

计算机系统结构实验报告 Lab02

FPGA 基础实验: 4-bit Adder

姚宣骋 520021910431

2022 年 3 月 15 日

摘要

本实验实现了 FPGA 基础实验中的四位全加器。该器件支持带进位的 4 位二进制数的加法，输入两个 4 位二进制数，输出表示输入的两个数相加的结果的一个 4 位二进制数，并且支持接收上一部分的进位结果与传递下一部分的进位结果。我们先通过语言内置的与、或、异或等简单的逻辑运算实现一位全加器，再通过只用一位全加器来进一步实现四位全加器。本实验通过软件仿真的形式进行实验结果的验证。

目录

1 实验目的	2
2 原理分析	2
2.1 一位全加器原理分析	2
2.2 四位全加器原理分析	3
3 功能实现	3
3.1 一位全加器功能实现	3

3.2 四位全加器功能实现	4
4 结果验证	4
5 总结反思	5
6 致谢	6
A 设计文件完整代码实现	6
A.1 一位全加器的代码实现	6
A.2 四位全加器的代码实现	6
B 激励文件完整代码实现	6

1 实验目的

本次实验有 4 个目的：

- 熟悉 Xilinx 逻辑设计工具 Vivado 的基本操作
- 掌握使用 VerilogHDL 进行简单的逻辑设计
- 理解一位全加器与四位全加器的工作原理
- 使用功能仿真验证功能实现的正确性

2 原理分析

2.1 一位全加器原理分析

一位全加器包括 a, b 和 c_i 三个 1 位输入端，用来承接两个输入数和上一部分的进位结果；还包括了 s 和 c_o 两个 1 位输出端，分别表示加法运算

结果和需要传递到下一部分的进位结果。用逻辑电路来表达，输入端和输出端的逻辑关系如下

$$s = a \oplus b \oplus c \quad (1)$$

$$c_o = (a \wedge b) \vee (a \wedge c_i) \vee (b \wedge c_i) \quad (2)$$

其中 \oplus 表示逻辑异或。

2.2 四位全加器原理分析

四位全加器包括 a, b 和 c_i 三个 4 位输入端，用来承接两个输入数和上一部分的进位结果；还包括了 s 和 c_o 两个 4 位输出端，分别表示加法运算结果和需要传递到下一部分的进位结果。

四位全加器在电路逻辑上可以视为四个一位全加器的串联组成，前一位的一位全加器的 C_o 端接入后一个一位全加器的 C_i 端，使得前一位的进位结果可以传递到下一个一位全加器中。然后将第一个一位全加器的 C_i 端当做整个四位全加器的 C_i 端，最后一个一位全加器的 C_o 端当做整个四位全加器的 C_o 端。并且将四位全加器的两个 a 和 b 输入分解成 4 个数字，分别分配给四个一位全加器，并把一位全加器的结果合并成一个 4 位二进制数作为输出 s

3 功能实现

3.1 一位全加器功能实现

根据 2.1 节中的输出端逻辑表达式，我们可以通过 Verilog 语言自带的 `and`, `or`, `xor` 逻辑运算单元实现代码表达，如下

```
1 wire s1, c1, c2, c3;  
2 and (c1, a, b),  
3      (c2, b, ci),
```

```

4      (c3, a, ci);
5
6  xor   (s1, a, b),
7      (s, s1, ci);
8
9  or    (co, c1, c2, c3);

```

完整的设计实现代码参见附录 A.1。

3.2 四位全加器功能实现

根据 2.2 节内容，我们可以将四个一位全加器串联从而实现四位全加器的功能。具体代码实现如下：

```

1  wire [2:0] ct;
2  adder_1bit a1(.a(a[0]), .b(b[0]), .ci(ci),
3  .s(s[0]), .co(ct[0])),
4      a2(.a(a[1]), .b(b[1]), .ci(ct[0]),
5      .s(s[1]), .co(ct[1])),
6      a3(.a(a[2]), .b(b[2]), .ci(ct[1]),
7      .s(s[2]), .co(ct[2])),
8      a4(.a(a[3]), .b(b[3]), .ci(ct[2]),
9      .s(s[3]), .co(co));

```

完整的代码实现参见附录 A.2。

4 结果验证

我们使用 Verilog 编写激励文件，采用软件仿真的形式对于四位全加器进行测试（代码实现参见附录 B）。用多组数据测试，结果如图 1 所示。从图 1 中我们可以看出，四位全加器的功能实现正确。

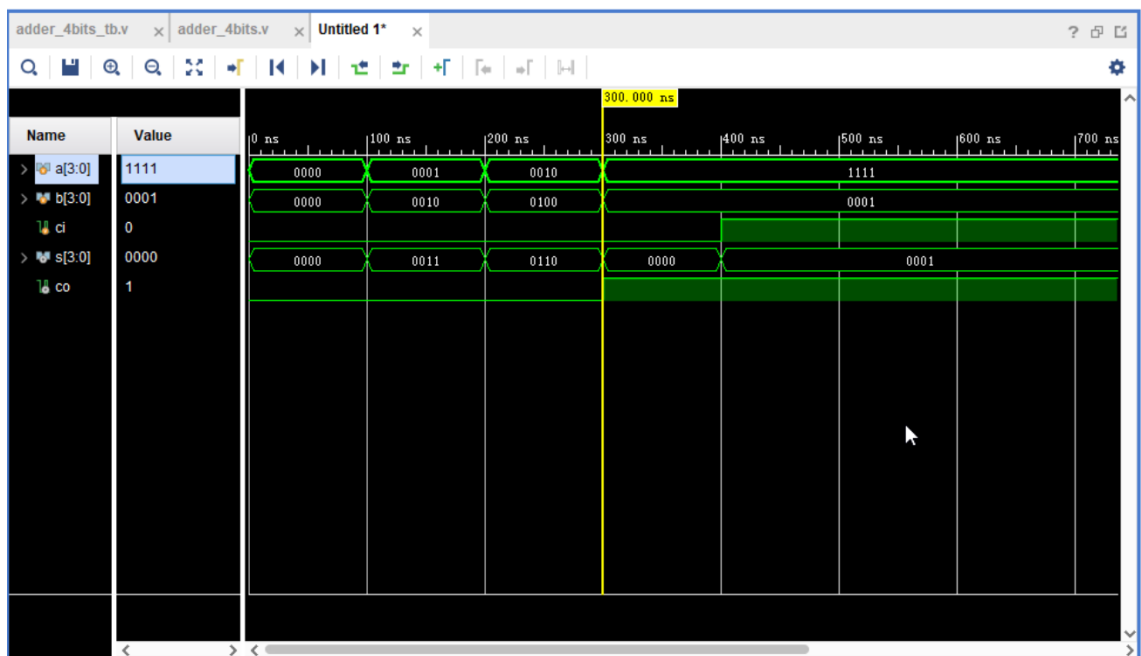


图 1: 四位全加器的测试结果图

5 总结反思

本实验实现了 FPGA 实验中四位全加器这一基础器件的设计与仿真测试。实验思路是从简至繁，自底向上，首先通过 Verilog 语言自带的逻辑元件进行操作，根据一位全加器的逻辑真值表模拟构建出有相同真值表的程序表达式，然后再对一位全加器进行串联，实现了四位全加器的功能结果。通过这次实验，我对于 Vivado 这一软件工具使用更加熟练，并对 Verilog 语言中的逻辑运算有了一点了解，以支撑我继续向下深入学习这种硬件逻辑语言。

除此之外，这种化繁为简，将一个复杂功能拆分成多个简单小功能组合的思想十分值得我去学习，可以运用到以后课程实验的思考与设计之中。让设计复杂功能器件成为可能。

6 致谢

感谢本次实验中指导老师在课程微信群里为同学们答疑解惑；
感谢上海交通大学网络信息中心提供的远程桌面资源；
感谢计算机科学与工程系相关老师对于课程指导书的编写以及对于课程的设计，让我们可以更快更好地学习相关知识，掌握相关技能；
感谢电子信息与电气工程学院提供的优秀的课程资源。

A 设计文件完整代码实现

A.1 一位全加器的代码实现

参见代码文件 `adder_1bit.v`。

A.2 四位全加器的代码实现

参见代码文件 `adder_4bits.v`。

B 激励文件完整代码实现

参见代码文件 `adder_4bits_tb.v`。