**1 实验目的和要求**

**1.1 实验目的**

（分点简要说明本次实验需要进行的工作和最终的目的）

To learn how to write Verilog testbench to simulate the ALU designed in Lab 3. Simulate the ALU with the following functions.

1)Simulate the 3-1 mux module by writing a 3-1 mux testbench, then check whether the output is correct.

2)Simulate the subtract module by writing a subtract testbench, then check whether the output is correct.

3)Simulate the ALU module by writing an ALU testbench, then check whether all the function works correctly.

**1.1 实验要求**

（说明本次实验的要求与任务）

To learn how to write Verilog testbench to simulate the ALU designed in Lab 3. Simulate the ALU with the following functions.

1)Simulate the 3-1 mux module by writing a 3-1 mux testbench, then check whether the output is correct.

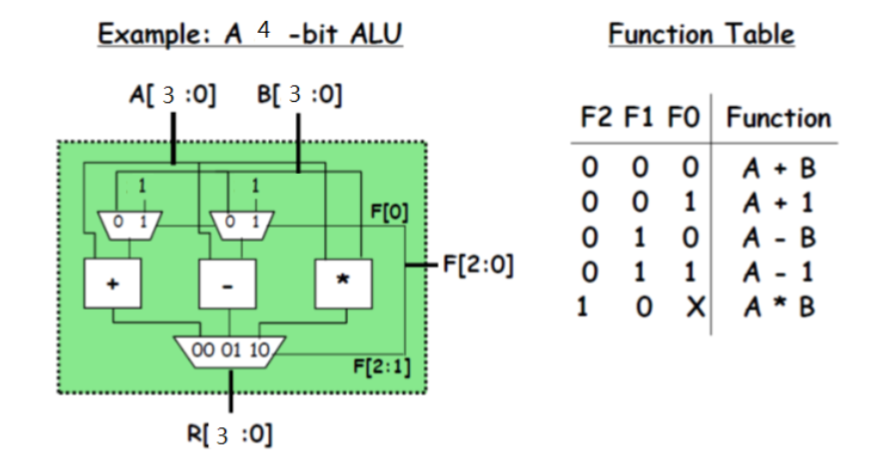
2)Simulate the subtract module by writing a subtract testbench, then check whether the output is correct.

3)Simulate the ALU module by writing an ALU testbench, then check whether all the function works correctly.

**2 实验原理**

（简要说明本次实验的理论，包括但不限于物理、数学或是算法方面的理论，电路原理图、算法框图等示意图也可以在此处给出）

HDL语言和算法原理

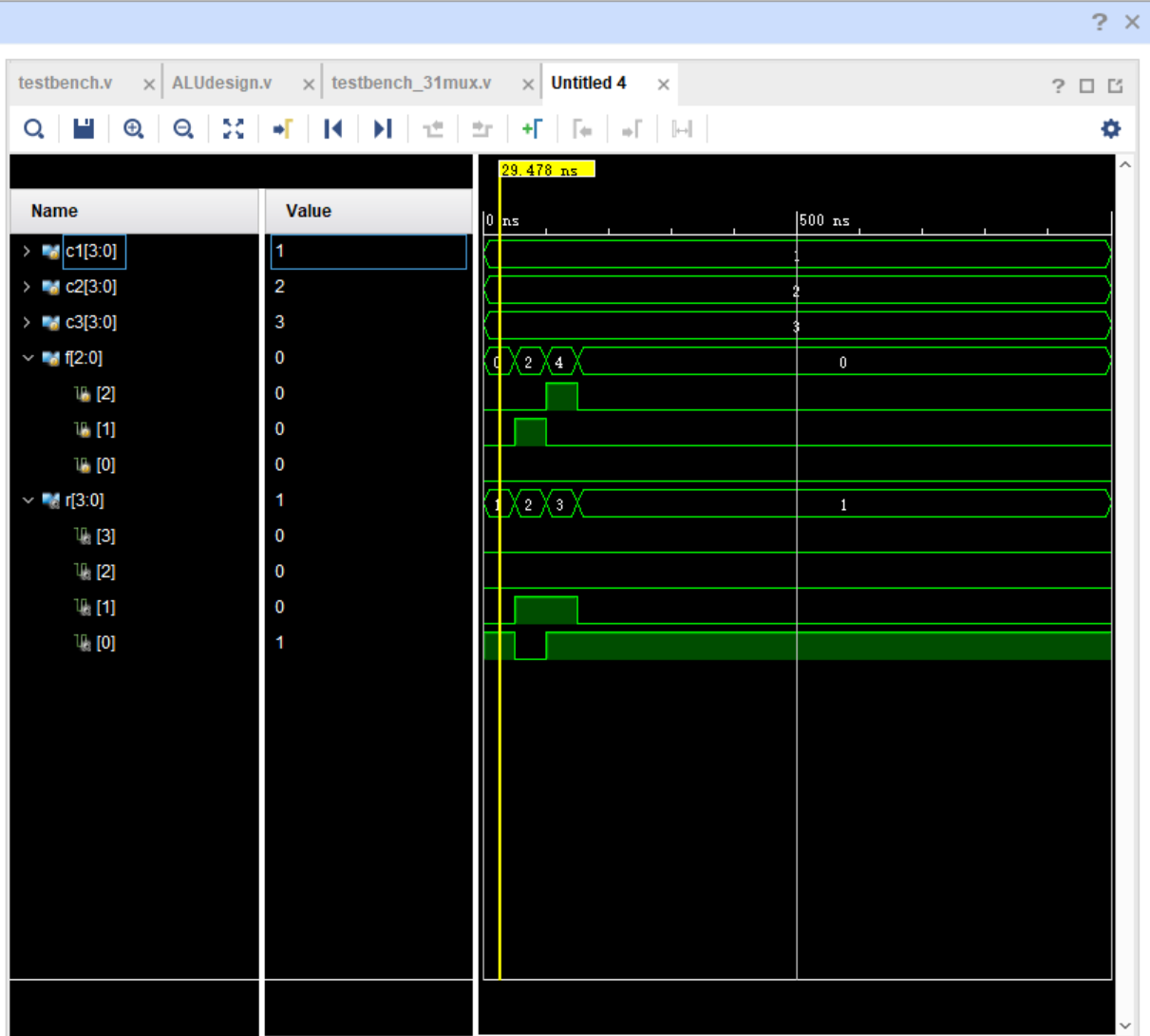


**3 实验内容**

（分点阐述实验步骤）

1. 使用vivado软件，打开lab2的工程项目
2. 分别编写4-bit 3-1 Mux, 4-bit Subtracter, and ALU模块的testbench代码。
3. 运行仿真，检测仿真结果是否正确。

**4 实验结果和分析**



图表 1 31mux仿真结果

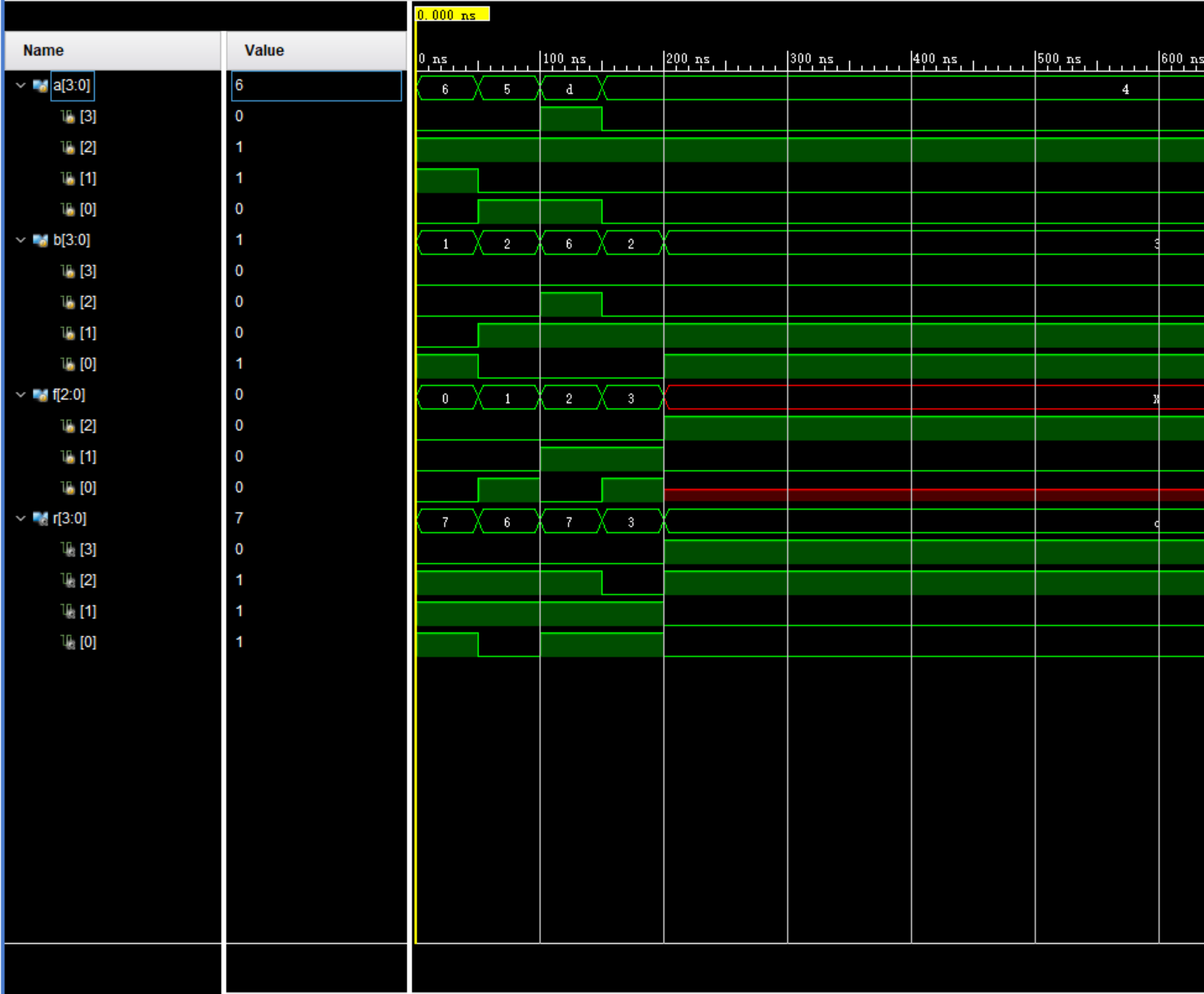
当f=000；010；100时，输出r分别选择为c1，c2，c3

可见仿真结果正确



图表 2 subtract仿真结果

分别测试了6个减法运算，仿真结果均正确



图表 3 ALU仿真结果

分别测试了6+1=7；5+1=6；d-6=7；4-1=3；4\*3=12五种运算，仿真结果均正确。

**5 实验结论**

（基于实验结果和分析，得出结论）

成功实现了所要求的31mux、subtract和ALU的仿真测试。

**6 源代码与分析**

module testbench\_31mux; //31选择器仿真代码

reg [3:0] c1,c2,c3;

reg [2:0]f;

wire [3:0]r;

mux31 m6(f,c1,c2,c3,r);

initial

begin

c1=4'b0001; //初始化信号，给三输入不同初值

c2=4'b0010;

c3=4'b0011;

f=3'b000;

#50;

c1=4'b0001;

c2=4'b0010;

c3=4'b0011;

f=3'b010; //改变输入f选择c2

#50;

c1=4'b0001;

c2=4'b0010;

c3=4'b0011;

f=3'b100; //改变输入f选择c3

#50;

c1=4'b0001;

c2=4'b0010;

c3=4'b0011;

f=3'b000; //改变输入f选择c1

end

endmodule

module testbench\_subtract; //减法器仿真代码

reg [3:0] a,b;

wire [3:0]y;

subtract\_4bit m3(a,b,y);

initial

begin

a=4'b0000; //0-1

b=4'b0001;

#50;

a=4'b1010; //a-5

b=4'b0101;

#50;

a=4'b1000; //8-d

b=4'b1101;

#50;

a=4'b0101; //5-b

b=4'b1011;

#50;

a=4'b0111; //7-9

b=4'b1001;

#50;

a=4'b0001; //1-f

b=4'b1111;

end

endmodule

module testbench\_ALU; //减法器仿真代码

reg [3:0] a,b;

reg [2:0]f;

wire [3:0]r;

ALUdesign m(f,a,b,r);

initial

begin

a=4'b0110; //a+b

b=4'b0001;

f=3'b000;

#50;

a=4'b0101; //a+1

b=4'b0010;

f=3'b001;

#50;

a=4'b1101; //a-b

b=4'b0110;

f=3'b010;

#50;

a=4'b0100; //a-1

b=4'b0010;

f=3'b011;

#50;

a=4'b0100; //a\*b

b=4'b0011;

f=3'b10x;

end

endmodule