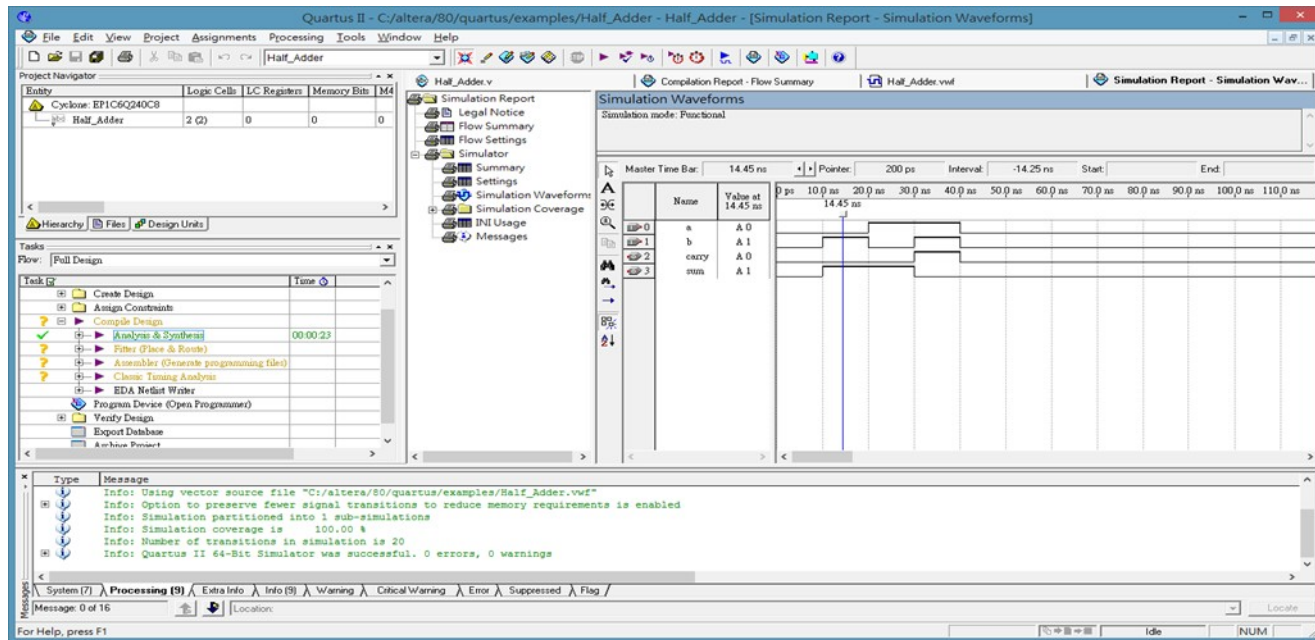
Behavioral (RTL) Description

Behavioral description

```
1. module OR_AND_BEHAVIORAL(IN, OUT);  
2.   input [3:0]  IN;  
3.   output      OUT;  
4.   reg         OUT;  
  
5.   always @(IN)  
6.   begin  
7.     OUT = (IN[0] | IN[1]) & (IN[2] | IN[3]);  
8.   end  
9. endmodule
```



Lab I - Simulation with waveform



a	b	sum	carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Lab II - Full Adder (1/2)

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

sum

$$\begin{aligned} &= \overline{a}\overline{b}c_{in} + a\overline{b}c_{in} + \overline{a}bc_{in} + abc_{in} \\ &= (\overline{a}b + a\overline{b})c_{in} + (\overline{a}b + ab)c_{in} \\ &= (a \oplus b)c_{in} + (a \oplus b)c_{in} \\ &= (a \oplus b) \oplus c_{in} \end{aligned}$$

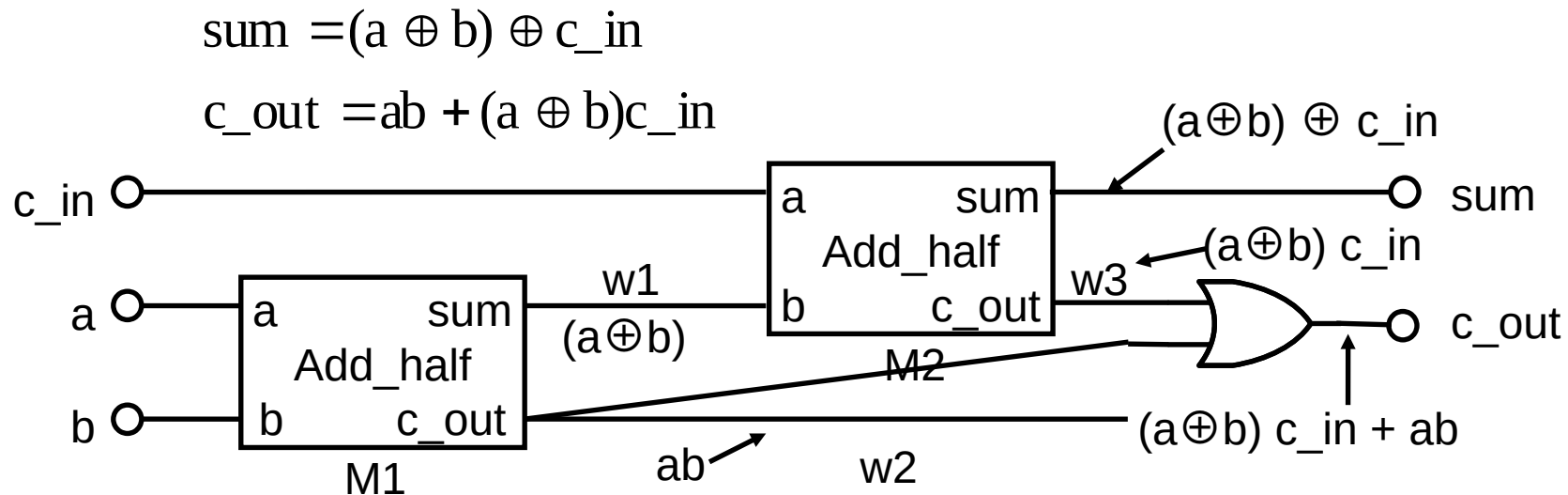
ab\c_in	0	1
00	0	0
01	0	1
11	1	1
10	0	1

c_out

$$\begin{aligned} &= ab + \overline{a}bc_{in} + a\overline{b}c_{in} \\ &= ab + (\overline{a}b + a\overline{b})c_{in} \\ &= ab + (a \oplus b)c_{in} \end{aligned}$$

Lab II - Full Adder (2/2)

You can implement a full adder with (2 half adder and 1 or gate).



Lab II - Full Adder

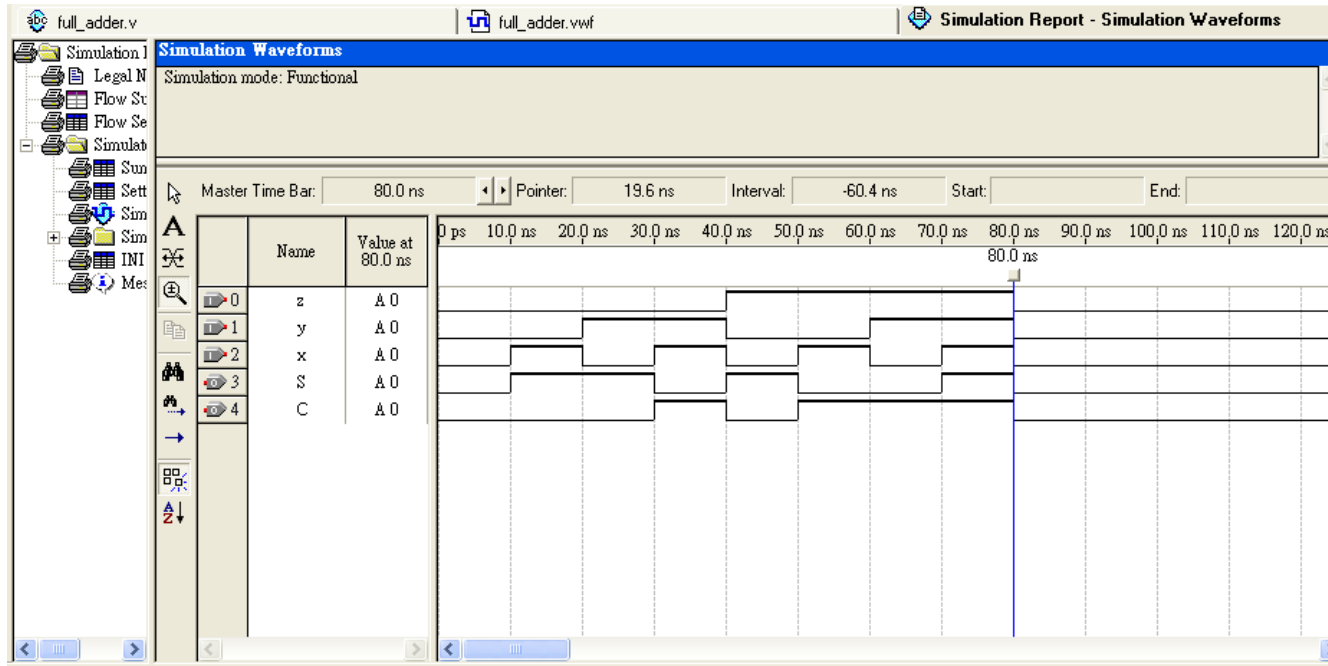
- **(a) Using Verilog to implement a 1-bit Full Adder**

Please implement with 3 methods

- 1) **Structural description(with 2 half adder + 1 or gate)**
- 2) **Data flow description**
- 3) **Behavioral description**

- **(b) Simulation with waveform**

Lab II - Simulation with waveform



z	y	x	S	C
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Notice

- 請勿在桌面建立 Project 及請勿命名中文資料夾
- Device family 請確認與 FPGA Chip 符合 (10M50DAF484C7G)
- Top module name & Project name 需要一致
- 確認 `module ... endmodule` 為 keyword 變成藍色字體