

# Lab0

## Hello FPGA

黃稚存



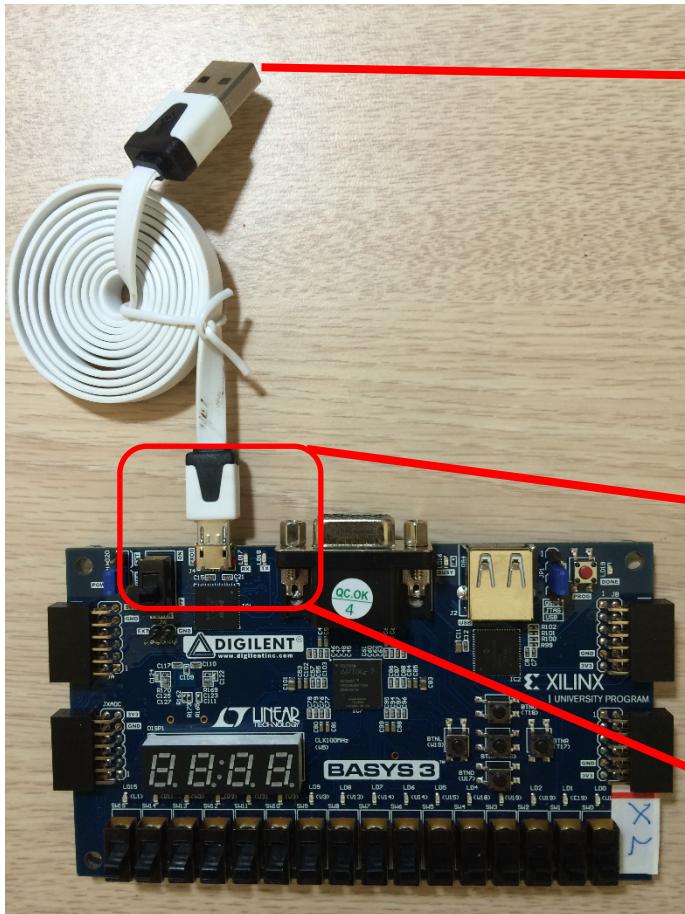
國立清華大學  
資訊工程學系

# Today's work

---

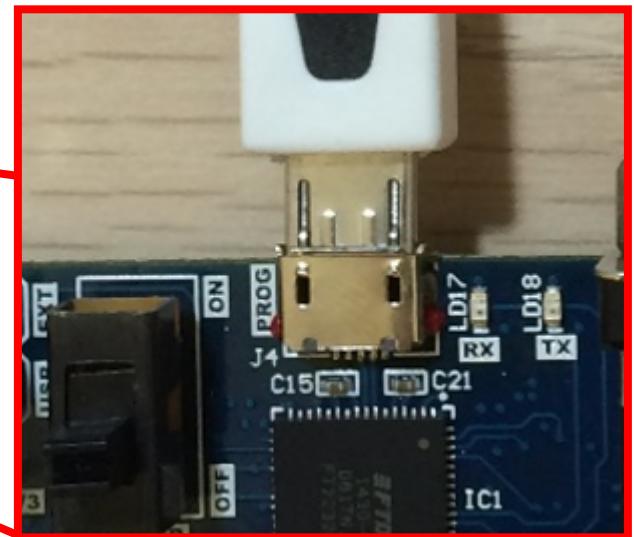
- 1. Get the FPGA board and USB cable
  - ◆ Sign your name after receiving the board
  - ◆ Make sure both of them work
- 2. Run lab0 and demo
  - ◆ Run simulation and play the game on the board
  - ◆ No report this time
  - ◆ You have to write a report start from lab1

# Attention



Only 5V DC

Please mind the *direction* !!



# Introduce Basys3 Demo Board

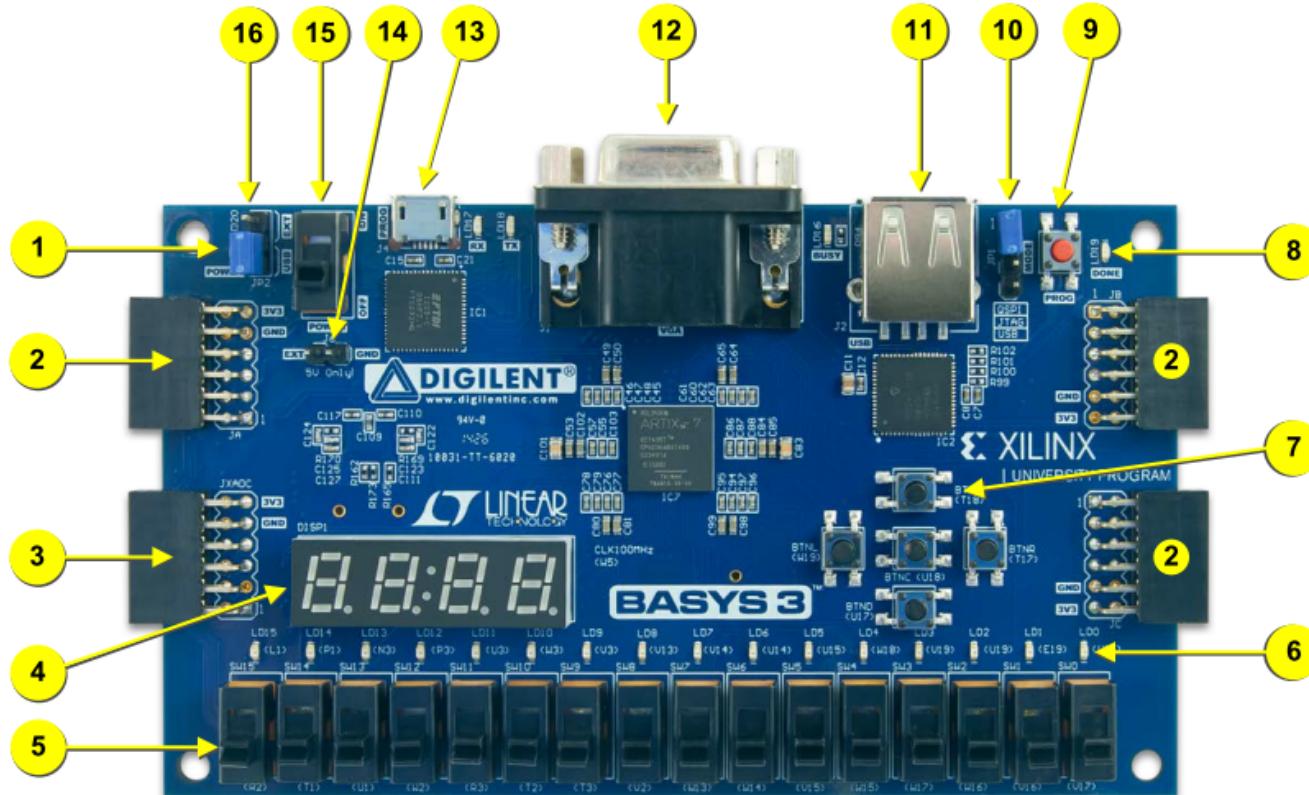


Figure 1. Basys3 FPGA board with callouts.

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

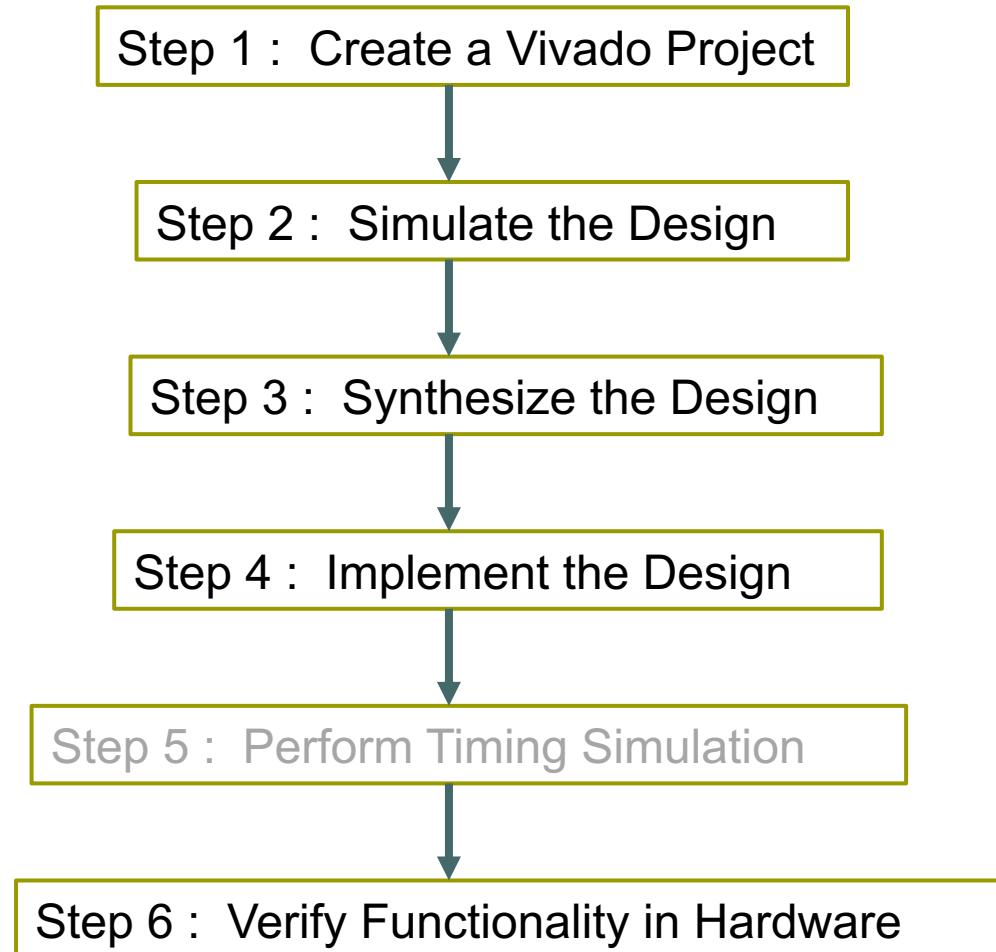
# Download Source

- You can download the lab materials (slides and source code) on iLMS.

The screenshot shows a course management system interface. On the left, a sidebar lists course functions: 課程公告 (Announcements), 上課教材 (Lesson Materials), 課堂整理 (Class Organization), 課程說明 (Course Description), 課程行事曆 (Course Calendar), 討論區 (Discussion Zone), and 小組專區 (Group专区). The '上課教材' item is highlighted with a red box. The main content area displays a list of lesson materials with the following details:

編號	標題	發表人	人氣	討論	更新時間
2007822	Supplementary: IEEE 1364-2005 Verilog Standard	黃稚存	2	0	09-11 23:28
2007820	Supplementary: Vivado Design Suite User Guide	黃稚存	1	0	09-11 23:27
2007819	Supplementary: Basys3 FPGA manual	黃稚存	1	0	09-11 23:25
2007817	01 A Quick Review of Verilog Modeling	黃稚存	2	0	09-11 23:24
2006618	00-1 Pre-Lab: How to Install Xilinx Vivado	黃稚存	51	0	09-10 22:58
2006604	00 Syllabus	黃稚存	31	0	09-10 22:43

# General Design Flow

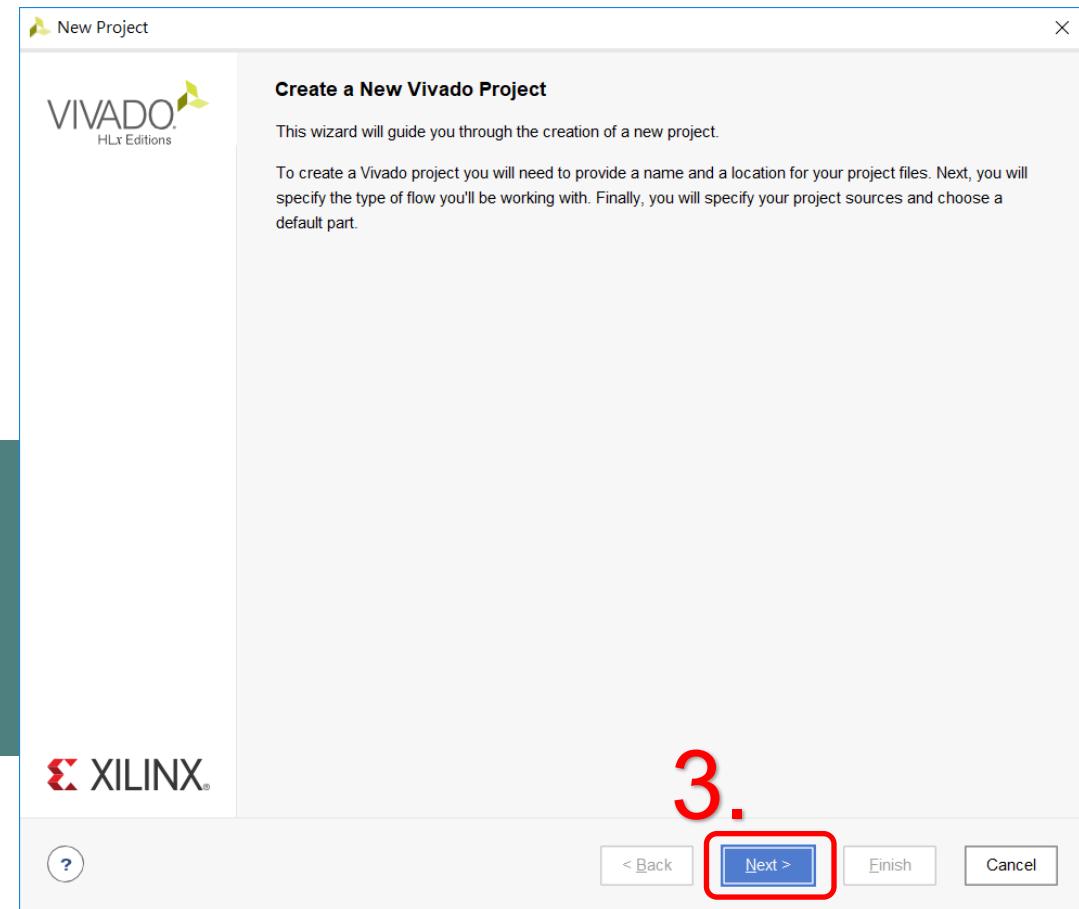
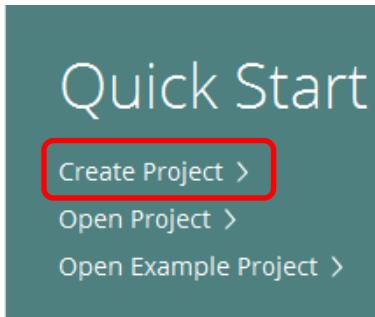


# Create New Project (1/8)

1.  
Click this icon



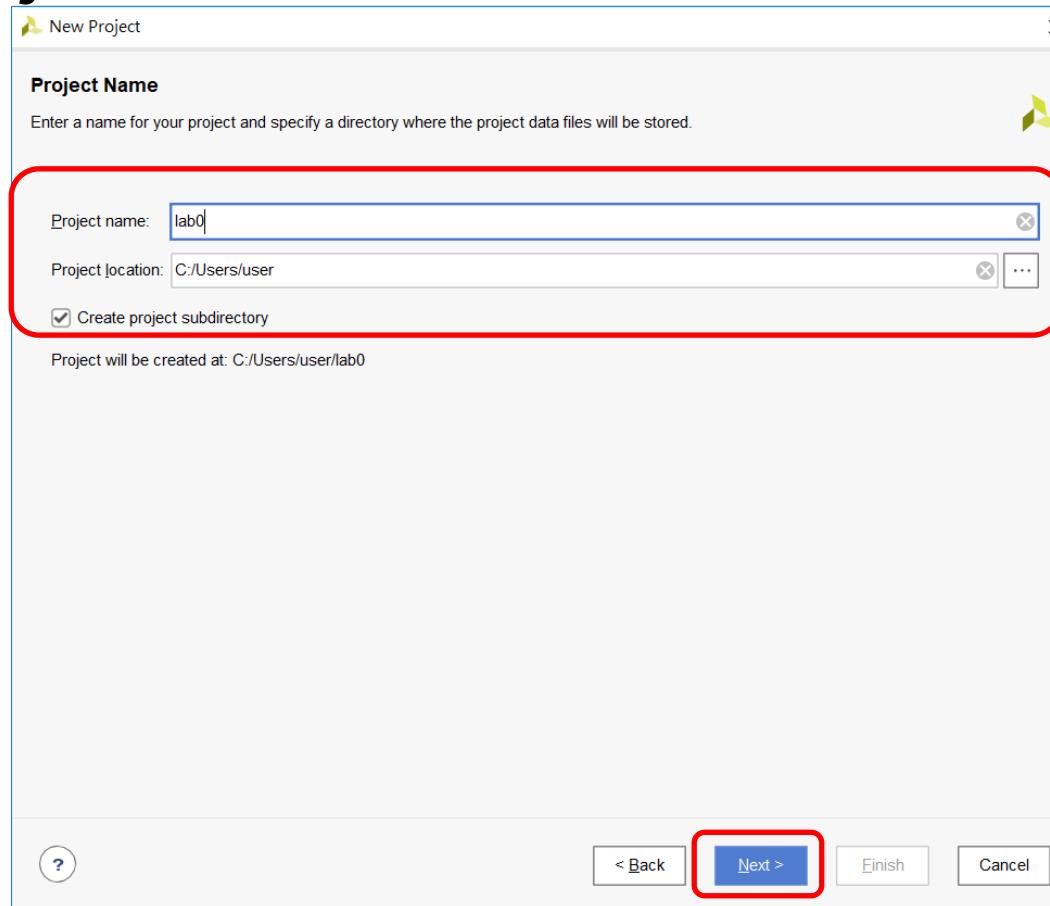
2.



3.

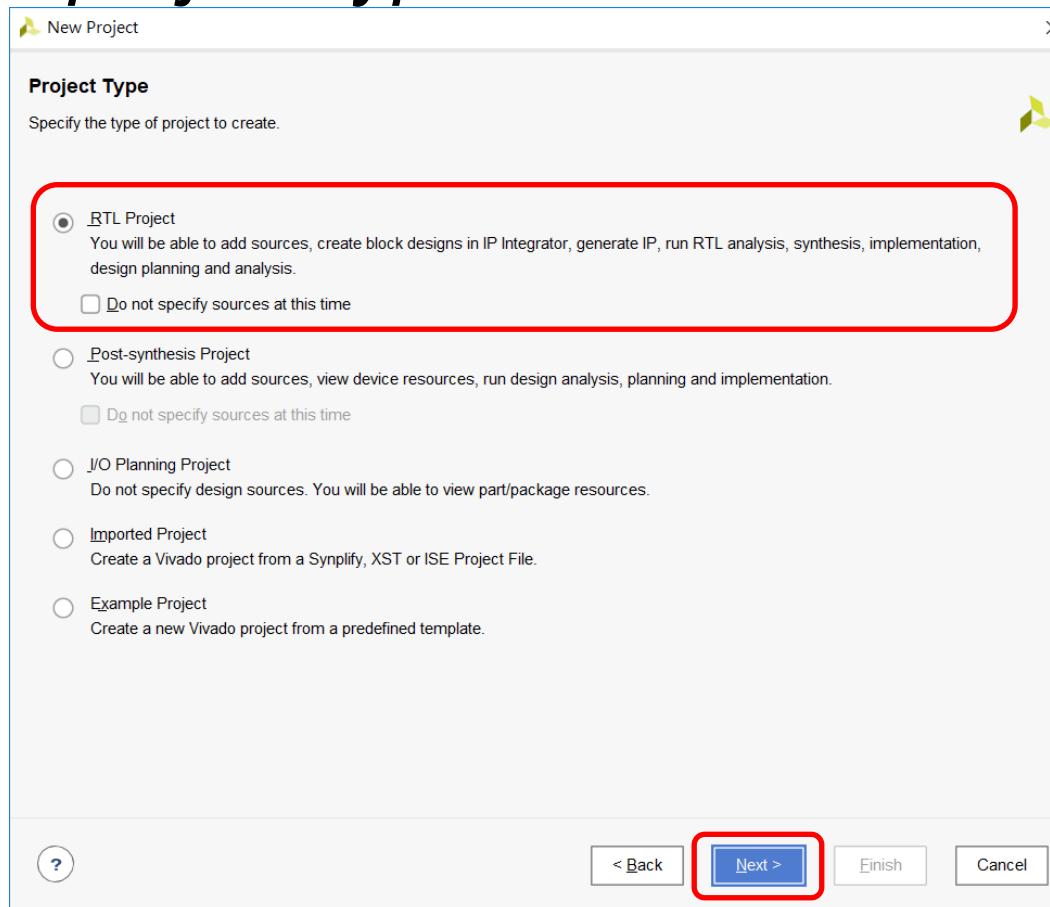
# Create New Project (2/8)

- Set *project name* and *location*



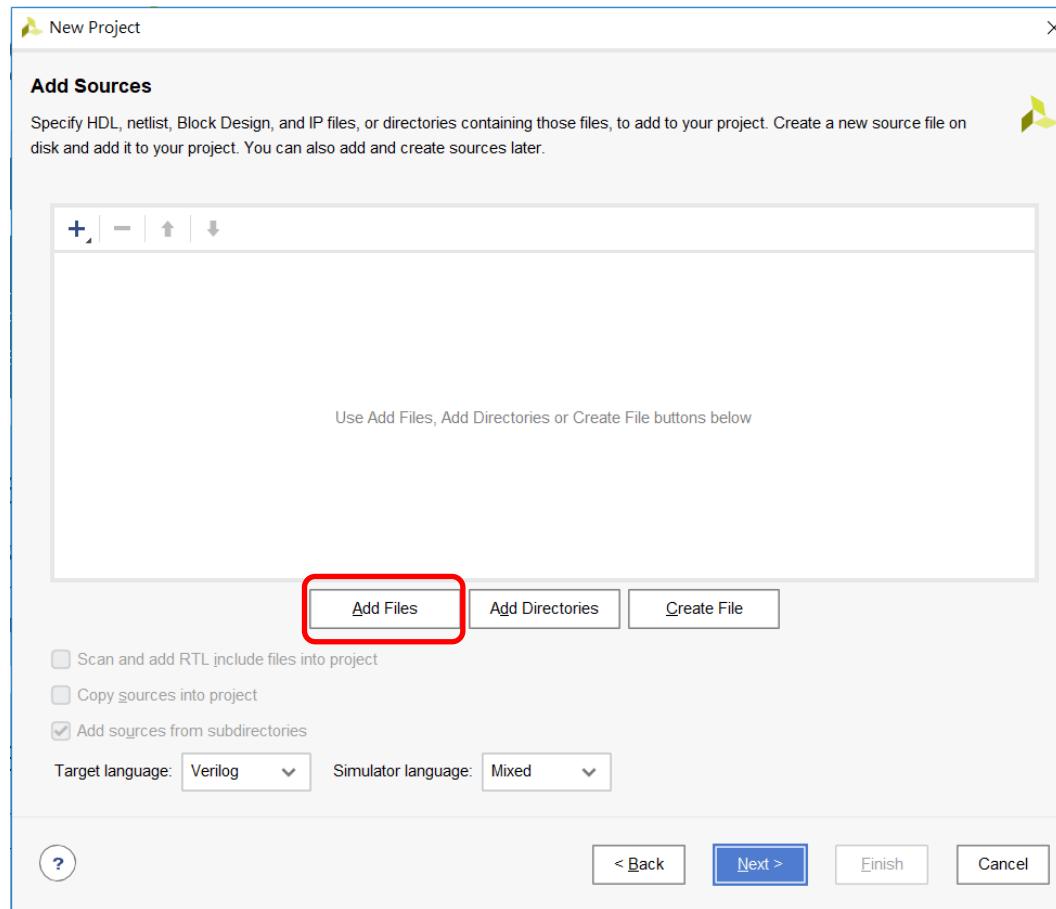
# Create New Project (3/8)

- Choose *project type*



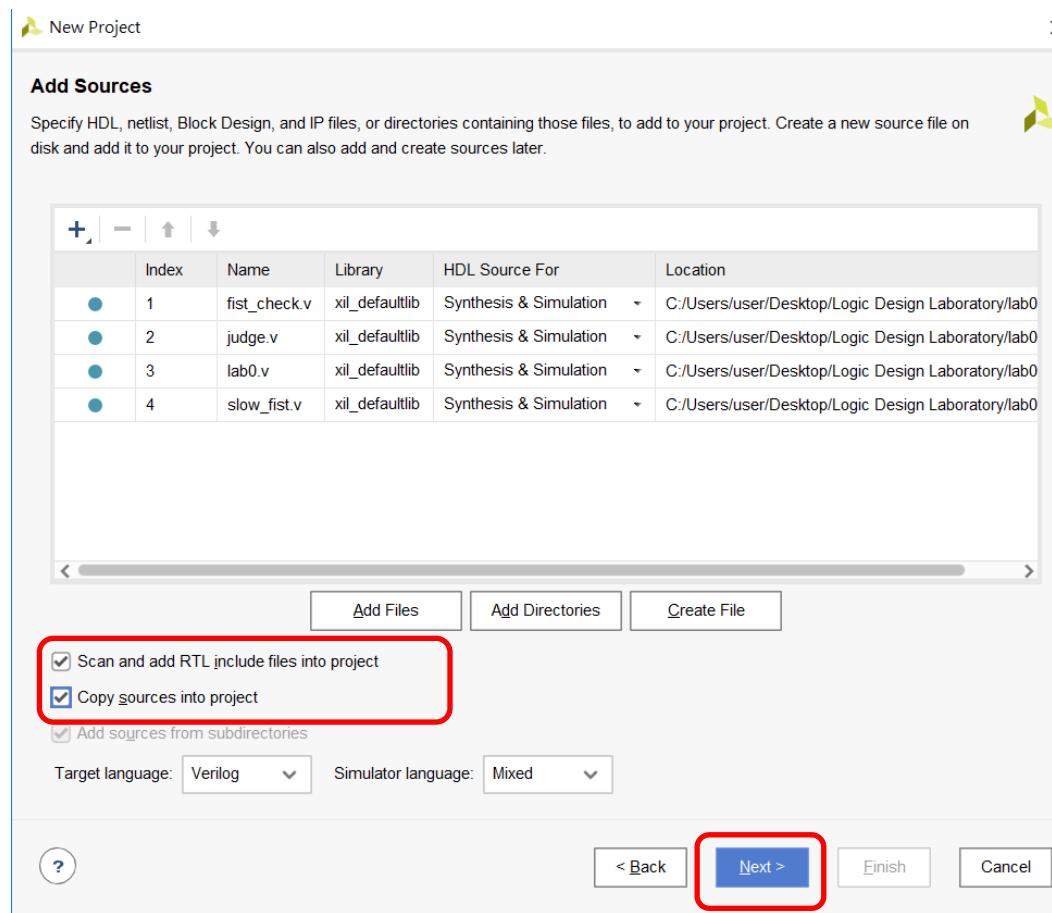
# Create New Project (4/8)

- Add all .v files expect “lab0\_t.v”

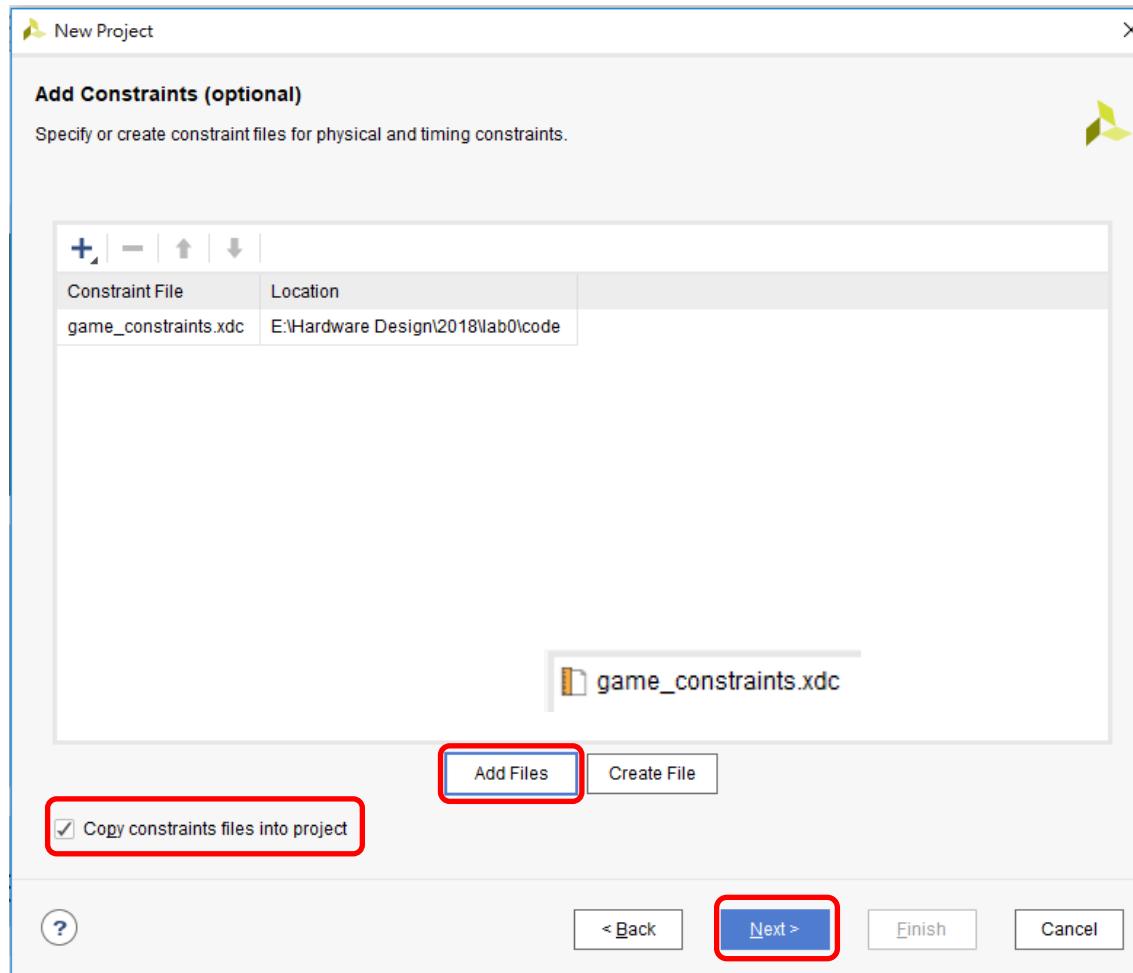


# Create New Project (5/8)

- Add all .v files expect “lab0\_t.v”



# Create New Project (6/8)

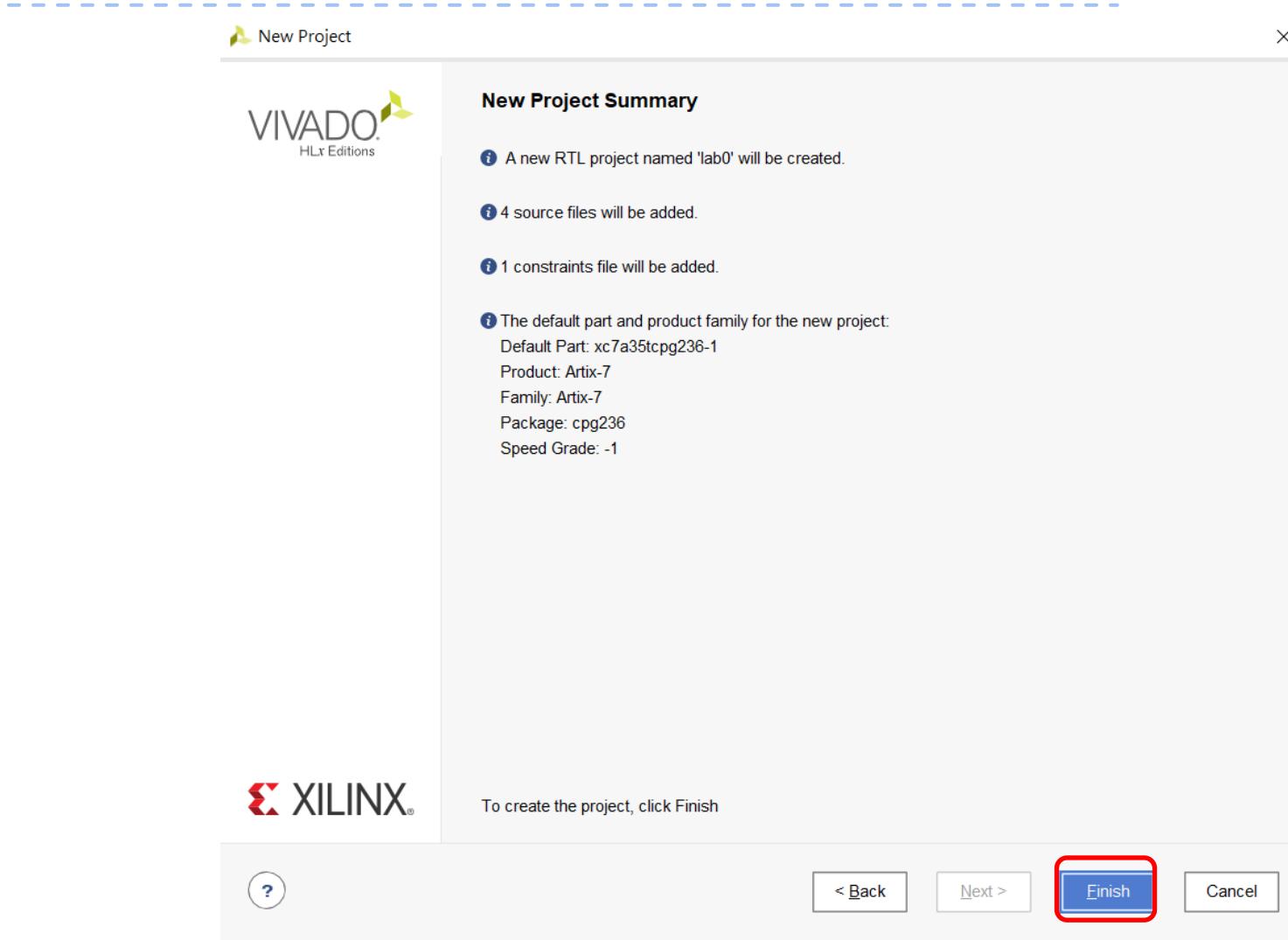


# Create New Project (7/8)

The screenshot shows the 'New Project' dialog with the title 'Default Part'. The heading 'Choose a default Xilinx part or board for your project.' is displayed. Below it, there are two tabs: 'Parts' (selected) and 'Boards'. A red box highlights the filter section at the top, which includes dropdowns for Category (All), Family (Artix-7), Package (cpg236), Speed (-1), Temperature (All Remaining), and Static power (All Remaining). A search bar labeled 'Search: Q-' is also present. A second red box highlights the table below, which lists three Xilinx parts: xc7a15tcpg236-1, xc7a35tcpg236-1, and xc7a50tcpg236-1. The xc7a35tcpg236-1 row is selected. At the bottom, there are buttons for '?', '< Back' (disabled), 'Next >', 'Finish' (disabled), and 'Cancel'.

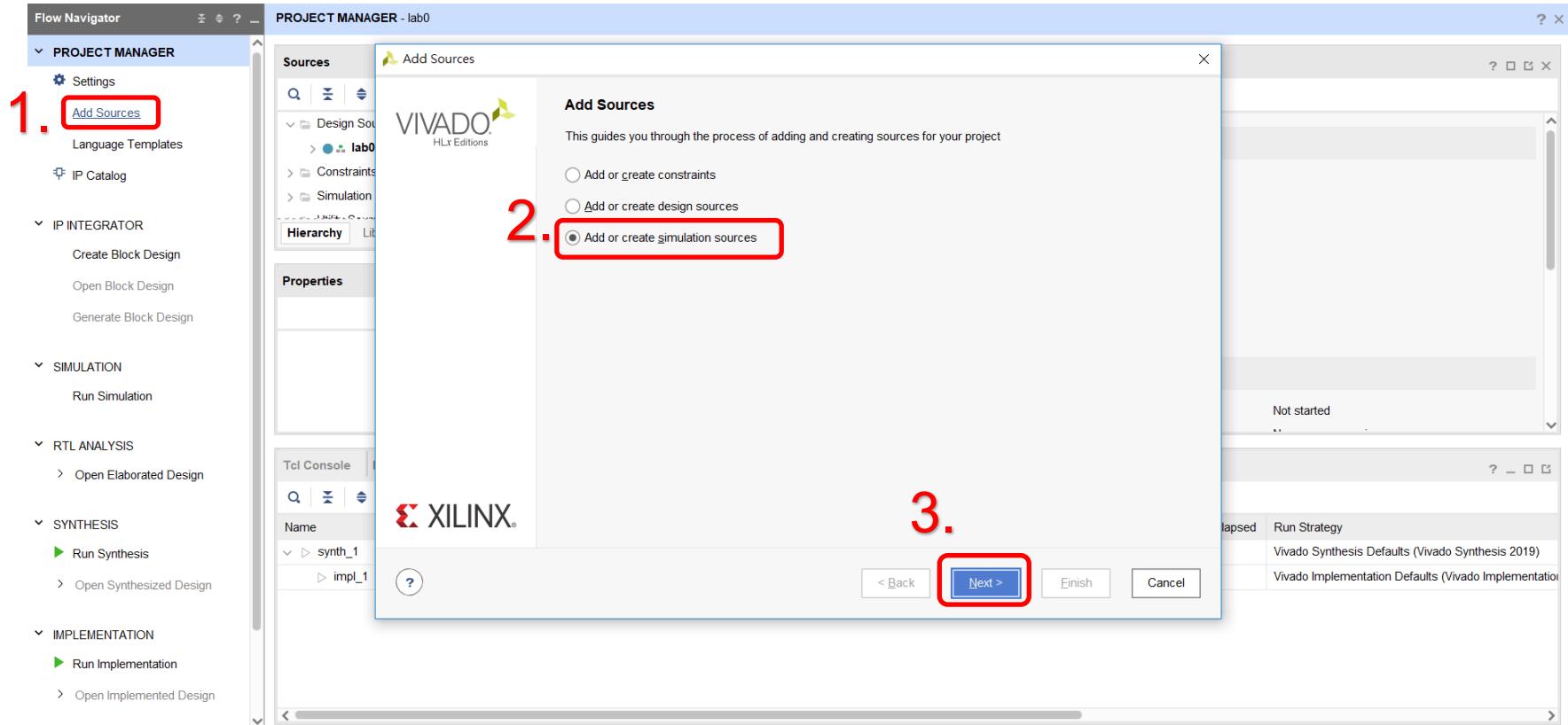
Part	I/O Pin Count	Available IOBs	LUT Elements	FlipFlops	Block RAMs	Ultra RAMs	DSPs	Gb Tran
xc7a15tcpg236-1	236	106	10400	20800	25	0	45	2
xc7a35tcpg236-1	236	106	20800	41600	50	0	90	2
xc7a50tcpg236-1	236	106	32600	65200	75	0	120	2

# Create New Project (8/8)

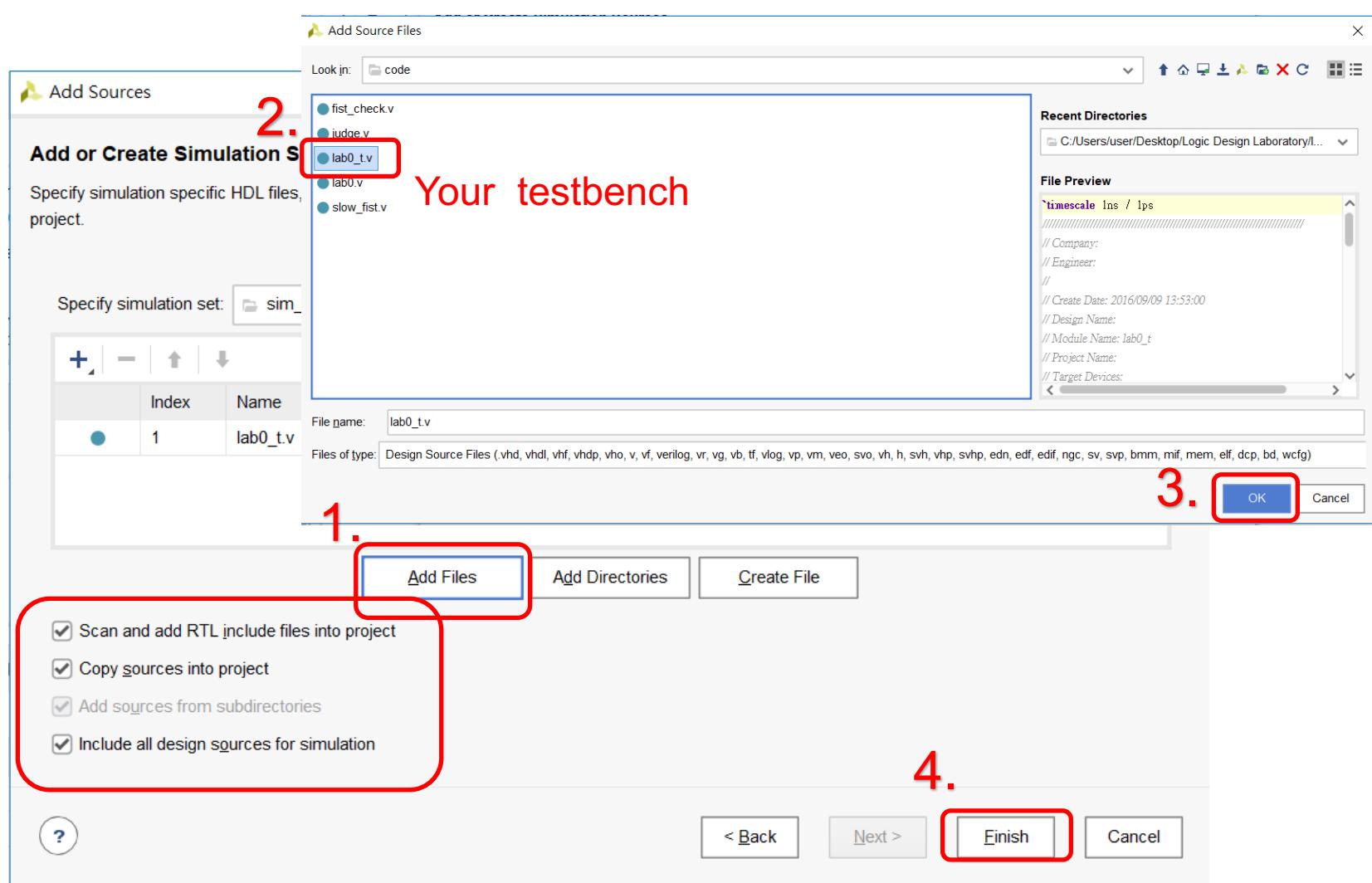


# Simulation (1/5): Add Sources

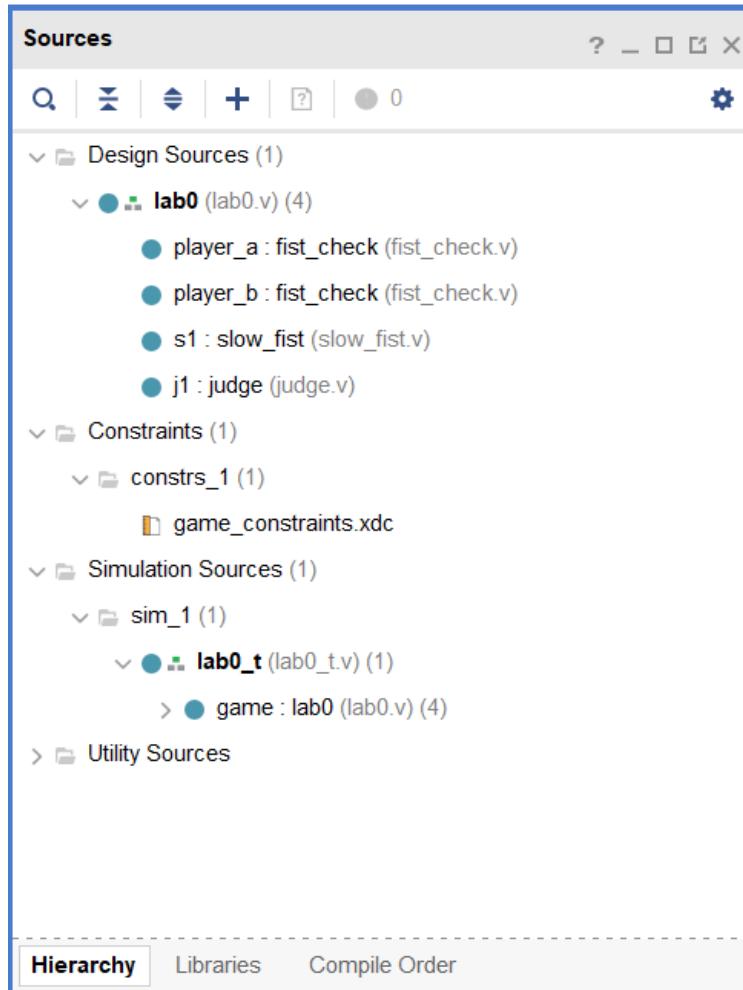
- Add sources (e.g., design, testbench, etc.)



# Simulation (2/5): Add Files



# Simulation (3/5): Add Files



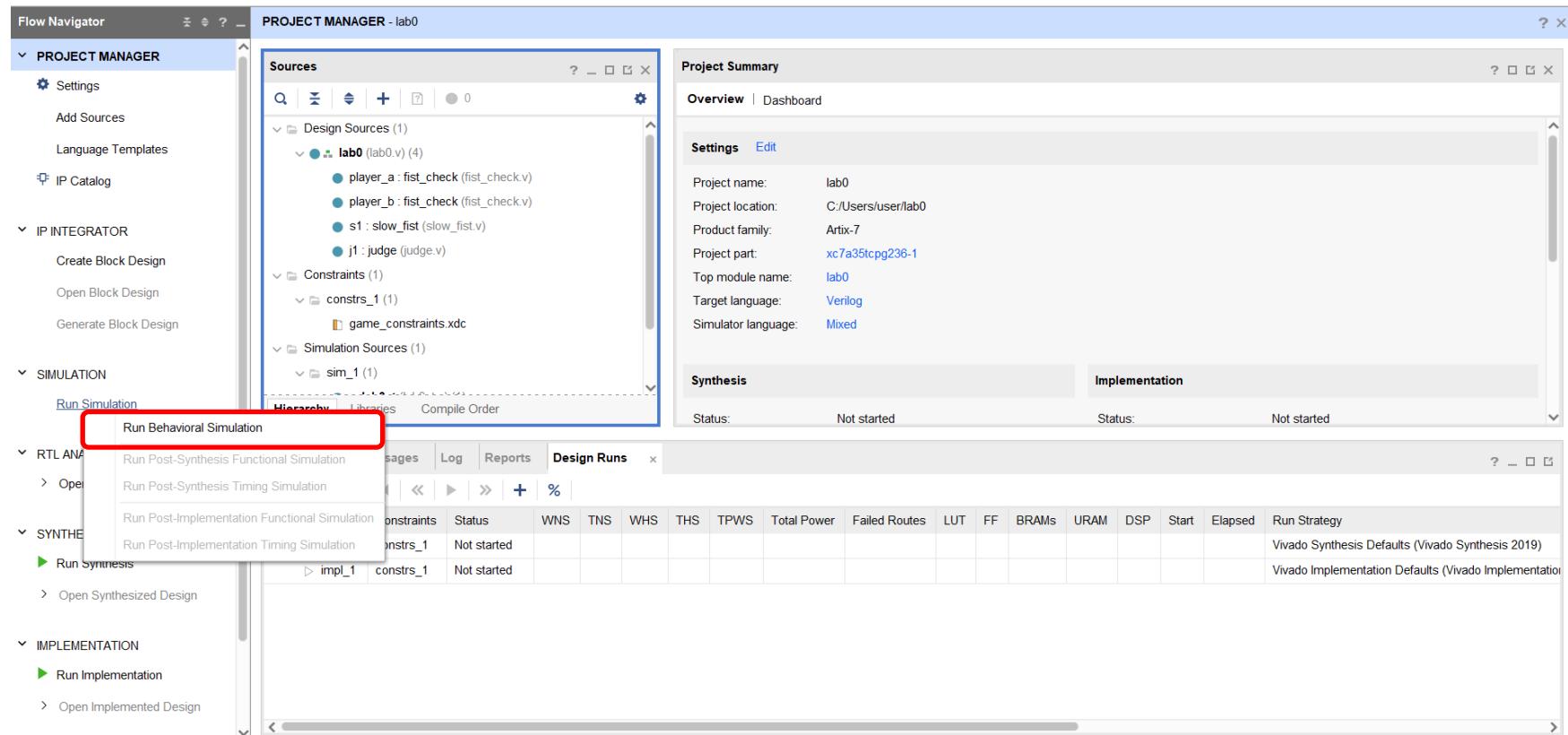
Hierarchy

Libraries

Compile Order

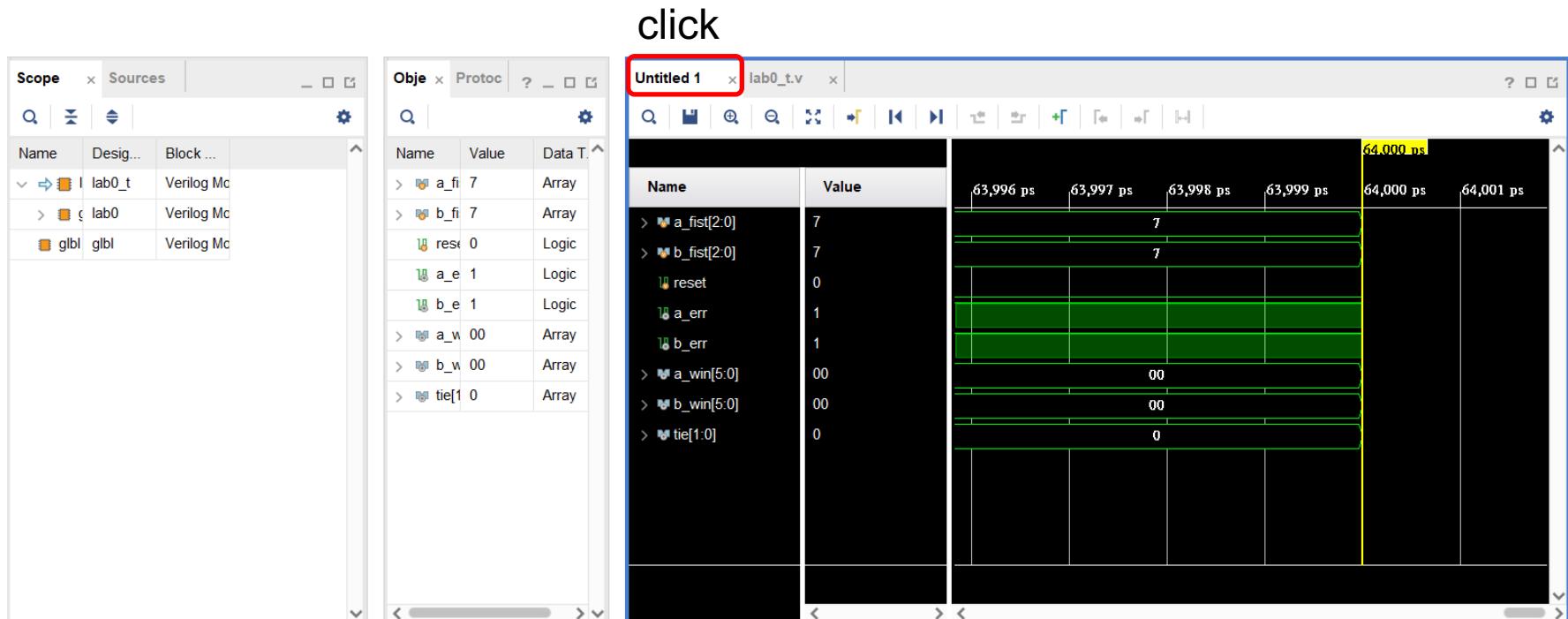
# Simulation (4/5): Run Simulation

- Run Simulation -> Run behavioral Simulation



# Simulation (5/5): Check Signals

- Show waveform

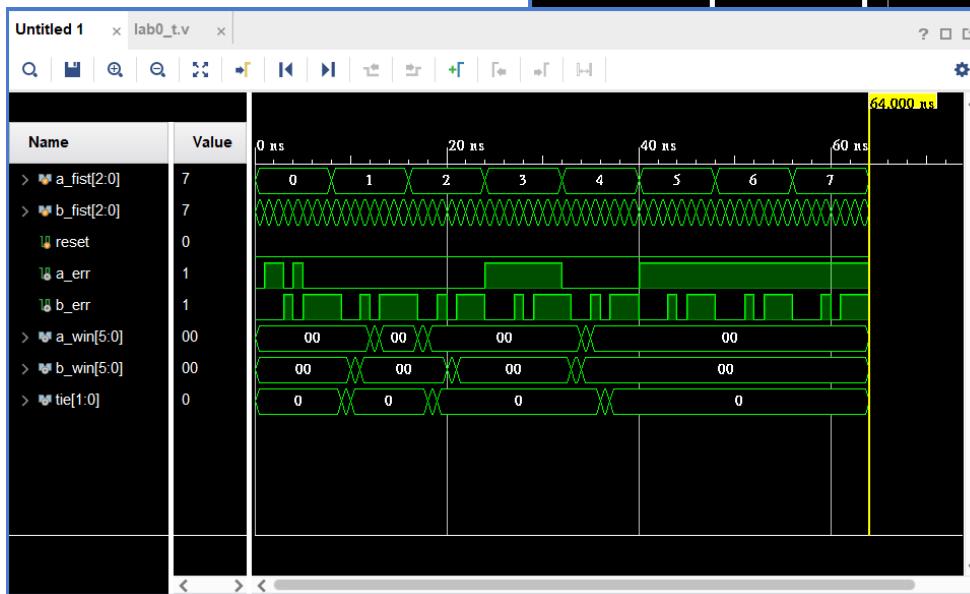
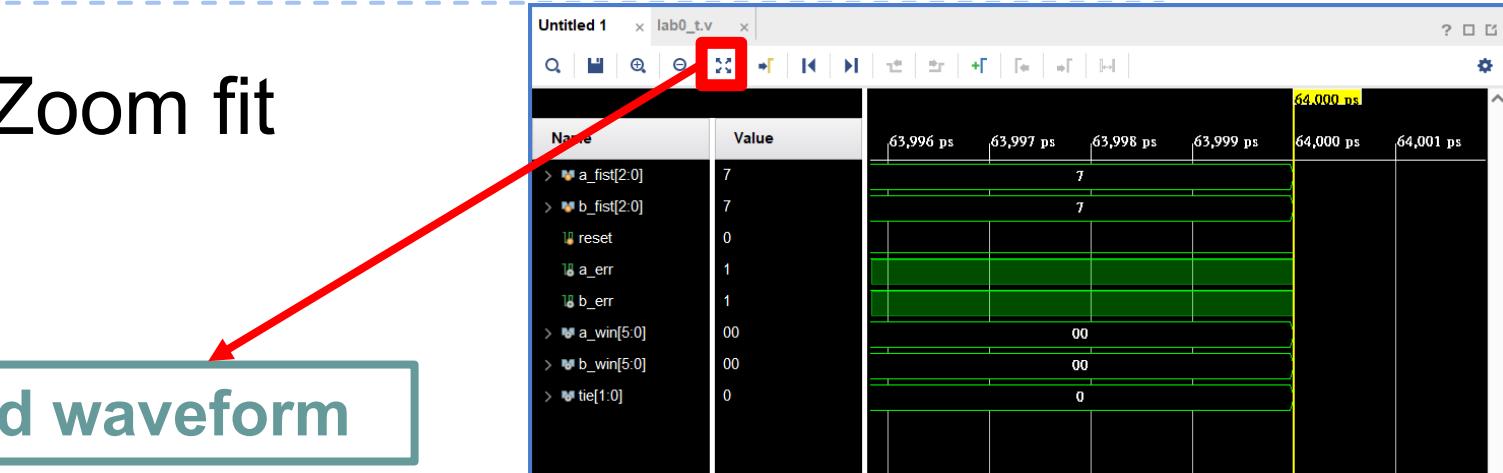


# Debugging (1/4)

Unfitted waveform

- Zoom fit

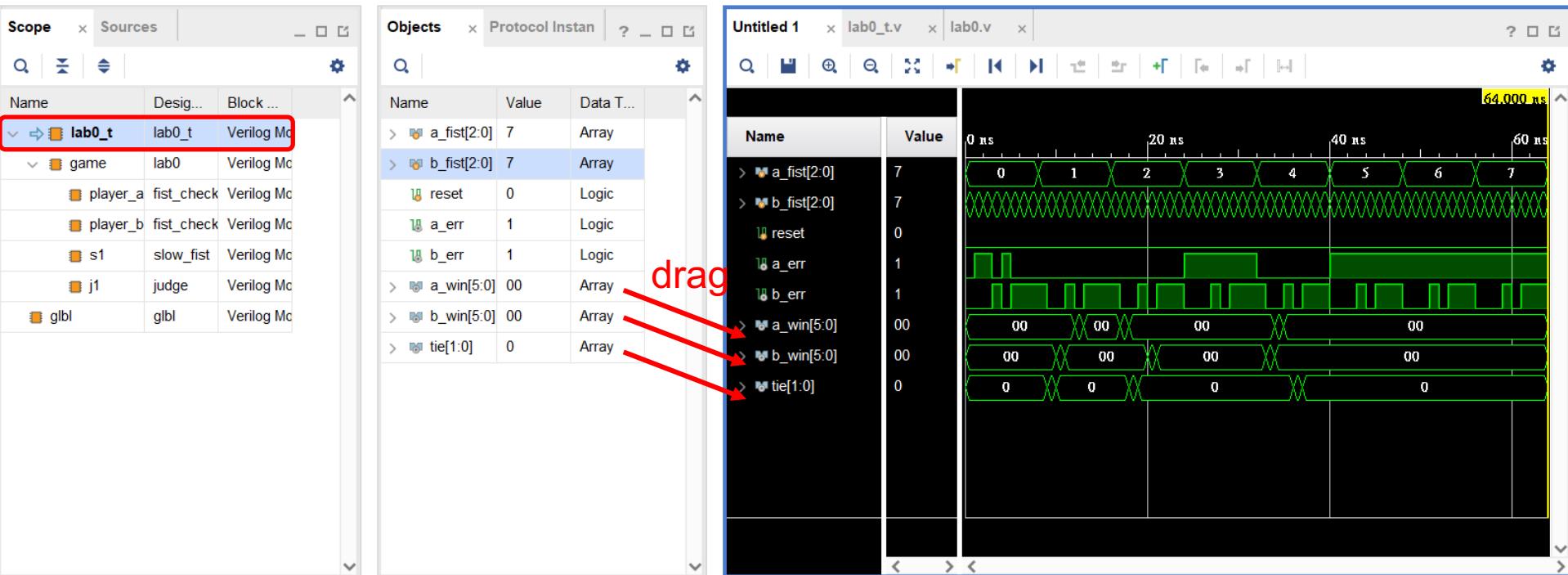
fitted waveform



# Debugging (2/4)

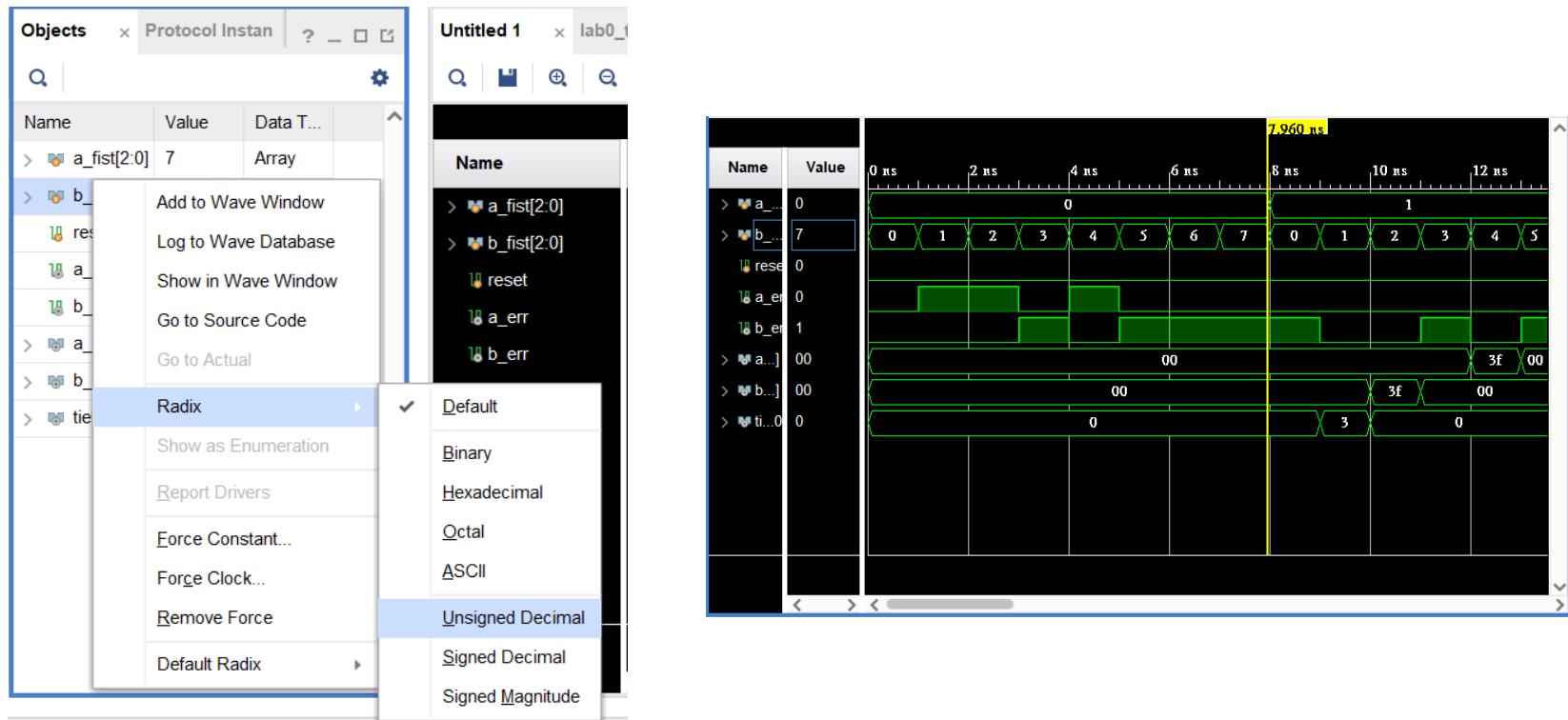
- Reorder (and/or modify) the signals by  and delete.

Choose the *block* that you want to see



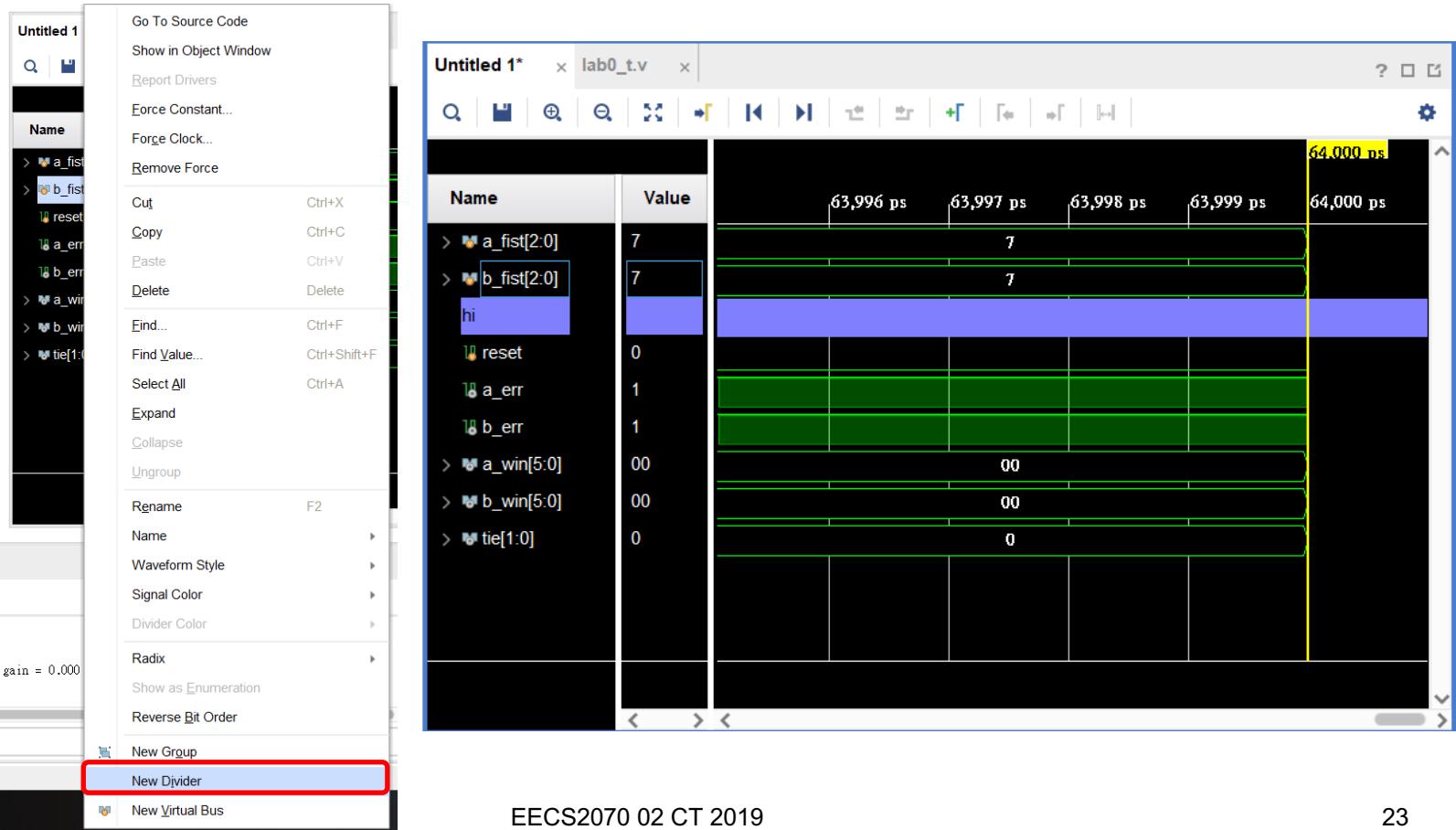
# Debugging (3/4)

- Right click to open the popup menu again, and select *Radix* > *Unsigned Decimal*



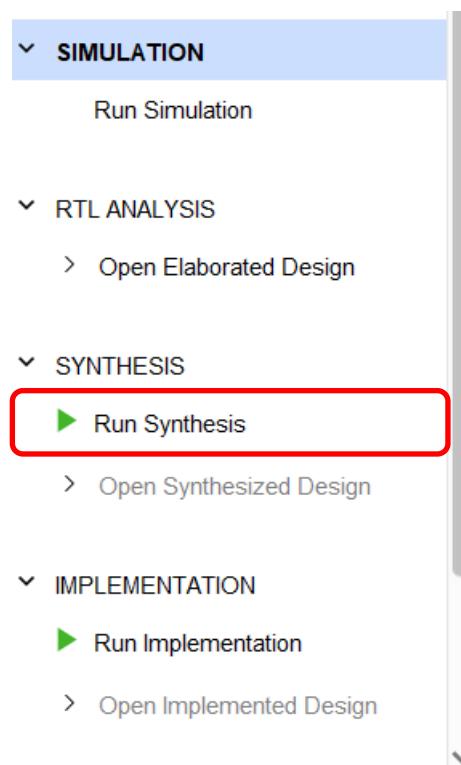
# Debugging (4/4)

- Right click to open the popup menu again, and select *New Divider*



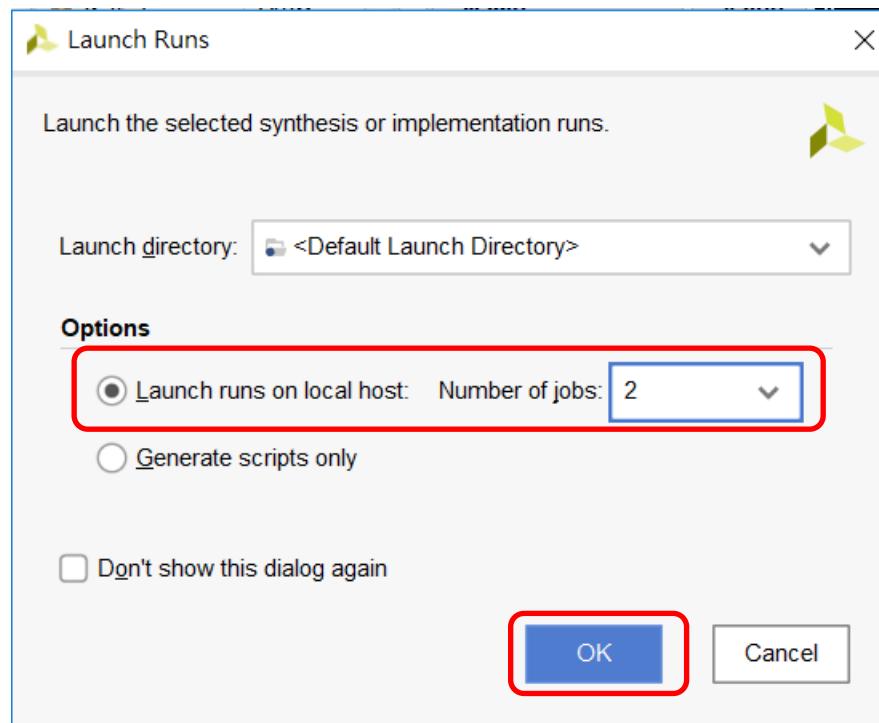
# Synthesize (1/3)

---



# Synthesize (2/3)

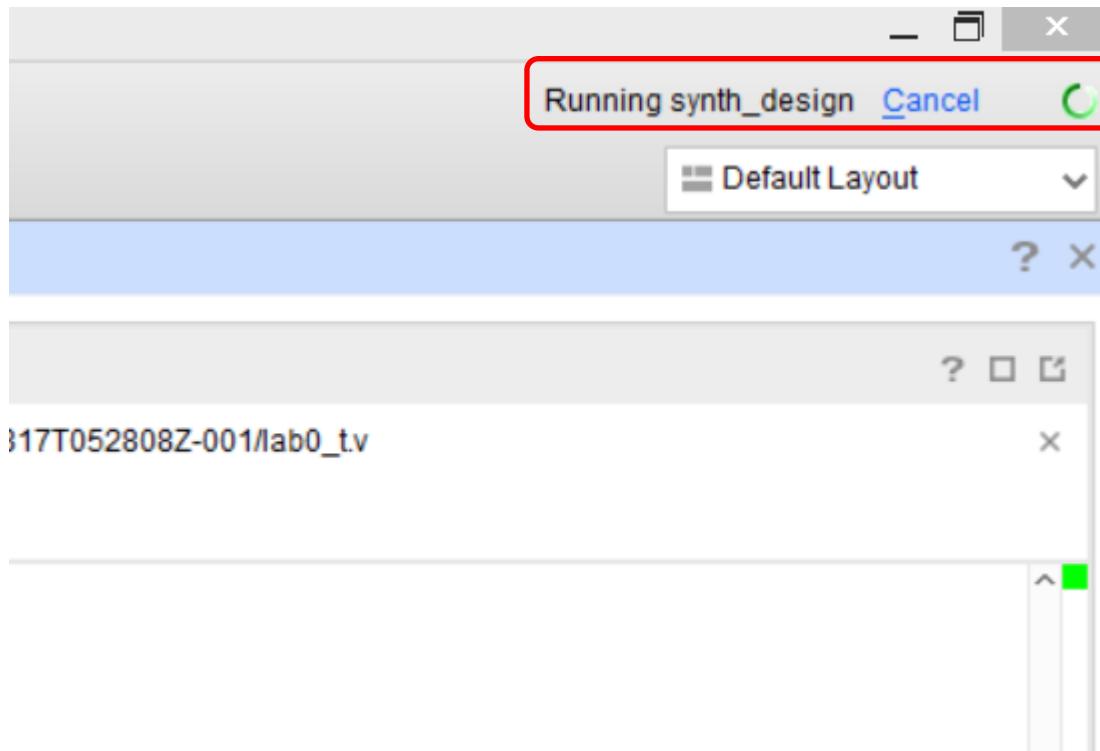
---



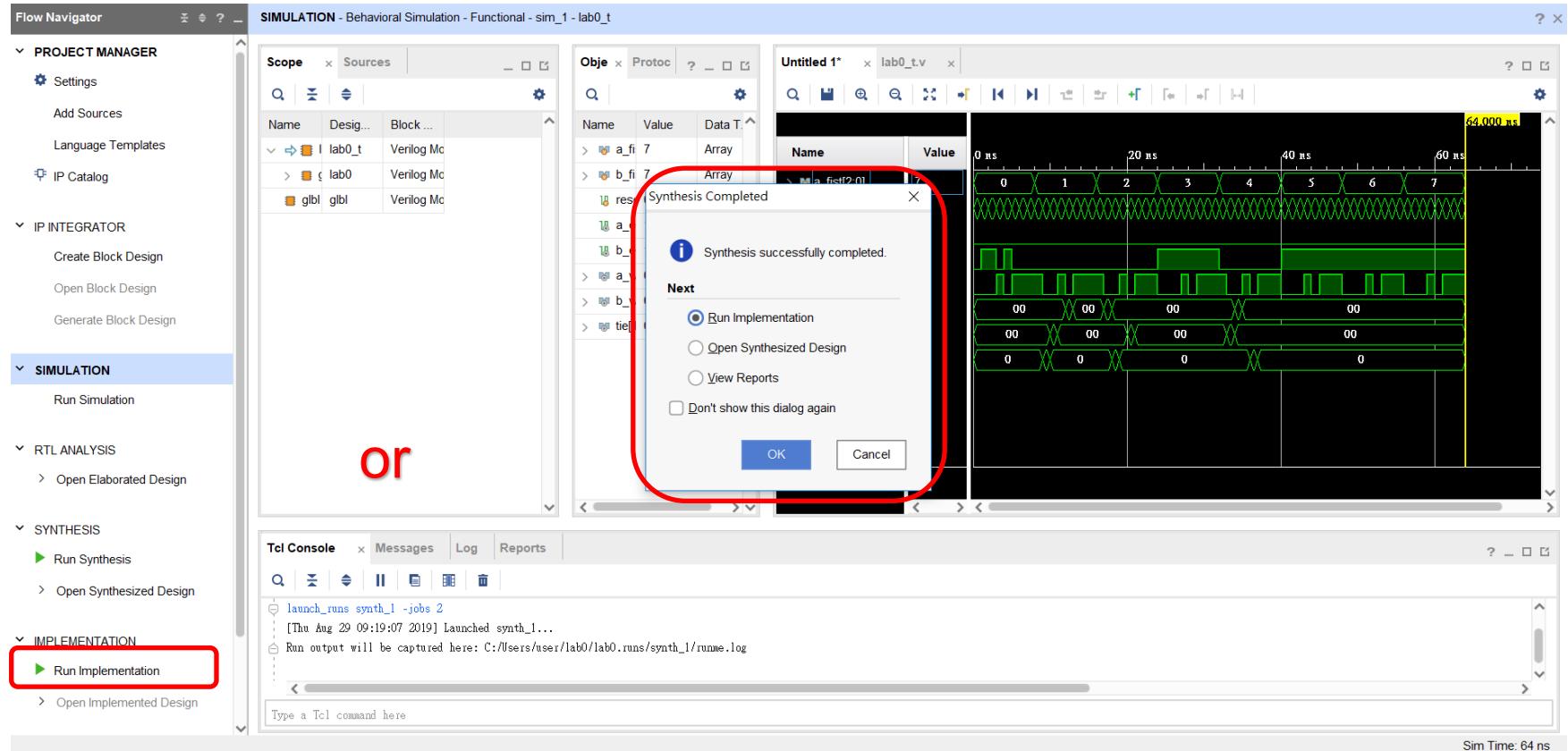
# Synthesize (3/3)

---

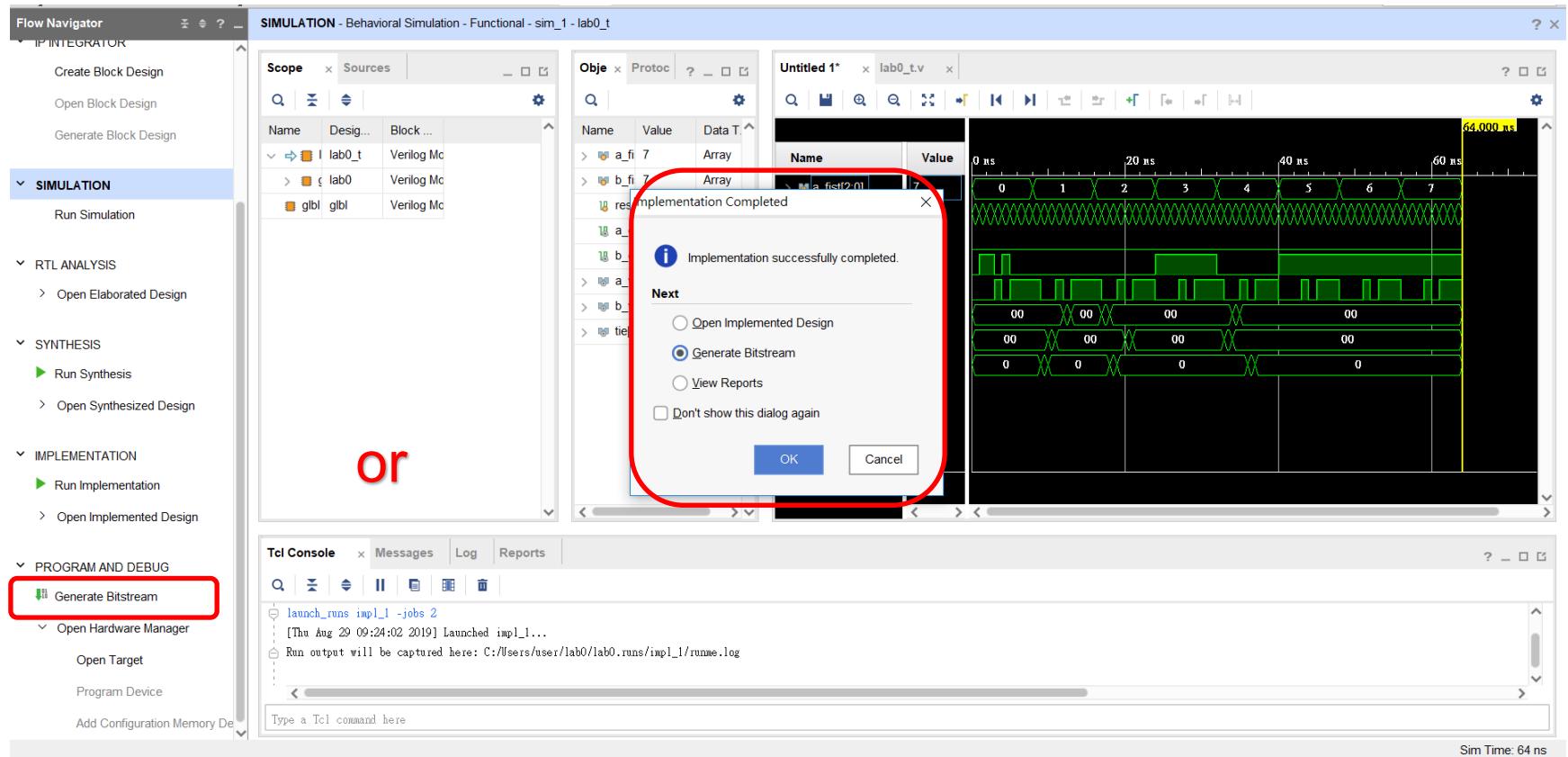
- Wait for it



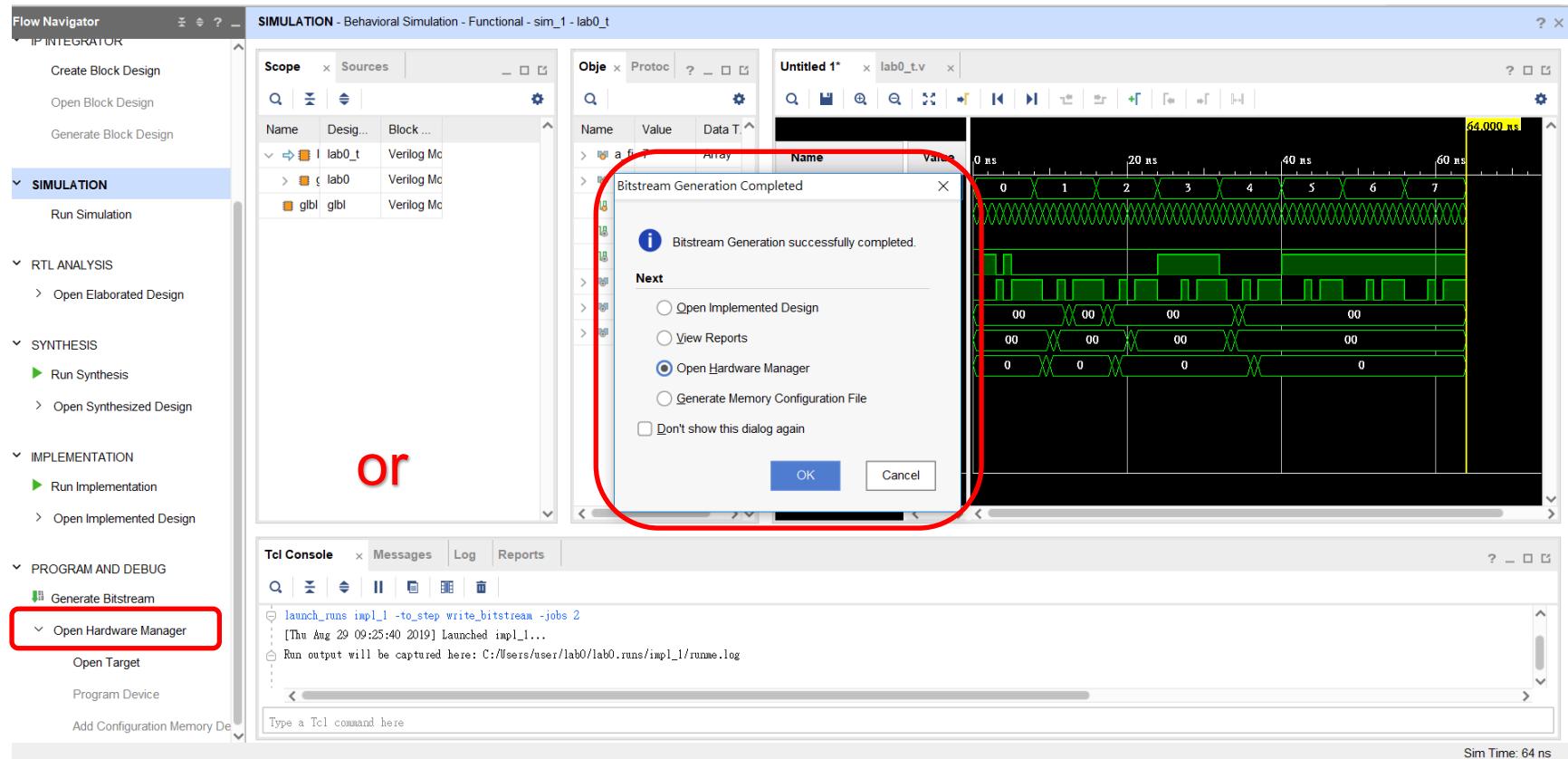
# Implement



# Generate Bitstream



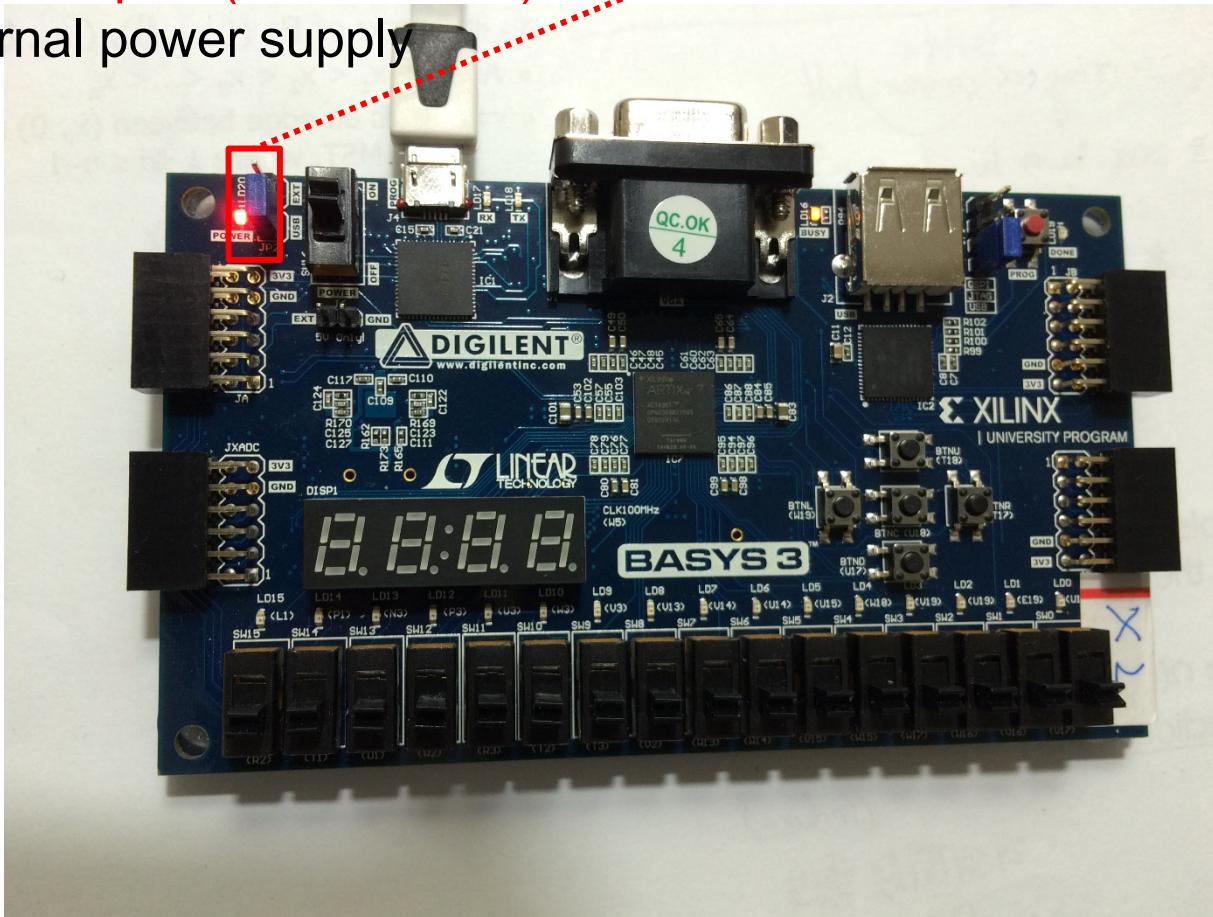
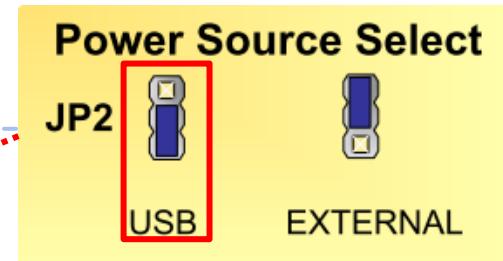
# Open Target (1/6)



# Open Target (2/6)

Receive “Power” from:

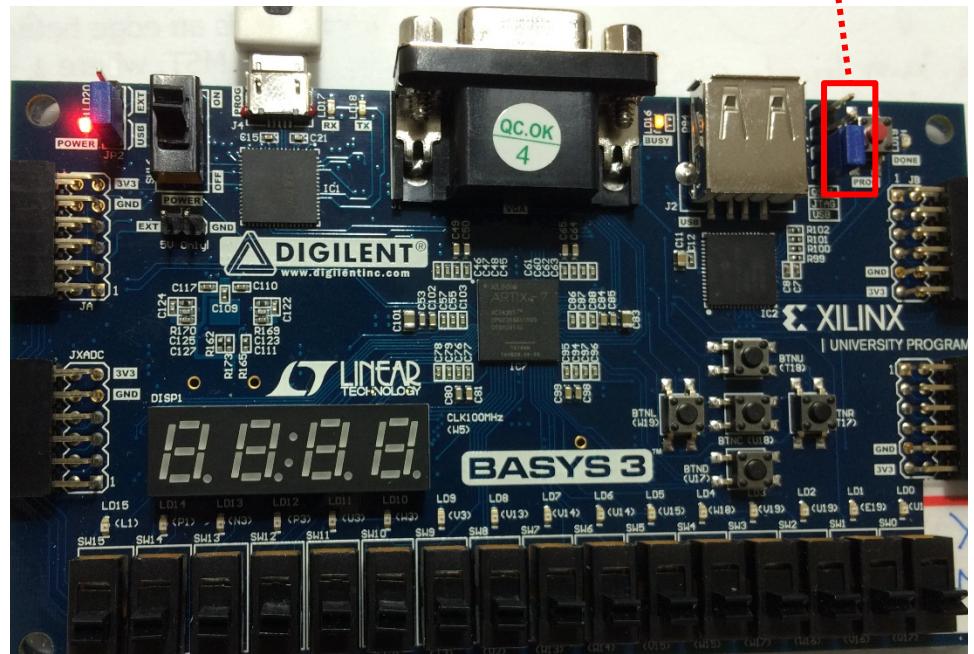
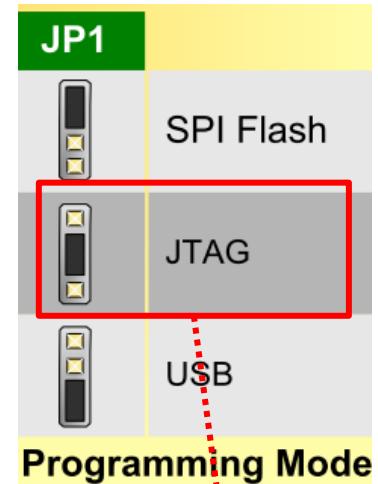
- the microUSB port (recommend)
- a 5V external power supply



# Open Target (3/6)

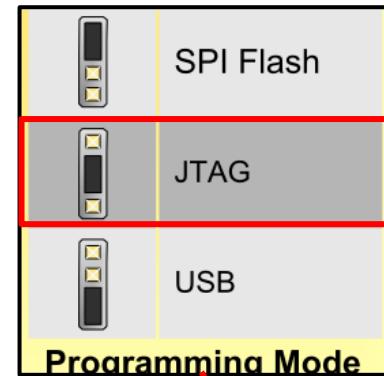
Receive “Bitstream” from:

- Flash
- JTAG (recommend)
- USB Drive

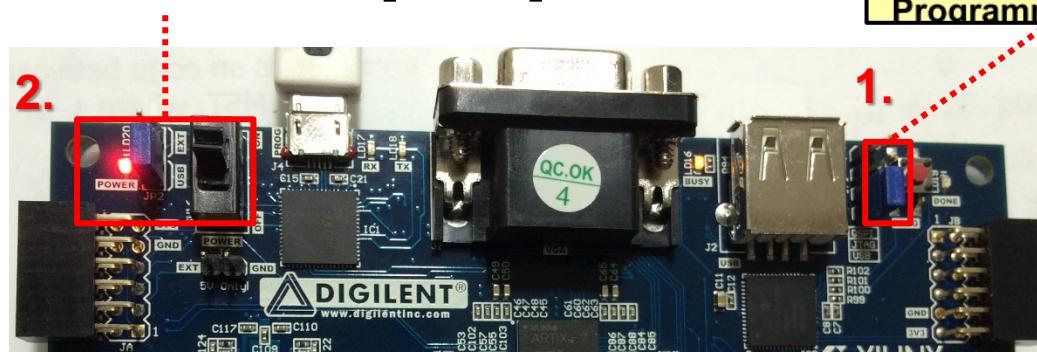


# Open Target (4/6)

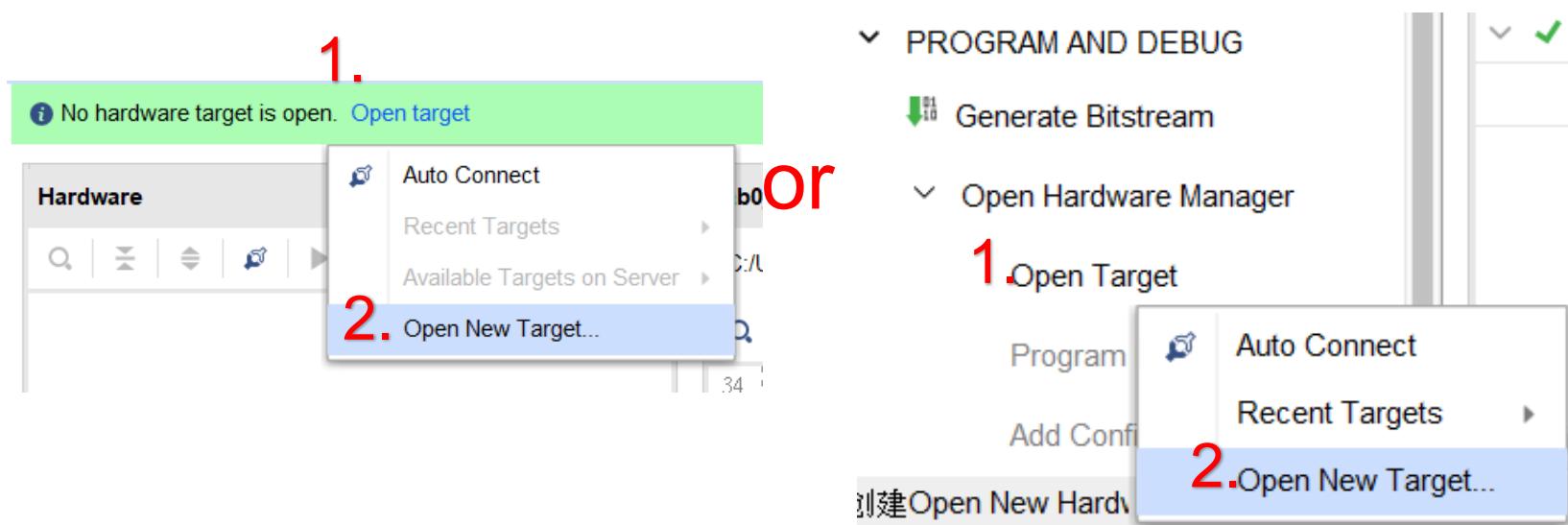
Step 1:  
Change blue jumper to [ **JTAG** ] mode



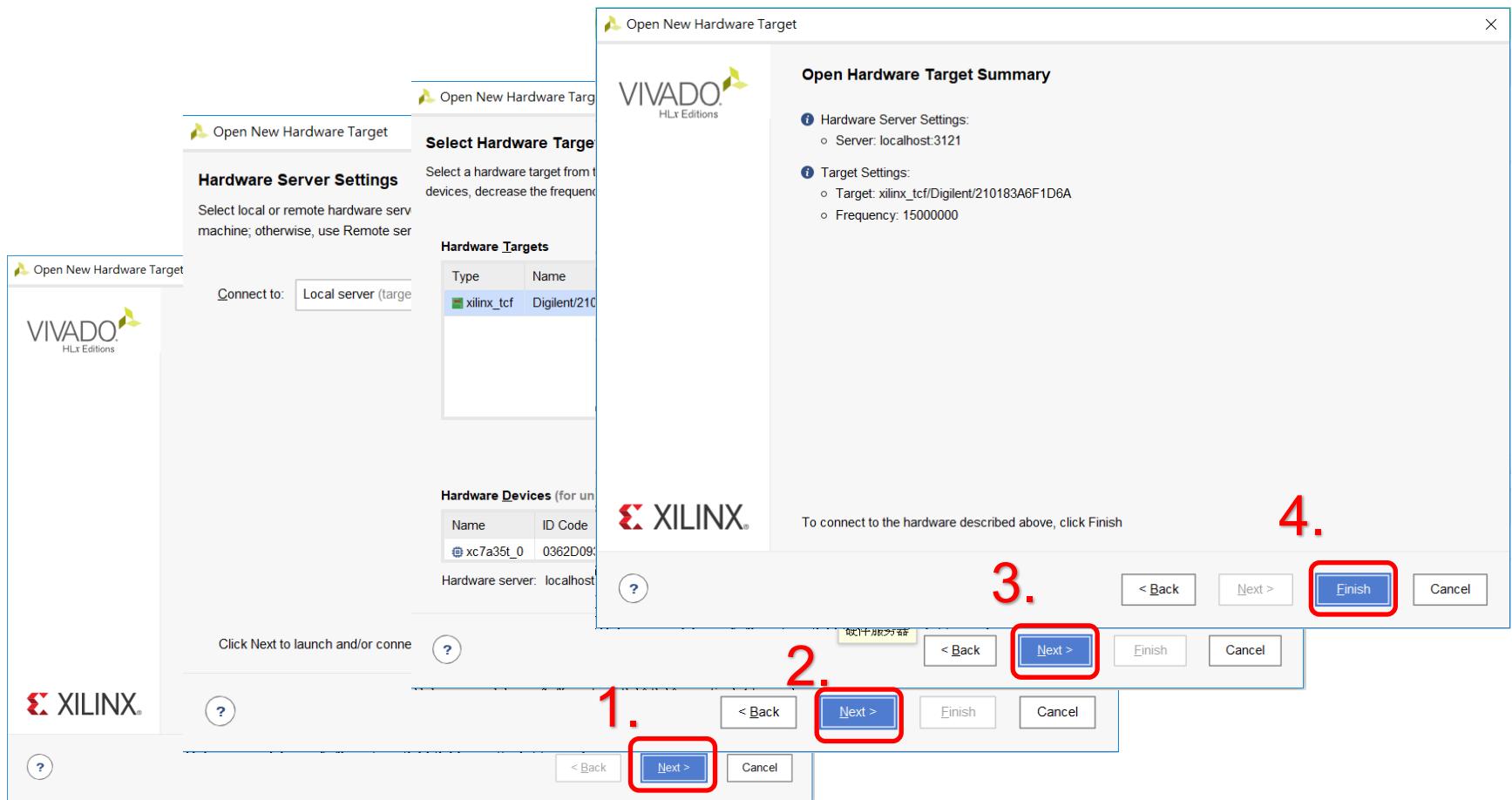
Step 2:  
Turn the power switch to [ **ON** ]



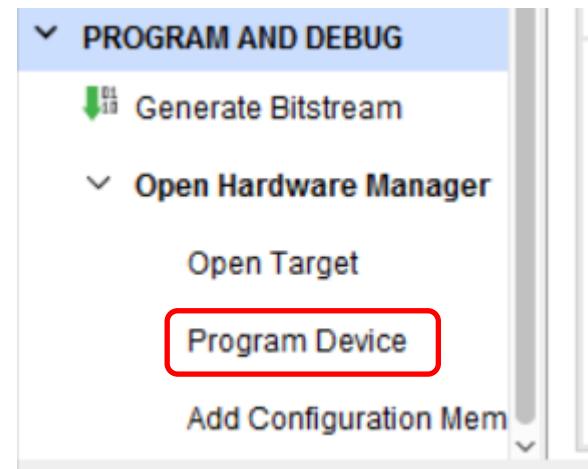
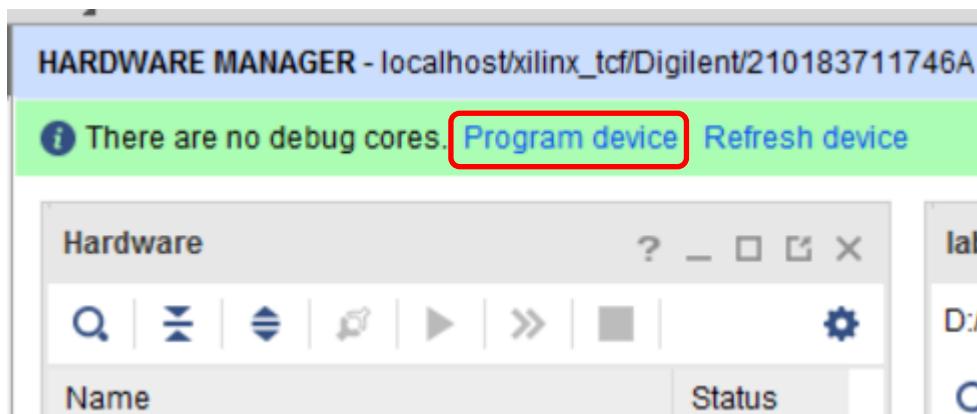
# Open Target (5/6)



# Open Target (6/6)



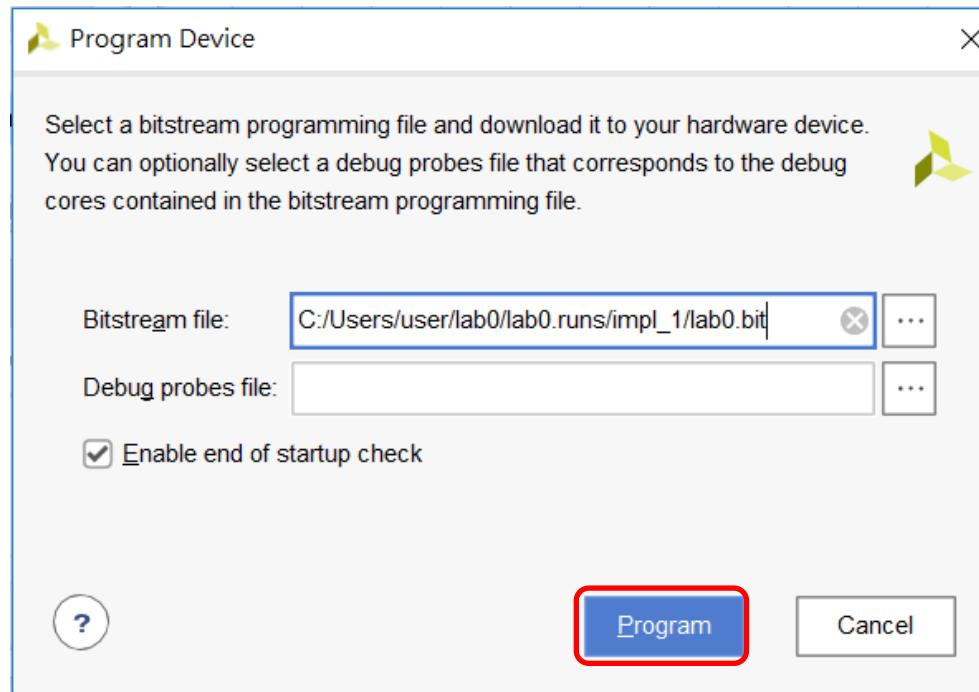
# Program Device (1/3)



or

# Program Device (2/3)

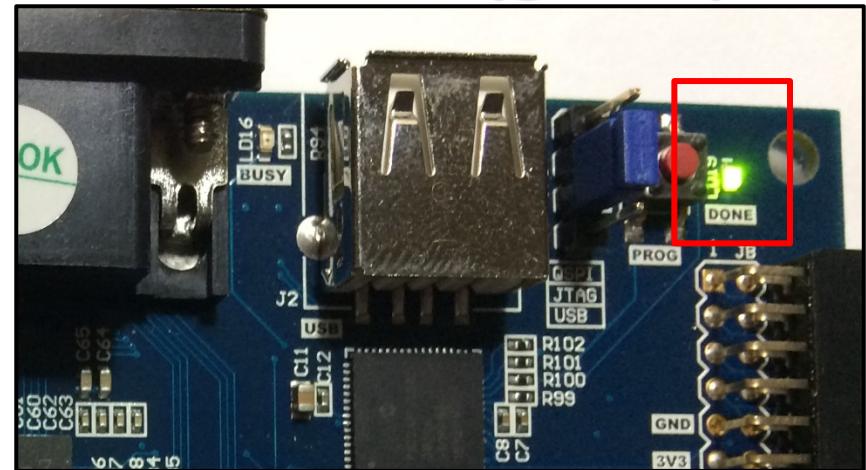
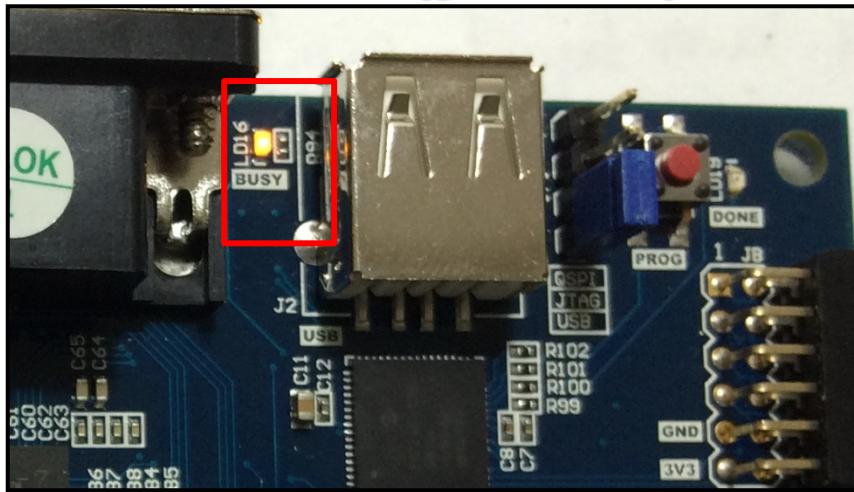
---



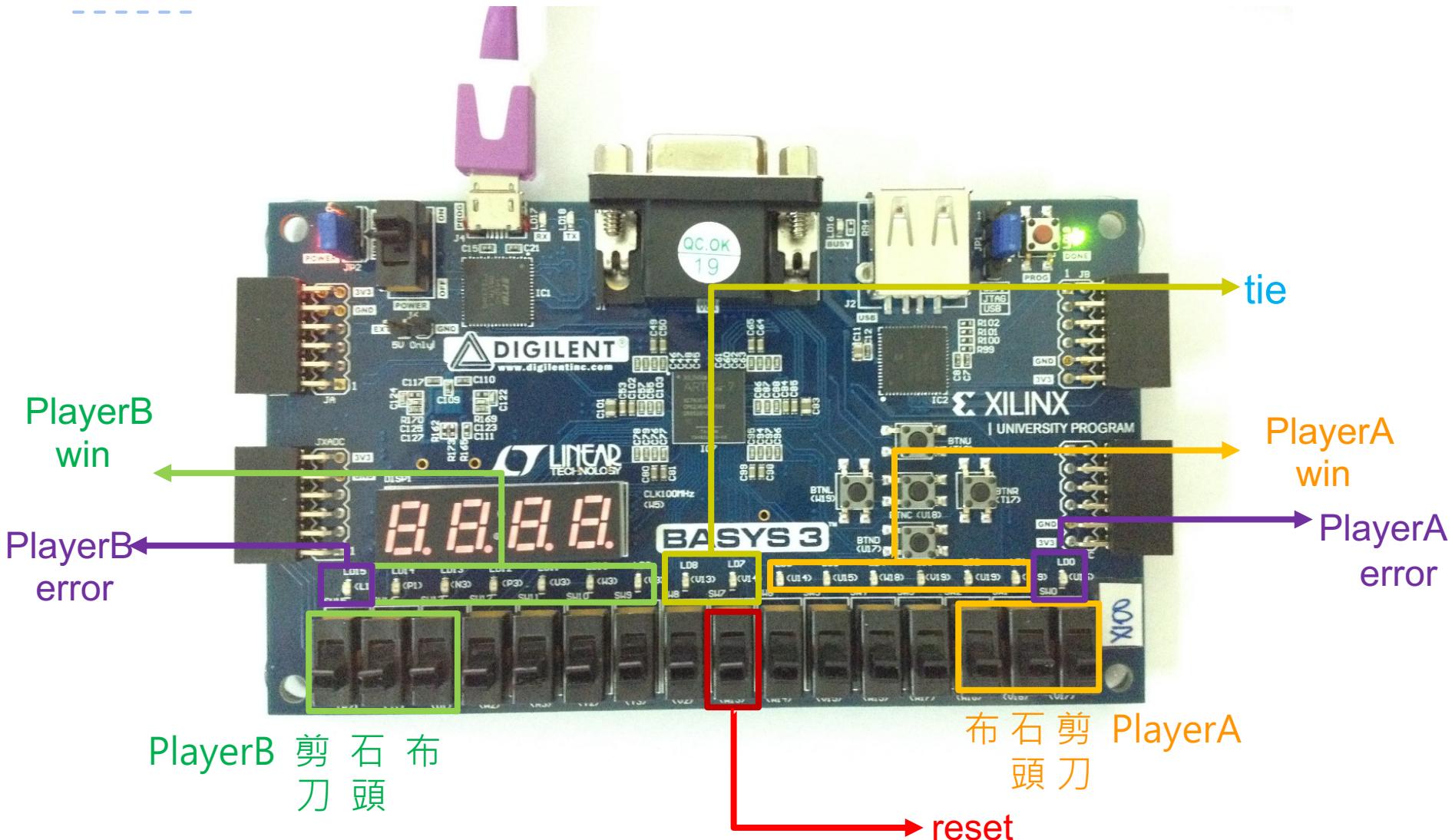
# Program Device (3/3)

- Wait for the programming is done

**BUSY (yellow)** → **DONE (green)**



# Game operating



# References

---

- All in *DocNav*  
(e.g., ug937-vivado-design-suite-simulation-tutorial.pdf)

