

安徽大学 20 21—20 22 学年第 2 学期

《 硬件描述语言与 FPGA 应用 》 考试试卷（期中卷）  
（闭卷 满分 100 分 时间 120 分钟）

考场登记表序号\_\_\_\_\_

一、单项选择题（每小题 2 分，共 20 分）

- 下列不属于 Verilog 行为级描述的是（ ）。  
A. 算法级                      B. RTL 级                      C. 系统级                      D. 门级
- 与电路原理图输入法相比，采用 Verilog HDL 设计复杂数字电路的特点不包括（ ）。  
A. 与工艺无关                      B. 周期长  
C. 容易移植                      D. 容易修改
- 下列关于模块的说法错误的是（ ）。  
A. 模块是可以进行层次嵌套的                      B. 每个语句和数据定义的最后必须有分号  
C. 模块可以是不可综合的                      D. 模块的内容位于 module 和 endmodule 之间
- reg 型数据的默认初始值为（ ）。  
A. 0                      B. 1                      C. x                      D. z
- 下列程序中，必须被定义为 reg 型的是（ ）。  
module example(o1, o2, a, b, c, d);  
input a, b, c, d;  
output o1, o2;  
    and u1(o2, c, d);  
    always @(a or b)  
        if (a) o1 = b; else o1 = 0;  
endmodule  
A. o1                      B. o2                      C. a 和 b                      D. c 和 d
- 若  $rega = 4'b1010$ ,  $regb = 4'b11x0$ , 则  $rega | regb$  的值为（ ）。  
A. 1                      B. 0                      C.  $11x0$                       D. 1110
- 若  $rega = 4'b1010$ ,  $regb = 4'b1x10$ , 则  $rega === regb$  的值为（ ）。  
A. 1                      B. 0                      C. x                      D. z

8. 下列程序运行后, regb 的值为 ( )。

```
module shift ();  
reg [7: 0] rega, regb;  
initial rega = 8'b00001100;  
    initial fork  
        regb <= rega << 5 ;  
    join  
endmodule
```

A. 11000000      B. 00000000      C. 10000000      D. 10000001

9. 下列关于非阻塞赋值 (如  $b \leq a$ ), 错误的是 ( )。

A. 赋值执行完成后再执行在顺序块内下一条语句      B. 块结束后才完成赋值操作  
C. b 的值并不是立刻就改变的      D. 非阻塞赋值不阻塞过程流

10. 四种循环语句中可以综合的是 ( )。

A. repeat      B. while      C. for      D. forever

## 二、判断题 (每小题 2 分, 共 10 分)

11. FPGA 因为本身的限制, 无法达到太高的系统性能。 ( )
12. 在 Verilog 模块中所有过程块、连续赋值语句、实例引用都是并行的。 ( )
13. fork join 语句的延迟时间是相互累加的结果。 ( )
14. 在 always 块内被赋值的信号可以定义为 wire 型, 也可以被定义为 reg 型。 ( )
15. 在 Verilog 中重复说明 case 项是合法的。 ( )

## 三、简答题 (每小题 5 分, 共 20 分)

16. 谈谈你对 FPGA 与 ASIC 优缺点的理解。
17. 简述什么是综合。
18. 简述 HDL 的设计流程。
19. 简述自顶向下设计的概念。

## 四、程序改错题 (每小题 2 分, 共 10 分) (任意答对 5 个或以上得 10 分)

```
Line1:    module mux4_to_1(out, i0, i1, i2, i3, s1, s0)  
Line2:    output out;
```

Line3:       output i0, i1, i2, i3;  
Line4:       input s1,   s0;  
Line5:       wire out;  
Line6:           always@(s1 or s0 or i0 or i1 or i2 or i3);  
Line7:       begin   case{s1, s0}  
Line8:                2'b00: out = i0;  
Line9:                2'b01: out = i1;  
Line10:               2'b10: out = i2;  
Line11:               2'b11: out = i3;  
Line12:               default: out = 4'b\_1111;  
Line13:               end  
Line14:       end  
Line15:

**五、程序填空题（第 25-29 题每小题 2 分，第 30 题 5 分，共 15 分）**

//下列程序将对两个32位总线变量进行按位异或。

//生成的32个异或门层次名为xor\_loop[0]. g1到xor\_loop[31]. g1。

module bitwise\_xor(out , i0 , i1);

    (25)   N= (26) ;

    (27) [N-1 : 0]   out;

    input [N-1 : 0]   i0, i1;

    (28)   j;

        (29)

            (30)

        begin: xor\_loop

            xor g1 (out[j], i0[j], i1[j]);

        end

endgenerate

**六、程序设计题（第31题10分，第32题15分，共25分）**

31. 使用条件语句（if\_else）设计一个四选一多路选择器mux4\_to\_1，输出端口为out，数据输入端口为i0，i1，i2，i3，地址端为s1，s0。

32. 设计一个八位乘法器。