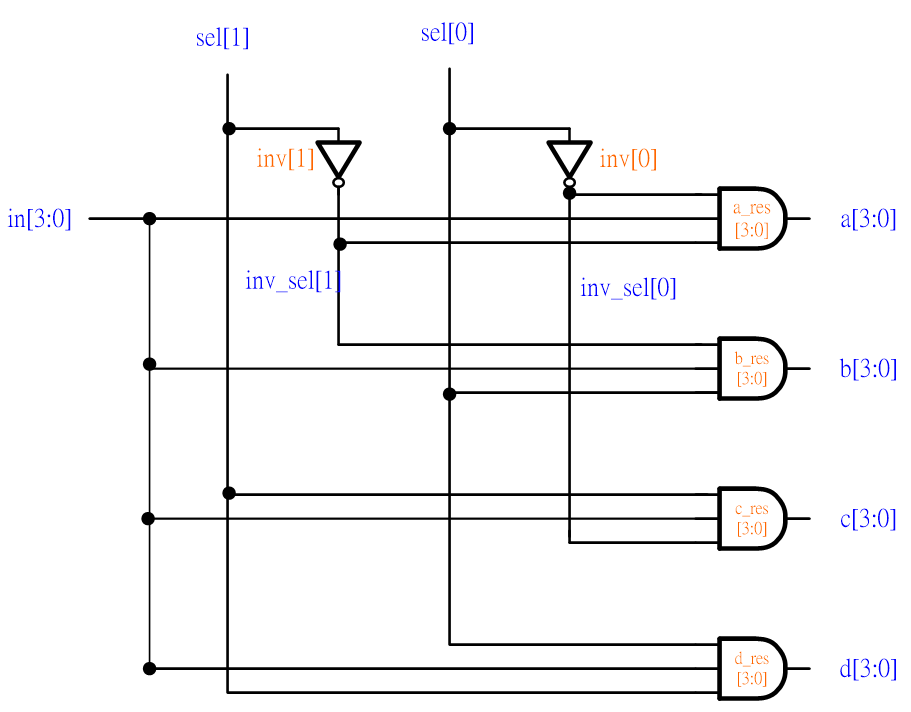
**Lab 1 Gate-Level Modeling**

**實驗報告**

組長:劉奇泓 109033135

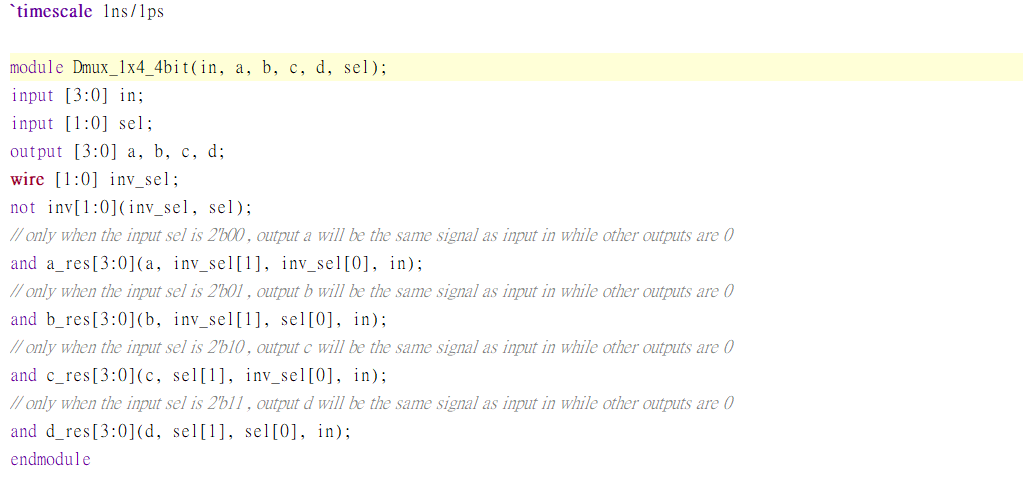
組員:洪聖祥109062315

1. 4-bit 1-to-4 de-multiplexer (DMUX)
2. 邏輯設計圖

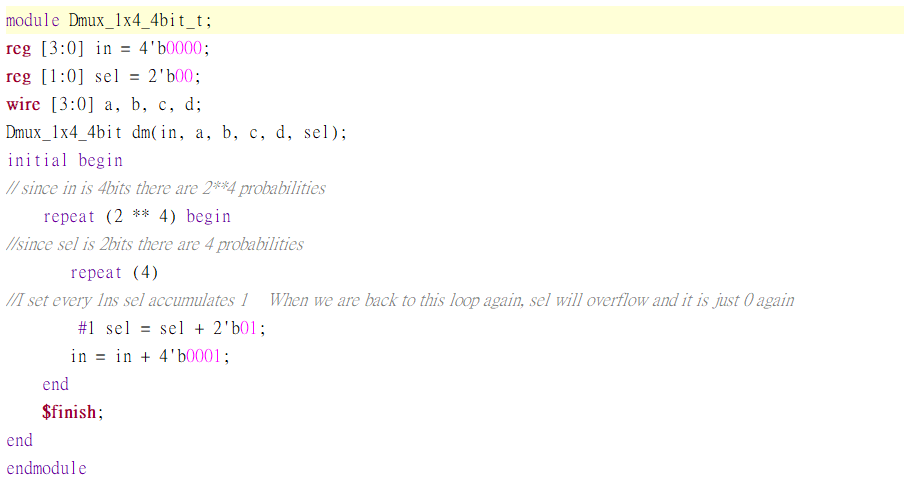


1. Verilog 程式碼

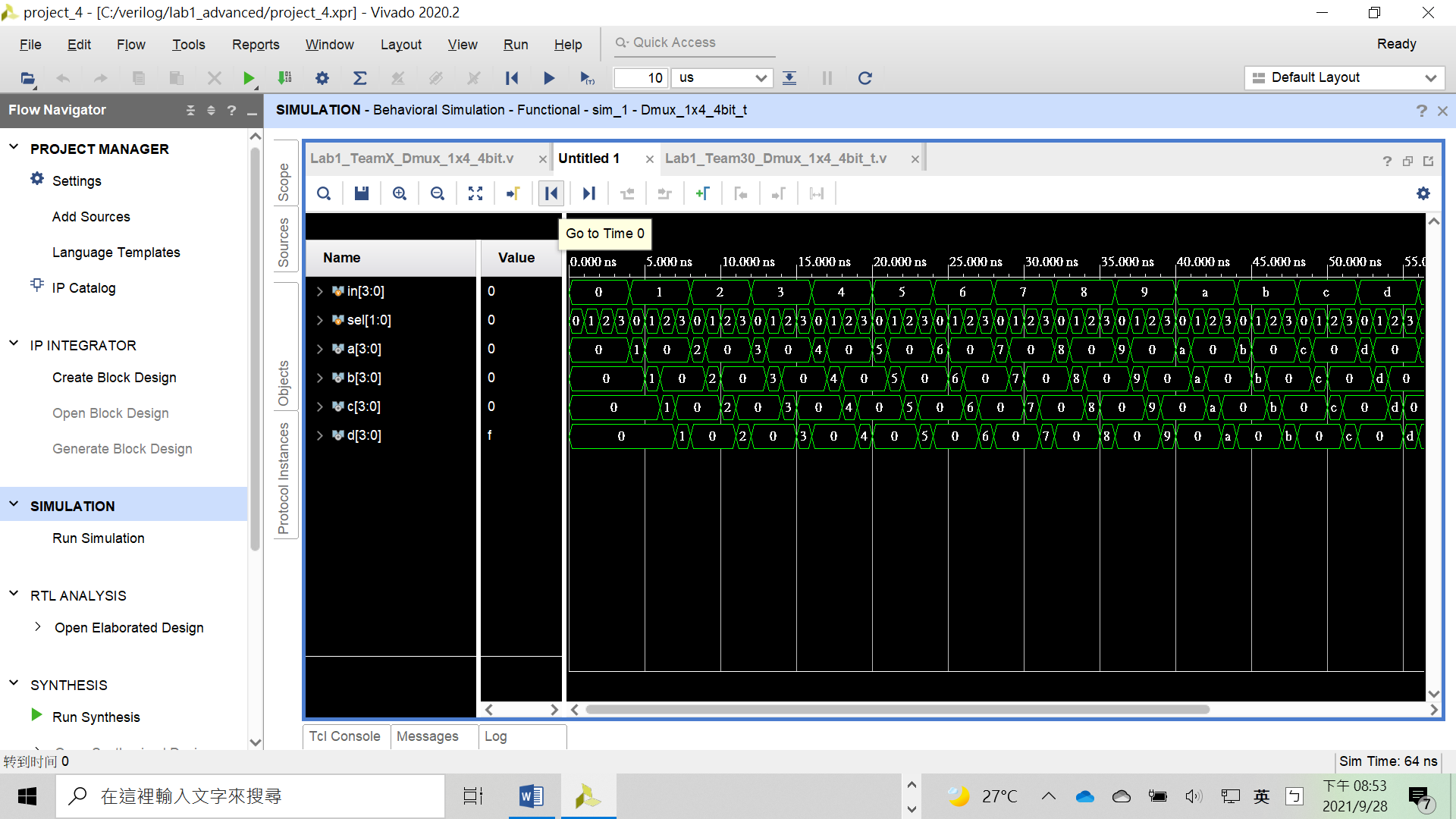
* Lab1\_Team30\_Dmux\_1x4\_4bit.v

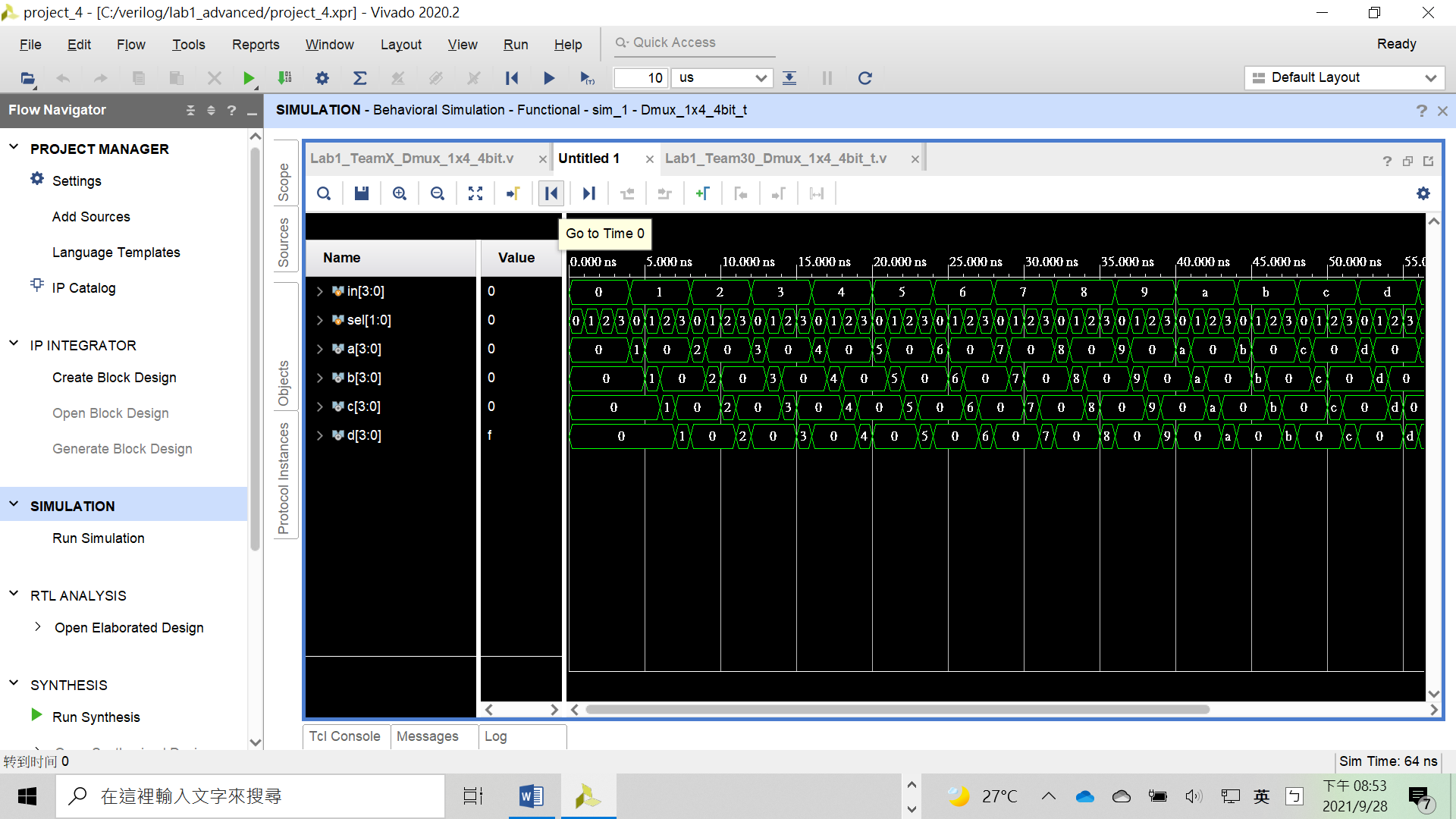


* testbench



1. 波形圖





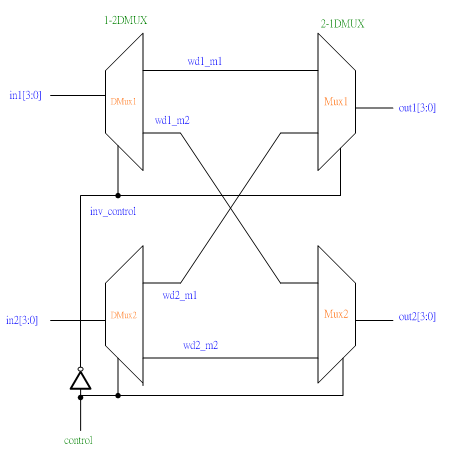
當sel = 2’b00時，a的值必須等於in其餘為0

當sel = 2’b01時，b的值必須等於in其餘為0

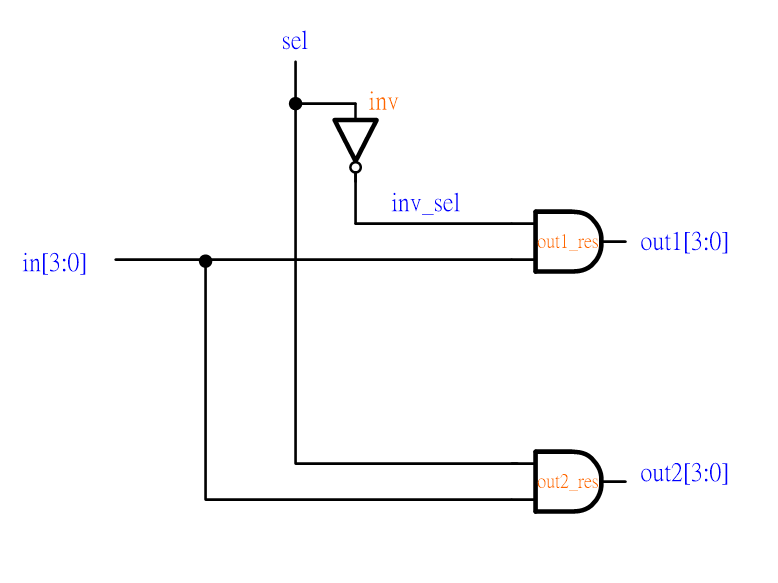
當sel = 2’b10時，c的值必須等於in其餘為0

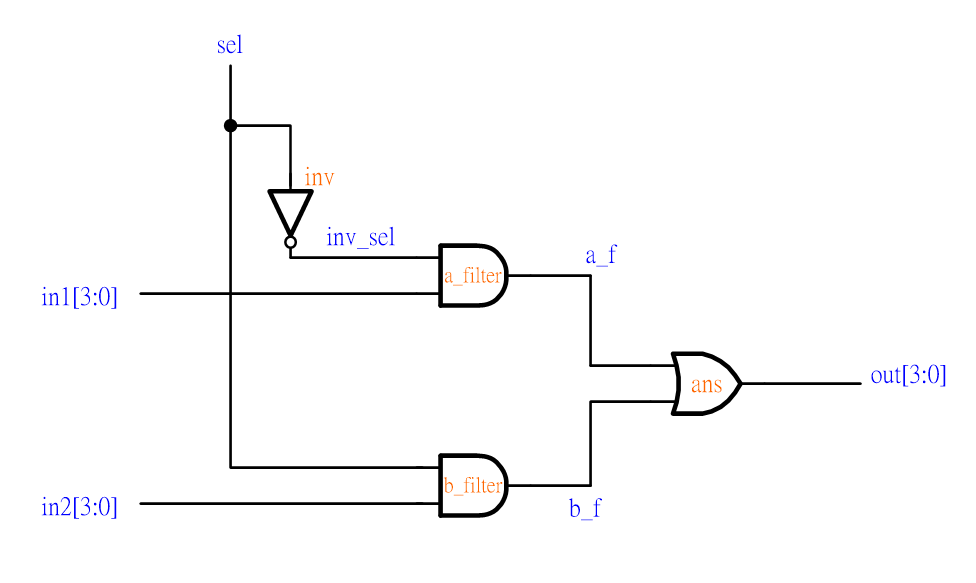
當sel = 2’b11時，d的值必須等於in其餘為0

1. 4-bit simple crossbar switch with MUX/DMUX
2. 邏輯設計圖



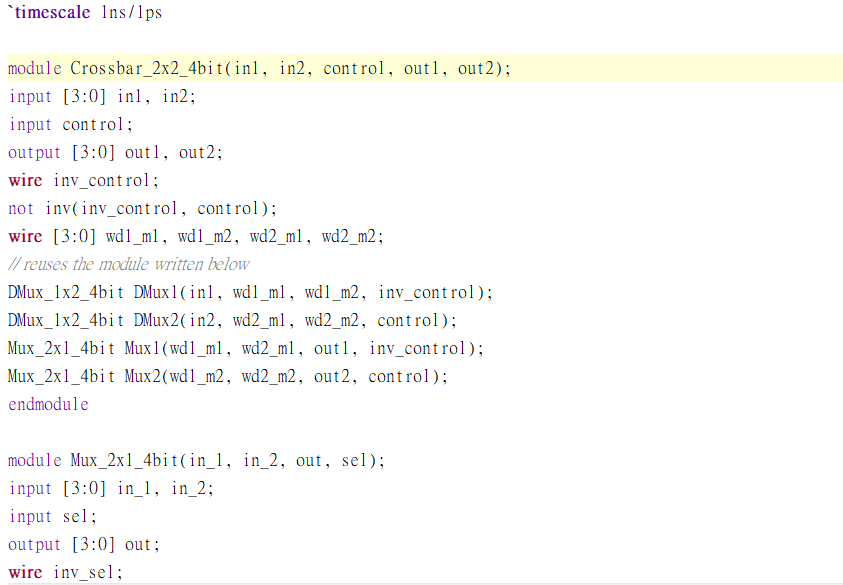
* 1-to-2 DMUX:

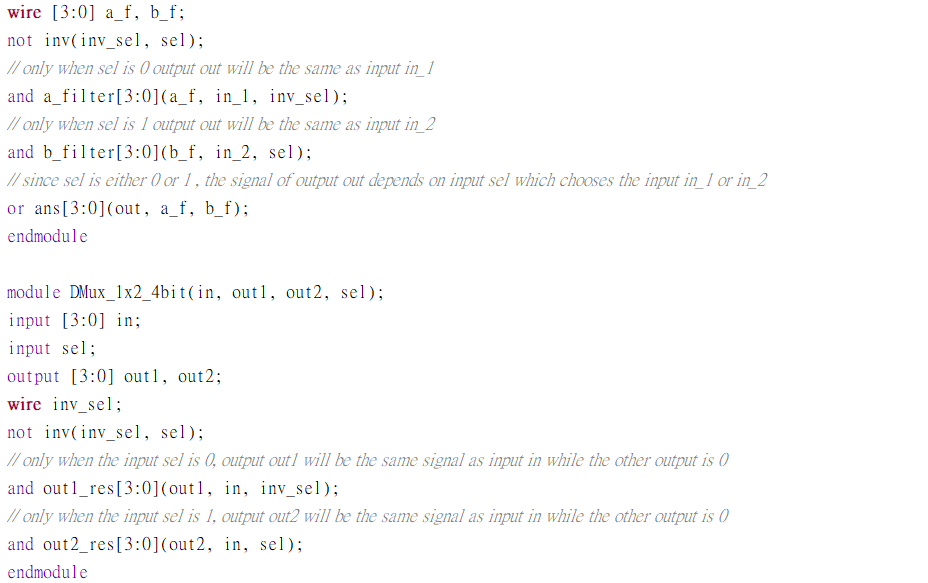


* 2-to-1 MUX:

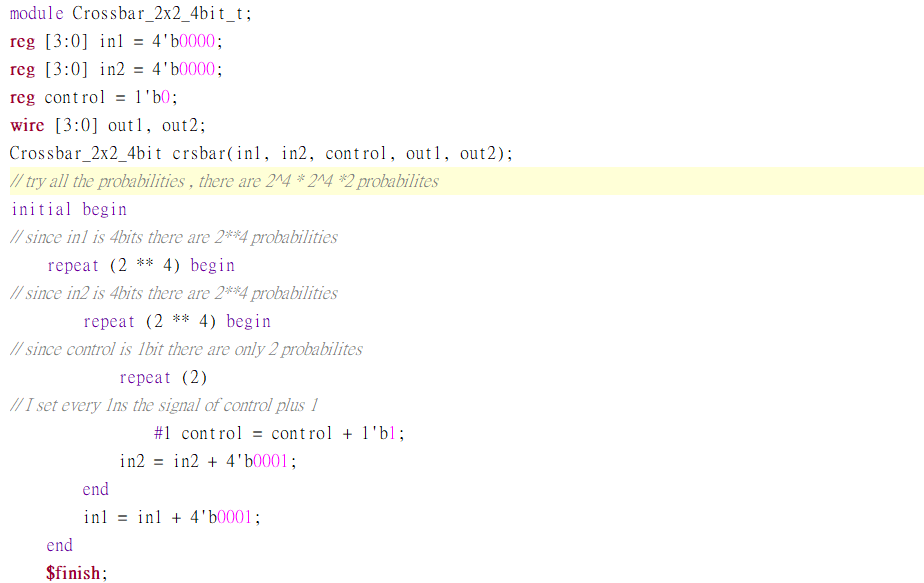
1. Verilog 程式碼

* Lab1\_Team30\_Crossbar\_2x2\_4bit.v

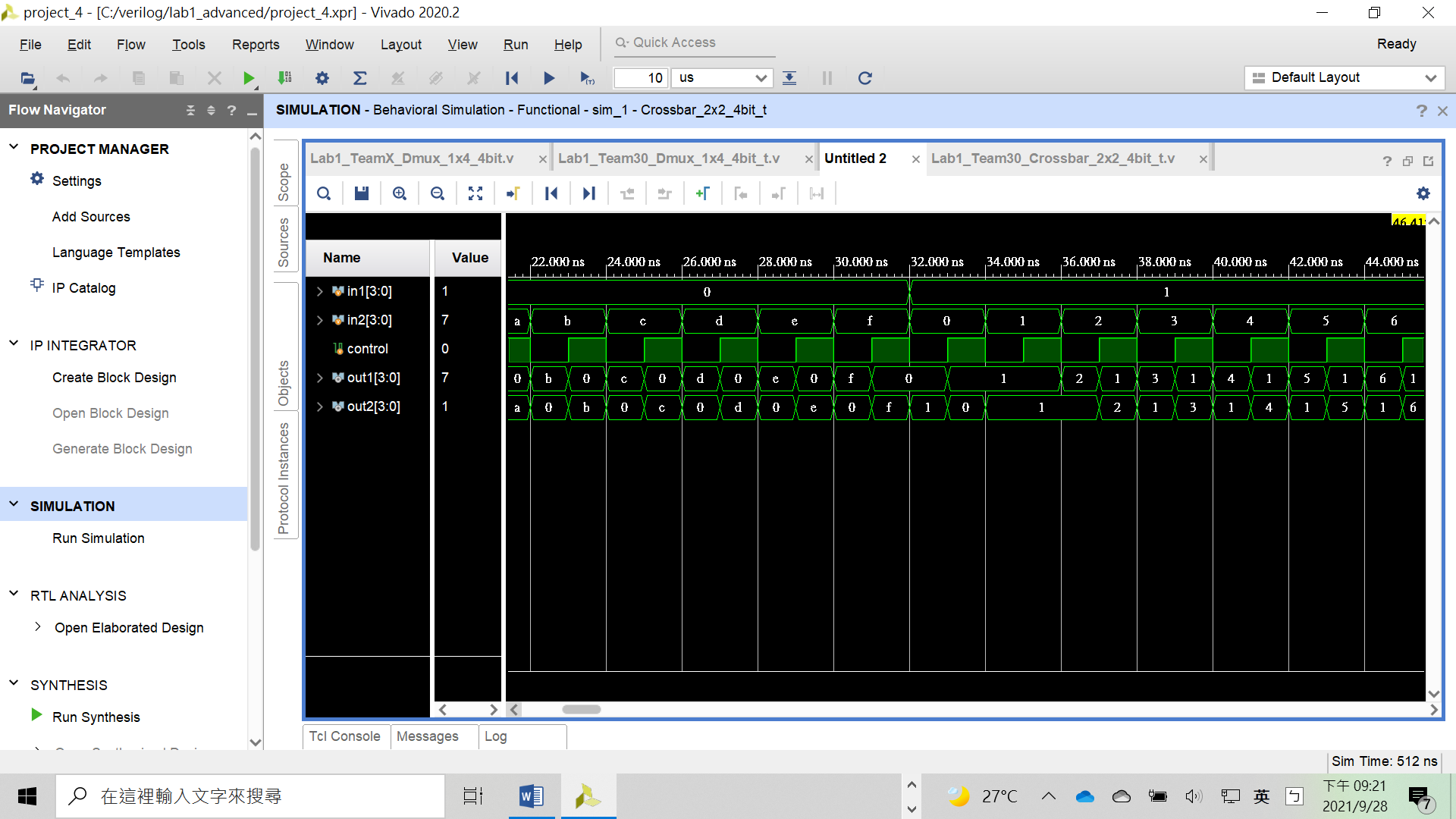




* testbench



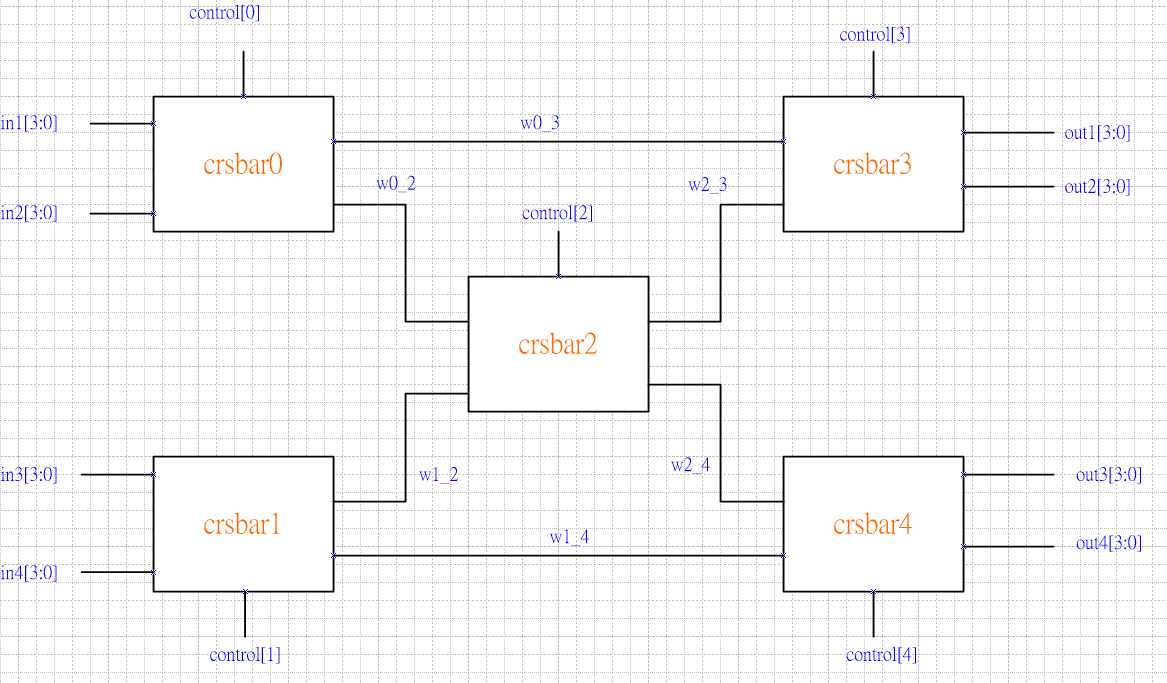


1. 波形圖

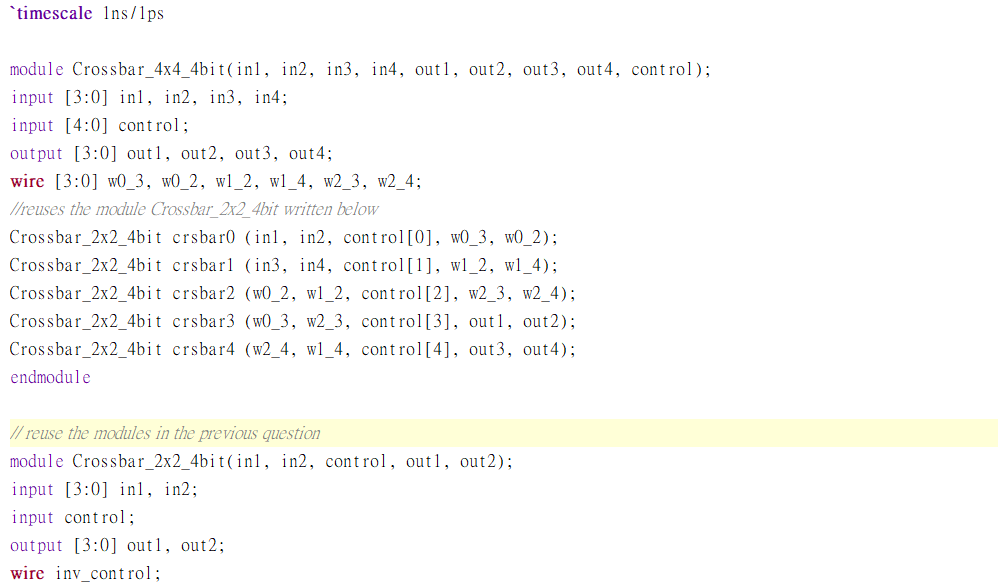
當control = 0， out2的值等於in1的值； out1的值等於in2的值

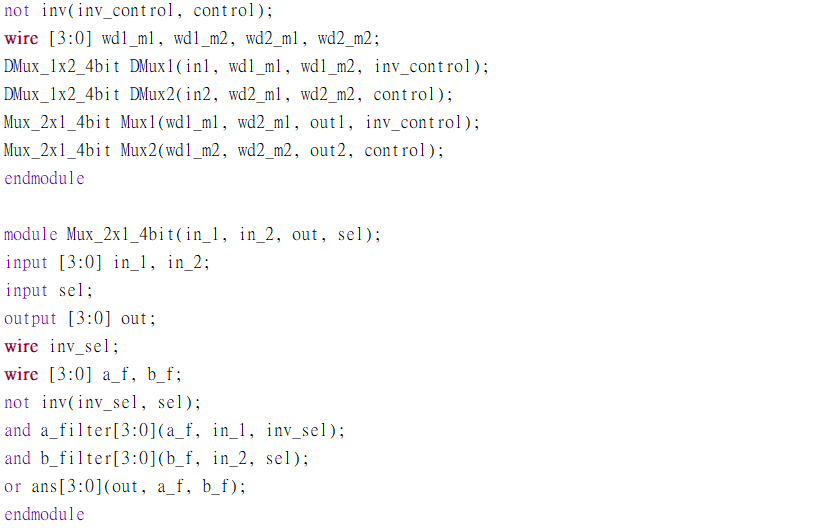
當control = 1， out2的值等於in2的值； out1的值等於in1的值

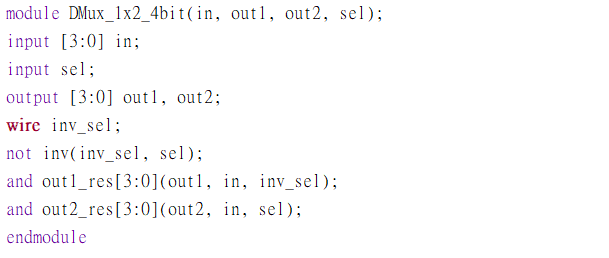
1. 4-bit 4x4crossbar with simple crossbar switch
2. 邏輯設計圖



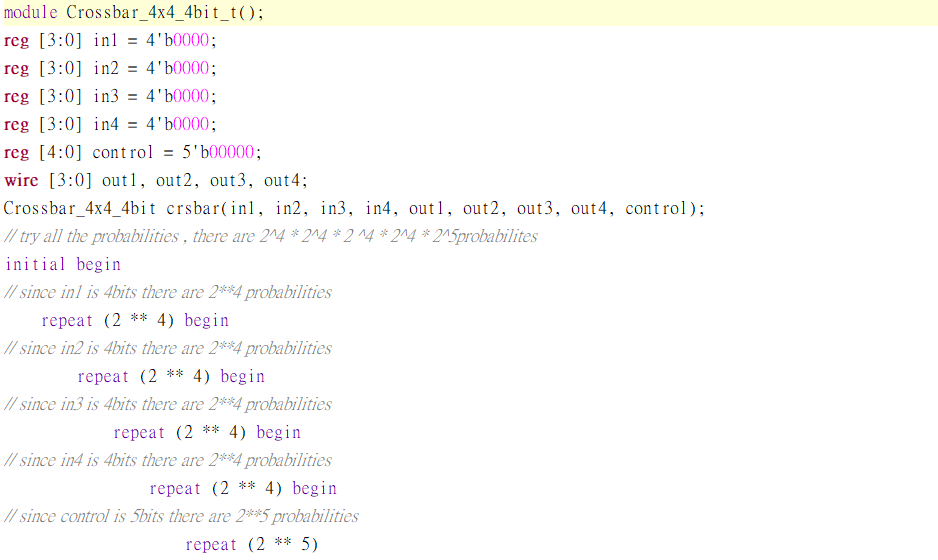
* + - crsbar : Question 2 製作的4-bit simple crossbar
  1. Verilog 程式碼
     + Lab1\_Team30\_Crossbar\_4x4\_4bit.v

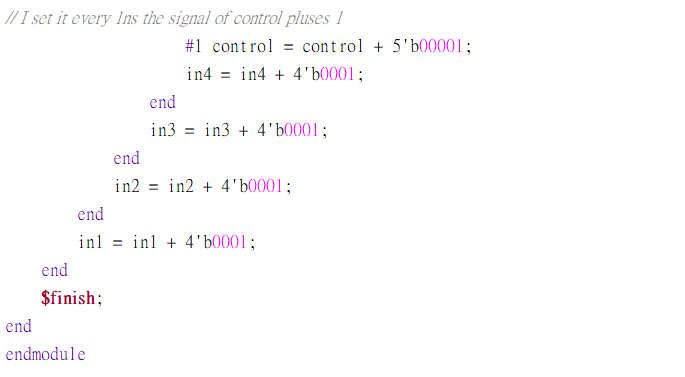




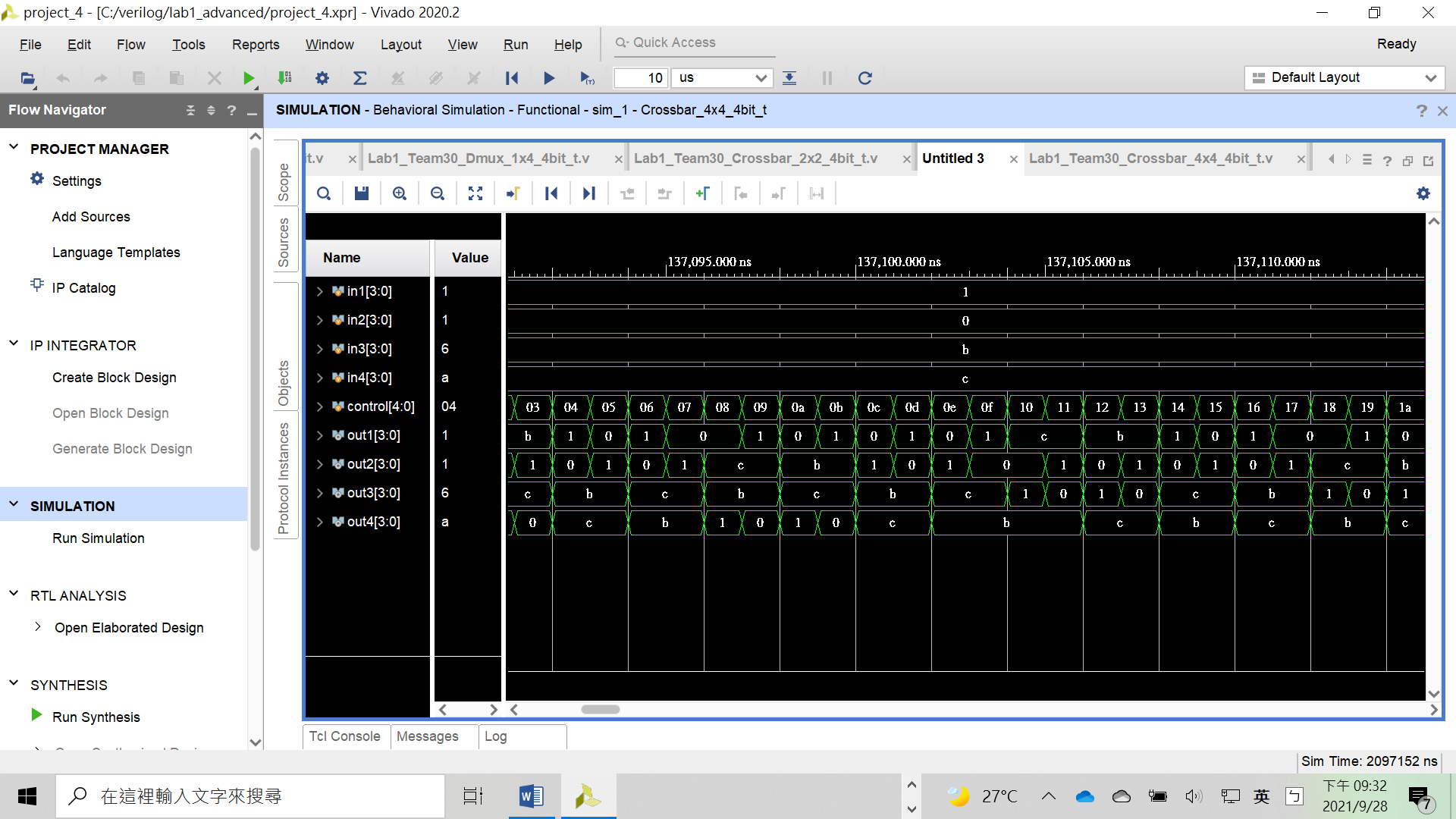


* + testbench



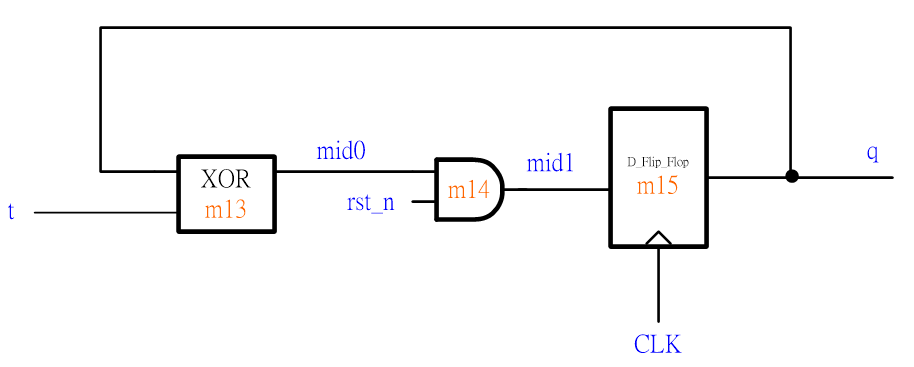


1. 波形圖

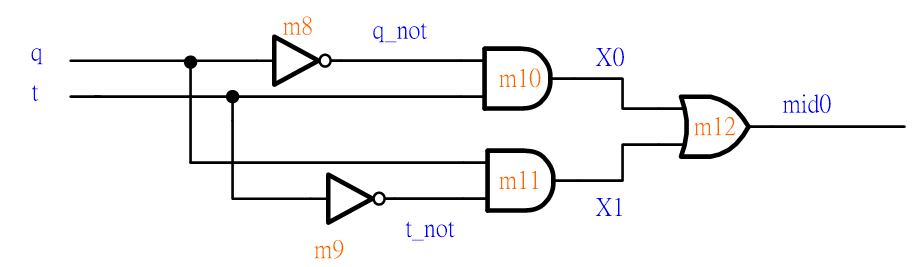


input有4個，理應當output會有24種組合。但從波形圖比對得知，有4種可能不會出現。當in1和in2的訊號同時傳到out3和out4以及當in3和in4的訊號同時傳到out1和out2在4\*4crossbar中是不可能出現的(共4種可能不會出現)。從邏輯設計圖得知，in1或in2經過crsbar0的output一定是w0\_3和w0\_2，但w0\_3是接到crsbar3，因此不可能兩個訊號都經過crsbar4而得到out3和out4。同理in3和in4。

1. 1-bit toggle flip flop (TFF)
2. 邏輯設計圖

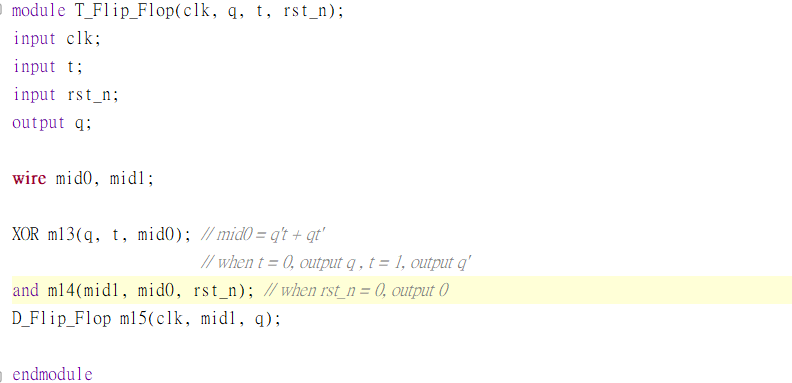


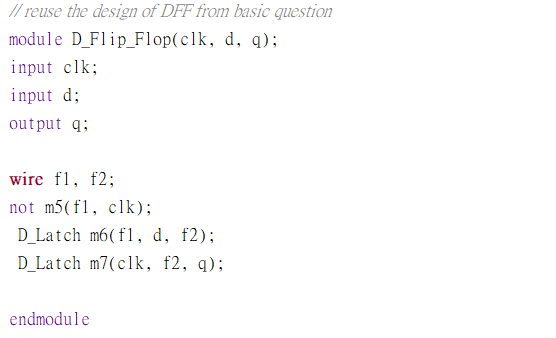
* D\_Flip\_Flop:使用basic question 2做出的DFF
* XOR:

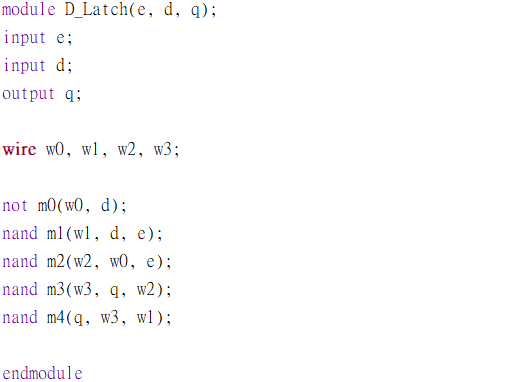


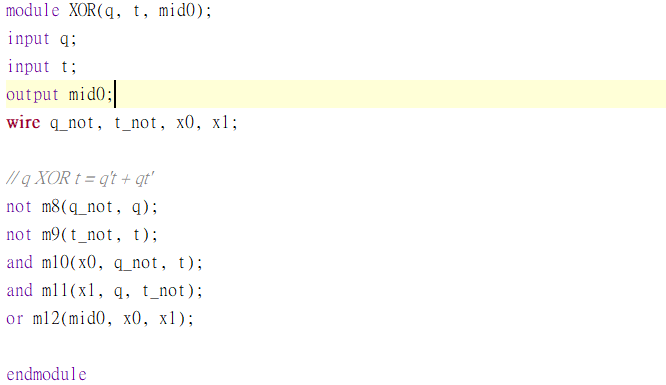
1. Verilog 程式碼

* Lab1\_Team30\_Toggle\_Flip\_Flop.v

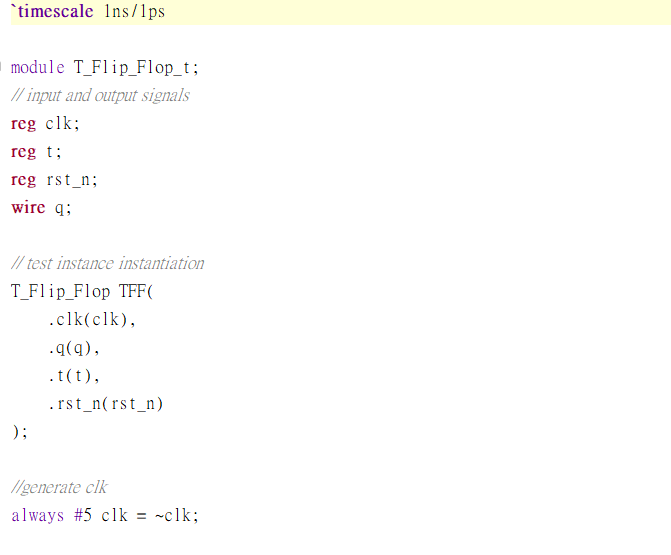


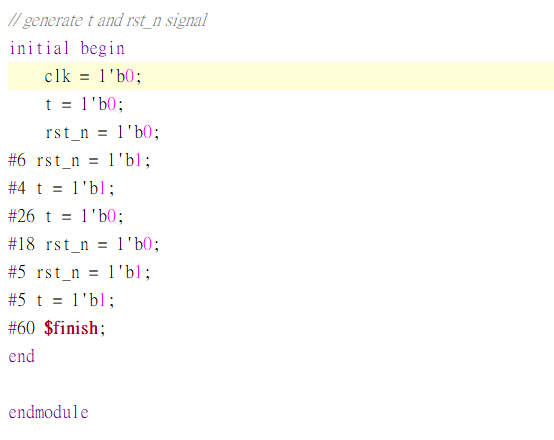




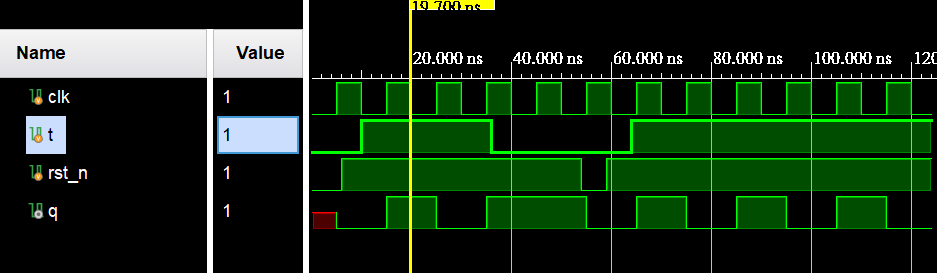


* testbench





1. 波形圖



t = 1, rst\_n = 1時，q在每個clk posedge complement

t = 0, rst\_n = 1時，q 不產生變化

rst\_n = 0且clk 處於posedge狀態時，q reset變成0

1. 工作分配

劉奇泓:Question 4 ，FPGA題的xdc.檔及連接，實驗報告製作

洪聖祥:Question1~3，FPGA題的程式碼，實驗報告中說明1~3題的波形圖

1. 心得感想

劉奇泓:這是邏設實驗課程中的第一個需要與組員合作的Lab，我在這一次實驗中學會了自己寫出完善的testbench，也學會了把verilog程式碼轉換成bitstream、寫入FPGA板的過程，希望之後的實驗我能對verilog越來越熟練，並更有效率地完成每個實驗!

洪聖祥:這次的lab需要用到大量的verilog寫各種module，在寫問題的過程中，我從本來只會粗淺的語法慢慢摸索並釐清各個module的邏輯，也第一次自己學會寫testbench。雖然在寫testbench以及燒錄fpga 時遇到瓶頸，但在跟組員討論以及自己查資料後我逐漸找到解決問題的方法。從這次的lab我學到團隊分工、溝通的重要。相信在經過這次lab的經驗下次lab能更迅速完成實驗。