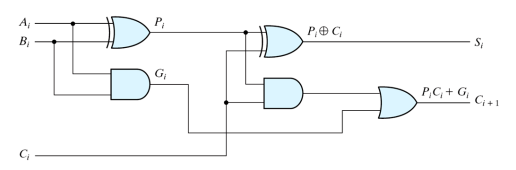
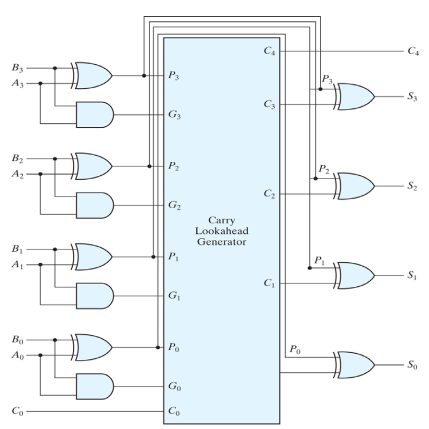
實用數位系統設計－HW1-1

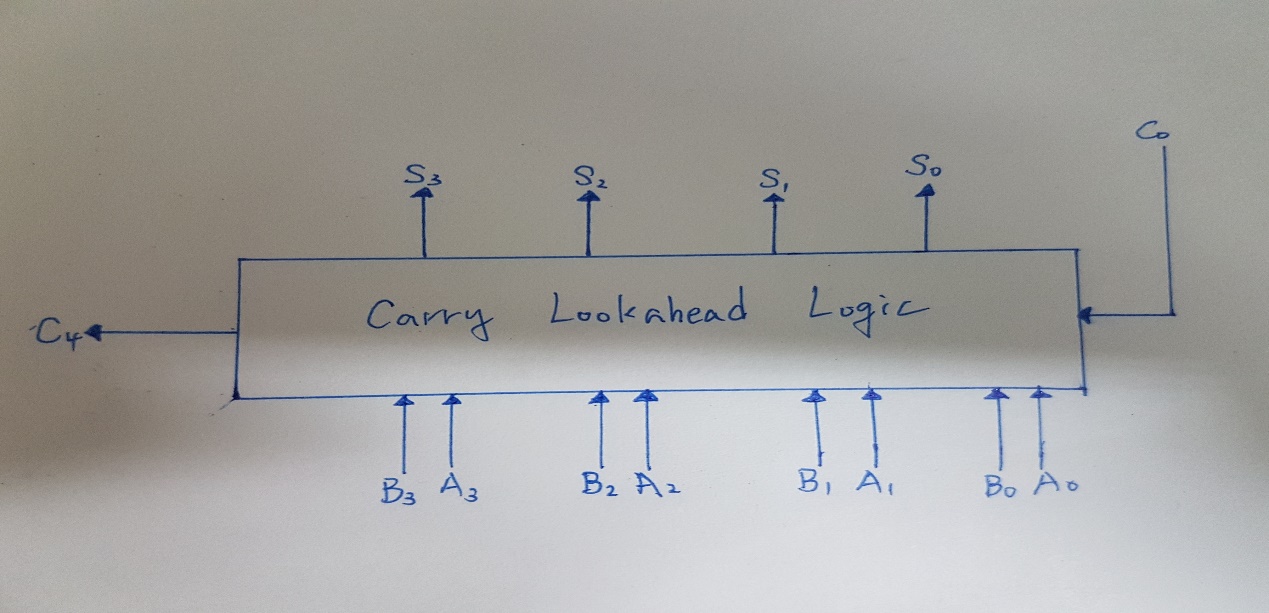
學號：B035020026 姓名：李冠霖

設計原理：

這次加法器我使用前瞻進位加法器實作，使用兩個4-bit的前瞻進位加法器構成一個8-bit的加法器。前瞻進位需要求出Generate, Propagate，公式為：

進而求出Sum和Carry：



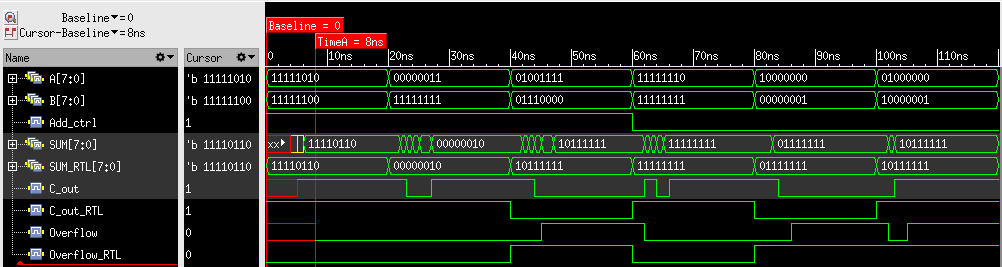
展開後同時計算才得已解決漣波進位加法器延遲的問題，同時因為展開算式，將會大幅增加邏輯閘的使用(使用面積增加)，尤其越高位的計算將會越複雜，也因此為了減少複雜度，又能獲得加速效果，採用了4+4bits，將低位4-bit的Carry作為高位4-bit的輸入，這邊會產生延遲，但速度已經比漣波進位快上許多！

此加法器採用的是有號數運算，在進行加法或減法的部分提供一個input (Add\_ctrl)控制，若進行減法，則先將減數(B)與(~Add\_ctrl)進行xor後進行相加即為A-B的結果。減法的Carry Out需再與(~Add\_ctrl)進行xor後才是正確輸出。

最後還有Overflow的檢測，取同號相加結果為異號者(正+正=負)，以及異號相減結果者與被減數異號者(正-負=負)為1，用卡諾圖化減則為

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AB PS | 00 | 01 | 10 | 11 |
| 00 |  | 1 |  |  |
| 01 |  |  |  | 1 |
| 10 |  |  | 1 |  |
| 11 | 1 |  |  |  |

A為被加數最高位，B為加數最高位，P為正負號(正=0)，S為總和最高位。

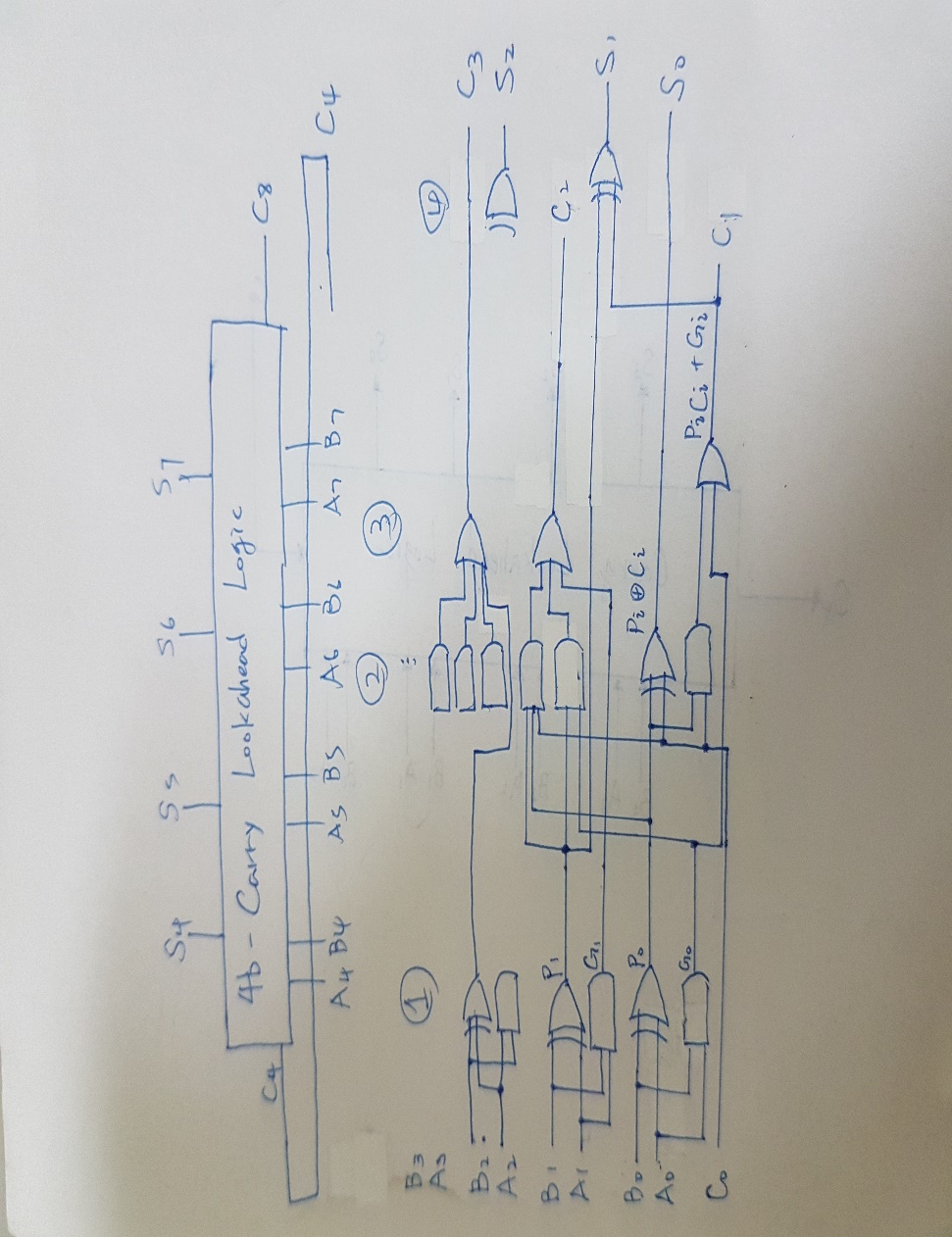
結果分析：

模擬指令：ncverilog HW1\_1\_tb.v +access+r

本加減法器可以顯示(-128)～(+127) (signed number)

使用六個輸入進行測試，分別為：

3+(-1)、(-6)+(-4)、79+112、(-2)-(-1)、(-128)-1、64-(-127)

測試同號異號的加減，並同時檢查是否可以正常處理overflow的狀況，從圖中可以看出在加入了delay的gate level code在overflow的波形有明顯的延遲後才顯示出正確的答案(RTL版本未加上delay)，也同時符合實際運作的情形，加入delay的方式為在每個基礎邏輯閘加上#1的delay，若不加上delay則在input進去後0秒就直接顯示，不符合實際電路運作狀況。

最長的路徑為Overflow的輸出，CLL產生3個delay，和產生S所需要的xor閘(1 delay)，以及前面判斷加減法的not+xor(2 delay)，最後的兩層判斷Overflow，3+1+2+2=8

心得討論：

對於第一次使用工作站，從架設環境到撰寫程式碼花了不少時間，加上非本科系，過去也沒有實際操作過實體電路，因此對硬體描述語言只停留在理論階段，在實際撰寫程式碼的時候，時常碰到語法錯誤導致到處都是bug，電路的設計也是經過查詢多個參考資料學習而來的結果。