

计算机系统结构实验报告 实验 2

颜培深 518030910094

2020 年 5 月 17 日

摘要

本实验首先实现了一个一位全加器，然后利用一位全加器设计了一个四位全加器，在加法器中仅利用与、或、异或三个操作，来仿真电路中的三种逻辑元件，达到实现四位加法功能的目的，实验通过软件仿真的方式进行结果验证。

目录

目录	1
1 实验目的	2
2 原理分析	2
2.1 一位全加器的设计	2
2.2 四位全加器的设计	2
3 功能实现	3
3.1 一位全加器的实现	3
3.2 四位全加器的实现	3
4 结果验证	4
5 总结与反思	4

1 实验目的

本次实验给出实验要求：

1. 熟悉 Xilinx 逻辑设计工具 Vivado 的基本操作
2. 掌握使用 VerilogHDL 进行简单的逻辑设计
3. 理解一位加法器以及四位加法器的工作原理
4. 使用功能仿真验证实现正确性

2 原理分析

2.1 一位全加器的设计

一位全加器模块包括三个输入 a, b, ci 和两个输出 s, co ，分别对应两个来自本位的输入和一个来自上一位的进位输入，两个输出分别是本位的输出和对下一位的进位输出。对应的真值表如表1所示。

a	b	ci	s	co
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

表 1: 一位全加器真值表

从而两个输出的对应逻辑计算公式为：

$$\begin{cases} co = (a \& b) | (a \& ci) | (b \& ci) \\ s = a \oplus b \oplus ci \end{cases}$$

2.2 四位全加器的设计

四位全加器可以看作是四个一位全加器的一个组合，我们可以定义四个一位全加器 A_0, A_1, A_2, A_3 ，将两个四位二进制数的第 0 位到第 3 位分别送入 A_0 到 A_3 ，然后将前一位的进位输出送入后一位的进位输入中，将最高位的进位输出看作是四位全加器的进位输出即可。原理图如图1所示。

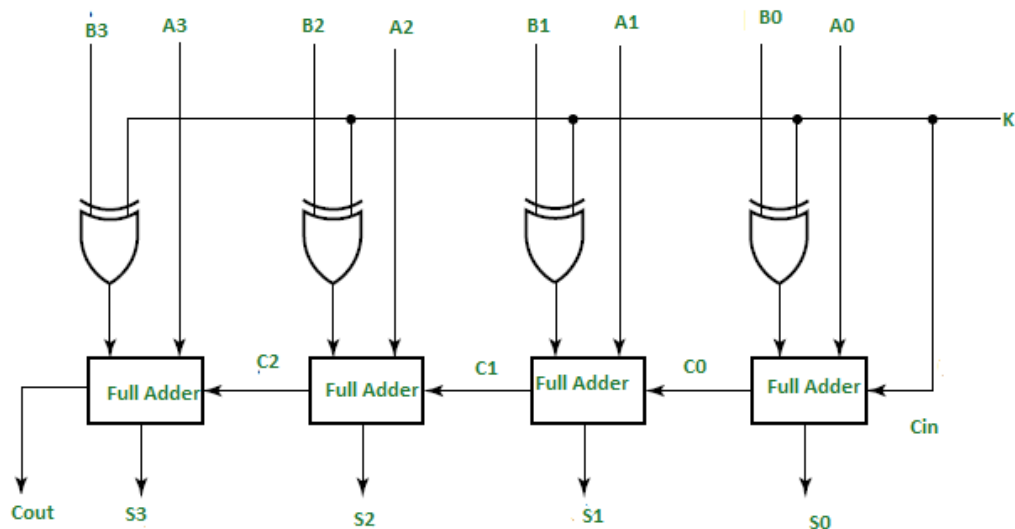


图 1: 四位全加器示意图

3 功能实现

3.1 一位全加器的实现

一位全加器中三个输入的逻辑操作直接利用代码中逻辑与、逻辑或、逻辑异或来表示，实现部分代码如下：

```

1 wire s1, c1, c2, c3;
2 and (c1, a, b),
3     (c2, b, ci),
4     (c3, a, ci);
5 xor (s1, a, b),
6     (s, s1, ci);
7 or (co, c1, c2, c3);

```

3.2 四位全加器的实现

四位全加器在组合前需实例化四个一位全加器，将两个四位二进制数每一位分别送入对应的一位全加器，然后将每一位的全加器进位输出接到下一位的全加器进位输入，实现部分代码如下：

```

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

```

4 结果验证

对于本实验中实现的四位全加器采用软件仿真的方法进行测试，用 Verilog 语言编写测试激励文件，进行仿真，观察两个输入的四位二进制数以及对应的四位二进制输出的数值变化，比对是否符合预期。

仿真得到的波形如图2所示。

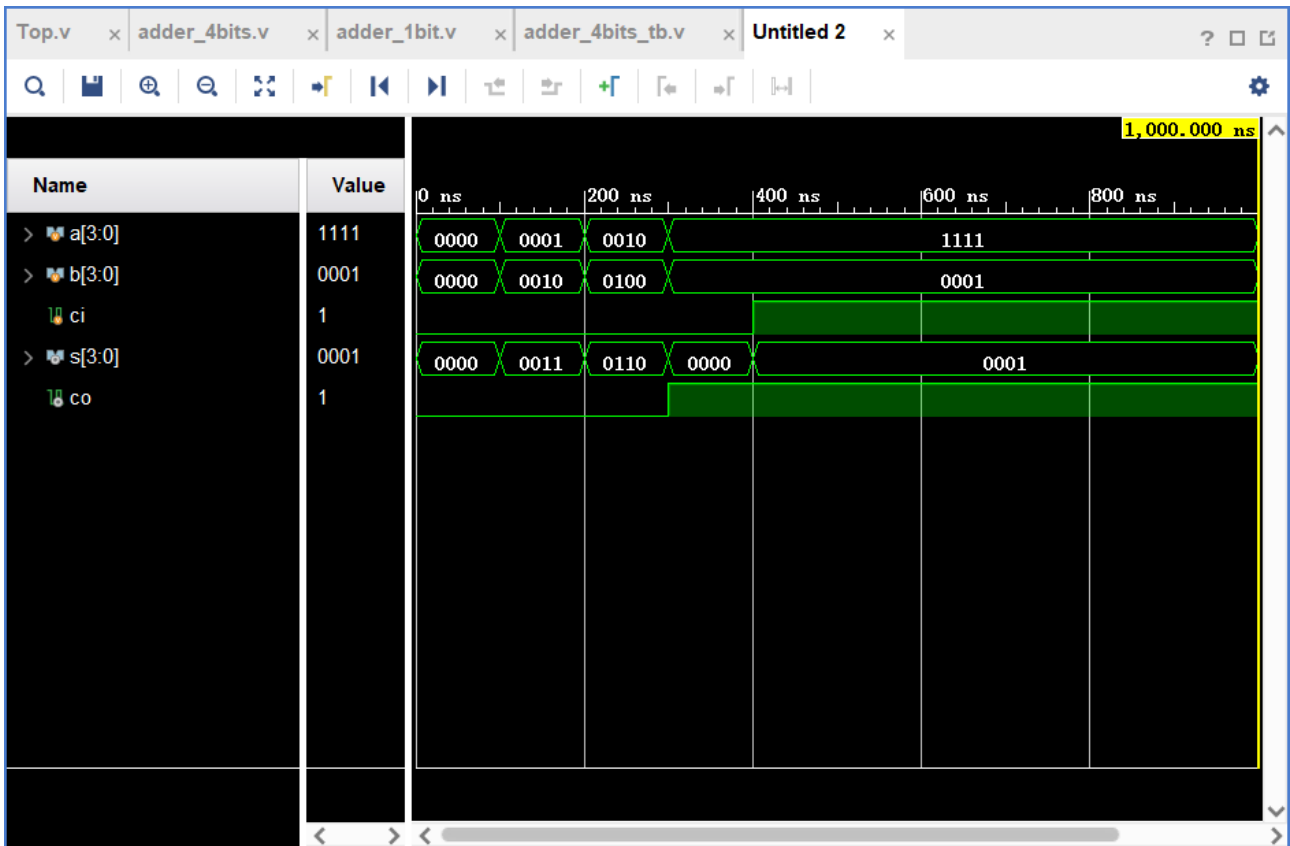


图 2: 四位全加器仿真波形图

通过比对数值发现四位全加器功能基本实现，同时在实验最后我们令该四位全加器的进位输入为 1 时也成功地计算出了最终结果，向我们证明了通过设计我们可以将四位加法器无限制地扩展到任意位数，证明本次实验验证成功。

5 总结与反思

本实验实现了四位全加器的简单仿真，对于四位全加器的基础：一位加法器的工作原理及硬件实现途径有了更好的了解，学会了如何利用与、或、异或三个操作来实现加法本位输出运算以及进位输出的运算，同时加深了对于仿真激励文件中延时操作多种写法的理解。借助这种化繁为简、以一推四的思想，我们同样的可以设计多位运算的乘法器，只是在基础一位运算器的基础上需要加入一些重要的辅助模块，有机会可以试试看。