

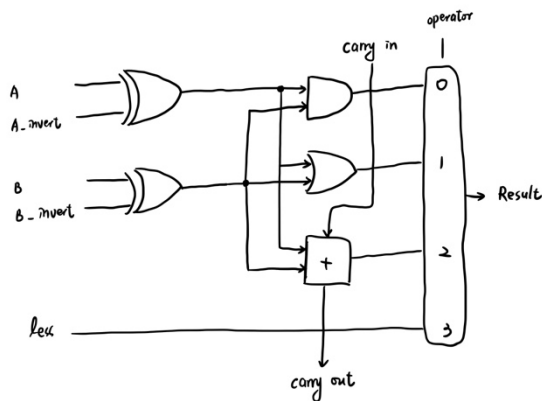
Computer Organization

Lab 1: 32-bit ALU

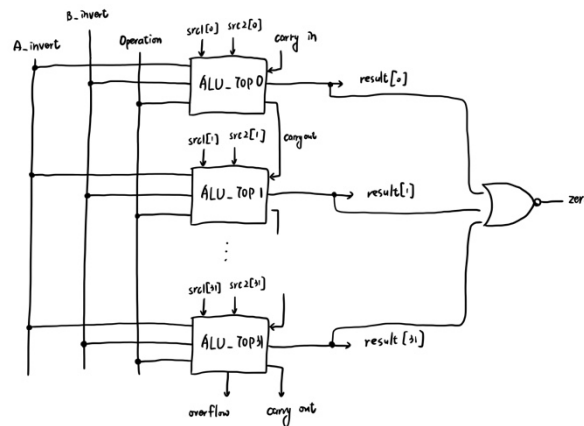
Student ID:110652011 Name: 郭佳翰

1. Architecture diagrams

ALU_TOP



ALU



2. Hardware module analysis

1. Adder: 全加器

2. ALU_top

- 用 XOR 完成 invert
- AND, OR: 直接設置 result 為 and 或 or.


- iii. ADD: 用 Adder 完成
- iv. SLT: 只取 adder 中 carry out 的部分，確認相減後是否為負。

3. ALU

- i. 用 submodule alu_top0~31
- ii. A_invert, B_invert, operator 取自 ALU_control[1:0]。
- iii. 額外宣告: wire r, co, 用來存 alu_top 的 result 與 carry out。
- iv. zero = (r === 32'd0)
- v. STL 中，result[0]為 co[31], 檢查是否為負。

3. Experimental result

All data are correct with old and new testcases.



```
ruccccc@Rues-MBP:~/Library/CloudStorage/OneDrive-國立陽明交通大學/大學/Co...  
└─┬ /C/OneDrive-國立陽明交通大學/大學/Co... | main 13 | 21:35:09  
> ./testbench  
VCD info: dumpfile testbench.vcd opened for output.  
*****  
Congratulation! All data are correct!  
*****  
└─┬ /C/OneDrive-國立陽明交通大學/大學/Co... | main 13 | 21:35:12  
>
```

4. Problems you met and solutions

- 1. 我平時使用的筆電是 M1 的 MAC，但 vivado 不支援 M1，所以改用 iverilog 和 gtkwave 來測試。
回到家用舊的 windows 電腦安裝仍舊不順利，installer 會在安裝過程中閃退，重複了幾次都一樣，所以打算之後繼續使用 iverilog 和 gtkwave 完成作業，遇到問題再向同學借電腦。
- 2. 因為是轉系生，大一時沒有修過數位電路設計，所以完全沒接觸過 verilog，不過在 google 與各路大神的部落格的幫助下，寫作業的過程不算太痛苦。
- 3. 在 alu 中，那 alu_top 的 result 不能直接用 reg，會報錯，所以額外宣告 wire vector r，再複製 r 到 result。

5. Summary

Verilog 好難。