

浙江大学实验报告

课程名称：电路与电子技术实验 II 指导老师：张伟
实验名称：一位全加器和四位加法器 同组学生：

专业：电子信息工程
姓名：
学号：
地点：紫金港东三 406
日期：2024 年 5 月 25 日

1 实验目的

1. 了解一位全加器的结构和工作原理；
2. 了解如何将一位全加器连接成四位加法器；
3. 学会使用 VHDL 语言描述数字电路；
4. 掌握 Quartus 软件的基本用法。

2 实验原理

2.1 一位全加器

使用 VHDL 语言描述一位全加器的结构如下：

```
1  --add1bit.vhd--
2  --引用库--
3  library IEEE;
4  use IEEE.STD_LOGIC_1164.ALL;
5  -- 实体定义 --
6  entity add1bit is
7      port(a : in std_logic;
8           b : in std_logic;
9           c0 : in std_logic;
10          s : out std_logic;
11          c1 : out std_logic );
12  end add1bit;
13  -- 架构定义 --
14  architecture behavioral of add1bit is
15  begin
16      c1 <= (a and b) or (a and c0) or
17            (c0 and b);
18      s <= a xor b xor c0;
19  end behavioral;
```

2.2 四位加法器

基本结构如下图所示：

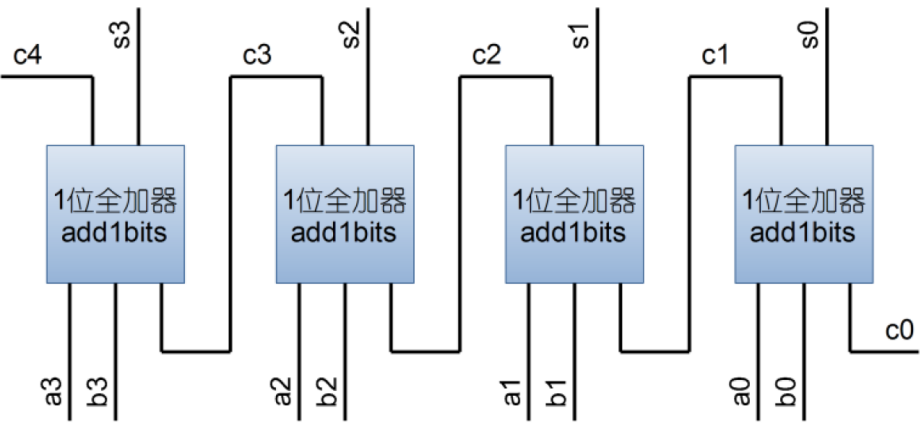
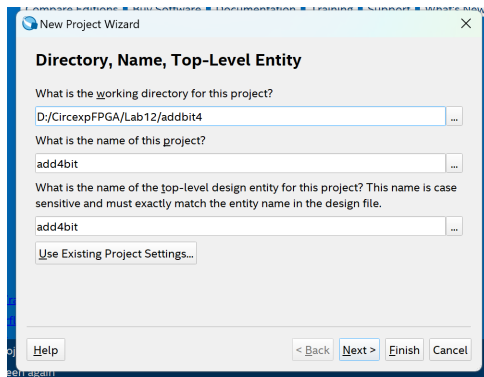
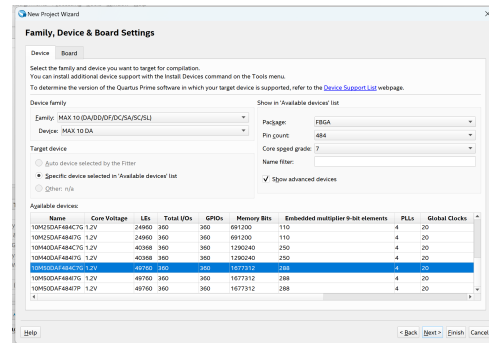


图 1: 四位加法器



(a) 新建工程



(b) 选择 FPGA 面板

图 2: 新建工程

由上图所示,该 4 位全加器由 4 个 1 位全加器组成,其中第一个全加器的进位输入由 C0 确定,其余的全加器的进位输入为前一个全加器的进位输出。则 VHDL 代码如下:

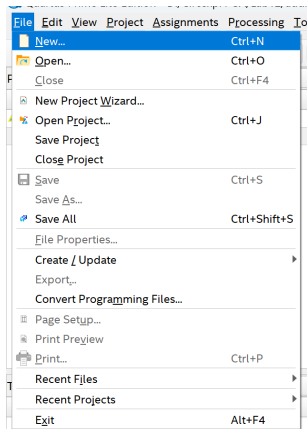
```

1  --add4bit.vhd--
2  --引用库--
3  library IEEE;
4  use IEEE.STD_LOGIC_1164.ALL;
5  -- 4位全加器实体定义 --
6  entity add4bit is
7      port(
8          a : in std_logic_vector(3
              downto 0);
9          b : in std_logic_vector(3
              downto 0);
10         c0 : in std_logic;
11         s : out std_logic_vector(3
              downto 0);
12         c4 : out std_logic
13     );
14 end add4bit;
15 -- 4位全加器架构定义 --
16 architecture Behavioral of add4bit is
17     -- 引用1位全加器 --
18     component add1bit
19         port(
20             a : in std_logic;
21             b : in std_logic;
22             c0 : in std_logic;
23             s : out std_logic;
24             c1 : out std_logic
25         );
26 end component;
27 -- 4位全加器的内部信号 --
28 signal ss : std_logic_vector(3
              downto 0);
29 signal c1, c2, c3, cc4 : std_logic;
30 -- 4位全加器的结构 --
31 begin
32     u0 : add1bit port map(a => a
33         (0), b => b(0), c0 => c0,
34         s => ss(0), c1 => c1);
35     u1 : add1bit port map(a => a
36         (1), b => b(1), c0 => c1,
37         s => ss(1), c1 => c2);
38     u2 : add1bit port map(a => a
39         (2), b => b(2), c0 => c2,
40         s => ss(2), c1 => c3);
41     u3 : add1bit port map(a => a
42         (3), b => b(3), c0 => c3,
43         s => ss(3), c1 => cc4);
44     s <= ss;
45     c4 <= cc4;
46 end Behavioral;

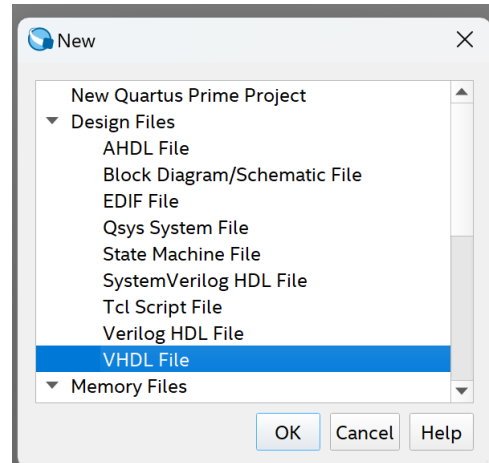
```

3 实验步骤

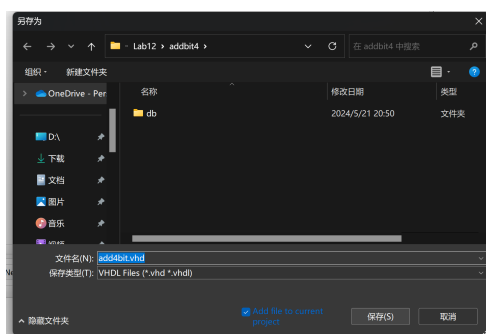
1. 打开 Quartus 软件，新建工程;
2. 在新建的工程中新建一个 VHDL 文件，将上述 add4bit 代码复制到 VHDL 文件中;



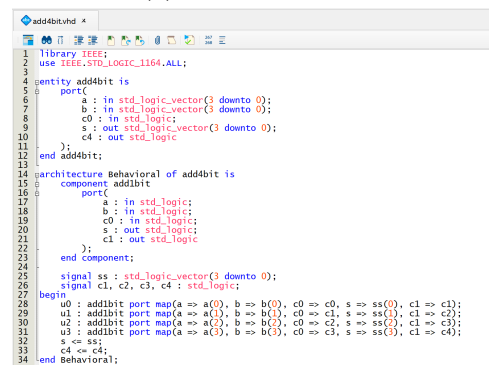
(a) 新建文件



(b) 选择 VHDL 文件



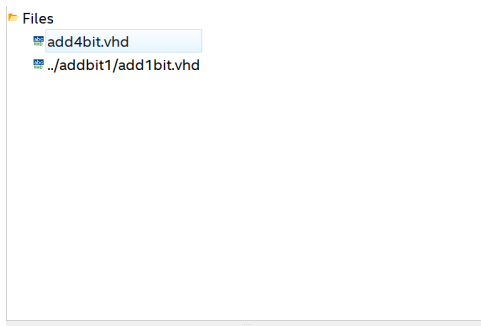
(c) 保存 add4bit.vhd 文件



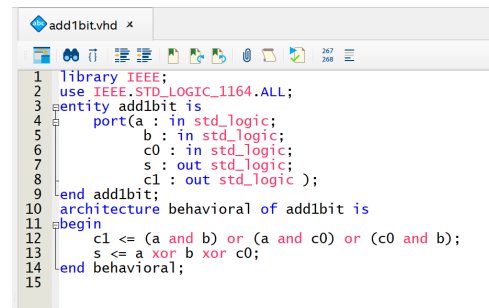
(d) 复制代码

图 3: 新建 VHDL 文件

3. 添加 add1bit.vhd 文件;



(a) 新建文件



(b) add1bit.vhd 文件

图 4: 添加 add1bit.vhd 文件

4. 编译工程;

Tasks		
Compilation		
Task		Time
0%	▶ Compile Design	00:00:02
4%	▶▶ Analysis & Synthesis	00:00:02
0%	▶▶ Fitter (Place & Route)	00:00:00
0%	▶▶ Assembler (Generate programming files)	00:00:00
0%	▶▶ Timing Analysis	00:00:00
0%	▶▶ EDA Netlist Writer	00:00:00
■ Edit Settings		
🔗 Program Device (Open Programmer)		

(a) 编译工程

Task		Time
✓	▶ Compile Design	00:00:41
✓	▶▶ Analysis & Synthesis	00:00:15
✓	▶▶ Fitter (Place & Route)	00:00:12
✓	▶▶ Assembler (Generate programming files)	00:00:05
✓	▶▶ Timing Analysis	00:00:07
✓	▶▶ EDA Netlist Writer	00:00:02
■ Edit Settings		
🔗 Program Device (Open Programmer)		

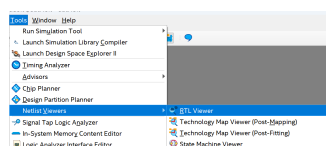
(b) 编译工程成功

5. 查看工程架构;

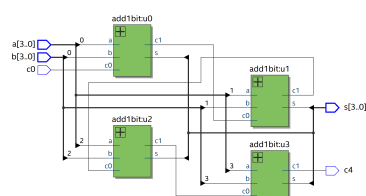
Entity:Instance	
MAX 10: 10M50DAF484C7G	
▼	add4bit
	add1bit:u0
	add1bit:u1
	add1bit:u2
	add1bit:u3

图 6: 工程架构

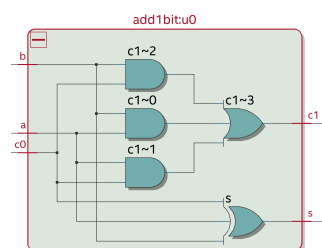
6. 查看 RTL 图;



(a) 查看 RTL 图



(b) add4bit 结构



(c) add1bit 结构

图 7: 查看 RTL 图

7. 对引脚进行排布:

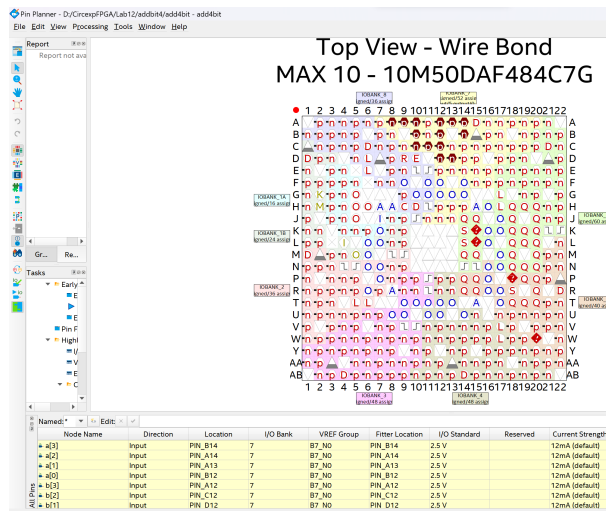


图 8: 引脚排布

8. 下载到 FPGA 板上;

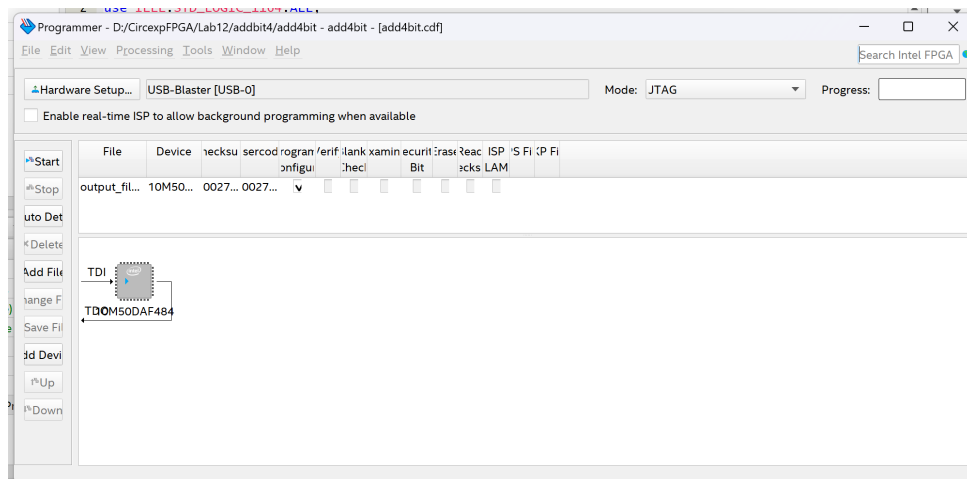
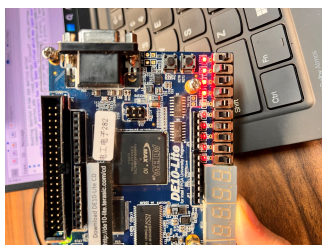
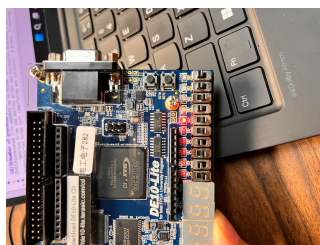


图 9: 下载到 FPGA 板上

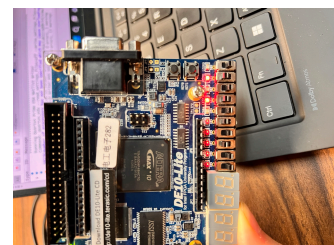
9. 测试结果;



(a) $0101 + 1010 + 0 = 1111$



(b) $0101 + 1010 + 1 = 10000$



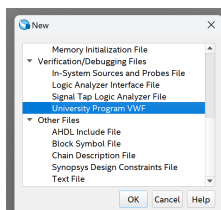
(c) $0101 + 0101 + 1 = 1010$

图 10: 测试结果

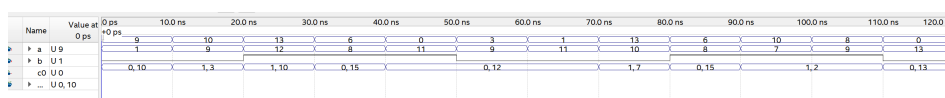
4 其他内容

对 VHDL 代码进行验证，使用 VWF 文件进行仿真：

1. 新建 VWF 文件；
2. 添加信号；
3. 修改波形；
4. 运行仿真；



(a) 新建 VWF 文件

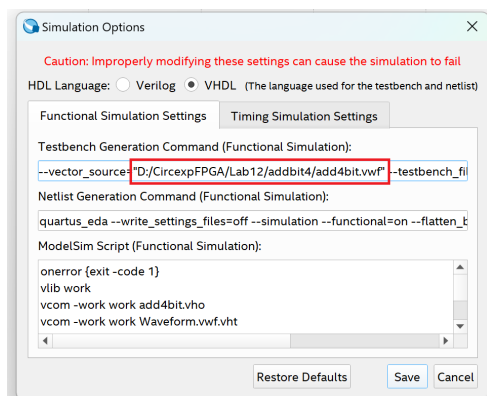


(b) 添加信号, 并设置波形

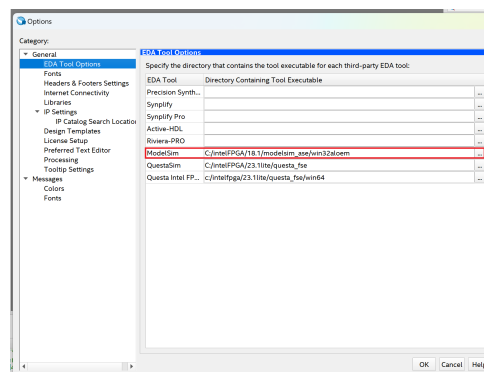
图 11: 仿真

在调试过程中报错，发现以下两点错误：

- 仿真命令中仿真文件名字错误
- 新版的 quartus 没有安装 modelsim，需要单独安装，并且需修改 modelsim 的 exe 文件路径



(a) 修改文件名



(b) 修改 modelsim 路径

图 12: 对结果进行仿真