

实验 1 熟悉 vivado 环境

班级：01 学号：2021k8009925006 姓名：冯浩瀚

一、实验目的

- 1.熟悉 Vivado 设计流程
- 2.掌握利用 Vivado 创建设计的方法（本次以实现 4 位加法器为例）
- 3.掌握编写 Testbench 的方法，以及行为仿真方法

二、实验环境

1. 操作系统：Windows 11
2. 软件及版本：Vivado 2017.4

三、原理说明

全加器代码：将加数、被加数、低位进位信号相加，给出求和结果以及进位信号。

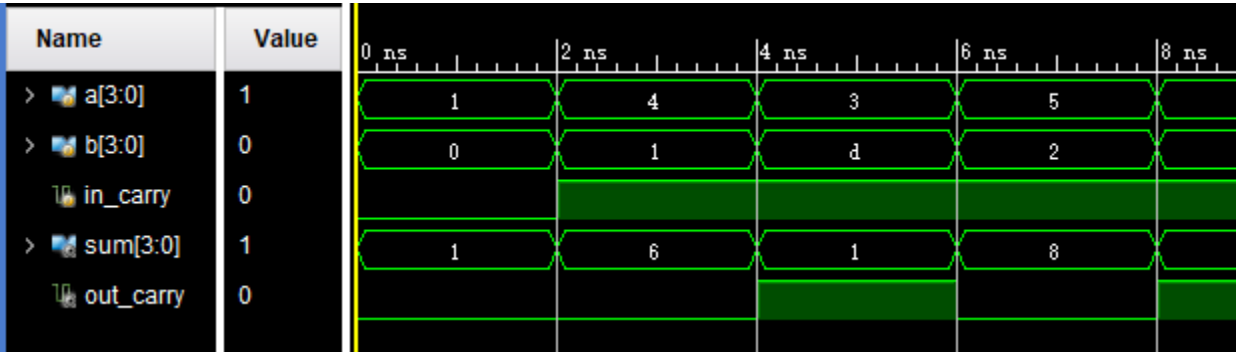
使用行为级描述，代码核心部分(assign {cout,out}=in_0+in_1+cin)使用 Verilog 的位拼接运算符：{信号 1, 信号 2}, 其中信号 1 是高位值，信号 2 是低位值，如果 in_0+in_1+cin 产生了溢出位，正好被参与拼接中的 cout 所接收，起到简化表达式的作用。

Testbench：随机产生两个 0-15 的整数作为加数和被加数以及一个 0-1 的整数作为低位进位信号，进行仿真模拟。

四、接口定义

I/O Port Definitions					
<div><div>+</div><div>−</div><div>↑</div><div>↓</div></div>					
Port Name	Direction	Bus	MSB	LSB	
in_0	input <div>▼</div>	<input checked="" type="checkbox"/>	3	0	
in_1	input <div>▼</div>	<input checked="" type="checkbox"/>	3	0	
cin	input <div>▼</div>	<input type="checkbox"/>	0	0	
out	output <div>▼</div>	<input checked="" type="checkbox"/>	3	0	
cout	output <div>▼</div>	<input type="checkbox"/>	0	0	

五、 调试结果及结果波形



六、 实验总结

本实验以实现 4 位加法器为例，熟悉了利用 Vivado 创建设计和编写 Testbench 并进行行为仿真的方法。虽为第一次使用 Vivado 但因实验整体难度较低加之有讲义辅助，完成过程中并未出现较大困难。

七、 源代码

加法器：

```
module add_4(  
    input [3:0] in_0,  
    input [3:0] in_1,  
    input cin,  
    output [3:0] out,  
    output cout  
);  
assign {cout,out}=in_0+in_1+cin;
```

```
endmodule
```

```
testbench:
```

```
module test_add_4( );
```

```
    reg [3:0] a;
```

```
    reg [3:0] b;
```

```
    reg in_carry;
```

```
    wire [3:0] sum;
```

```
    wire out_carry;
```

```
    add_4 instance_add_4(
```

```
        .in_0(a),
```

```
        .in_1(b),
```

```
        .cin(in_carry),
```

```
        .out(sum),
```

```
        .cout(out_carry)
```

```
    );
```

```
    initial begin
```

```
        a=4'h1;
```

```
        b=4'h0;
```

```
        in_carry=1'b0;
```

```
    end
```

```
    always begin
```

```
        #2;
```

```
        a=$random()%16;
```

```
        b=$random()%16;
```

```
        in_carry=$random()%2;
```

```
    end
```

```
endmodule
```