國立高雄科技大學

電子工程系(第一校區)

硬體描述語言

Lab 2-2

指導教授：陳銘志

班　　級：電子三甲

學生姓名：蕭詠釗

學　　號：C111112132

Lab 2-2.

1. **根據上周完成的 4-bit Ripple Carry Full Adder，接續完成這周的課堂作業。**

   要求：

   1. **在 1-bit Full Adder 內，其 XOR 的 gate delay=5 ; AND 的 gate delay=3。跑出波形並驗證其結果是否正確。**

      (Hint : 也就是 rise, fall, turn-off delay 皆相同於 gate delay)

   2. **其 4-bit Ripple Carry Full Adder 計算 a=1011, b=1101，產生結果輸出，需經過多少時間？**

----------第一題作答區----------

主(電路)程式

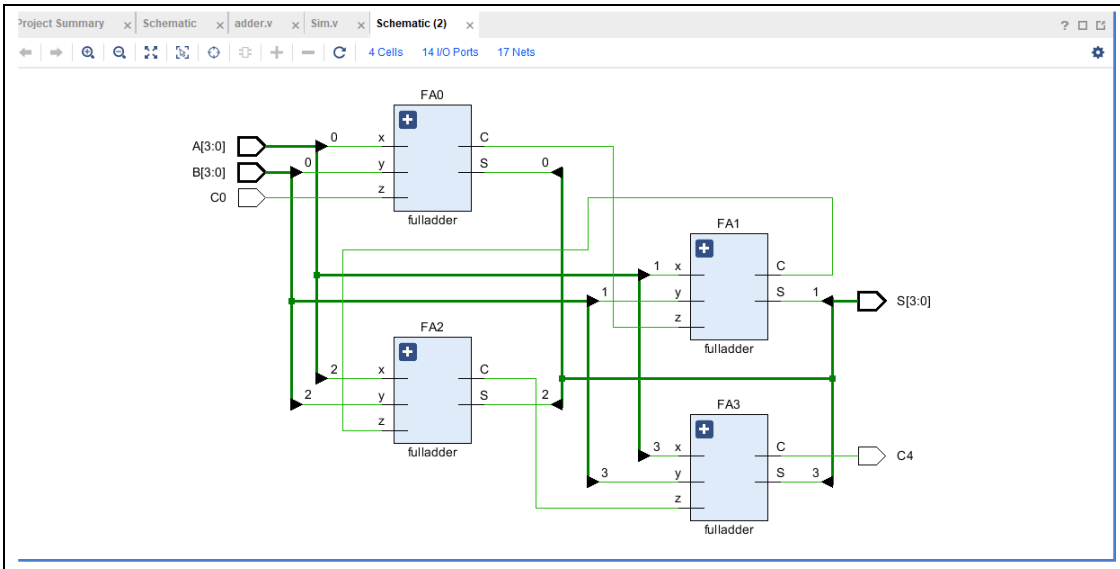
```
`timescale 1ns / 1ps



module fulladd(sum, c_out, a, b, c_in);
// I/O port declarations
output sum, c_out;
input a, b, c_in;
// Internal nets
wire s1, c1, s2;
// Instantiate logic gate primitives
xor #5 (s1, a, b);
and #3 (c1, a, b);
xor #5 (sum, s1, c_in);
and #3 (s2, s1, c_in);
or (c_out, s2, c1);
endmodule



module fulladd4(sum, c_out, a, b, c_in);
// I/O port declarations
output [3:0] sum;
output c_out;
input[3:0] a, b;
input c_in;
// Internal nets
wire c1, c2, c3;
```

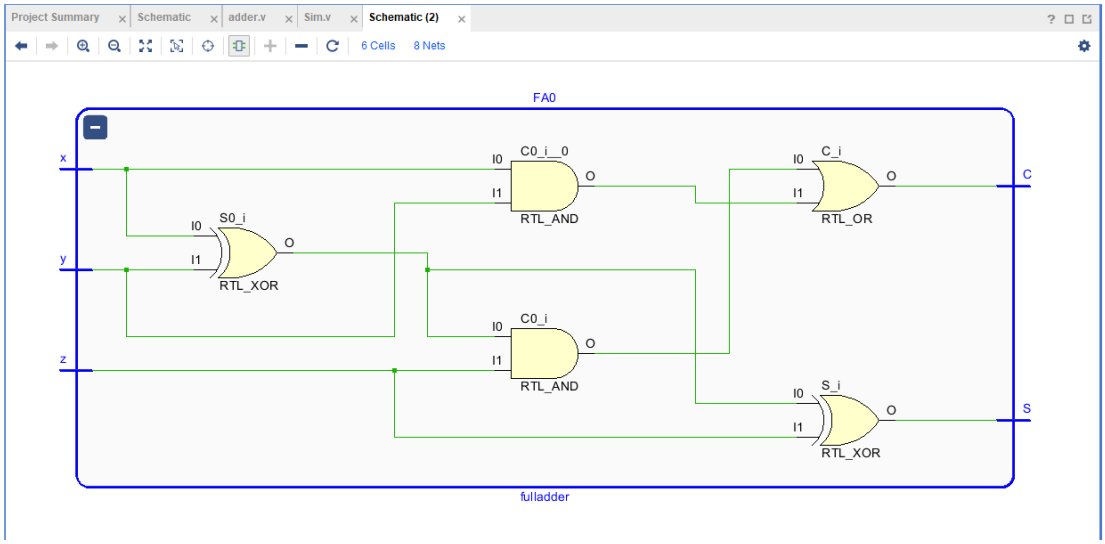
```
// Instantiate four 1-bit full adders.
fulladd fa0(sum[0], c1, a[0], b[0], c_in);
fulladd fa1(sum[1], c2, a[1], b[1], c1);
fulladd fa2(sum[2], c3, a[2], b[2], c2);
fulladd fa3(sum[3], c_out, a[3], b[3], c3);
endmodule
```

測試程式(testbench)

```
`timescale 1ns / 1ps
module stimulus;
// Set up variables
reg [3:0] a, b;
reg c_in;
wire [3:0] sum;
wire c_out;
// Instantiate the 4-bit full adder. call it FA1_4
fulladd4 FA1_4(sum, c_out, a, b, c_in);
initial
begin
$monitor($time," A= %b, B=%b, C_IN= %b, --- C_OUT= %b, SUM= %b\n",
a, b, c_in, c_out, sum);
end
// Stimulate inputs
initial
begin

  a = 4'b1011; b = 4'b1101;c_in=1'b0;



end
endmodule
```

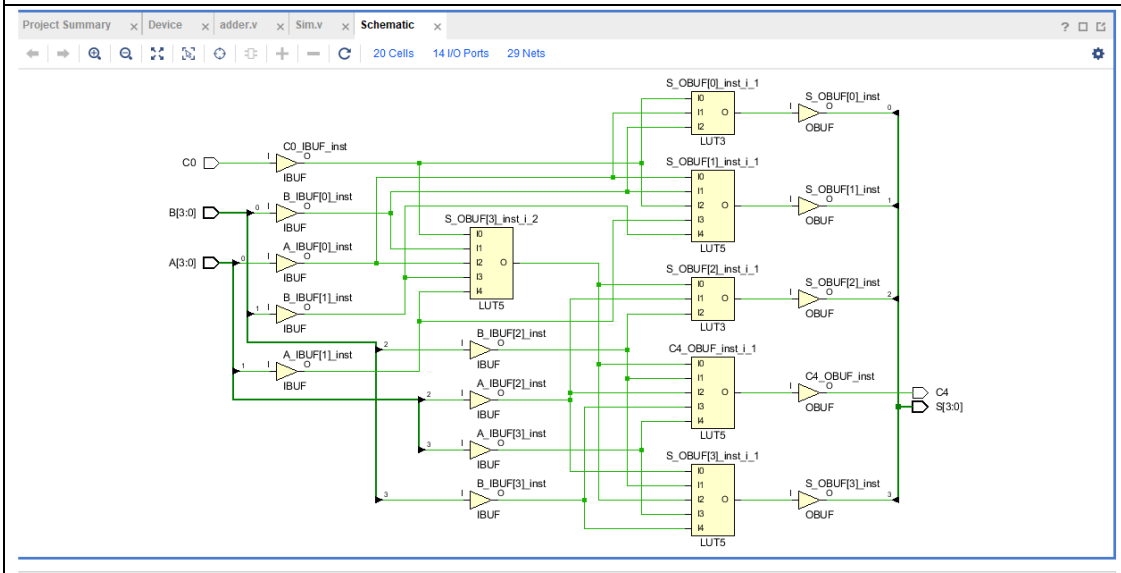
RTL schematic





示意圖請一層一列並標示是第幾層(第三層)

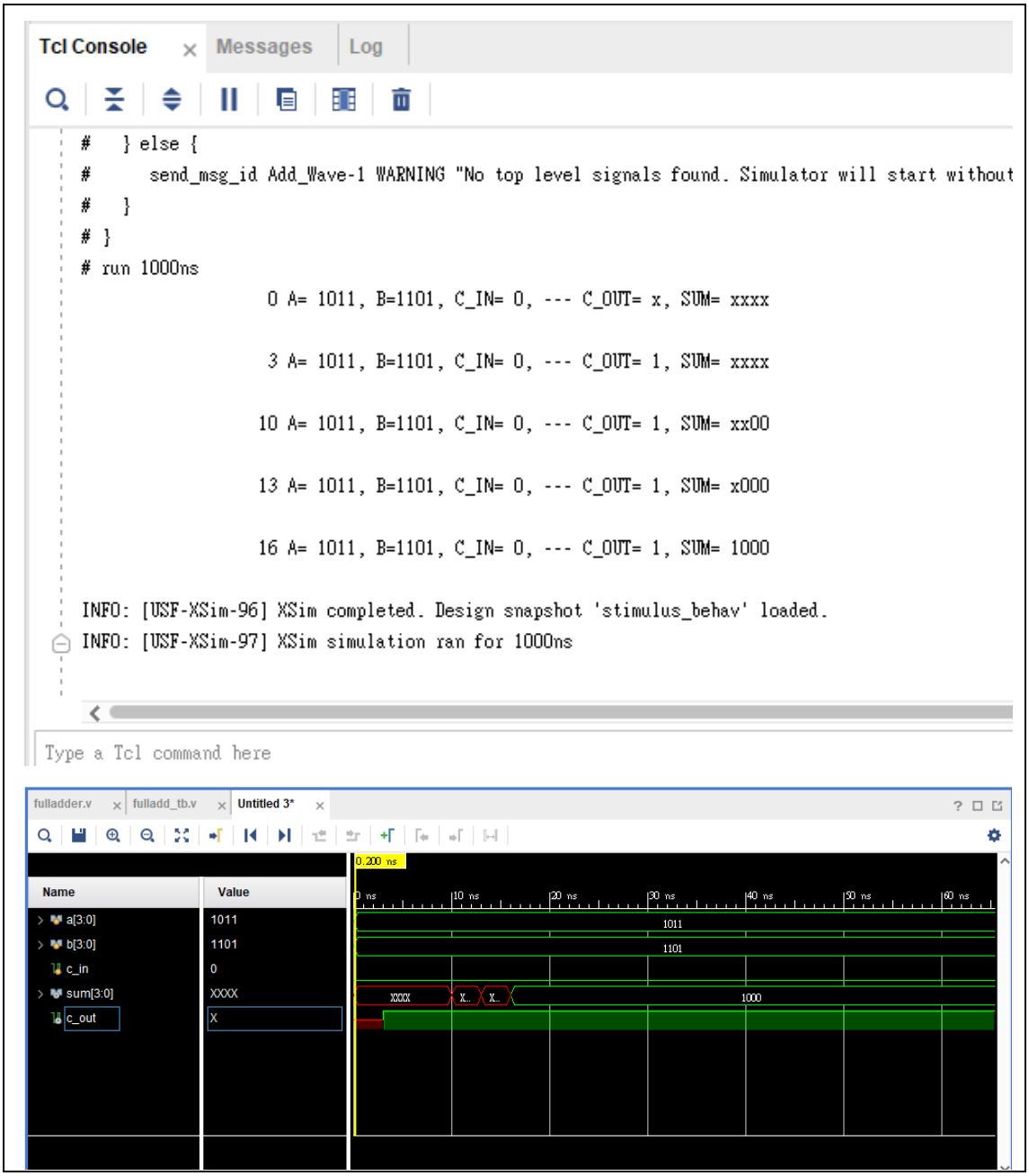
Technology Schematic



示意圖請一層一列並標示是第幾層(第二層)

示意圖請一層一列並標示是第幾層(第三層)

Behavioral Simulation Waveform(含 $monitor)



----------作答區結束----------

(作業請於兩週內交，遲交零分計算)