

# 浙江大学实验报告

专业：电气工程及其自动化

姓名：潘谷雨

学号：3220102382

地点：紫金港东三 406

课程名称：电路与电子技术实验 指导老师：张伟 成绩：\_\_\_\_\_

实验名称：DE10-Lite开发板数字逻辑设计实验 同组学生姓名：杨骥恺

## 一. 实验目的

1. 初步了解FPGA工程的开发过程；
2. 认识DE10-Lite开发板；
3. 学习Quartus-Prime集成开发软件的应用和FPGA软件的开发。

## 二. 实验仪器

MAX10M50DAF484C7G芯片，Quartus-Prime集成开发软件。

## 三. 实验原理

### 1. 一位全加器

输入A, B, 前一位进位 $C_{i-1}$ 。真值表如下左图，画出 $S_i$ 与 $C_i$ 卡诺图如下右图。

$[A, B, C_{i-1}] \rightarrow [C_i, S_i]$

$[0, 0, 0] \rightarrow [0, 0]$ ;

$[0, 0, 1] \rightarrow [0, 1]$ ;

$[0, 1, 0] \rightarrow [0, 1]$ ;

$[0, 1, 1] \rightarrow [1, 0]$ ;

$[1, 0, 0] \rightarrow [0, 1]$ ;

$[1, 0, 1] \rightarrow [1, 0]$ ;

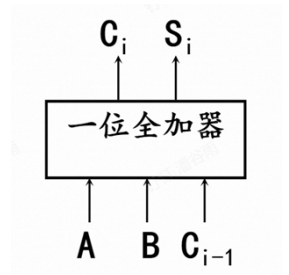
$[1, 1, 0] \rightarrow [1, 0]$ ;

$[1, 1, 1] \rightarrow [1, 1]$ ;

$S_i$	$BC_{i-1}$			
	00	01	11	10
0	0	1	0	1
1	1	0	1	0

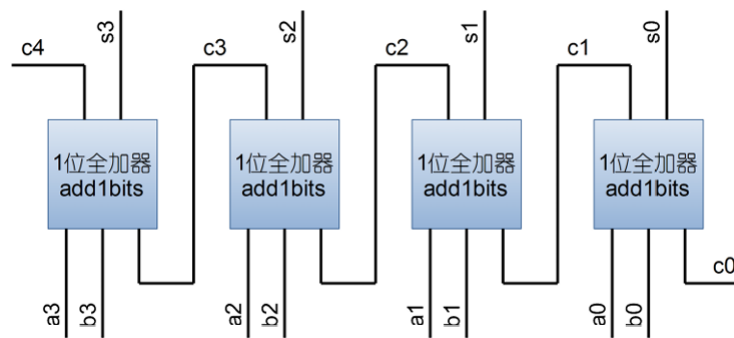
$C_i$	$BC_{i-1}$			
	00	01	11	10
0	0	0	1	0
1	0	1	1	1

故输出 $S_i$ 满足：  $S_i = A \oplus B \oplus C_{i-1}$ ，输出 $C_i$ 满足：  $C_i = AB + BC_{i-1} + AC_{i-1}$ 。



### 2. 四位全加器

四位全加器由4个一位全加器级联而成，原理如下图所示。



## 四. 实验步骤与结果

### 1. 一位全加器

#### (1) 新建项目

启动Quartus软件，选择File > New Project Wizard...菜单命令，输入项目文件夹、项目名称和顶层实体名，新建空项目，器件选择10M50DAF484C7G芯片。

#### (2) 编写设计文件

新建设计文件，类型选择VHDL File，另存为add1bit.vhd，输入1位全加器的VHDL源程序后如图1.1所示。

实验名称：DE10-Lite开发板数字逻辑设计实验 姓名：潘谷雨 学号：3220102382

```

1 library IEEE;
2 use IEEE.STD_LOGIC_1164.ALL;--使用了IEEE库中的STD_LOGIC_1164包
3
4 -- 定义一个实体add1bit, 用于实现1位二进制数的加法
5 entity add1bit is
6 port(
7     a:in std_logic; -- 输入: 1位二进制数a
8     b:in std_logic; -- 输入: 1位二进制数b
9     c0:in std_logic; -- 输入: 来自低位的进位
10    s:out std_logic; -- 输出: 本位的和
11    c1:out std_logic; -- 输出: 向高位的进位
12 );
13 end add1bit;
14
15 architecture behavior of add1bit is
16 begin
17     c1<=(a and b)or(a and c0)or(c0 and b); --当a、b、c0中任意两个或全部为1时, c1为1产生进位
18     s<=a xor b xor c0;-- 本位和为三输入的异或
19 end behavior;

```

图1.1 1位全加器的VHDL源程序

### (3) 项目编译

执行Processing>Start Compilation 菜单命令。

### (4) 管脚定义

①等到编译不报错, 执行Assignments>Pin planner 菜单命令, 进入引脚分配编辑器窗口, 如图1.2所示。将3个拨码开关SW2 (PIN\_D12)、SW1 (PIN\_C11)、SW0 (PIN\_C10) 分别作为全加器的输入a, b, c0, 将2个LED灯LEDR1 (PIN\_A9)、LEDR0 (PIN\_A8) 作为全加器的输出c1, s。

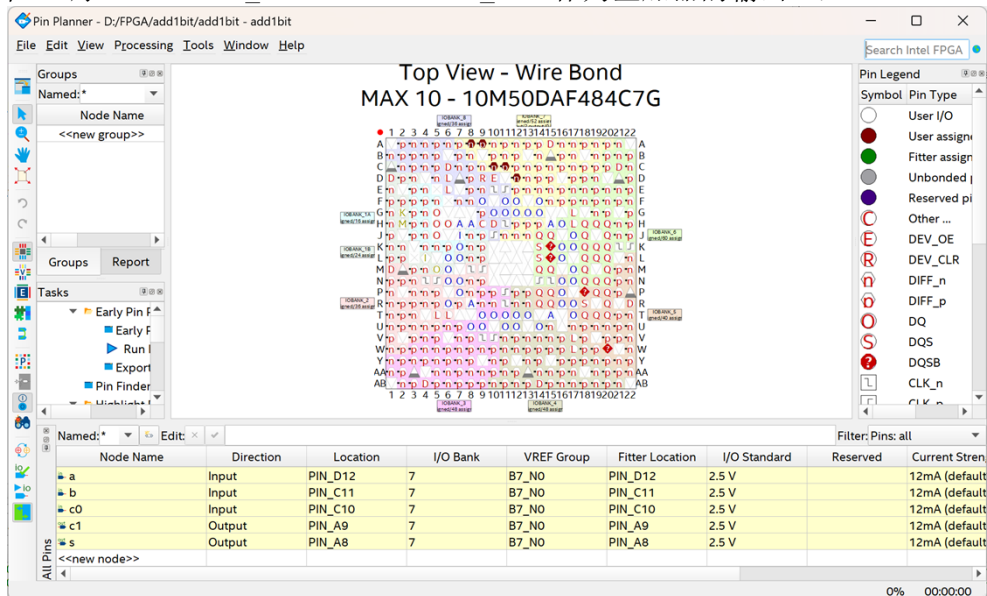


图1.2 引脚分配编辑器窗口

②进行未用引脚的设置, 执行Assignments>Device菜单命令, 点击Device and Pin Options...按钮, 选中Unused Pins 条目, 在Reserve all unused pins 下拉列表中选择As input tri-stated设置项。

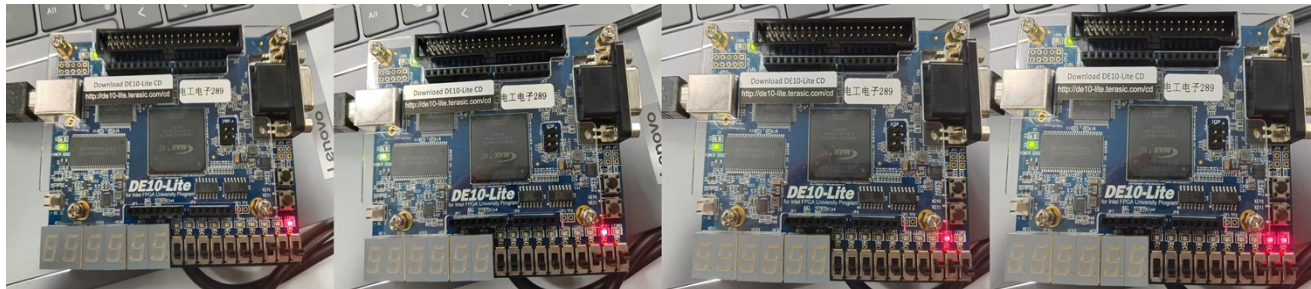
### (5) 重新编译

重新对项目进行完整编译后生成add1bit.sof文件, 经程序下载后在DE10-Lite开发板上验证程序。结果如下图所示。

①开关全部置零无灯亮, 输出00; abc0=001, 输出01; abc0=010, 输出01; abc0=011, 输出10。



②abc0=100, 输出01; abc0=101, 输出10; abc0=110, 输出10; abc0=111, 输出11。



符合一位全加器功能。

## 2. 四位全加器

### (1) 新建项目

启动Quartus软件，选择File > New Project Wizard...菜单命令，输入项目文件夹、项目名称和顶层实体名，新建空项目，器件选择10M50DAF484C7G芯片。

### (2) 编写设计文件

新建设计文件，类型选择VHDL File，另存为add4bit.vhd，输入4位全加器的VHDL源程序后如图2.1所示。将add1bit.vhd粘贴至add4bit.vhd同一目录中。

```
1 library IEEE;
2 use IEEE.STD_LOGIC_1164.ALL;
3
4 -- 定义名为add4bit的实体
5 entity add4bit is
6 port(
7   a:in std_logic_vector(3 downto 0); -- 输入: 4位二进制数a
8   b:in std_logic_vector(3 downto 0); -- 输入: 4位二进制数b
9   c0:in std_logic; -- 输入: 初始进位位
10  s:out std_logic_vector(3 downto 0); -- 输出: 4位二进制数的和
11  c4:out std_logic -- 输出: 最终进位位
12 );
13 end add4bit;
14
15 --为add4bit实体定义行为结构体
16 architecture behavior of add4bit is
17   --声明一个组件add1bit, 用于实现1位二进制数的加法
18   component add1bit
19   port(
20     a:in std_logic; -- 输入: 1位二进制数a
21     b:in std_logic; -- 输入: 1位二进制数b
22     c0:in std_logic; -- 输入: 进位输入
23     s:out std_logic; -- 输出: 1位二进制数的和
24     c1:out std_logic -- 输出: 进位输出
25   );
26 end component;
27
28 -- 声明信号用于在各组件间传递进位信息和部分和。
29 signal ss:std_logic_vector(3 downto 0); -- 用于暂存4位加法的部分和
30 signal c1,c2,c3,cc4:std_logic; -- 用于传递进位信号
31
32 begin
33   u0:add1bit port map(a=>a(0),b=>b(0),c0=>c0,s=>ss(0),c1=>c1); -- 实例化add1bit组件, 处理最低位的加法
34   u1:add1bit port map(a=>a(1),b=>b(1),c0=>c1,s=>ss(1),c1=>c2); -- 每个组件从前一位接收进位信号, 处理该位加法
35   u2:add1bit port map(a=>a(2),b=>b(2),c0=>c2,s=>ss(2),c1=>c3);
36   u4:add1bit port map(a=>a(3),b=>b(3),c0=>c3,s=>ss(3),c1=>cc4);
37   s<=ss; -- 将部分和信号赋值给最终的和输出s
38   c4<=cc4; -- 将最高位的进位输出赋值给c4
39 end behavior;
```

图2.1 4位全加器的VHDL源程序

### (3) 项目编译

执行Processing>Start Compilation 菜单命令。

### (4) 管脚定义

①等到编译不报错，执行Assignments>Pin planner 菜单命令，进入引脚分配编辑器窗口，如图2.2所示。拨码开关SW0 (PIN\_C10) 作为全加器的输入c0，4个拨码开关SW1-SW4分别作为全加器的输入a[0]-a[3]，4个拨码开关SW5-SW8分别作为全加器的输入b[0]-b[3]，4个LED灯LEDR0-LEDR3作为全加器的输出s[0]-s[3]，LEDR3作为全加器的输出c4。



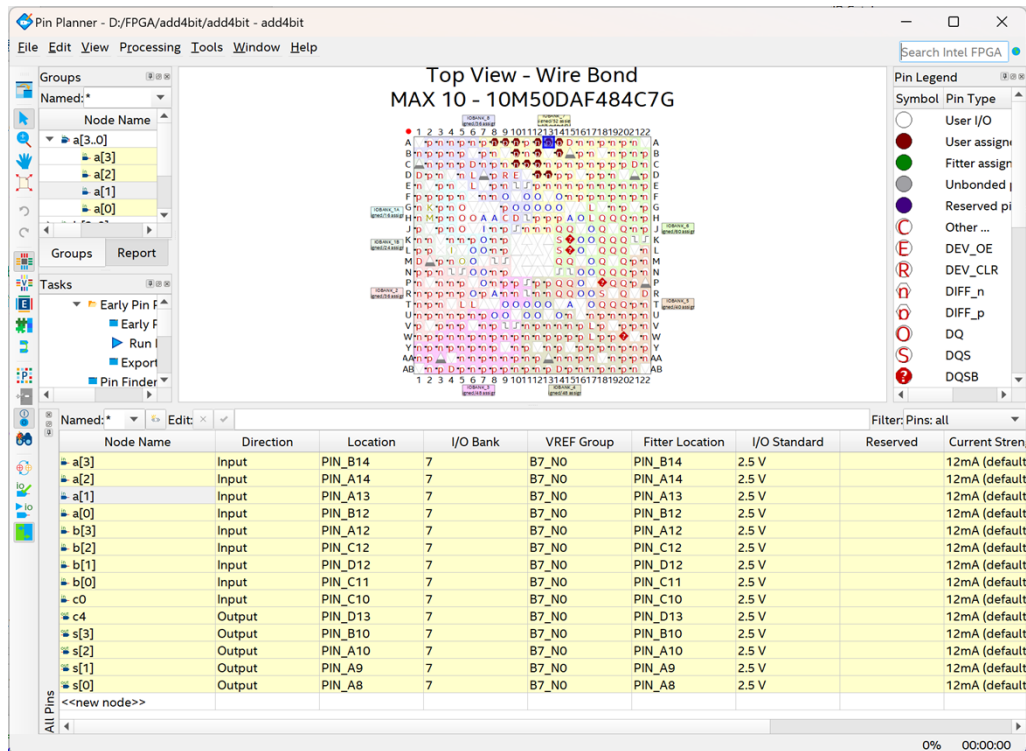


图2.2 引脚分配编辑器窗口

②进行未用引脚的设置，执行Assignments>Device菜单命令，点击Device and Pin Options...按钮，选中Unused Pins 条目，在Reserve all unused pins 下拉列表中选择As input tri-stated设置项。

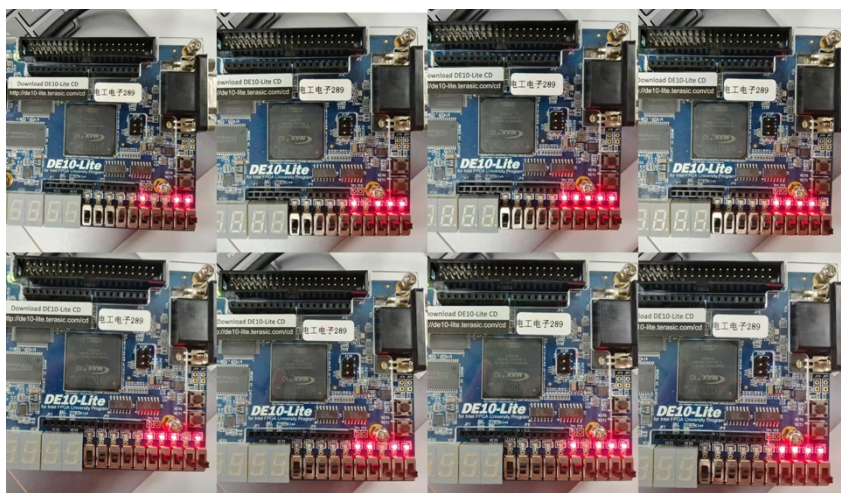
#### (5) 重新编译

重新对项目进行完整编译后生成add4bit.sof文件，经程序下载后在DE10-Lite开发板上验证程序。按顺序调整开关，每次改变一位，部分结果如下图所示。

①开关全部置零无灯亮，输出00000；c0置1，输出00001；a[0]置1，输出00010；b[0]置1，输出00011；a[1]置1，输出00101；a[2]置1，输出01001；a[3]置1，输出10001。



②接上述步骤，b[1]置1，输出10011；b[2]置1，输出10111；b[3]置1，输出11111，c0置0，输出11110；a[0]置0，输出11101；a[1]置0，输出11011；a[2]置0，输出10111；a[3]置0，输出01111。



符合四位全加器功能。

实验名称：DE10-Lite开发板数字逻辑设计实验 姓名：潘谷雨 学号：3220102382

