

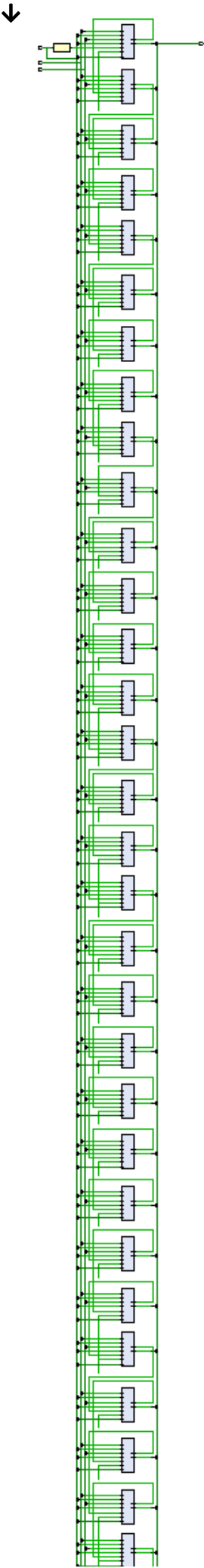
Computer Organization Lab1

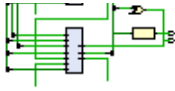
1. Architecture Diagrams

1 bit ALU



32 bit ALU





2. Answer the following Question

How overflow is calculated?

- 我們觀察一下什麼時候會發生 overflow
- 第一種情況: 兩個正數相加, 且 sign bit 有進位, 最終 sign bit 為 1
- 第二種情況: 兩個負數相加, 且 sign bit 有進位, 最終 sign bit 為 0
- 寫成表格會長這樣

a_i	b_i	carry_in	carry_out
0	0	1	0
1	1	0	1

- 會發現 $\text{overflow} = \text{carry_in} \wedge \text{carry_out}$
- 但是在 slt overflow 要特別判斷
- 這種情況不會發生 overflow
- 修正得到 $\text{overflow} = ((\text{carry_in} \wedge \text{carry_out}) \&\& !(\text{operation} == 2'b11))$

Explain why ALU control signal of SUB is 0110 and NOR is 1100 ?

- 我們觀察一下 0110 是啥意思
- 代表 $a_invert = 0, b_invert = 1, operation = 10$
- 觀察一下 $a - b = a + b' + 1$, 而在 bit_ALU, $operation = 10$ 代表 | (OR)
- 所以這也就是為啥我們要那樣設定 SUB 的原因
- 因為可以剛好湊出 $a + b'$
- 此外, 我們來觀察一下 1100 是啥意思
- 代表 $a_invert = 1, b_invert = 1, operation = 00$
- 觀察一下 $a \text{ nor } b = (a \text{ or } b)' = a' \text{ and } b'$, 而在 bit_ALU, $operation = 00$ 代表 & (AND)
- 所以這也就是為啥我們要那樣設定 NOR 的原因
- 因為可以剛好湊出 $a' \text{ and } b'$

(2 con'd) If you assign different signal to these operation, what problems you may encountered?

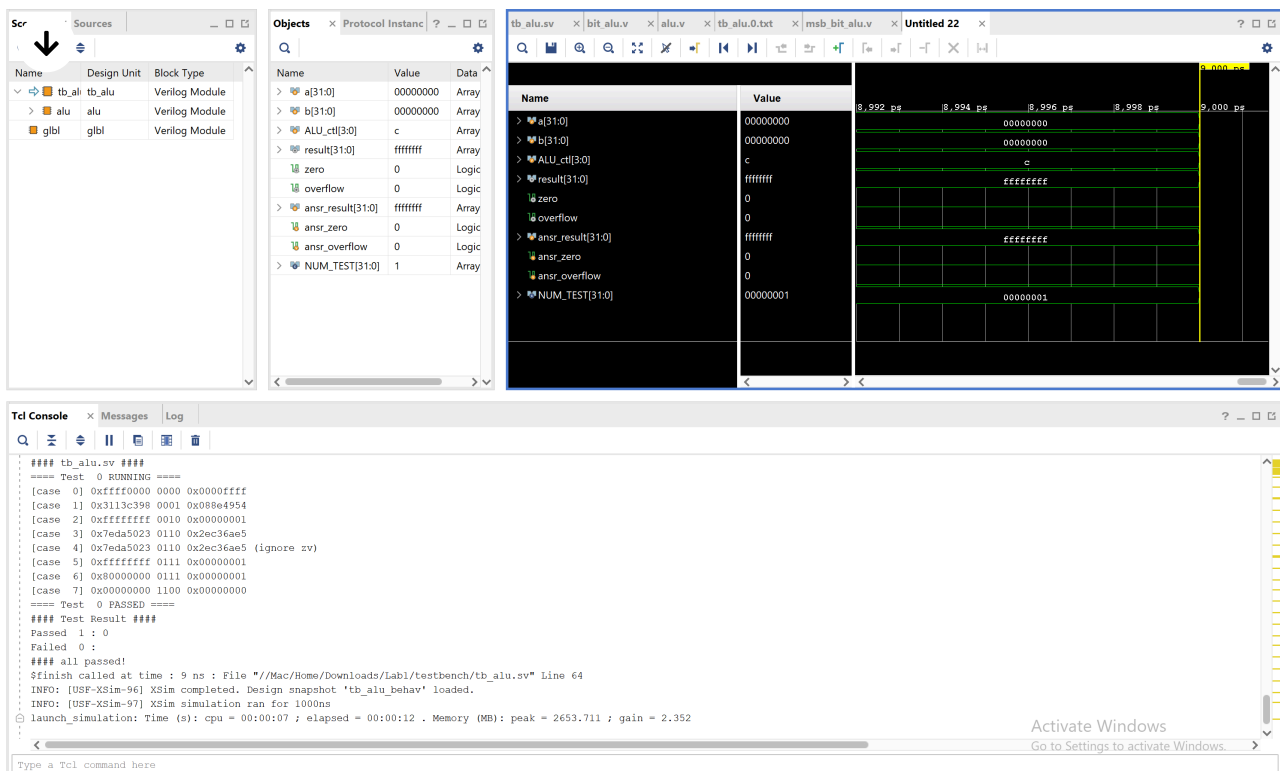
- 我們可能會因此得到錯誤的布林運算式
- 導致我們算出非預期的答案

True or false: Because the register file is both read and written on the same clock cycle, any MIPS datapath using edge-triggered writes must have more than one copy of the register file. Explain your answer.

- 錯的
- 我們會依照訊號去 triggered 寫入或讀取
- 不會同時讀入且讀取

3. Experimental Result

Show the waveform screen shot of the testbench tb_alu.0.txt result.



What other cases you've tested? Why you choose them?

- 2147483647 1 ADD -2147483648 0 0

- 因為我們只能表達 $2^{31} - 1 \sim -2^{31}$ 的數字

4. Problems Encountered & Solution (optional)

- verilog 不會寫
- 參考線上教學
- 菜鳥教程 (<https://www.runoob.com/w3cnote/verilog-tutorial.html>).
- Verilog HDL 教學講義 (https://hom-wang.gitbooks.io/verilog-hdl/content/Chapter_01.html).
- 天璇加法器 (https://www.youtube.com/watch?v=fl4eChygKV8&list=PLkH9pBMaZuHQ0_P26d8ctZSd9trPajCml&index=11).

5. Feedback (optional)

- no comment