

# 中山大学数据科学与计算机学院本科生实验报告

## (2018 学年第二学期)

课程名称：数字电路与逻辑设计实验

任课教师：郭雪梅

助教：林鸿鑫

年级&班级	18 级计科 7 班	专业(方向)	计算机类
学号	18340181	姓名	谢俊杰
电话	16607657742	Email	xiej8@mail2.sysu.edu.cn
开始日期	2019/5/11	完成日期	2019/5/11

### 一、实验题目

使用 Vivado IP 核实现组合逻辑电路

### 二、实验目的

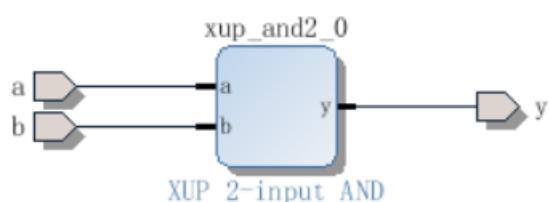
- 掌握使用 Vivado 实现组合逻辑电路的方法，并验证其逻辑功能。
- 熟悉 Vivado 的使用。

### 三、实验内容

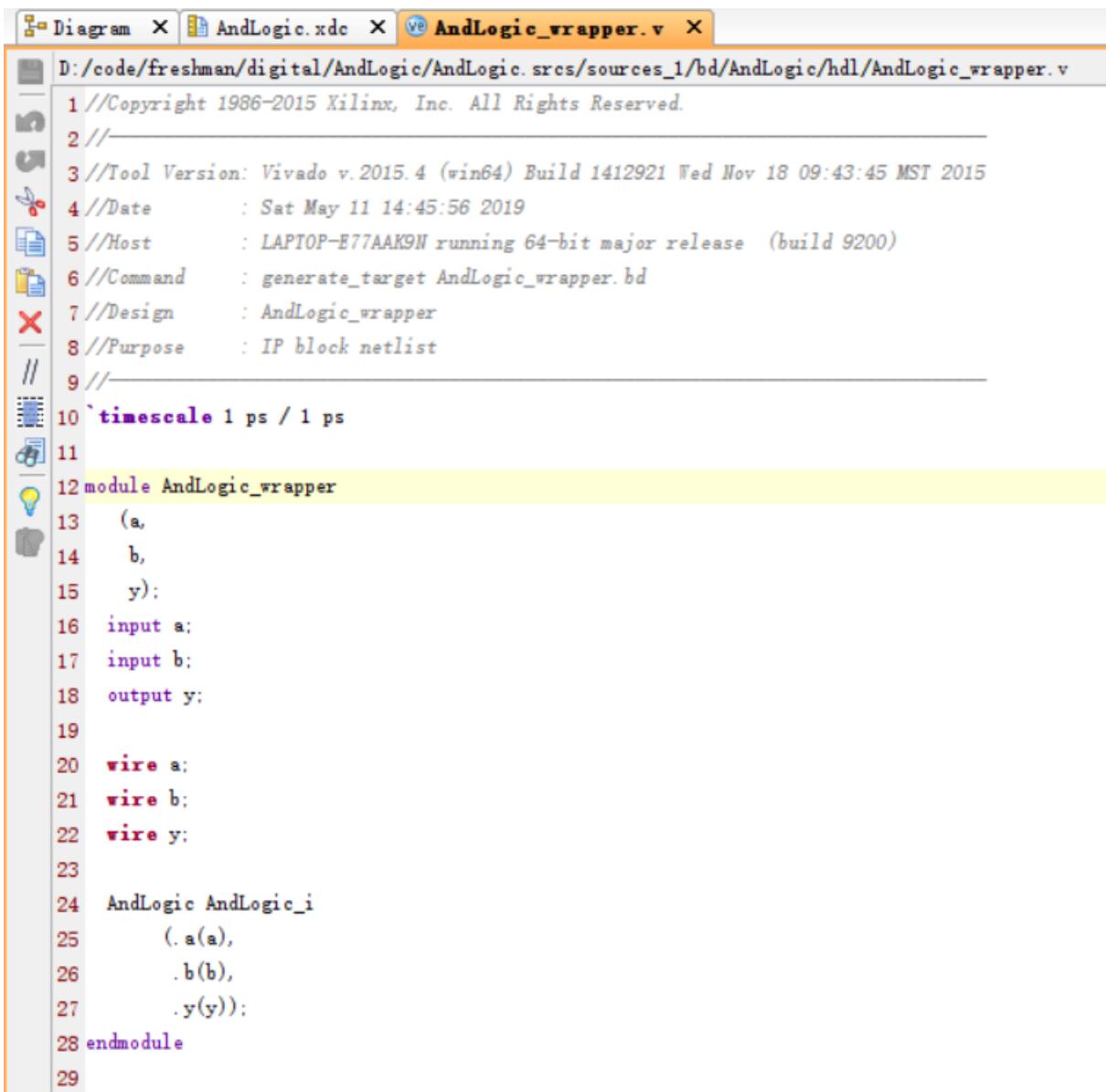
#### 1. 实验步骤

生成与门逻辑：

- (1) 新建工程文件，添加2输入与门IP核，并完成输入输出设计的原理图：



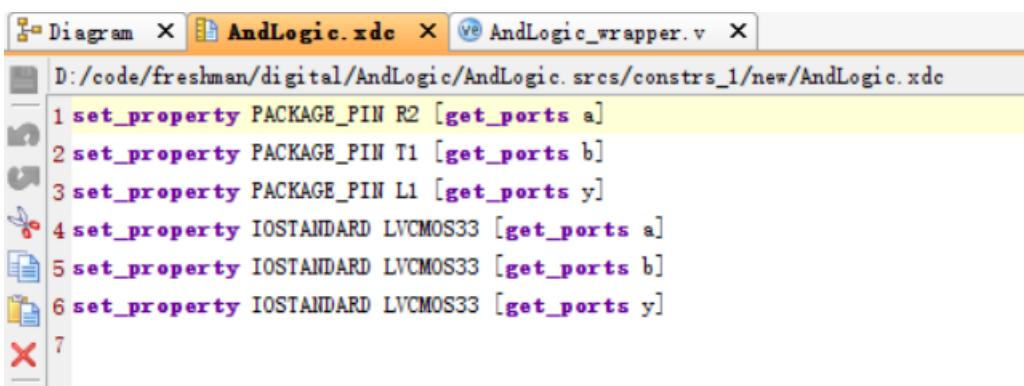
- (2) 在原理图基础上生成顶层文件AndLogic\_wrapper.v，内容如图：



The screenshot shows the Vivado IDE interface with three tabs open: Diagram, AndLogic.xdc, and AndLogic\_wrapper.v. The AndLogic\_wrapper.v tab is active, displaying Verilog code for a wrapper module. The code defines a module AndLogic\_wrapper with inputs a and b, and output y. It uses an AndLogic IP block (AndLogic\_i) to implement the logic. The code is as follows:

```
1 //Copyright 1986-2015 Xilinx, Inc. All Rights Reserved.
2 //
3 //Tool Version: Vivado v.2015.4 (win64) Build 1412921 Wed Nov 18 09:43:45 MST 2015
4 //Date      : Sat May 11 14:45:56 2019
5 //Host      : LAPTOP-E77AAK9N running 64-bit major release (build 9200)
6 //Command   : generate_target AndLogic_wrapper.bd
7 //Design    : AndLogic_wrapper
8 //Purpose   : IP block netlist
9 //
10 `timescale 1 ps / 1 ps
11
12 module AndLogic_wrapper
13     (a,
14      b,
15      y);
16     input a;
17     input b;
18     output y;
19
20     wire a;
21     wire b;
22     wire y;
23
24     AndLogic AndLogic_i
25         (.a(a),
26          .b(b),
27          .y(y));
28 endmodule
29
```

(3) 添加约束文件，分配管脚，约束文件如图：



The screenshot shows the Vivado IDE interface with three tabs open: Diagram, AndLogic.xdc, and AndLogic\_wrapper.v. The AndLogic.xdc tab is active, displaying an XDC (Hardware Description Constraint) file. The file sets properties for pins R2, T1, and L1, specifying they are PACKAGE\_PINS and have IOSTANDARD LVCMS33. The code is as follows:

```
1 set_property PACKAGE_PIN R2 [get_ports a]
2 set_property PACKAGE_PIN T1 [get_ports b]
3 set_property PACKAGE_PIN L1 [get_ports y]
4 set_property IOSTANDARD LVCMS33 [get_ports a]
5 set_property IOSTANDARD LVCMS33 [get_ports b]
6 set_property IOSTANDARD LVCMS33 [get_ports y]
```

(4) 生成Bitstream后，烧写到Basys3实验板。

实现输出端为3位的二进制数的与门逻辑：

(1) 新建工程文件，添加n输入与门IP核，并完成输入输出设计的原理图：



(2) 在原理图基础上生成顶层文件AndLogic\_wrapper.v，内容如图：

```
Diagram X AndLogic.xdc X AndLogic_wrapper.v X
D:/code/freshman/digital/AndLogic/AndLogic.scs/sources_1/bd/AndLogic/hdl/AndLogic_wrapper.v
1 //Copyright 1986-2015 Xilinx, Inc. All Rights Reserved.
2 //
3 //Tool Version: Vivado v.2015.4 (win64) Build 1412921 Wed Nov 18 09:43:45 MST 2015
4 //Date       : Sat May 11 15:08:19 2019
5 //Host       : LAPTOP-E77AAK9N running 64-bit major release (build 9200)
6 //Command    : generate_target AndLogic_wrapper.bd
7 //Design     : AndLogic_wrapper
8 //Purpose    : IP block netlist
9 //
10 `timescale 1 ps / 1 ps
11
12 module AndLogic_wrapper
13   (a,
14   b,
15   y);
16   input [2:0]a;
17   input [2:0]b;
18   output [2:0]y;
19
20   wire [2:0]a;
21   wire [2:0]b;
22   wire [2:0]y;
23
24   AndLogic AndLogic_i
25     (.a(a),
26      .b(b),
27      .y(y));
28 endmodule
29
```

(3) 添加约束文件，分配管脚，约束文件如图：

```

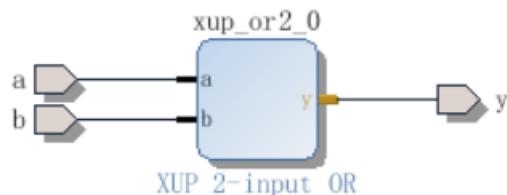
1 set_property PACKAGE_PIN R2 [get_ports a]
2 set_property PACKAGE_PIN T1 [get_ports b]
3 set_property PACKAGE_PIN L1 [get_ports y]
4 set_property IOSTANDARD LVCMOS33 [get_ports {a[2]}]
5 set_property IOSTANDARD LVCMOS33 [get_ports {a[1]}]
6 set_property IOSTANDARD LVCMOS33 [get_ports {a[0]}]
7 set_property IOSTANDARD LVCMOS33 [get_ports {b[2]}]
8 set_property IOSTANDARD LVCMOS33 [get_ports {b[1]}]
9 set_property IOSTANDARD LVCMOS33 [get_ports {b[0]}]
10 set_property IOSTANDARD LVCMOS33 [get_ports {y[2]}]
11 set_property IOSTANDARD LVCMOS33 [get_ports {y[1]}]
12 set_property IOSTANDARD LVCMOS33 [get_ports {y[0]}]
13
14 set_property PACKAGE_PIN R2 [get_ports {a[2]}]
15 set_property PACKAGE_PIN T1 [get_ports {a[1]}]
16 set_property PACKAGE_PIN U1 [get_ports {a[0]}]
17 set_property PACKAGE_PIN W2 [get_ports {b[2]}]
18 set_property PACKAGE_PIN R3 [get_ports {b[1]}]
19 set_property PACKAGE_PIN T2 [get_ports {b[0]}]
20 set_property PACKAGE_PIN L1 [get_ports {y[2]}]
21 set_property PACKAGE_PIN P1 [get_ports {y[1]}]
22 set_property PACKAGE_PIN N3 [get_ports {y[0]}]
23

```

(4) 生成Bitstream后，烧写到Basys3实验板。

实现2输入或门逻辑：

(1) 新建工程文件，添加2输入或门IP核，并完成输入输出设计的原理图：



(2) 在原理图基础上生成顶层文件OrLogic\_wrapper.v，内容如图：

The screenshot shows the Vivado IDE interface with the file 'OrLogic\_wrapper.v' open. The code is a Verilog module definition for an OR logic block. It includes comments about the tool version (Vivado v.2015.4), date (Sat May 11 15:18:30 2019), host (LAPTOP-E77AAK9N), command (generate\_target), design (OrLogic\_wrapper), and purpose (IP block netlist). The module has three ports: a, b, and y. It uses an OrLogic\_i IP block to implement the logic. The code ends with an endmodule statement.

```
1 //Copyright 1986-2015 Xilinx, Inc. All Rights Reserved.
2 //
3 //Tool Version: Vivado v.2015.4 (win64) Build 1412921 Wed Nov 18 09:43:45 MST 2015
4 //Date      : Sat May 11 15:18:30 2019
5 //Host      : LAPTOP-E77AAK9N running 64-bit major release (build 9200)
6 //Command   : generate_target OrLogic_wrapper.bd
7 //Design    : OrLogic_wrapper
8 //Purpose   : IP block netlist
//
9 //
10 `timescale 1 ps / 1 ps
11
12 module OrLogic_wrapper
13     (a,
14      b,
15      y);
16     input a;
17     input b;
18     output y;
19
20     wire a;
21     wire b;
22     wire y;
23
24     OrLogic OrLogic_i
25         (.a(a),
26          .b(b),
27          .y(y));
28 endmodule
29
```

(3) 添加约束文件，分配管脚，约束文件如图：

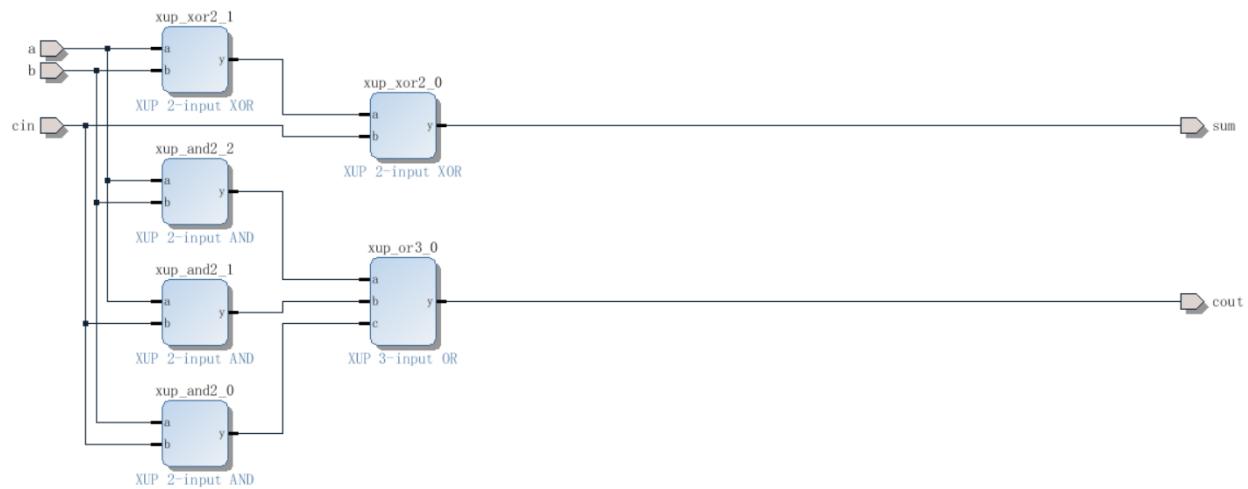
The screenshot shows the Vivado IDE interface with the file 'OrLogic.xdc' open. The code defines six set\_property commands for pins R2, T1, L1, and three instances of IOSTANDARD LVCMOS33. The first three lines set PACKAGE\_PIN properties for pins R2, T1, and L1 respectively, each followed by a [get\_ports] expression. The next three lines set IOSTANDARD properties for the same pins, also followed by [get\_ports] expressions. The final line is a comment starting with '#'. The code ends with a line number '7'.

```
1 set_property PACKAGE_PIN R2 [get_ports a]
2 set_property PACKAGE_PIN T1 [get_ports b]
3 set_property PACKAGE_PIN L1 [get_ports y]
4 set_property IOSTANDARD LVCMOS33 [get_ports a]
5 set_property IOSTANDARD LVCMOS33 [get_ports b]
6 set_property IOSTANDARD LVCMOS33 [get_ports y]
7
```

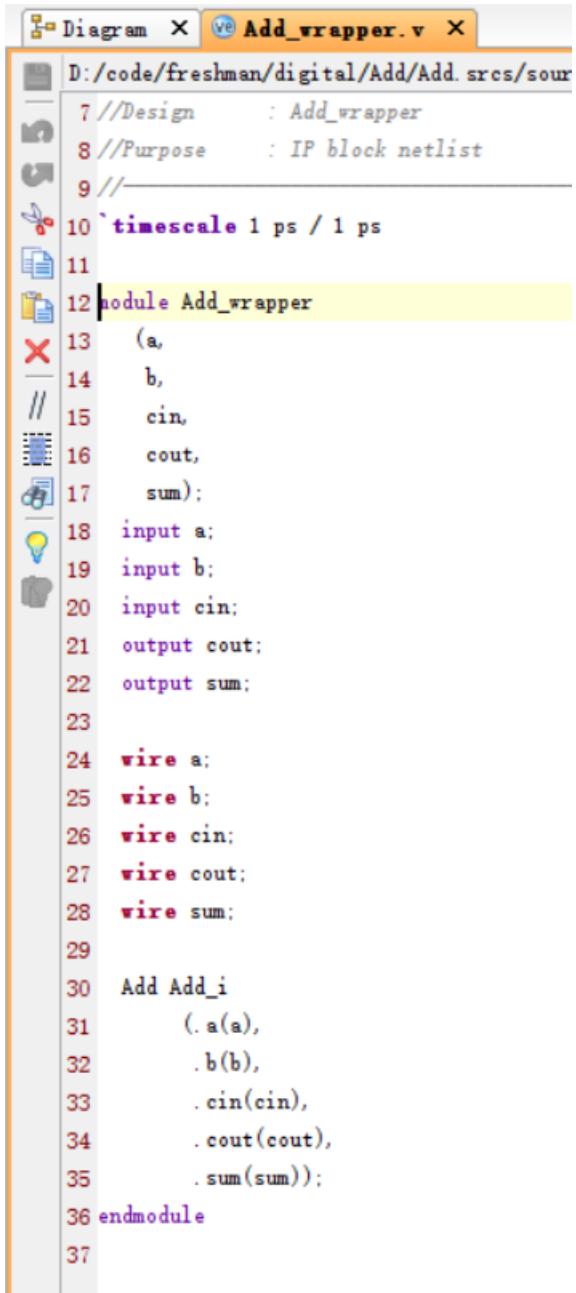
(4) 生成Bitstream后，烧写到Basys3实验板。

实现全加器：

(1) 新建工程文件，添加2输入与门IP核、2输入异或门、3输入或门，并完成输入输出设计的原理图：



(2) 在原理图基础上生成顶层文件Add\_wrapper.v，内容如图：



The screenshot shows a code editor window with the following details:

- Title Bar:** Diagram X Add\_wrapper.v X
- File Path:** D:/code/freshman/digital/Add/Add.sr~~c~~s/sour
- Code Content:**

```
7 //Design      : Add_wrapper
8 //Purpose     : IP block netlist
9 //
10 `timescale 1 ps / 1 ps
11
12 module Add_wrapper
13   (a,
14   b,
15   cin,
16   cout,
17   sum);
18   input a;
19   input b;
20   input cin;
21   output cout;
22   output sum;
23
24   wire a;
25   wire b;
26   wire cin;
27   wire cout;
28   wire sum;
29
30   Add Add_i
31     (.a(a),
32     .b(b),
33     .cin(cin),
34     .cout(cout),
35     .sum(sum));
36 endmodule
37
```

(3) 添加约束文件，分配管脚，约束文件如图：

```

Diagram × Add_wrapper.v × Add.xdc ×
D:/code/freshman/digital/Add/Add.srcts/constrs_1/new/Add.xdc
1 set_property PACKAGE_PIN V17 [get_ports a]
2 set_property PACKAGE_PIN V16 [get_ports b]
3 set_property PACKAGE_PIN W16 [get_ports cin]
4 set_property PACKAGE_PIN E19 [get_ports cout]
5 set_property PACKAGE_PIN U16 [get_ports sum]
6 set_property IOSTANDARD LVCMOS33 [get_ports a]
7 set_property IOSTANDARD LVCMOS33 [get_ports b]
8 set_property IOSTANDARD LVCMOS33 [get_ports cin]
9 set_property IOSTANDARD LVCMOS33 [get_ports cout]
10 set_property IOSTANDARD LVCMOS33 [get_ports sum]
11

```

(4) 生成Bitstream后，烧写到Basys3实验板。

#### 四、实验结果

生成与门逻辑实验中，烧写到实验板后，只有当R2和T1两个开关都置于1时，灯L1才会亮，其他情况下L1熄灭，符合2输入与门逻辑功能。

实现2输入或门逻辑实验中，烧写到实验板后，只有当R2和T1两个开关都置于0时，灯L1才会熄灭，其他情况下L1亮，符合2输入或门逻辑功能。

实现全加器实验中，烧写到实验板后，开关V17、V16、W16可置于0、1，灯U16和E19的亮灭情况如下图真值表所示：

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

1	1	0	0	1
1	1	1	1	1

经验证，符合全加器的逻辑功能。

## 五、实验感想

在第一次使用 Vivado 的实验后，对该软件的使用并不了解，在实验后仍然不懂如何去使用，所以导致在第二次（即该次）实验中，需要常常看回上一次的实验步骤和操作流程，但是在此次实验后，经过在实验课和课后的几次重复实验后，也逐渐熟悉 Vivado 的一部分功能使用，也明白到这些软件的熟悉需要重复的练习。