

# 數位邏輯設計

Verilog

2021.05.14

# Module

**Module**：基本組成單元

語法協定：

每行結束後需有分號，表示敘述完畢。

單行註解：`//註解`

多行註解：`/*註解*/`

數字規格：`<size>'<base format><number>`

ex:     4'b1111         4bit 二進位數 = 1111  
      16'd255         16bit 十進位數 = 255  
          //base format default為十進位

```
module module_name (輸出入埠 型態 名稱,  
                   輸出入埠 型態 名稱); //需與硬體連接之輸出入埠  
  
    參數定義;         //無需與硬體連接之輸出入埠  
    參數初值設定;  
  
    電路功能 敘述; //每個敘述如果大於兩行，用begin end連接  
  
endmodule
```

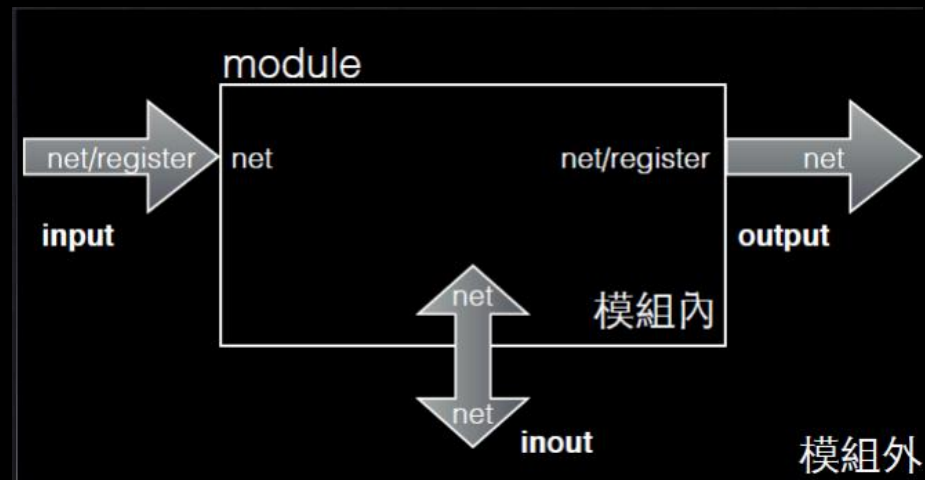
# Input/Output

- 接線方向
- Input : wire only
- Output : reg /wire
- Inout : wire only

# Wire/Register

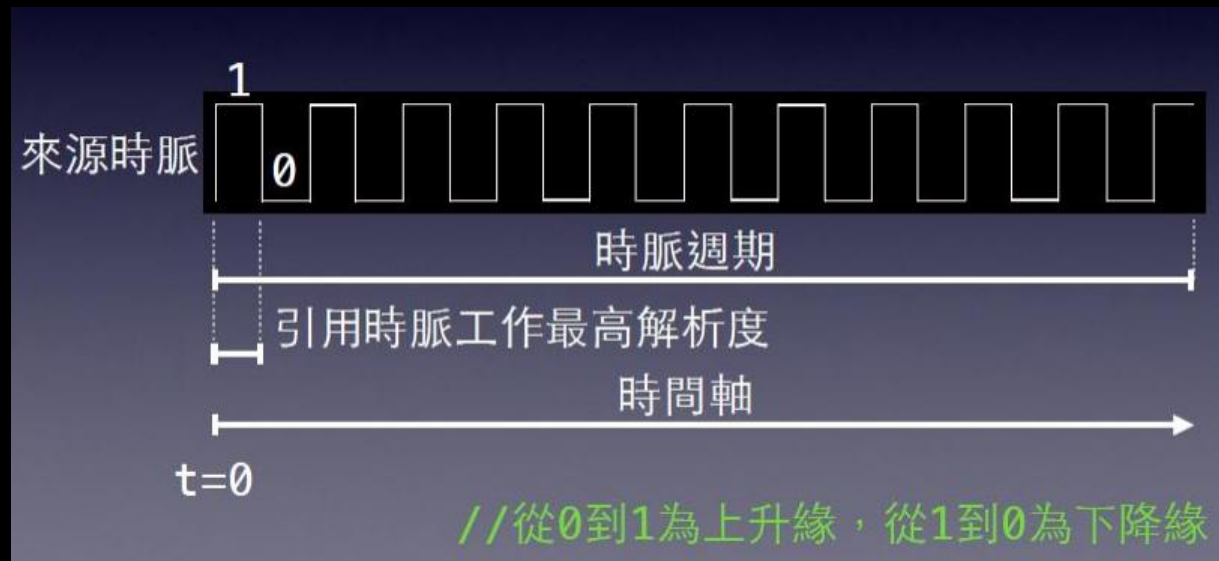
- 接線型態

- 接線(Net)：wire 用於連接硬體
- 暫存器(Register)：reg 資料儲存空間，也可以如接線般交流
- 參數(Parameter)：parameter，設定定值



# Clock

- 在硬體語言中，所有的依據都是以硬體引入的時脈(clock)做為來源的依據。



# 條件判斷

- If –else

```
if(<expression>) <begin>
    <statement_or_null>
<end>
else if(<expression>)
<begin>
    <statement_or_null>
<end>
.....
else <begin>
    <statement_or_null>
<end>
```

- case

```
case(<expression>)
    Case1 : <begin>
        <statement_or_null>
    <end>
    Case2:(<expression>) <begin>
        <statement_or_null>
    <end>
    .....
    Default: <begin>
        <statement_or_null>
    <end>
endcase
```

# 練習(除頻)

```
module clock_divide( input clk,          // 屬性為wire，為預設（預設為1MHz時脈）
                    output reg clk2 // 需指定值，使用暫存器
                    );
    reg [19:0] cnt; // 不需連接模組外之參數，不用輸出入埠
    //初值設定
    initial begin
        clk2 = 1'b0;
        cnt  = 20'b0;
    end
    //創造一秒的時脈
    always @(posedge clk)begin          //觸發條件
        if (cnt == 999999) cnt <= 0; //counter數到999999把其歸0，表示1MHz去數一秒的總數
        else cnt <= cnt+1;
    end
    always @(posedge clk)begin
        if (cnt <= 499999) clk2 <= 0; //duty cycle 50%
        else clk2 <= 1;                //1MHz的計數器數1000000所以在一半的地方做反相
    end
    //判斷上能用 <=，就不要用 <
endmodule
```

**VIVADO**



# Installing Vivado and Digilent Board Files

- Vivado 設計套件，是 FPGA 廠商 Xilinx (賽靈思) 公司 2012 年發布的集成設計環境，幫助實現 FPGA 整合與開發。(前身為ISE)
- Digilent Board Files 為 Vivado設計環境提供 Digilent 板卡訊息幫助快速實現板卡整合。

# Vivado 設計套件

- Vivado 為 RTL 設計提供了一個合成，整合和驗證除錯的工作環境。
- Vivado 軟體包含
  - IP Integrator : 用於設計嵌入式處理器系統的硬體部分。您可以指定 ARM Cortex-A9 處理器內核，IP 周邊及這些組件的互連及其相應的詳細配置。
  - Software Development Kit : SDK 是一個集成的開發環境，與 Vivado 互補，用於 C / C++ 嵌入式軟體應用程序創建和驗證。 SDK 是基於Eclipse 開源框架所構建的。
  - Vivado 邏輯分析儀 : 分析&除錯硬體設計。

# 到 Xilinx website 下載WebPack版本

The screenshot shows a web browser window displaying the Xilinx website. The page title is "Downloads" and the URL is "https://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/vivado-design-tools/archive.html". The Xilinx logo is visible at the top left, and navigation links for Applications, Products, Developers, Support, and About are at the top right. The main content area is divided into two columns. The left column contains three download links for Vivado HLx 2016.4 WebPACK and Editions, each with its MD5 SUM Value. The right column contains an "Important Information" section with a warning icon and a link to AR#68334. Below this, there is a section for "Vivado Lab Solutions - 2016.4 Full Product Installation" with a table of download types and last updated dates.

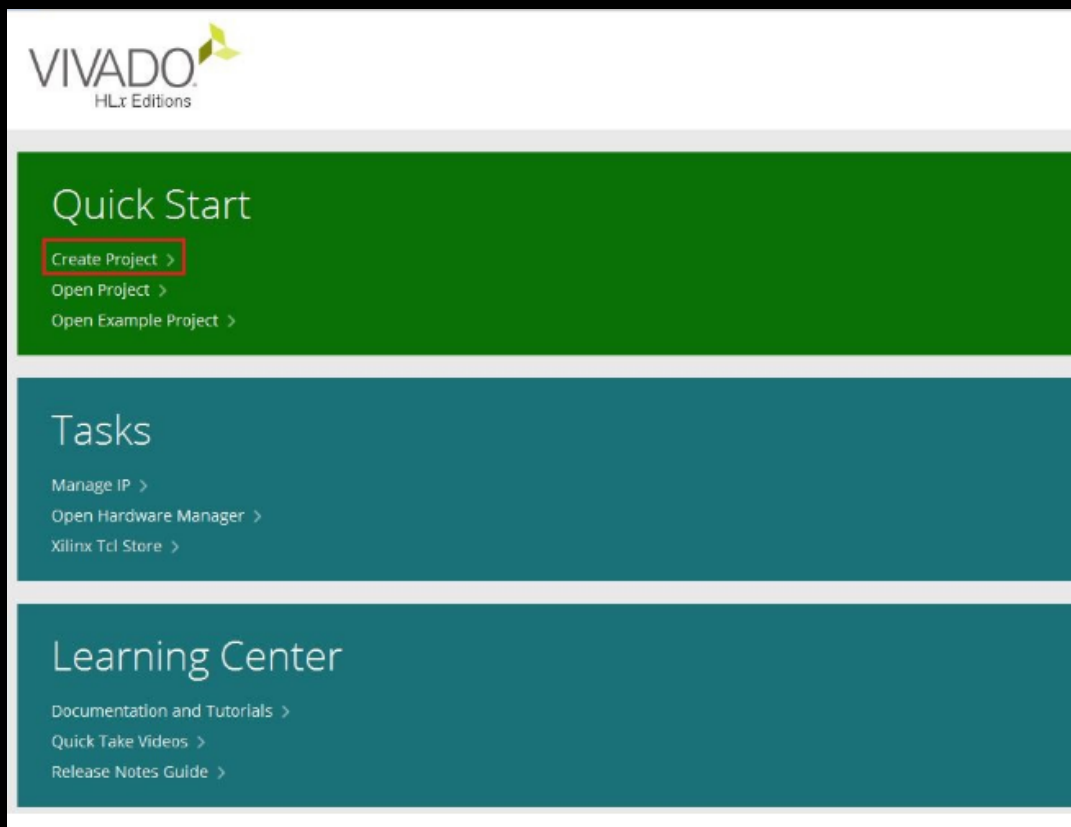
Download Type	Last Updated
Full Product Installation	Jan 25, 2017

**Important Information**

Vivado Lab Edition is a new, compact, and standalone product targeted for use in the lab environments. It provides for programming and logic/serial IO debug of all Vivado supported devices. Lab Edition requires no certificate or activation license key and supports 64- and 32-bit OS platforms.

<https://www.xilinx.com/support/download.html>

# 開啟專案



VIVADO<sup>®</sup>  
HLx Editions

## Quick Start

- [Create Project >](#)
- [Open Project >](#)
- [Open Example Project >](#)

## Tasks

- [Manage IP >](#)
- [Open Hardware Manager >](#)
- [Xilinx Tcl Store >](#)

## Learning Center

- [Documentation and Tutorials >](#)
- [Quick Take Videos >](#)
- [Release Notes Guide >](#)

New Project x

**Project Name**

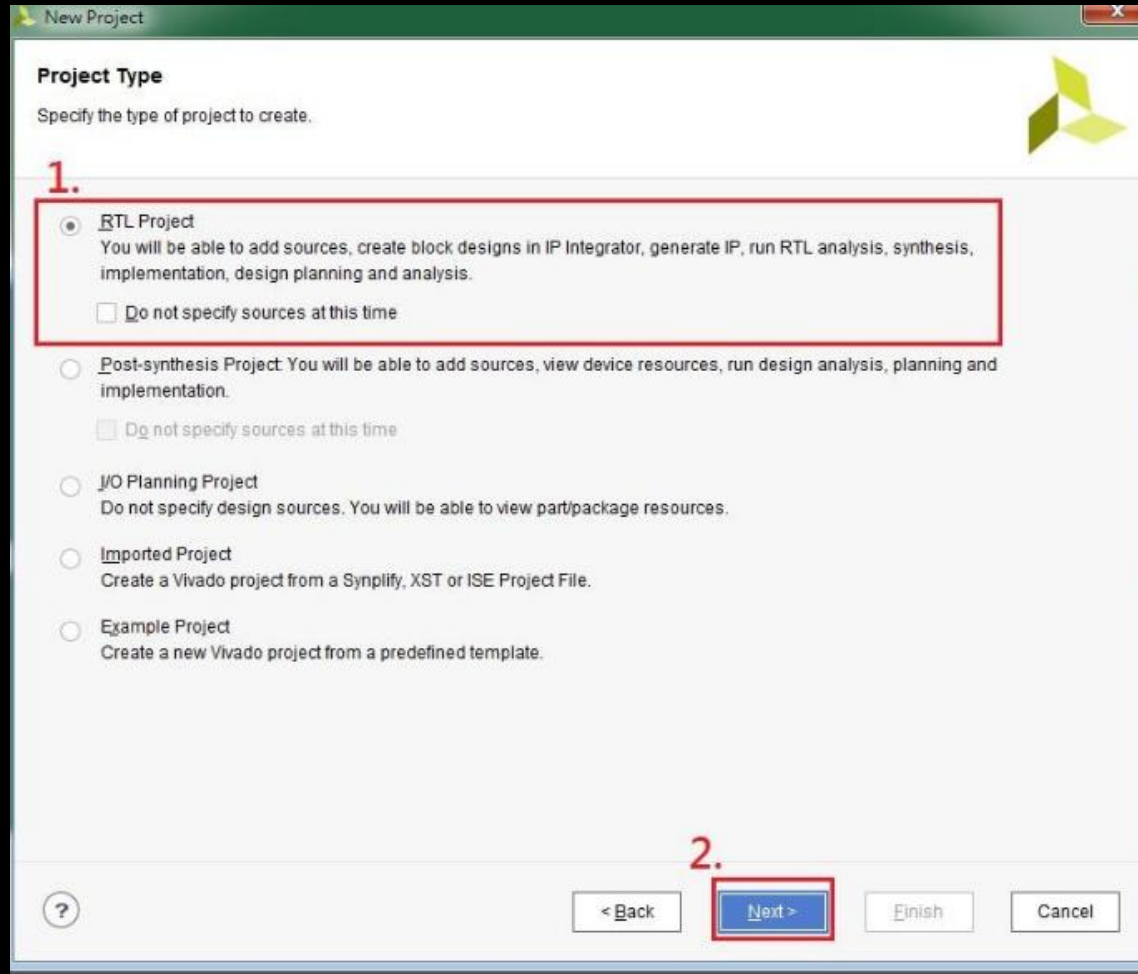
Enter a name for your project and specify a directory where the project data files will be stored.

Project name: 專案名稱(只能用英文)

Project location: 檔案位置(只能用英文)

Create project subdirectory

Project will be created at: C:/Users/DI/Desktop/day3/Hello\_world/Hello\_world



- **RTL : Register Transfer Level**,是一種對同步數位電路的抽象模型，這種模型是根據數位訊號在硬體暫存器、存儲器、組合邏輯裝置和匯流排等邏輯單元之間的流動，以及其邏輯代數運作方式來確定的。  
(from 維基百科)

# 添加與撰寫 Verilog 程式

Project Manager -> Add sources 增加檔案

The screenshot displays the Vivado 2017.4 Project Manager interface. The 'Project Manager' window is active, showing the 'Sources' panel on the left and the 'Project Summary' panel on the right. The 'Add Sources' button in the 'Sources' panel is highlighted with a red rectangle. The 'Project Summary' panel shows the following details:

- Project name: project\_1
- Project location: D:/Zybo\_training/pl\_led
- Product family: Zynq-7000
- Project part: Zybo Z7-20 (xc7z020ctg400-1)
- Top module name: Not defined
- Target language: Verilog
- Simulator language: Mixed

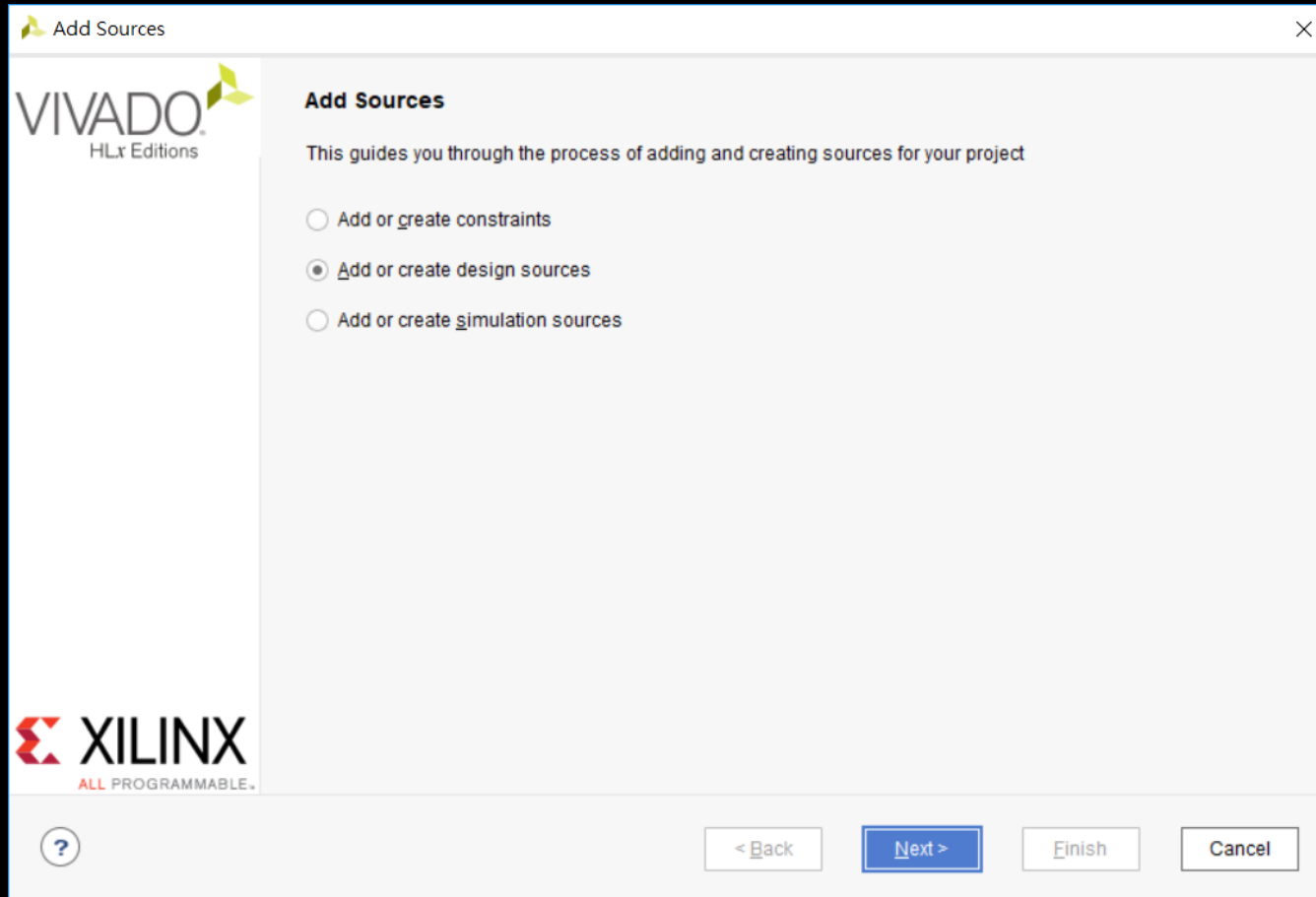
The 'Board Part' section shows:

- Display name: Zybo Z7-20
- Board part name: digilentinc.com:zybo-z7-20:part0:1.0

The 'Design Runs' table at the bottom shows the following data:

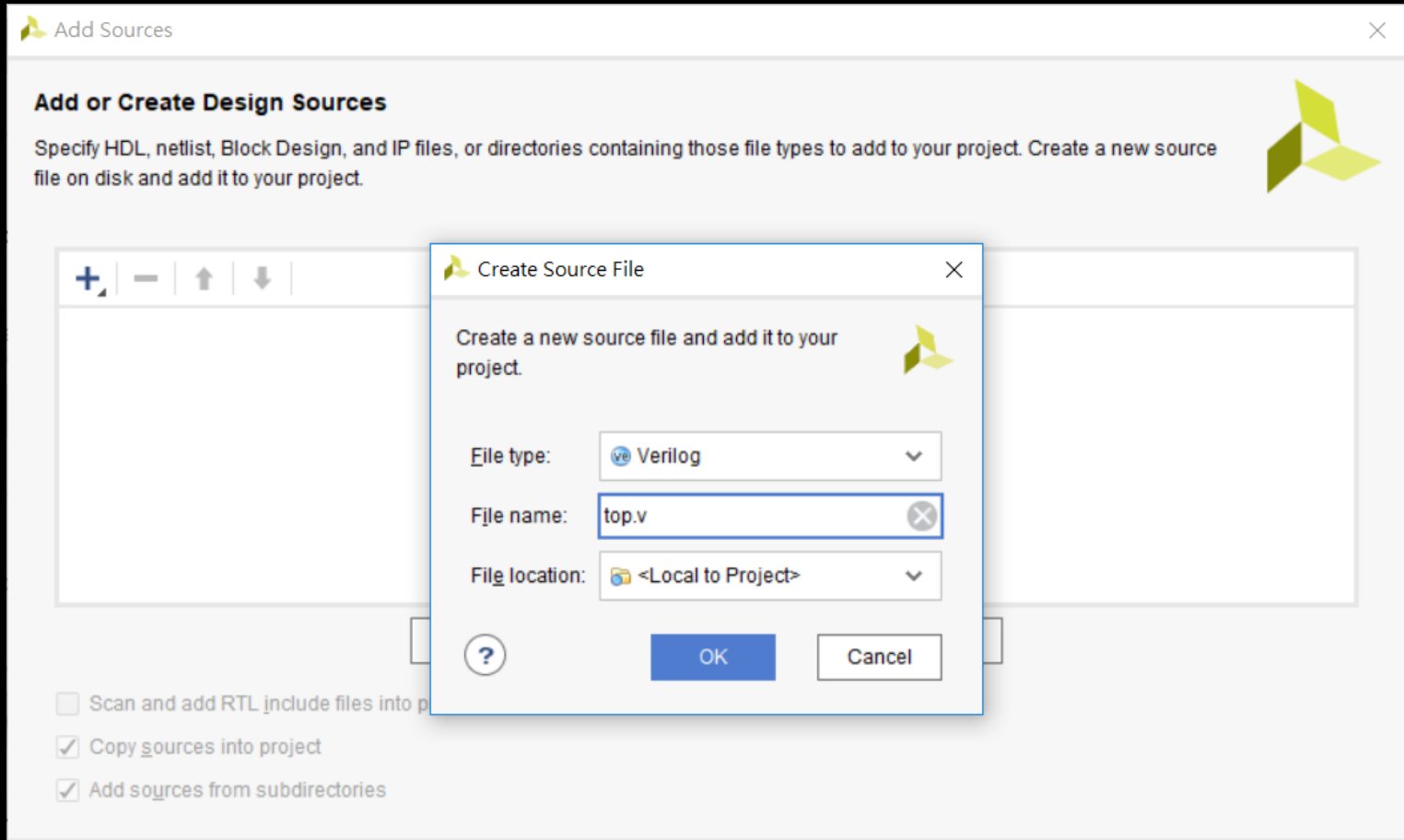
Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAMS	URAM	DSP	Start	Elapsed	Run Strategy
synth_1	constrs_1	Not started															Vivado Synthesis Defaults (Vivado Synthesis 2017)
impl_1	constrs_1	Not started															Vivado Implementation Defaults (Vivado Implementa

# 新增設計

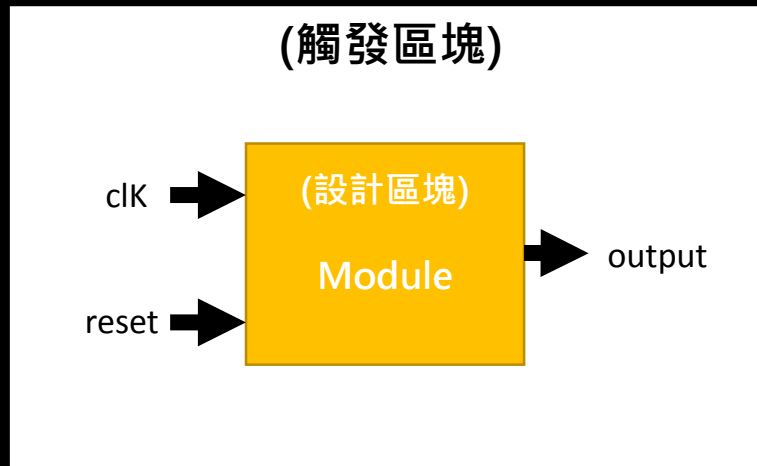




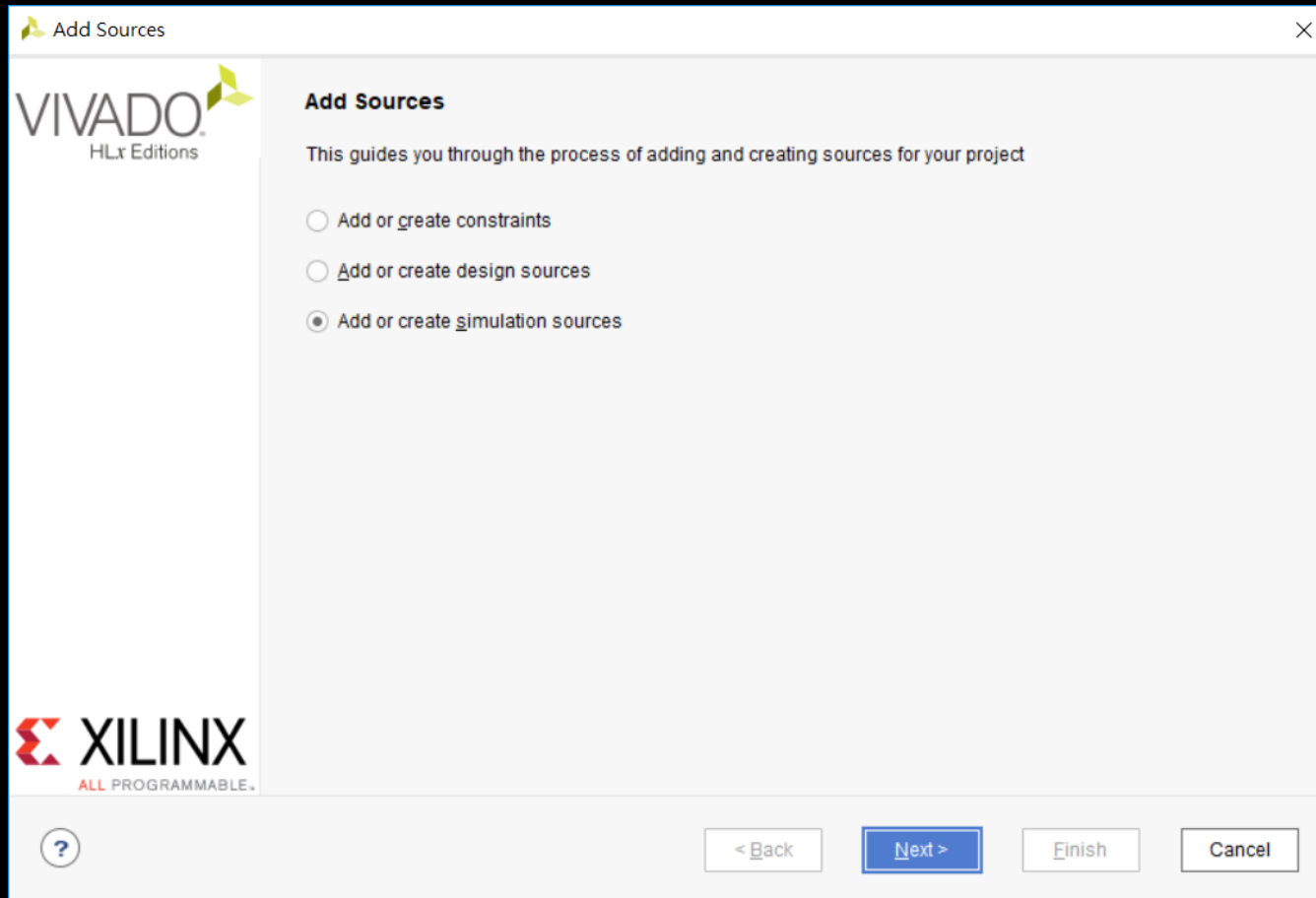
# Create File 新增 top.v 主程式



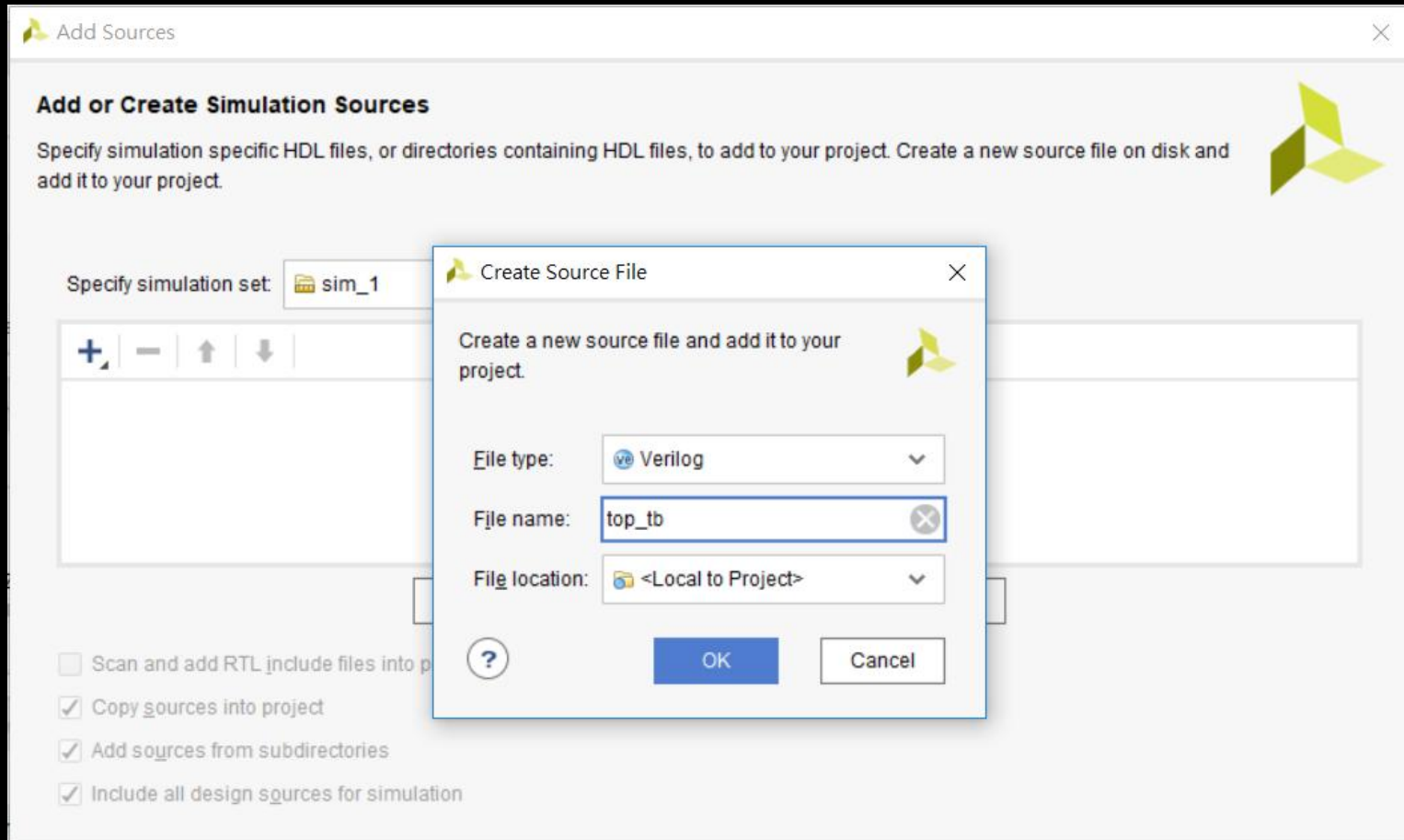
# Test Bench



# 設定模擬



# 設定模擬: Create file



# 設定模擬:完成 test bench

project\_2 - [C:/Users/S4-805-04/Desktop/058/project\_2/project\_2\_xpr] - Vivado 2018.3

File Edit Flow Tools Reports Window Layout View Run Help Q: Quick Access Ready

Flow Navigator SIMULATION - Behavioral Simulation - Functional - sim\_1 - ts

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
- Open Target
- Program Device
- Add Configuration Memory Device

Scope Sources x

- Design Sources (1)
  - TEST (TEST.v)
- Constraints
- Simulation Sources (1)
  - sim\_1 (1)
    - ts (ts.v) (1)
      - u1: TEST (TEST.v)
- Utility Sources

TEST.v x ts.v x Untitled 5 x

C:/Users/S4-805-04/Desktop/058/project\_2/srcs/sim\_1/new/ts.v

```
1 ;
2
3
4 // Company:
5 // Engineer:
6 // Create Date: 2019/05/09 15:30:42
7 // Design Name:
8 // Module Name: ts
9 // Project Name:
10 // Target Devices:
11 // Tool Versions:
12 // Description:
13 //
14 // Dependencies:
15 //
16 // Revision:
17 // Revision 0.01 - File Created
18 // Additional Comments:
19 //
20
21
22
23 module ts;
24     reg clk;
25     wire [7:0]cnt;
26     wire clk2;
27     TEST u1(
28         .clk(clk),
29         .cnt(cnt),
30         .clk2(clk2)
31     );
32
33     initial begin
34         clk = 1'b0;
35     end
36
37     //50MHz clk
38     always #10 clk = ~clk;
39 endmodule
```

程式所參照的時間單位  
Time Scale的單位/精度

timescale 1ns / 10ps

Current time: 6286973810 ns... Cancel 21:0 Insert Sim Time: 6286988150 ns Verilog

# 設定模擬: Run Simulation

The screenshot displays the Vivado 2017.4 software interface. The 'PROJECT MANAGER - project\_1' window is open, showing the 'Sources' pane with 'Simulation Sources (1)' containing 'sim\_1 (1)' and 'top\_tb (top\_tb.v) (1)'. The 'Run Simulation' option in the 'SIMULATION' section of the left-hand 'Flow Navigator' is highlighted with a red box. A context menu is overlaid on the 'Run Simulation' option, listing several simulation types: 'Run Behavioral Simulation', 'Run Post-Synthesis Functional Simulation', 'Run Post-Synthesis Timing Simulation', 'Run Post-Implementation Functional Simulation', and 'Run Post-Implementation Timing Simulation'. The 'Run Behavioral Simulation' option is also highlighted with a red box. The background shows the 'top\_tb.v' source file with Verilog code, including an 'initial' block and a reset signal assignment.

```
39     end
40     .
41     initial
42     begin
43         clk <= 1'b0;
44         PL_CPU_RESET <= 0;
45         #100 PL_CPU_RESET <= 1;
```

RAMs	URAM	DSP	Start	Elapsed	Run Strategy
					Vivado Synthesis Defaults (Vivado Synthesis 2017)
					Vivado Implementation Defaults (Vivado Implementa

51:0 插入 Verilog

# 設定模擬: Simulation result

The screenshot displays the Vivado 2017.4 simulation environment. The main window shows the simulation results for a behavioral simulation of a Verilog module named 'top\_tb'. The simulation is running at a time scale of 10 us.

**Scope Table:**

Name	Design Unit	Block Type
top_tb	top_tb	Verilog Module
led_inst	top	Verilog Module
gbl	gbl	Verilog Module

**Objects Table:**

Name	Value	Data T...
clk	0	Logic
PL_CPU_R...	0	Logic
LED[3:0]	0	Array

**Waveform:** The waveform shows the simulation results for the 'clk', 'PL\_CPU\_RESET', and 'LED[3:0]' signals. The 'clk' signal is a periodic square wave with a period of 500 ns. The 'PL\_CPU\_RESET' signal is a single pulse. The 'LED[3:0]' signal is a constant 0. The simulation time scale is 500 ns, and the total simulation time is 1,000 ns.

**Tcl Console:**

```
INFO: [USF-XSim-96] XSim completed. Design snapshot 'top_tb_behav' loaded.  
INFO: [USF-XSim-97] XSim simulation ran for 1000ns  
launch_simulation: Time (s): cpu = 00:00:07 ; elapsed = 00:00:18 . Memory (MB): peak = 837.703 ; gain = 20.168
```

Sim Time: 1 us

# 作業

- If  $\text{clk}=50\text{Mhz}$ , how to generate  $\text{clk2}=10\text{ hz}$ ?
- If  $\text{clk}=100\text{Mhz}$ , how to generate  $\text{clk2}=50\text{M hz}$ ?



# 作業

- 完成兩題除頻的模擬
- 將結果截圖，並加入解釋
- 5/28 10:00 前，以PDF 寄到 信箱 [gain514@g.ncu.edu.tw](mailto:gain514@g.ncu.edu.tw)
- 郵件主旨 數位邏輯設計\_你的名字