

電 腦 輔 助 VLSI 設 計

Home work 1

405402133 林亞辰

(一) 程式碼說明：

hw1.vhd | hw1.vwf | Compilation Report - Flow Summary | Simulation Report - Simulation Waveforms

```

1 Library ieee;
2 use ieee.std_logic_1164.all;
3 use ieee.numeric_bit.all;
4
5 entity hwl is
6     port(
7         A, B, C : in unsigned(3 downto 0);
8         S : in bit_vector(1 downto 0);
9         Y : out unsigned(3 downto 0)
10    );
11 end hwl;
12
13 architecture My_ckt_1 of hwl is
14 signal X : unsigned(7 downto 0); // 此變數用來協助作乘法運算。
15 begin
16     process(A, B, C, S, X) // 這裡事後想想好像有點多此一舉，不過因為不影響結果最後就沒刪了。
17     begin
18         X <= "00000000"; // 設定初值，因為若不設定的話，雖不影响結果，卻會有 Warning 警告要設初值。
19         Y <= "0000";
20         case S is
21             when "00" => // 這裡使用 "case" 和 "when" 的語法來代替，一方面因版面方便讓 code 区分情況，一方面避免用字
22                 Y <= A OR B; // 量畫面而破壞清晰
23             when "01" => // 剛好支援 OR Instruction，直接用
24                 if A > B then
25                     if A > C then
26                         Y <= A;
27                     else
28                         Y <= C;

```

光建立 Code 中會用到的 Library；因為並非會有 "unsigned" type，要加入 numeric_bit library。

宣告並非。第一個要注意明顯是 "hwl" entity 要與此端名相同，因為 compiler 執行是依此尋找起始執行處的。

因為變數 A, B, C 在往後的 code 中需要作運算，所以設為 unsigned type 而非 std_logic_vector。而 S 只用作判斷 operation，因此用 bit_vector 即可。

這裡事後想想好像有點多此一舉，不過因為不影響結果最後就沒刪了。

這段初值，因為若不設定的話，雖不影响結果，卻會有 Warning 警告要設初值。

Y 其實沒必要設初值，但為了測試用，所以預設為 0。

這裡使用 "case" 和 "when" 的語法來代替，一方面因版面方便讓 code 区分情況，一方面避免用字量畫面而破壞清晰。

剛好支援 OR Instruction，直接用

hw1.vhd* | hw1.vwf* | Compilation Report - Flow Summary | Simulation Report - Simulation Waveforms

```

28         Y <= C;
29     end if;
30     else
31         if B > C then
32             Y <= B;
33         else
34             Y <= C;
35         end if;
36     end if;
37     when "10" =>
38         X <= A * B; // 相乘的部分原本計劃用 for 循環一個 bit 一個 bit 乘，但想到若 overflow
39         for i in 3 downto 0 loop // 還要用浮點轉換，不如直接要乘一個 8-bit amy 及正並擴大級相乘，簡短一點比較明瞭。最後再用 for 循環取出後 4 bits 在入 Y 中。
40             Y(i) <= X(i);
41         end loop;
42     when others =>
43         Y <= A + C;
44     end case;
45 end process;
46 end My_ckt_1; // 別忘了最後都要記得寫 "end"。
47
48
49
50
51
52
53
54

```

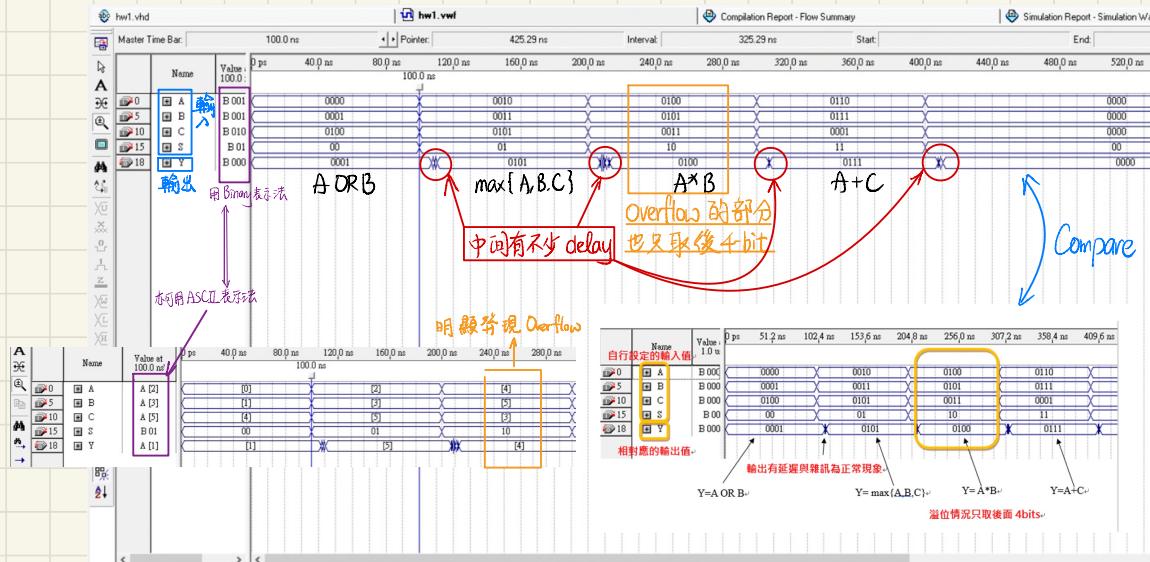
// 指 Maximum 的 "01" case 中，利用 "if, else" 逆反覆判斷尋找最大值。

原本上調查有無更直接的指令，但似乎沒有，因此用 Nested if 判斷。

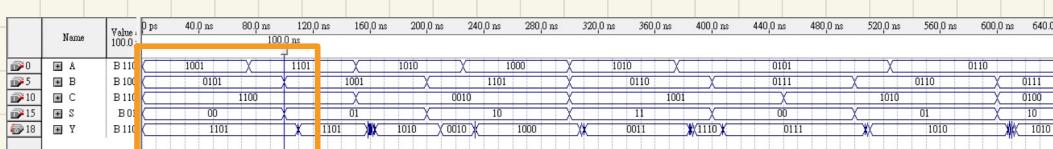
// 相乘的部分原本計劃用 for 循環一個 bit 一個 bit 乘，但想到若 overflow 還要用浮點轉換，不如直接要乘一個 8-bit amy 及正並擴大級相乘，簡短一點比較明瞭。最後再用 for 循環取出後 4 bits 在入 Y 中。

// 別忘了最後都要記得寫 "end"。

(二) 模擬結果說明



⇒ 執行輸出之圖形依據 Reference output 相同，只是我的程式碼中雜訊較多，推估是在作判斷的時候花了較多時間因此 delay 較多。



⇒ 將 A 以每 75 ns Random, B 以每 100 ns Random, C 以每 150 ns Random 後，輸出結果經驗算皆正確。在方框的 100 ns 中可發現，雖然 A 值改變，但因 A OR B 結果不變，因此丫輸出時長為 $100 + \Delta(\text{delay}) \approx 100$ ns，且沒有中間的雜訊，因此推估訊號雜訊和 Context switch 有關。

(三) 結論及心得：

- 難題：
① 撰寫 Code 時，最大的難題在於起初的設立 library 與變數。
雖然已由 Hint 指出用 unsigned type 來設變數，但卻誤用成 std_logic_1164 library，導致 error，後來再次查看上課講議才得知 unsigned type 要用 numeric_bit library，也成功測試出結果。
- ② 在程式碼解析有提到 A^*B 的部分，因該考量取 X 為 8-bit 變數的話會比直接大數相乘要浪費 Memory space，但又因複雜度之時間 delay 考量，決定直接以空間換取時間比較優良。
- ③ 其實說最最最難在於不熟悉軟體，因為程式碼早就完成了卻不會操作 Compilation 与 Simulation，反覆上網找資料，回頭看 Chapter I 的 pdf 講議，甚至三番兩次的尋問助教（在此再次感謝助教的協助），最後圓跑出來且無誤時，著實感動。

心得：寫此 Homework 的途中反覆回憶上課所教的程式碼，一再的思考推敲各 Code 之間的差異，尋找適合此 Homework 的 Answer，過程十分有趣，包含思考大數乘法、想用 and、or、xor 來解決問題等等。
當然也學到了 VLSI 基本的設計方式，實際操作有趣許多。

-End

P.S. 在此附上不知道該如何解決的 Warning 8

The screenshot shows a VHDL editor interface with three tabs: 'hw1.vhd', 'hw1.wlf', and 'Compilation Report - Flow'. The 'hw1.vhd' tab contains the following VHDL code:

```
1 Library ieee;
2 use ieee.std_logic_1164.all;
3 use ieee.numeric_bit.all;
4
5 entity hw1 is
6   port(
7     A, B, C : in unsigned( 3 downto 0 );
8     S : in bit_vector( 1 downto 0 );
9     Y : out unsigned( 3 downto 0 )
10    );
11 end hw1;
12
13 architecture My_ckt_1 of hw1 is
14   signal X : unsigned( 7 downto 0 );
15 begin
16   process( A, B, C, S, X )
17   begin
18     X <= "00000000";
19     ...
20   end;
21 end architecture;
```

The 'Compilation Report - Flow' tab shows a warning window with the following details:

Type	Message
⚠	Warning: Feature LogicLock is not available with your current license
⚠	Warning: No exact pin location assignment(s) for 18 pins of 18 total pins
⚠	Warning: Some pins have incomplete I/O assignments. Refer to the I/O Assignment Warnings report for details
⚠	Warning: Found 4 output pins without output pin load capacitance assignment
⚠	Warning: The Reserve All Unused Pins setting has not been specified, and will default to 'As output driving ground'.