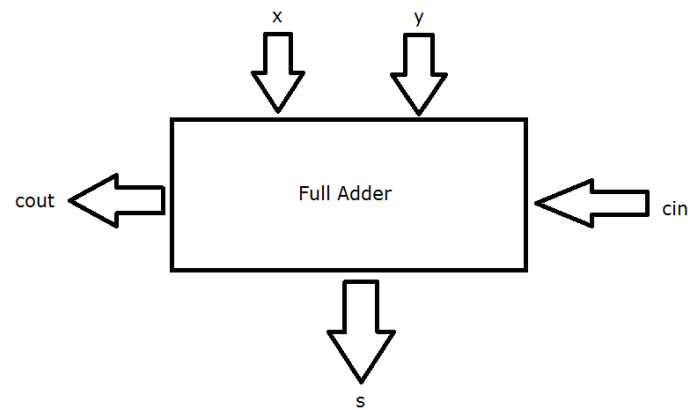


Lab 1: Verilog HDL

1. Full Adder

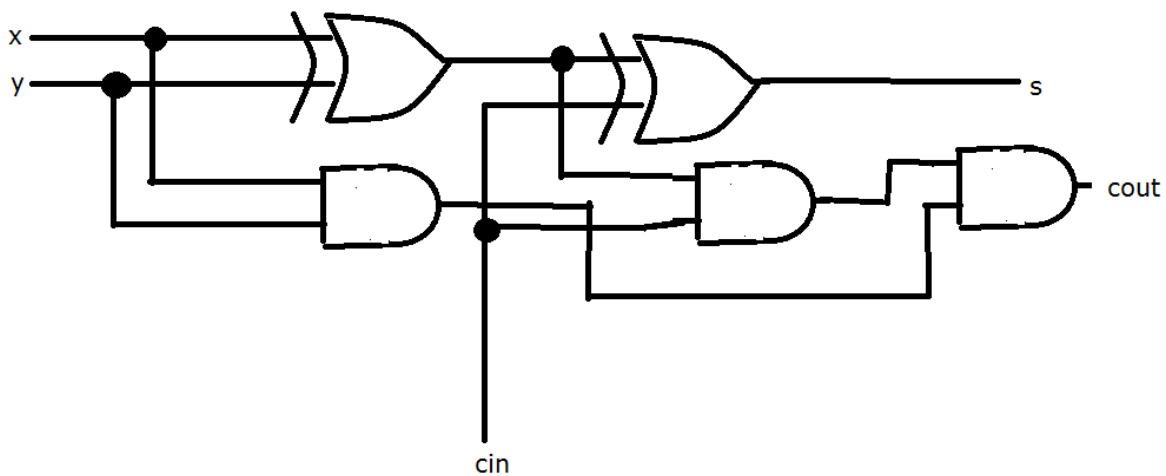
Design Specification

- (1) Input: x, y, cin.
- (2) Output: cout, s.
- (3) Block diagram:



Design Implementation

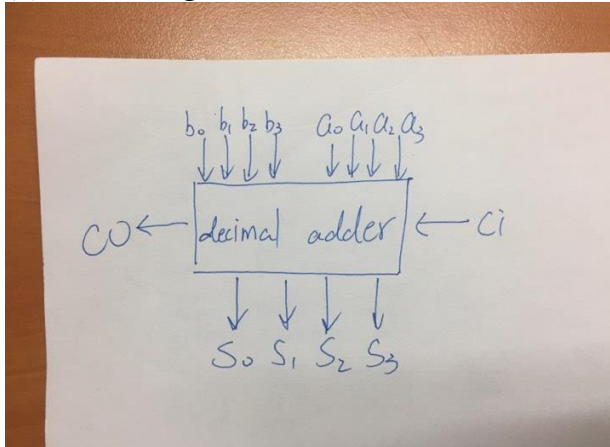
- (1) Logic function:
 $s = x \oplus y \oplus \text{cin};$
 $\text{cout} = (x \& y) \mid (\text{cin} \& (x \oplus y));$
- (2) Logic Diagram:



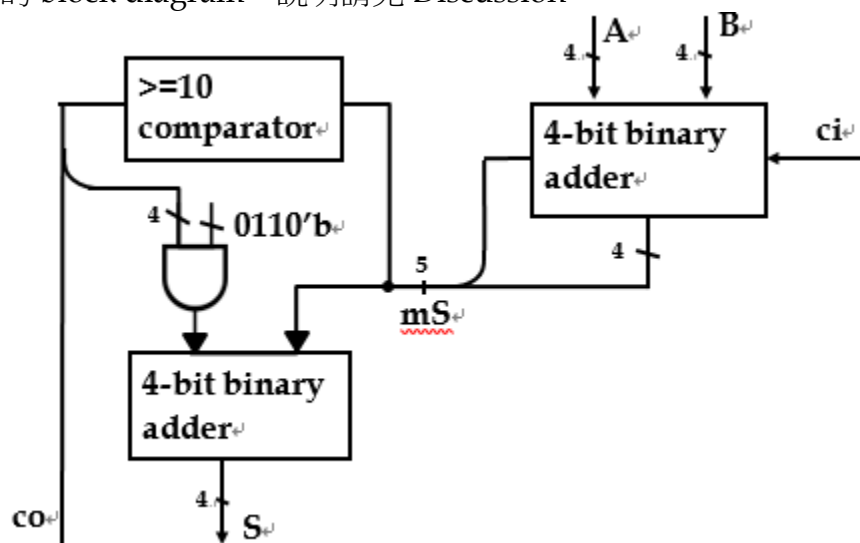
2. single digit decimal adder

Design Specification

- (1) Input: $a[3:0]$, $b[3:0]$, ci .
- (2) Output: $s[3:0]$, co .
- (3) Block diagram:



詳細的 block diagram，說明請見 Discussion



Design Implementation

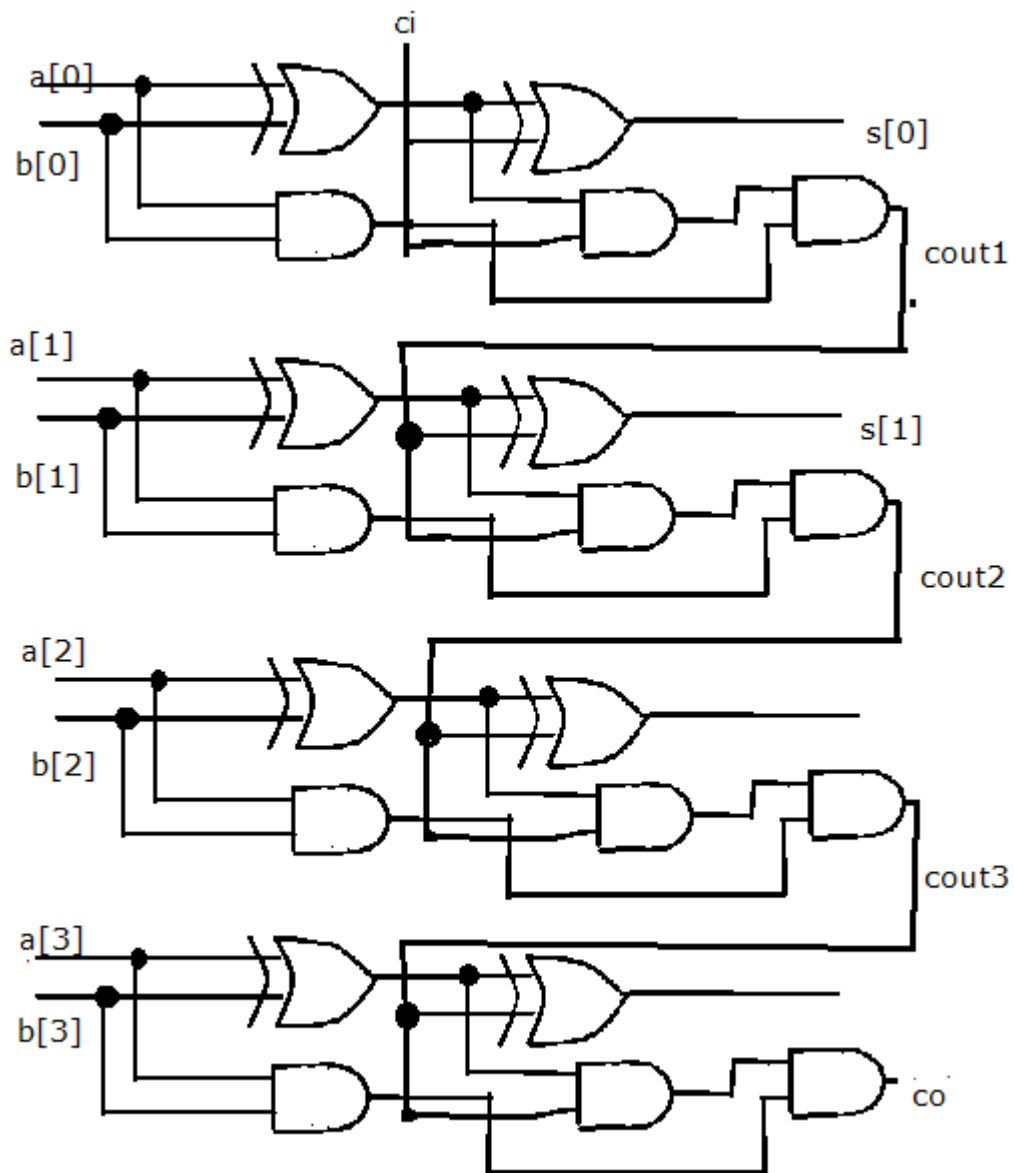
- (1) Logic Function:

我用 Ripple Carry Adder 實作本題，而每個 Full Adder 輸出中有一進位值，在我的 Verilog 程式碼中是以 wire 儲存此值

$$s[i] = a[i] \oplus b[i] \oplus co[i];$$

$$co = (a[3] \& b[3]) \mid (ci[3] \& (a[3] \oplus b[3]));$$

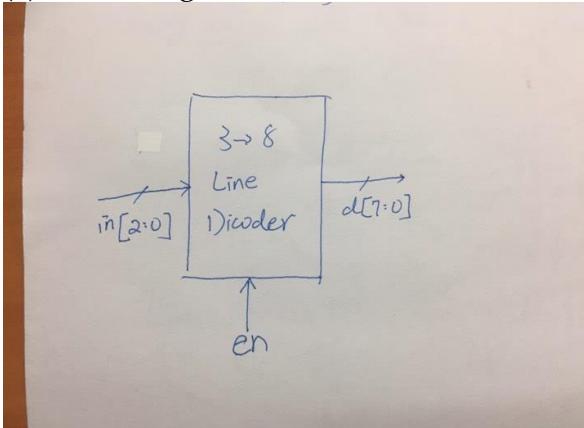
- (2) Logic Diagram: (Full adder 部分)



3. 3-to-8-line decoder with enable

Design Specification

- (1) input in[2:0], en.
- (2) Output d[7:0].
- (3) Block diagram:

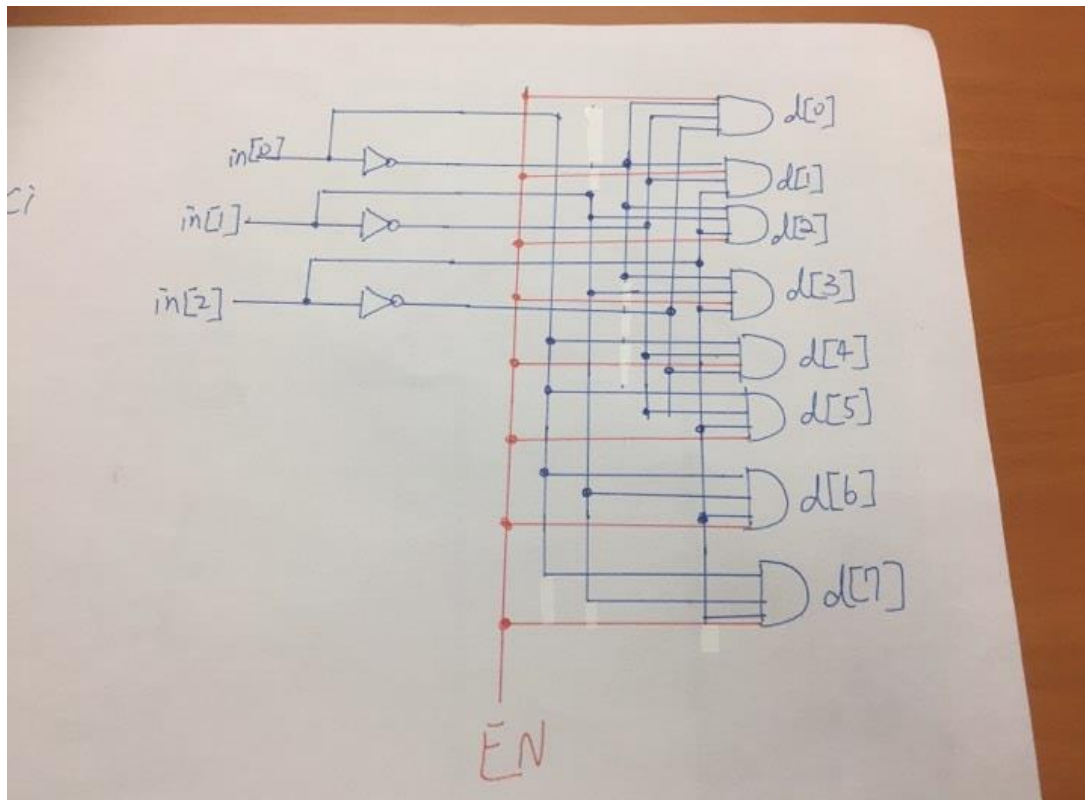


Design Implementation

- (1) Logic function

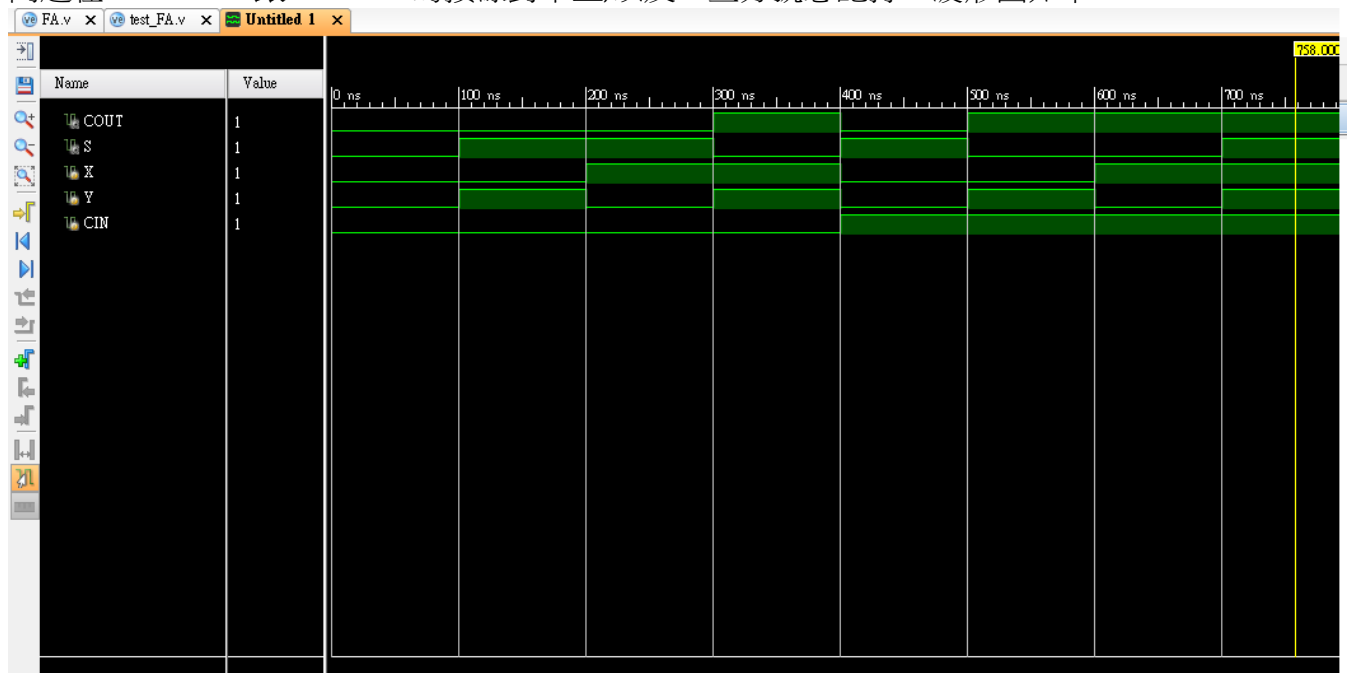
$$\begin{aligned} d[0] &= en \& (\sim in[2]) \& (\sim in[1]) \& (\sim in[0]); \\ d[1] &= en \& (\sim in[2]) \& (\sim in[1]) \& (in[0]); \\ d[2] &= en \& (\sim in[2]) \& (in[1]) \& (\sim in[0]); \\ d[3] &= en \& (\sim in[2]) \& (in[1]) \& (in[0]); \\ d[4] &= en \& (in[2]) \& (\sim in[1]) \& (\sim in[0]); \\ d[5] &= en \& (in[2]) \& (\sim in[1]) \& (in[0]); \\ d[6] &= en \& (in[2]) \& (in[1]) \& (\sim in[0]); \\ d[7] &= en \& (in[2]) \& (in[1]) \& (in[0]); \end{aligned}$$

- (2) Logic diagram

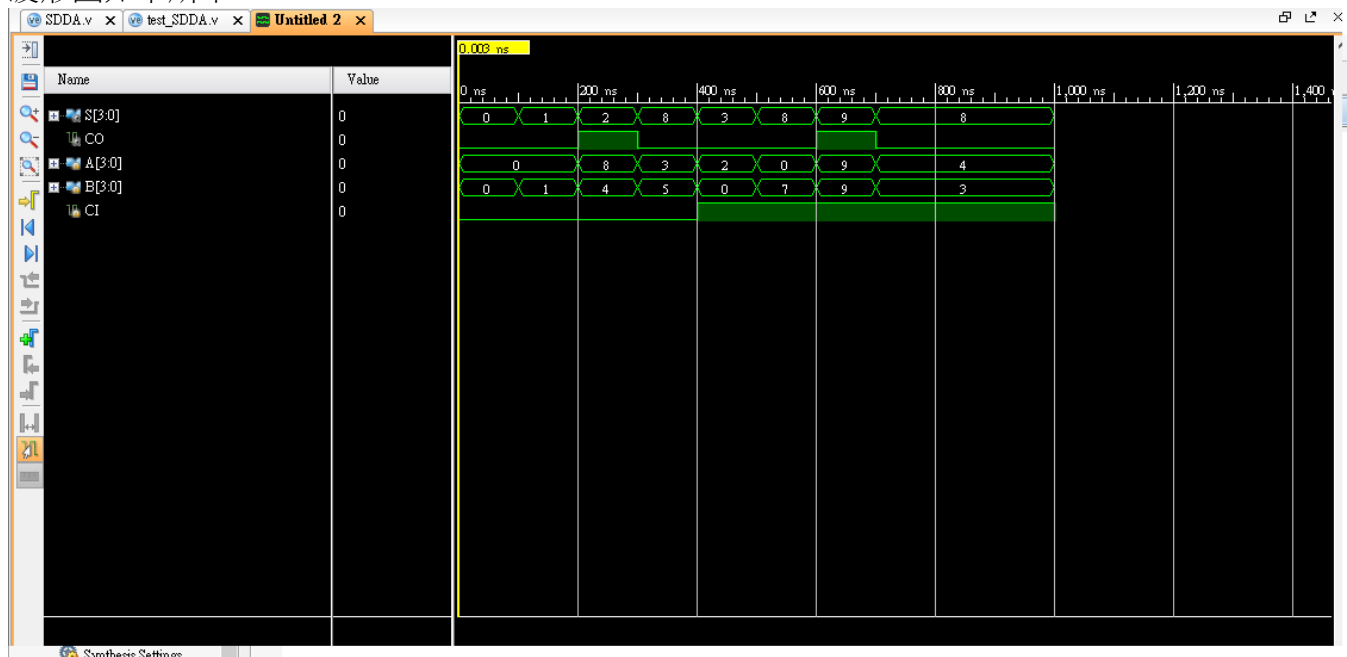


Discussion

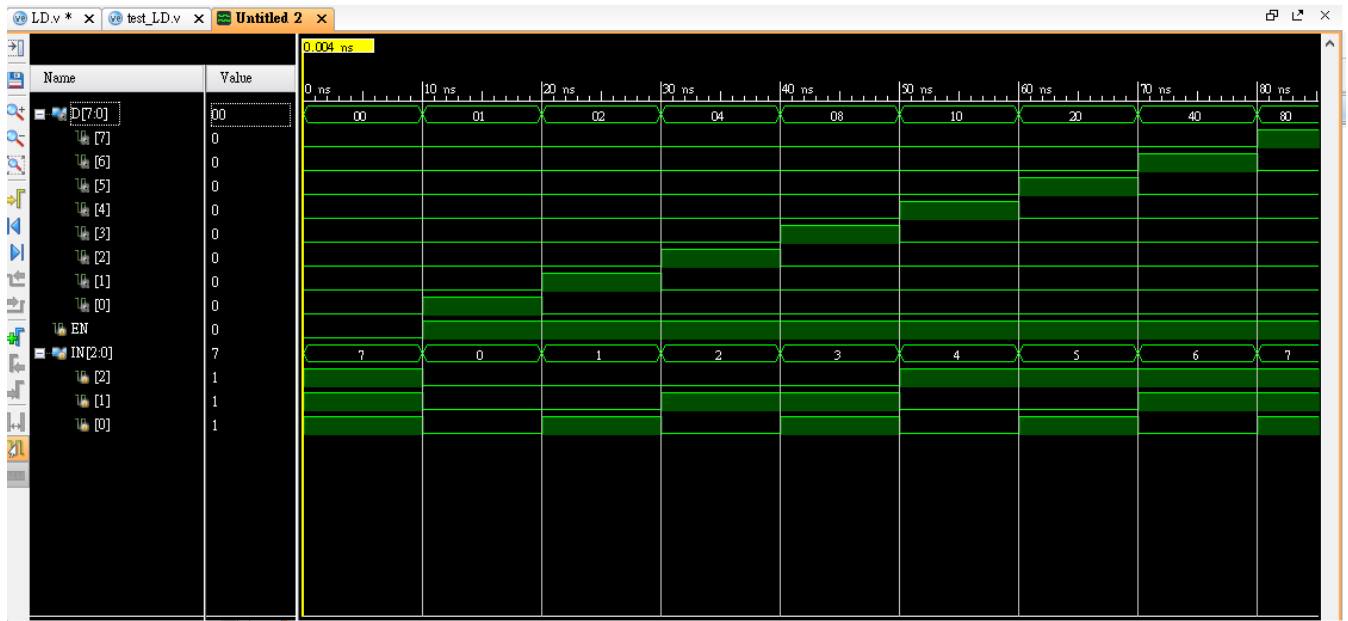
實驗一跑了好幾次波形都是 Z 跟 X，最後幸好有熱心助人的李佳璋同學幫忙 debug 才發現問題在 testbench 跟 module 的接線對不上，以及一些分號忘記打。波形圖如下：



實驗二的加法器：先把兩數相加，為了避免溢位把輸出設成五個數(即 code 中的 mS)。接著讓和經過一個大於比較器，若大於 10 就把 co 接成 4 個 bit 並與 -10(以 2 補數表示即 0110)逐位元過 and 閘，並與原數相加，就相當於減 10 的過程了。反之若 co 小於 10 就維持輸出不變。波形圖如下所示：



實驗三是很基本的組合邏輯複習，一個 decoder 就是各個最小項之和。接著把致能接到每個 and 閘上就可以了。上學期的講義內有 code。經模擬後波形圖如下：



Conclusion

初學 Verilog，我會建議與我做一些相同的事情：

1. 買一本中文書來看:因為 Verilog 不像 C 語言是一種程式語言，而是電路描述語言，而且老師上課所教的不一定能包含到實作時種種問題。所以我會建議買一本中文的書來看。目前品質好的中文書為數不少，而且相較原文的便宜。
2. 與同學或其他人討論: 不會的地方就問助教或同學，那怕真正的 bug 只是很細節的東西 (像 port 名字打錯或忘記加分號)。尤其學長姐給的家傳的 code 一定要讀懂，理解他人的程式碼對練習或創意的發想很有幫助。

如果你很幸運認識就讀高職電子電機科的朋友,問爆他們就對了。

References

《Verilog 硬體描述語言數位電路》，鄭信源，儒林圖書公司(2016)。