

# 实验四 verilog 数字系统设计基础 实验报告

16337233 王凯祺

2017 年 4 月 10 日

## 1 实验目的

1. 掌握 verilog 数字系统设计语言
2. 能使用 verilog 数字系统设计出一个简单的全减器

## 2 实验原理

1. 列出全减器真值表
2. 由真值表写函数式
3. 对函数式进行化简
4. 使用结构化描述的建模方式、数据流描述的建模方式、行为描述的建模方式为全减器建模，完成一个全减器的 verilog 设计

## 3 实验仪器

Vivado 2015.3 Basys 3 实验板

## 4 实验内容

使用结构化描述的建模方式、数据流描述的建模方式、行为描述的建模方式为全减器建模，完成一个全减器的 verilog 设计

## 5 实验设计

### 5.1 真值表

a	b	cin	sum	cout
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

### 5.2 表达式

$$sum = (\bar{a}\bar{b}cin) \vee (\bar{a}b\bar{c}in) \vee (a\bar{b}\bar{c}in) \vee (ab\bar{c}in) = a \oplus b \oplus cin$$

$$cout = (\bar{a}\bar{b}cin) \vee (\bar{a}b\bar{c}in) \vee (\bar{a}b\bar{c}in) \vee (ab\bar{c}in) = (\bar{a}\bar{b}) \vee (\bar{a}cin) \vee (b\bar{c}in)$$

### 5.3 建模

#### 5.3.1 结构化描述的建模方式

```
1 module minus_1(input a, input b, input cin, output sum, output cout);
2     wire s1, t1, t2, t3, na;
3     xor x1(s1, a, b);
4     xor x2(sum, s1, cin);
5     not n1(na, a);
6     and a1(t1, na, b);
7     and a2(t2, na, cin);
8     and a3(t3, b, cin);
9     or o1(cout, t1, t2, t3);
10 endmodule
```

#### 5.3.2 数据流描述的建模方式

```
1 module minus_2(input a, input b, input cin, output sum, output cout);
2     wire s1, t1, t2, t3, na;
3     assign s1 = a ^ b;
4     assign sum = s1 ^ cin;
5     assign na = ~a;
6     assign t1 = na & b;
7     assign t2 = na & cin;
8     assign t3 = b & cin;
9     assign cout = t1 | t2 | t3;
```

```
10 endmodule
```

### 5.3.3 行为描述的建模方式

```
1 module minus_3(input a, input b, input cin, output sum, output cout);
2     reg sum;
3     reg cout;
4     always @ (a or b or cin) begin
5         sum = a ^ b ^ cin;
6         cout = (~a & b) | (~a & cin) | (b & cin);
7     end
8 endmodule
```

### 5.3.4 仿真文件

```
1 module minus_tb();
2     reg a;
3     reg b;
4     reg cin;
5     wire sum1, sum2, sum3;
6     wire cout1, cout2, cout3;
7     minus_1 uut1(.a(a), .b(b), .cin(cin), .sum(sum1), .cout(cout1));
8     minus_2 uut2(.a(a), .b(b), .cin(cin), .sum(sum2), .cout(cout2));
9     minus_3 uut3(.a(a), .b(b), .cin(cin), .sum(sum3), .cout(cout3));
10    initial begin
11        a = 0;
12        b = 0;
13        cin = 0;
14        #2;
15    end
16    integer i, j;
17    always begin
18        for (i = 0; i < 8; i = i + 1) begin
19            j = i;
20            cin = j & 1;
21            j = j >> 1;
22            b = j & 1;
23            j = j >> 1;
24            a = j & 1;
25            #2;
26        end
27    end
28 endmodule
```

### 5.3.5 仿真波形图

