实验五 EDA 作业二

电 25 吴晨聪 2022010311

# 实验目的

1. 熟练掌握面向 FPGA 的简单数字系统的设计流程。
2. 学习编写测试文件对设计电路进行仿真验证。
3. 熟悉实验装置——实验板，掌握板上外设的工作原理。

# 预习任务

1. 根据实验任务中的步骤提示，写出要用到的电路模块及其功能。

(1)1位二进制全加器——完成两个二进制位及进位的运算；

(2)2位二进制全加器——由两个1位二进制全加器构成，可以实现两个2位二进制位及1个进位的运算；

(3)4位二进制全加器——由四个1位二进制全加器构成，可以实现两个4位二进制位及1个进位的运算；

(4)原码转补码——由于原码不可以直接进行加减运算，因此必须要将输入的原码转换为补码；

(5)补码转原码——输出应为原码，因此要把四位全加器输出的补码转换为原码；

(6)二进制运算器——可以完成两个3位带符号二进制数的运算并输出反码；

(7)二选一数据选择器——使用数据选择器控制7448“灭零”引脚是否“灭零”；

(8)四选一数据选择器——使用数据选择器控制亮起的LED管；

(9)显示译码器——利用系统自带的显示译码器芯片并搭配部分自己设计的逻辑达到根据时钟信号的变化轮流显示4个值的效果。

1. 阅读网络学堂中的“FPGA 实验板说明书”了解实验板上的外设资源，并掌握其工作原理。
2. 进入实验室之前，可预先完成电路的设计输入。

# 实验内容

在可编程逻辑器件上设计一个运算电路，可以实现S＝M+N。M和N为3位二进制数，其中1位是符号位，2位是有效数字。要求用原理图的输入方式完成。

用实验板上的拨码开关模拟运算数（原码输入），用发光二极管表示运算数

的正负标志；用数码管显示运算数、运算结果（原码）及运算结果的正负标志。

具体内容及步骤如下：

## 用门电路设计一个1位二进制全加器，将其封装成1位全加器模块。

定义A、B为加数，CI为来自低位的进位，S为和，CO为高位进位。可列出1位全加器的真值表:

表 1 1位全加器真值表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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 |

可以得出输入输出逻辑式

逻辑电路图如图所示：

一張含有 文字, 圖表, 螢幕擷取畫面, 行 的圖片

自動產生的描述

图 1 1位全加器模块设计

一張含有 文字, 數字, 字型, 行 的圖片

自動產生的描述

图 2 1位加法器功能验证

一張含有 文字, 螢幕擷取畫面, 字型, 圖表 的圖片

自動產生的描述

图3 封装成1位全加器模块

## 以1中已封装的1位全加器模块为基础实现一个4位二进制全加器，并仿真检查功能正确与否（仿真工具不限）。

利用上述1位全加器模块，将四个1位全加器连接成4位全加器。将每个1位全加器的进位输出连接到下一个1位全加器的进位输入，确保进位的连续传递。

一張含有 文字, 圖表, 螢幕擷取畫面, 數字 的圖片

自動產生的描述

图 4 4位全加器模块设计

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

自動產生的描述

图 5 4位加法器功能验证

结果正确，说明各个全加器搭建正确。

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述

图6 封装成4位全加器模块

## 以2中的4位全加器模块为基础实现一个二进制运算器，可以完成运算S＝M+N编写测试文件，使用Modelsim仿真验证运算器的功能。

原码转补码模块:

一張含有 文字, 圖表, 行, 方案 的圖片

自動產生的描述

图 7 原码转补码模块设计

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

自動產生的描述

图 8 原码转补码功能验证

注意输入“100”时的输出情况，此时输出是“0000”符合要求。

补码转原码模块:

一張含有 文字, 圖表, 數字, 行 的圖片

自動產生的描述

图 9补码转原码模块设计

一張含有 文字, 螢幕擷取畫面, 數字, 行 的圖片

自動產生的描述

图 10 补码转原码功能验证

结果正确，说明补码转原码模块搭建正确。

二进制运算器模块:

输入为A（3位原码）、B（3位原码）。使用原码转补码模块，对A和B进行原码到补码的转换。再使用四位全加器，对补码的A和B进行二进制加法运算。最后，使用补码转原码模块，对S进行补码到原码的转换，得到运算结果。

一張含有 文字, 圖表, 數字, 行 的圖片

自動產生的描述

图 11二进制运算器模块设计

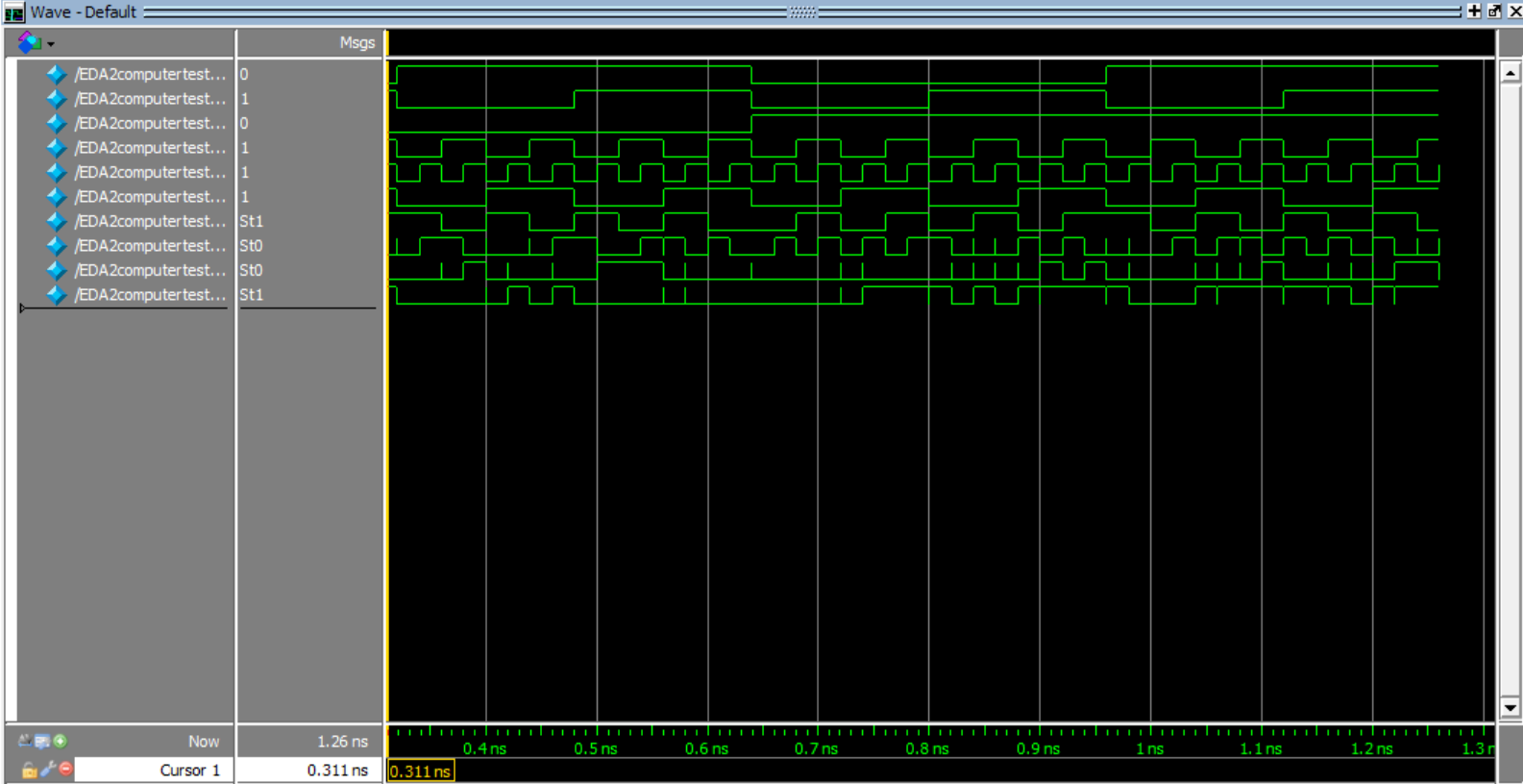


图 12 二进制运算器功能验证

结果正确

## 设计一个4位数码管的扫描显示电路。

(1) 将运算器的两个运算数和运算结果根据拨码开关 DIP1、DIP2的状态在4位数码管上轮流显示，如表1所示。（提示：可以选用库中的译码器 7448或自行设计译码器）与此同时，运算数的正负标志在发光二极管上显示。

(2) 运算结果的正负标志在数码管1显示时，正数无显示、负数显示“—”。

一張含有 文字, 字型, 數字, 行 的圖片

自動產生的描述

顶层电路图:

一張含有 文字, 圖表, 數字, 平行 的圖片

自動產生的描述

图 13 4位数码管的扫描显示电路顶层电路图

红圈内部分: 实现根据DIP1、DIP2拨码开关状态决定4位数码管上DIG1~3哪一个会亮起。

输入为DIP1、DIP2；输出为DIG1~3，直接用逻辑门实现，输出为 1 时数码管亮起，输出为 0 时数码管熄灭。

红圈外部分: 实现根据DIP1、DIP2拨码开关状态决定4位数码管上DIG1~3所显示的内容。

(1) 输出M、N、S

D0、D1、D2、D3分别对应S、符号位、N、M，使用四个四选一数据选择器，其输入端分别连结上述对应端口，并把输出接7448的A~D输入端，四选一数据选择器不使用的输入端接地处理。

7448的𝐿TN引脚接 Vcc，BIN’ 引脚，将其与一个二选一数据选择器相连，在DIP1、DIP2输入不为10时接 Vcc，直接输出。

7448的OA~OF引脚直接输出，而OG引脚因要输出符号位，需要与一个二选一数据选择器相连，在DIP1、DIP2输入不为10时接 Vcc，直接输出。

(2) 输出符号位

需要输出符号位时，DIP1、DIP2为10，此时，四个数据选择器输出结果都为 0，因此 7448 的A~D输入端也都为 0，7448 的BIN’引脚，在DIP1、DIP2为10时接地，因此 OA~OF 对应的数码管不会亮起，OG输出为加法器的符号位。

一張含有 文字, 螢幕擷取畫面, 圖表, 數字 的圖片

自動產生的描述

图 14 4位数码管的扫描显示电路功能验证

DIP1、DIP2为00时，DIG3亮起，010+011=5，M=5: acdfg亮起。

DIP1、DIP2为01时，DIG2亮起，010+110=0，N=2: abdeg亮起。

DIP1、DIP2为10时，DIG1亮起，011+100=3，符号位= +: 全灭。

DIP1、DIP2为11时，DIG0亮起，100+110=0，S=0: abcdef亮起。

# 遇到的问题及解决方法

1. 开始实现M+N=S加法器时并未关注到M、N输入是原码，因此并未搭建封装原码转补码，补码转原码的模块。发现该问题后搭建了能够避免输出负零的转换模块。

2. 在编写测试文件的时候发现若把输入的M、N赋予3位二进制位的空间，那么如果利用循环for(m=0;m<8;m=m+1)的时候永远无法跳出循环，原因是for循环在执行完m=7的循环之后还会将m的数值加1来判断是否还符合循环的条件，而此时m再加1就会变回000从而永远无法跳出循环。合适的解决方法就是直接为M、N赋值，不要使用for循环编写测试文件。