國立中興大學　110 學年度 第 2 學期　期末考試命題紙

授課教師	范志鵬　先生 隸屬電機系（所）	科目資料	名稱 FPGA 實驗課 科目代碼：	考試時間	中華民國　111　年 6　月　15　日 下午 3:10 時起至 5:00 共 110 分鐘	修習人數	

1. (20%) Let us define that X=4'b1100 , Y=4'b1010, W=4'b010x, A=4'b1011, write the following answers ?

(a). | A **(b).** X || Y **(c).** X | Y **(d).** X >>> 1 **(e).** Y <<< 1

(f). Y = = = W **(g).** ~Y **(h).** {2{A}, 2'b100, 3{Y[1]}, 2{A[0]} } **(i).** X & Y **(j).** X && Y

2. (20%) Some Verilog HDL constructs are listed as follows,

(a) #12 **(b) bufif0** **(c) ? : ** **(d) task** **(e) fork join**

(f) $display **(g) initial** **(h) >>** **(i) case** **(j) { }**

(2-1) Which are the logic synthesizable codes (可邏輯合成) ?

(2-2) Which are not the logic synthesizable codes (不可邏輯合成) ?

3. (5%) The following Verilog codes have one major coding error. How to correct the codes ?

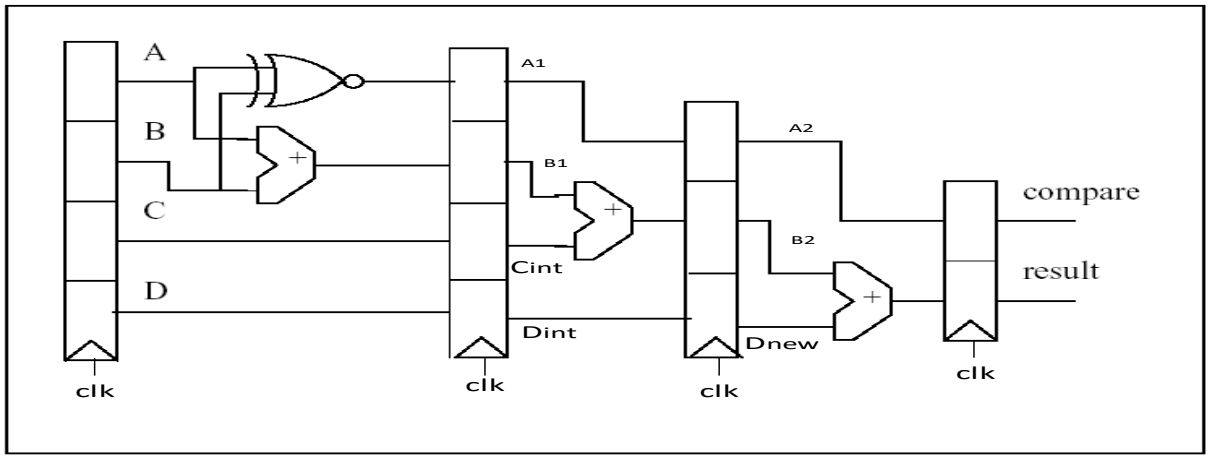
```
always @(x or i1)
  if(x==0)
  out=i1;

always @(y or i2)
  if(y==1)
  out=i2;
```

4. (5%) The following combinational logic has modeling problems. How to correct the codes ?

```
always @( a or b or w)
        w <= x & ( y | z );
```

5. (10%) Write the Verilog codes of the following pipelined circuit.



6. **(10%)** Please describe and plot the FPGA-based design flow.

****請注意** ： 背面還有題目**

7. (10%) Write the correct order (正確操作次序) with (a)(b)(c)(d) for Xilinx FPGA design flow.

(a) Implementation (Place & Route)　　(b) Generate the bit file, and download it to the chip

(c) Logic Synthesis　　(d) Write Verilog HDL codes

8. (10%) Verilog Coding Styles - Please answer the following descriptions with "Yes" or "No":

(a) We can mix level and edge sensitive in one always block.

(b) We can mix positive and negative edge-triggered flip-flops.

(c) Use continuous assign statements for simple combinational logic.

(d) Use nonblocking assignment for sequential logic.

(e) We cannot mix blocking and nonblocking assignments in one always block.

9.　(a) (5%) 實驗課的 FPGA 平台有如下方的跑馬燈 Verilog codes :

```
always @ (posedge clk)
    begin
      if (rst)
        out_r <= 12'b1111_1111_1110;
      else if (COUNT==4'b0001)
        out_r <= {out_r[0], out_r[11:1]};
      else
        out_r<=out_r;
    end
```

其中暫存器 out_r 控制 12 顆 LED 燈的亮暗, **假設邏輯 1(高準位) 會讓 LED 燈亮起來;**
當 COUNT 為 4'b0001 時, 此時 12 顆 LED 燈的狀態為:

(a) 暗燈向左跑
(b) 暗燈向右跑
(c) 亮燈向左跑
(d) 亮燈向右跑
(e) 全亮
(f) 全暗
(h)　以上皆非

(b) (5%) 若要兩個 LED **亮燈**同時由:

- **第 6 顆 LED 向左跑,**
- **第 5 顆 LED 向右跑,**

程式碼要如何修改 ?

LED 位置 與 out_r 的對應關係如下 :

LED 位置	11	10	9	8	7	6	5	4	3	2	1	0
out_r	11	10	9	8	7	6	5	4	3	2	1	0