

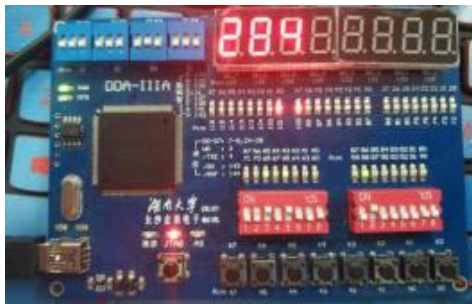
三位数码管实验日志与报告

一、日志

1.1、3 位数码管实验下载：（2021.3.2 完成）

操作说明：首先按照实验要求安装 quartus II。然后下载并安装 USB-blaster 驱动。连接实验板，打开 quartus II，点击 **【Tool】->【programmer】** 命令。单击 **【Add File】** 加入 **【scan_led3.sof】**，单击 **【Hardware setup】** 按钮，在硬件设备列表里选择 **【USB_Blaster】** 并双击，选择完成后返回上一界面，单击 **【Start】**。

现象描述：下载后的结果如图所示：



实际上在实际中是一种频闪的状态，这是由于 clk3 的时钟信号的频率只有 1Khz，已经可以产生视觉暂留。可以通过下方的拨码开关对显示数值进行调整，拨码开关每四位一组，满足“8421”。修改左上方的时钟信号的档位可以对时钟信号的频率进行微调，发现是进行频闪，且顺序为从左到右依次闪烁。

1.2、示波器测量位选信号和时钟信号：

时钟信号：



其中的黄色是时钟信号，可以看出时钟信号是一个十分规整的方波，占空比大约为 50%，频率为 1KHZ，这与时钟信号的特性有关。

位选信号：



可以看出位选信号也是一个方波，但占空比不一定为 50%，而是 60%，这是因为我们这里是三个数码管循环显示，在计数器的时候产生的是 0,1,2,3，其比值为 2:1:1:1，其中 0 表示了一个数码管，1 表示了一个数码管，2,3 共同表示了一个数码管，导致最终的三个数码管的时间之比为 2:1:2。其频率为 200HZ。

二、报告

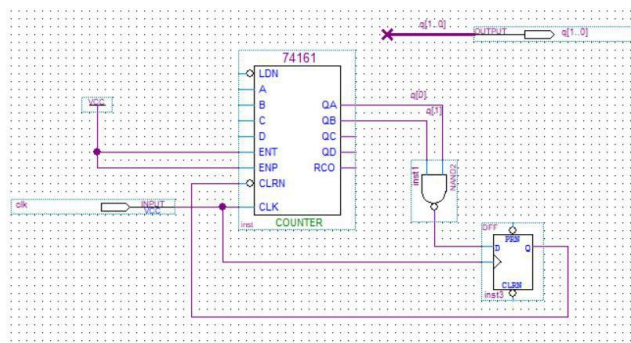
2.1、实验目的：

目标：熟悉 quartusII 的数字系统设计流程,DDA 系列数字系统实验平台的使用，图形输入、文本输入（硬件描述语言）、层次设计的过程，三位数码管的使用方法、代码编写和引脚设计，以及图形输入的注意事项和画图技巧。

资源：windows 10 专业版的荣耀MagicBook,DDA-IIIA CyloneII EP2C5T144C8N quartus-II 软件。

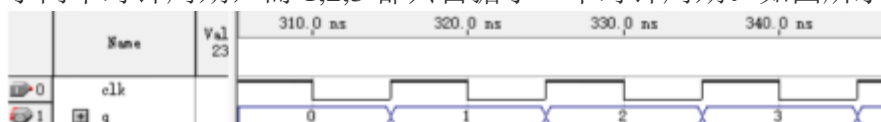
2.2、设计

2.2.1、模 4 计数器电路图：



Node Name	Direction	Location	I/O Bank	VREF Group
clk	Input	PIN_91	3	B3_N0
q[1]	Output	PIN_118	2	B2_N0
q[0]	Output	PIN_115	2	B2_N0

如图所示，即为一个模 4 计数器，其中由于仅仅是用来计数，所以 a~d 不需要输入，而由于是用来进行连续的输出，所以直接将两个使能端 ent 和 enp 设置为 vcc 高电平，使其永远成立。而 qa 和 qb 是输出端，其中 qa 是低位，qb 是高位，并进行输出。取 nand 后当且仅当两者全为 1 的时候会输出 0，而当他输出 0 的时候触发器 dff 会产生一个时钟周期的停留，所以展示出来的结果就是数字 0 占据了两个时钟周期，而 1,2,3 都只占据了一个时钟周期。如图所示：



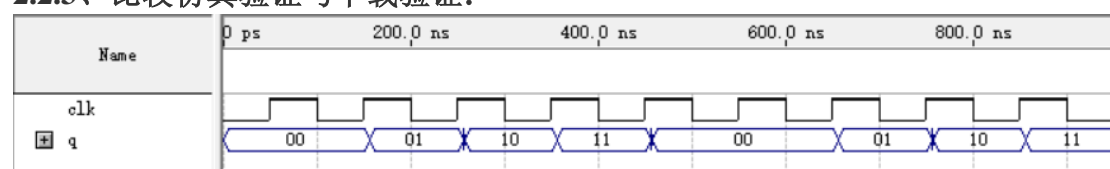
2.2.2、模 4 计数器下载现象描述:

首先设置引脚, 这里为了可以直观显示, 将始终的引脚设置为 91, 将输出的引脚设置为二极管 115、118, 然后进行全编译, 再进行下载, 下载结果为:



可以看出在设置时钟信号为最低档的时候, 可以很明显的看到二极管 118 和 115 在交替闪烁, 闪烁的规律大致为 00 (全灭) -> 01 -> 10 -> 11 (全亮), 并且 00 的时间明显要长于其他情况。闪烁的规律与仿真波形符合, 由此可以证明电路行为正确, 并得出结论: 74161 模块是一个计数模块, 通过 `clrn` 端口进行置零, 并进行计数, 最高可以达到 15。

2.2.3、比较仿真验证与下载验证:



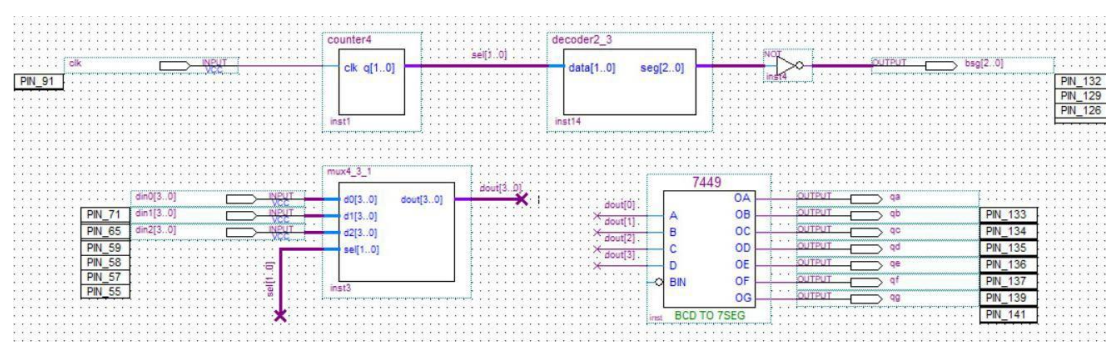
仿真验证分为功能验证和时序验证。时序仿真会有延迟但是更加符合真实情况。但是实验时一般采用功能仿真。

下载验证是将所写好的 `mof` 文件通过 USB 接口下载到板子上, 然后板子开始运行程序, 通过观察板子上出现的现行, 比方说二极管发光, 数码管显示等来变相观察代码是否正确。

总的来说, 一般先进行仿真验证再进行下载验证, 一方面是因为通过仿真波形图可以比实际下载更轻松更直接的找出可能出现的错误; 另一方面是因为如果程序本身出现了结果上的问题, 直接下载到板子上可能会导致板子出现错误, 发生损坏等。

2.3、实现

2.3.1、顶层电路



Node Name	Direction	Location	I/O Bank	VREF Group
bsg[2]	Output	PIN_132	2	B2_N1
bsg[1]	Output	PIN_129	2	B2_N1
bsg[0]	Output	PIN_126	2	B2_N0
clk	Input	PIN_91	3	B3_N0
din0[3]	Input	PIN_71	4	B4_N0
din0[2]	Input	PIN_70	4	B4_N0
din0[1]	Input	PIN_69	4	B4_N0
din0[0]	Input	PIN_67	4	B4_N0
din1[3]	Input	PIN_65	4	B4_N0
din1[2]	Input	PIN_64	4	B4_N0
din1[1]	Input	PIN_63	4	B4_N0
din1[0]	Input	PIN_60	4	B4_N0
din2[3]	Input	PIN_59	4	B4_N0
din2[2]	Input	PIN_58	4	B4_N1
din2[1]	Input	PIN_57	4	B4_N1
din2[0]	Input	PIN_55	4	B4_N1
qa	Output	PIN_133	2	B2_N1
qb	Output	PIN_134	2	B2_N1
qc	Output	PIN_135	2	B2_N1
qd	Output	PIN_136	2	B2_N1
qe	Output	PIN_137	2	B2_N1
qf	Output	PIN_139	2	B2_N1
qg	Output	PIN_141	2	B2_N1

可以看出整个电路大致是通过四个模块组成，左上角的是一个 counter4 模块，即上文的模 4 计数器；右上角的是一个 2-3 译码器，将模 4 计数器产生的两位信号进行译码，译码成一个三位的信号用来控制三个位选信号，以此来控制三个数码管进行显示；左下角是一个 4-3-1 模块，是自己写出来用来通过拨码开关对数码管上的显示数字进行控制的模块，其产生的数据 dout 将会传给右下角的模块；右下角的模块是一个 7449 芯片，用来进行 4-7 的 BCD 译码，以此来对段选信号进行控制，通过从 ABCD 输入的一个四位二进制数，对其进行编译解码，是之可以在数码管上进行显示，比方说数字 0 展示在数码管上就要求段选信号 a~g 的值是 1111110。

2.3.2 下载实作截图

结果如图：

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
a	
b	
c	
d	
e	
f	

以下为输入 0~f 时的符号显示结果：

输入数值	译码情况	段选信号
0	(0000)	1111110
1	(0001)	0110000
2	(0010)	1101101
3	(0011)	1111001

4	(0100)	0110011
5	(0101)	1011011
6	(0110)	0011111
7	(0111)	1110000
8	(1000)	1111111
9	(1001)	1110011
a	(1010)	0001101
b	(1011)	0011001
c	(1100)	0100011
d	(1101)	1001011
e	(1110)	0001111
f	(1111)	0000000

2.3.3 优化

这里可以很明显的看出，在 quartusII 里面的 7449 芯片不能完全处理 0-f 的所有情况，所以需要在下面对电路图进行改进，需要改进显示 6,9 以及 a-f 的情况，在这里我通过设置新的模块来达到优化的目的。

由于需要在不删除 7449 芯片的时候进行优化，所以想到可以自己新建一个模块，模块将判断是否可以使用 7449 芯片，以及在 7449 芯片不能使用的时候进行补充替换。由此先得到 6,9 以及 a~f 的段选信号表示：

输入数值 段选信号值

6 (0110)	1011111
9 (1001)	1111011
a (1010)	1110111
b (1011)	0011111
c (1100)	1001110
d (1101)	0111101
e (1110)	1001111
f (1111)	1000111

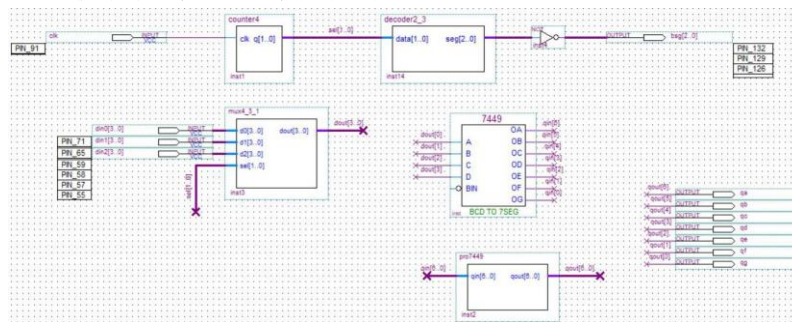
设计思路：这里是通过直接加入一个 youhua 模块，当 d（数据）的值不是 6,9,af 的时候，就直接输出输入的数据，将数据导入到 7449 模块中进行译码；而如果满足 6,9,af 的时候，就直接在该模块中进行译码。同时输出一个 dd 判断信号，用来判断是通过那个模块得到的数值，再通过该判断信号对三态门进行控制，将两个模块所得到的值进行汇总融合便可以达到 0~f 全显示的目的。

```

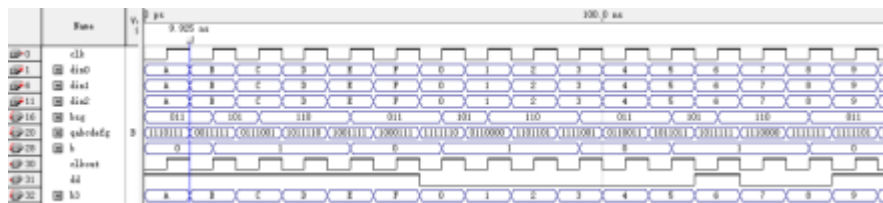
library ieee;
use ieee.std_logic_1164.all;
entity youhua is
port (din:std_logic_vector(3 downto 0);
      d0,d1,d2,d3,dd,da0,da1,da2,da3,da4,da5,da6:out_std_logic);
end youhua;
architecture rtl of youhua is
begin
  process(d)
  begin
    if(d="0110") then
      dd<='1';
      da6<='1';
      da5<='1';
      da4<='1';
      da3<='1';
      da2<='1';
      da1<='0';
      da0<='1';
    elsif(d="1001") then
      dd<='1';
      da6<='1';
      da5<='1';
      da4<='1';
      da3<='0';
      da2<='0';
      da1<='0';
      da0<='1';
    elsif(d="1010") then
      dd<='1';
      da6<='1';
      da5<='1';
      da4<='1';
      da3<='0';
      da2<='1';
      da1<='1';
      da0<='1';
    elsif(d="1011") then
      dd<='1';
      da6<='1';
      da5<='1';
      da4<='1';
      da3<='1';
      da2<='1';
      da1<='1';
      da0<='1';
    elsif(d="1100") then
      dd<='1';
      da6<='1';
      da5<='1';
      da4<='1';
      da3<='1';
      da2<='0';
      da1<='0';
      da0<='1';
    elsif(d="1101") then
      dd<='1';
      da6<='1';
      da5<='1';
      da4<='1';
      da3<='1';
      da2<='1';
      da1<='1';
      da0<='1';
    elsif(d="1110") then
      dd<='1';
      da6<='1';
      da5<='1';
      da4<='1';
      da3<='1';
      da2<='1';
      da1<='1';
      da0<='1';
    elsif(d="1111") then
      dd<='1';
      da6<='1';
      da5<='1';
      da4<='1';
      da3<='1';
      da2<='1';
      da1<='1';
      da0<='1';
    end if;
  end process;
  d1<=d(1);
  d2<=d(2);
  d3<=d(3);
end rtl;

```

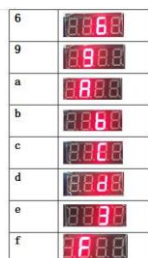

经过优化后最终电路图如下：



仿真结果为：



优化后的下载实作截图：



通过下载的结果可知，符合仿真结果，实验正确。

三、总结

3.1、学到的知识点：

学习到了对数码管的使用方法，以及段选信号和位选信号的作用究竟是什么。可以自己使用数码管进行简单的编译和制造调用，为以后的多种输出，输出数量超过了板子所能承受的最大数量的时候，可以通过数码管进行结果展示。还熟悉了下载的方法。

3.2、掌握的技能：

学会了quartusII的数字系统设计流程，DDA系列数字系统实验平台的使用，图形输入、文本输入（硬件描述语言）、层次设计的过程，熟悉了三数码管的使用方法、代码编写和引脚设计，以及图形输入的注意事项和画图技巧。

3.3、人文：

相比以前的实验，现在的实验更加注重的自学能力以及合作能力。老师只是起一个引导作用，对于实验知识点的理解和实际操作都需要自己认真学习。实验过程总是会出现许多不可预料的错误，此时就体现了团队合作的重要性。

3.4、使用实验室过程中的安全责任方面的感悟：

在使用实验仪器之前，一定要了解实验仪器的性能特性、熟练掌握操作方法和操作技巧，严格按照老师要求进行。在离开实验室前，一定要关灯、关门、关空调、将桌椅归位、带走垃圾。