

中国科学技术大学计算机学院
《数字电路实验》报告



实验题目：Verilog 硬件描述语言

学生姓名：陈鸿绪

学生学号：PB21000224

完成日期：10.28

计算机实验教学中心制

2020 年 09 月

【实验题目】

Verilog 硬件描述语言

【实验目的】

掌握 Verilog HDL 常用语法；

熟练阅读并理解 Verilog 代码；

可以设计较复杂的数字功能电路；

能够将 Verilog 代码与实际硬件相对应；

【实验环境】

vlab.ustc.edu.cn

verilog.ustc.edu.cn

【实验练习】

T1. 阅读以下 Verilog 代码，找出其语法错误，并进行修改

```
module test(  
    input a,  
    output b);  
    if(a) b = 1'b0;  
    else b = 1'b1;  
endmodule
```

[解]：if else 语句一般出现在 always 语句的过程语句部分，而不能直接在模块内部单独出现。且在 always 语句中的被赋值变量一般是 reg 变量，不可以是 wire 型变量。修改后如下：

```
module test(  
    input a,  
    output reg b);  
    always@(*)  
    begin  
        if(a) b = 1'b0;  
        else b = 1'b1;  
    end  
endmodule
```

T2. 阅读以下 Verilog 代码，将空白部分补充完整

```
module test(  
    input [4:0] a,  
    _____);  
    always@(*)  
        b = a;  
    _____
```

[解]：output 中的 b 由于在 always 语句中被赋值，所以 b 应该

是 reg 类型变量，且由于 a 有 5 bits, 故 b 也应该有 5 bits。

代码如下所示：

```
module test(  
    input [4:0] a,  
    output reg [4:0] b);  
    always@(*)  
        b = a;  
endmodule
```

T3. 阅读以下 Verilog 代码, 写出当 $a = 8'b0011_0011$, $b = 8'b1111_0000$ 时各输出信号的值。

```
module test(  
    input [7:0] a, b,  
    output [7:0] c, d, e, f, g, h, i, j, k );  
    assign c = a & b;  
    assign d = a | b;  
    assign e = a ^ b;  
    assign f = ~a;  
    assign g = {a[3:0], b[3:0]};  
    assign h = a >> 3;  
    assign i = &b;  
    assign j = (a > b) ? a : b;
```

```
    assign k = a - b;

endmodule
```

[解]: 当 $a = 8'b0011_0011$, $b = 8'b1111_0000$ 时

1. $c = a \& b = 8'b0011_0000$
2. $d = a \mid b = 8'b1111_0011$
3. $e = a \wedge b = 8'b1100_0011$
4. $f = \sim a = 8'b1100_1100$
5. $g = \{a[3:0], b[3:0]\} = 8'b0011_0000$
6. $h = a \gg 3 = 8'b0000_0110$;
7. $i = \&b = 8'b'0000_0000$;

注: 事实上这一道题是有问题的, i 有 8 位, 而赋值的数只有 1 位, i 得到的值是高阻态更加合理。

8. $j = (a > b) ? a : b = 8'b1111_0000$;
9. $k = a - b = 8'b0100_0011$;

题目 4. 阅读以下 Verilog 代码, 找出代码中的语法错误, 并修改

```
module sub_test(

input a,b,

output reg c);

    assign c = (a<b)? a : b;

endmodule
```

```

module test(
    input a,b,c,
    output o);

    reg temp;

    sub_test(.a(a),.b(b),temp);

    sub_test(temp,c,.c(o));

endmodule

```

[解]：assign 中被赋值的数据类型应该是 wire 型，故 c 为 wire 类型，故 temp 也为 wire 类型；两种模块调用方式不可以混用；模块调用需要有模块名；故修改后的 Verilog 代码如下：

```

module sub_test(
    input a,b,
    output c);

    assign c = (a<b)? a : b;

endmodule

module test(
    input a,b,c,
    output o);

    wire temp;

    sub_test sub_test1(.a(a),.b(b),.c(temp));

    sub_test sub_test2(temp,c,o);

```

```
endmodul
```

T5. 阅读以下 Verilog 代码，找出其中的语法错误，说明错误原因，并进行修改。

```
module sub_test(  
    input a,b);  
    output o;  
        assign o = a + b;  
endmodule  
  
module test(  
    input a,b,  
    output c);  
    always@(*)  
        begin  
            sub_test sub_test(a,b,c);  
        end  
endmodule
```

[解]：1. output 参数与 input 参数应该在同一个括号内

2. 调用的模块一般不可以放在 always 语句当中

修改后代码如下：

```
module sub_test(  
    input a,b,  
    output o);  
        assign o = a + b;  
endmodule
```

```
input a,b  
output o);  
    assign o = a + b;  
endmodule  
  
module test(  
input a,b,  
output c);  
    sub_test sub_test(a,b,c);  
endmodule
```

【总结与思考】

1. 我掌握 Verilog HDL 常用语法;并且可以熟练阅读并理解 Verilog 代码;同时学会了设计较复杂的数字功能电路;能够将 Verilog 代码与实际硬件相对应;
2. 难易程度: 适中
3. 任务量: 偏小
4. 改进建议: 有让同学编写 Verilog 代码的题目。