

浙江大学实验报告

课程名称: 数字系统 任课老师: 李宇波

实验名称: Xilinx Development Software 实验日期: 2023/3/24

1 实验目的和要求

1.1 实验目的

To learn how to use the Xilinx circuit design software to design and implement digital circuits.

学会如何使用 Xilinx 电路设计软件去设计和实现数字电路。

1.2 实验要求

a. Design a 2-1 Mux with the Xilinx software. Connect the input a, b, s to three switches. Connect the y output to an LED. Implement the circuit onto the FPGA board to test and verify that it works correctly.

- 1) Design it with AND/OR/NOT gate level Verilog description.
- 2) Design it with Boolean function Verilog description.
- 3) Design it with Always statement and if-else statement.
- 4) Design it with always statement and case statement.

b. Design 4-1 Mux by using hierarchy structure of the module 2-1 Mux designed above.

a. 使用 Xilinx 的软件设计一个 2-1 Mux。用三个开关分别连接输入 a, b, s; 选择一个 LED 连接输出 y。在 FPGA 板实现电路, 并进行测试验证。

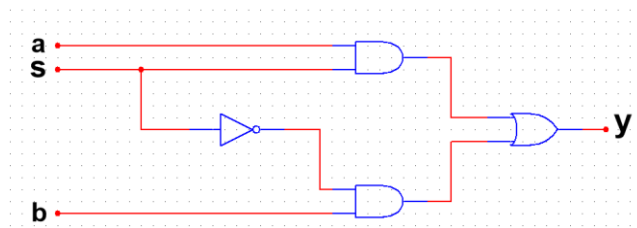
- 1) 使用与或非门级 Verilog 描述
- 2) 使用布尔函数 Verilog 描述
- 3) 使用 always 和 if-else 表达式
- 4) 使用 always 和 case 表达式

b. 通过不同层次结构设计, 用 2-1 Mux 模块设计 4-1 Mux

2 实现代码

2.1 2-1 Mux

2.1.1 And/Or/Not gate Level



图一 2-1 Mux 电路图

```
module mux(
    input a,
    input b,
    input s,
    output y
);
    wire t0, t1, sbar;

    and(t0, a, s), (t1, b, sbar);
    not(sbar, s);
    or(y, t0, t1);

endmodule
```

2.1.2 Boolean function

```
module mux(
    input a,
    input b,
    input s,
    output y
);
    assign y = (s)? a:b;

endmodule
```

2.1.3 Always statement and if-else statement

```
module mux(
    input a,
    input b,
    input s,
    output reg y
);
    always @(*)
    begin
        if(s)
            y=a;
        else
            y=b;
    end
endmodule
```

2.1.4 always statement and case statement

```
module mux(  
    input a,  
    input b,  
    input s,  
    output reg y  
);  
always @(*)  
begin  
    case(s)  
        1'b0: y=a;  
        1'b1: y=b;  
    endcase  
end  
  
endmodule
```

2.2 4-1 Mux

```
module mux4to1(  
    input s0,  
    input s1,  
    input w0,  
    input w1,  
    input w2,  
    input w3,  
    input w4,  
    output y  
);  
wire y1,y2;  
mux mux1(.a(w0),.b(w1),.s(s0),.y(y1));  
mux mux2(.a(w2),.b(w3),.s(s0),.y(y2));  
mux mux3(.a(y1),.b(y2),.s(s1),.y(y));  
endmodule
```