

ALU and Shifter

10801128 陳俊鴻

Q1: 請簡略說明你所新增的檔案以及其功能。倘若你沒有新增檔案，則請簡略說明你的 ALU.v 中設計了哪些電路。

A1: ALU_LSB.v 和 ALU_MSB.v 是我新增的兩個檔案，因為 result[0] 的計算牽涉到 set，然而 set 是透過第15號 1-bit ALU (ALU_MSB) 之 full adder 求得，這個過程會牽涉到整個 carry chain，故第0號 1-bit ALU (ALU_LSB) 必須先產出第0個 carry 才能計算出 set，因此 result[0] 的計算必須在 ALU_MSB 之後。至於 ALU_MSB 既需要算出 set 又需要算出 overflow，output 與其他 1-bit ALU 很不一樣，因此也獨立出來運行。

Q2: 假設在 ALU 設計中不需要實作減法運算，請問哪幾條控制訊號可以刪除？假設 ALU 設計中不需要 NOR 運算，請問哪幾條控制訊號可以刪除？

A2: 若排除減法的話，B invert 便沒有意義，因為對於 and, or, add, slt 而言可以 operation 決定，至於 nor 則可以 A invert 決定。若排除 nor 的話，A invert 便沒有意義，因為對於 and, or, add, slt 而言可以 operation 決定，至於 slt 則可以 B invert 和 operation 決定。

Q3: 試算出你所設計的 16-bit ALU 之 propagation gate delay，並說明之。

$$A3: t_{pd} = \max(t_{PHL}, t_{PLH})$$

$$\text{ALU_LSB: } 20 (\text{xor}) + 40 (\text{full adder}) = 60$$

$$\text{ALU_1bit: } 20 (\text{xor}) + 40 (\text{full adder}) + 10 (\text{and}) + 10 (\text{or}) = 80$$

$$\text{ALU_MSB: } 20 (\text{xor}) + 40 (\text{full adder}) + 10 (\text{and}) + 10 (\text{or}) = 80$$

$$\begin{aligned} \text{Total delay} &= 60 (\text{ALU_LSB}) + 15 * 80 (1\text{-bit ALU}) + 80 (\text{ALU_MSB}) + 2 * 10 (\text{or}) + 10 (\text{not}) \\ &= 1370 \end{aligned}$$

Q4: 請寫出你在 Lab2 中所碰到的困難，以及實作完成後的心得。

A4: Lab1都寫出來了，Lab2好像沒什麼困難，心得的話Lab1倒是比較多，Lab2太快寫完了沒什麼想法。

Q5: 請簡單說明你是如何修改ALU，使在執行“set on less than”時，不致因發生溢位而產生錯誤結果。

A5: 因為 slt 的溢位僅發生在 positive - negative 和 negative - positive ，故發生溢位時，set 和 zero 必相反，在計算完 ALU_MSD 後，將 set 換為 set xor overflow 即可矯正。

Q6: 請以你設計Barrel shifter電路的方法，畫一張4-bit Barrel shifter的電路圖。

A6: 抱歉，因為這是我第一次畫電路圖，上網爬到只有網頁版的繪圖工具，所以做的可能蠻簡陋的。

