

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：叶子昂

学生学号：PB20020586

完成时间：2021年11月14日

实验题目

verilog硬件描述语言

实验目的

- 掌握Verilog HDL的常用语法
- 能够熟练阅读并理解Verilog代码
- 能够设计较为复杂的数字功能电路
- 能够将Verilog代码与实际硬件相对应
- 能够对Verilog代码有基本的纠错能力
- 学会硬件编程的思想与软件编程有所区分

实验环境

- 装有windows，能连校园网的笔记本电脑
- 本地的Logisim，和Vivado软件
- 配置好环境的VS Code用以编写verilog代码
- vlab.ustc.edu.cn综合平台

实验练习

题目一

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  
*/
```

- 阅读代码有如下错误：在非always块中使用if-else
- 如要使用if-else需使用always，与此同时output b应改为output reg b
- 修改代码如下：

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

题目二

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;  
_____  
*/
```

- 阅读代码首先发现缺少endmodule，再由b=a可知b与a位宽匹配故有output reg [4:0] b
- 代码如下：

```
module test(  
    input [7:0] a,  
    output reg [7:0] b  
);  
    always@(*)  
    begin  
        b=a;  
    end  
endmodule
```

题目三

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  
*/
```

- 根据各运算符含义得到各结果为
c=8b'0011_0000
d=8b'1111_0011
e=8b'1100_0011
f=8b'1100_1100
g=8b'0011_0000
h=8b'0000_0110
i=8b'0000_0000

```
j=8'b1111_0000
k=8'b0100_0011
```

题目四

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
*/
```

- 阅读代码发现如下错误
 - sub_test中assign的左值为reg类型
 - test中实例化sub_test使用reg类型的temp
 - 模块实例化方法不统一，基于端口以及基于位置的实例化混用。
- 修正代码如下：

```
module sub_test (
    input a,b,
    output c
);
    assign c=(a<b)?a:b;
endmodule

module top_module (
    input a,b,c,
    output o
);
    wire temp;
    sub_test test1(.a(a),.b(b),.c(temp));
    sub_test test2(temp,c,o);
endmodule
```

题目五

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
*/

```

- 阅读代码发现有如下错误
 - sub_test中output o未在端口处声明
 - 实例化不可以在always中进行。
- 修改代码如下：

```

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

module top_module (
    input a,b,
    output c
);
    sub_test test(a,b,c);
endmodule

```

总结与思考

- 本次实验的收获
 - 通过本次实验我了解更多的如Verilog关键字，代码结构，数据类型以及运算符。
 - 能够流畅阅读Verilog代码并分析其作用
 - 能够将Verilog中的代码与实际电路相对应，清楚代码综合形成的电路
 - 对Verilog代码有基本的纠错能力得到锻炼，能快速发现错误代码中的基本语法问题
- 本次实验的难易程度
 - 有关Verilog语法的内容较多，但本次练习题相对单一总体较为简单。
- 本次实验的任务量
 - 本次题目较少且相对简单任务量较少。
- 对本次实验的建议
 - 本次实验练习题多为改错，希望可以多一些设计方面的题目。