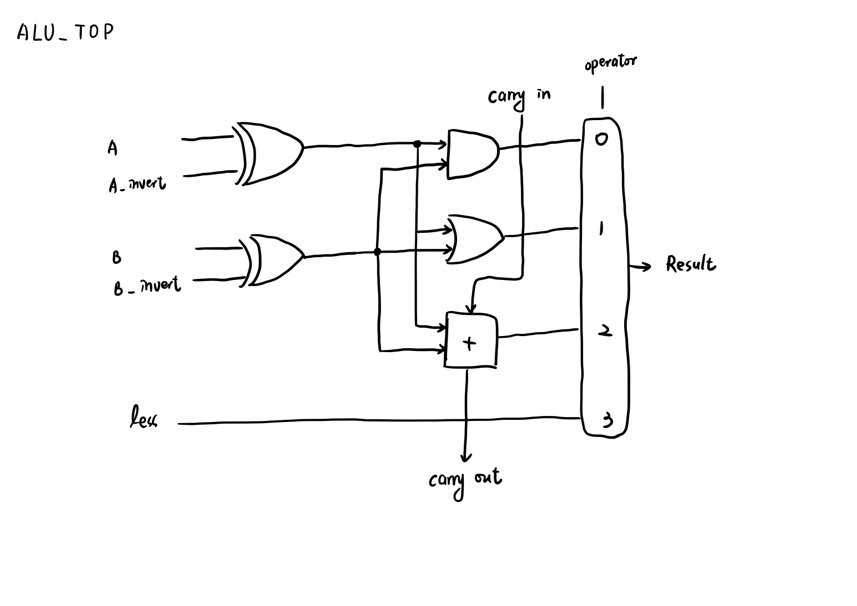
**Computer OrganizationLab 1: 32-bit ALU**

Student ID:110652011 Name: 郭佳翰

1. Architecture diagrams



一張含有 圖表, 圖解 的圖片

自動產生的描述

1. Hardware module analysis
2. Adder: 全加器
3. ALU\_top
   1. 用XOR完成invert
   2. AND, OR: 直接設置result為and或or.
   3. ADD: 用Adder完成
   4. SLT: 只取adder中carry out 的部分，確認相減後是否為負。
4. ALU
   1. 用submodule alu\_top0~31
   2. A\_invert, B\_invert, operator取自ALU\_control[1:0]。
   3. 額外宣告: wire r, co, 用來存alu\_top的result與carry out。
   4. zero = (r === 32’d0)
   5. STL中，result[0]為co[31], 檢查是否為負。
5. Experimental result

All data are correct with old and new testcases.

一張含有 文字 的圖片

自動產生的描述

1. 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。

* 1. 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。
  2. 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。

1. Summary

Verilog好難。