

数字逻辑实验

点亮数字人生

实验报告

数字逻辑实验 A 班 计 86

任一 2018011423

2020 年 3 月 19 日

实验环境	
操作系统:	Windows10 家庭版 18362.72
QuartusII 版本:	Quartus II 13.0 sp1
ModelSim 版本:	Modelsim SE-64 10.7

1 实验概述

本实验实现了使用 VHDL 语言，控制 FPGA 以显示数字数列，分别能够显示自然数列 (16 进制与 10 进制)、偶数列和奇数列。在本地仿真后，本实验在 JieLabs 在线实验平台上进行了进一步的仿真和测试，达到了预期结果。

其中，DigitalLife 文件夹下为本地测试的实验文件。在本文件夹的 DigitalLife.vhd 文件中，我实现了自然数数列、偶数列、奇数列对应的带译码的数码管和不带译码的数码管共 6 种输出。DigitalLife_tb.vhd 则是用来测试的 testbench 文件。

此外，DigitalLife_Online 文件夹下，有 JieLabs 在线实验平台上，本实验使用的仿真代码。考虑到在线实验平台的 FPGA 只有 16 个可用的管脚，在本文件夹下的 VHDL 文件中，本实验选择实现了不带译码的自然数数列、带译码的偶数列和奇数列。

2 实验结果

2.1 仿真截图

2.1.1 DigitalLife(本地版本) 仿真结果

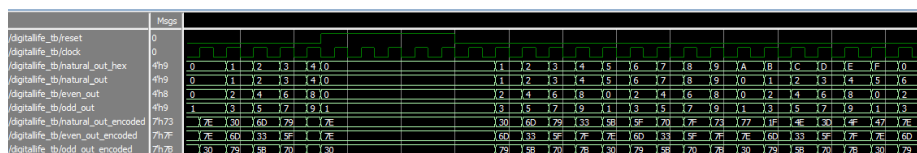


图 1: testbench 对 DigitalLife 文件夹下的文件进行的仿真结果波形图 (放大后清晰可见)

在上图中，可以看到随着 clock 信号每次上升沿的出现，各个数列 (十六进制自然数列，十进制自然数列、奇数列、偶数列) 都发生一次变化。在 reset 信号从 0 变为 1 时，各信号回到了初始状态，reset 信号从 1 变为 0 时，各信号继续随时钟变化而变化。其中后三个信号为编码后的数列，对应没有译码的数码管使用，在本次仿真中其数值没有具体含义。¹

¹ 为什么图中第一次遇到上升沿时，波形没有变化？详见 3.1 中的说明。

2.1.2 DigitalLife_Online 仿真结果

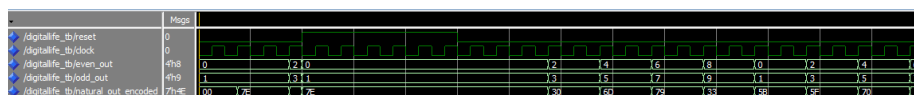


图 2: testbench 对 DigitalLife_Online 文件夹下的文件进行的仿真结果波形图 (放大后清晰可见)

在上图中, 可以看到随着 clock 信号每 2 次上升沿的出现 (每 2 个时钟周期进行一次数列值的更新), 各个数列 (奇数列、偶数列) 都发生一次变化。在 reset 信号从 0 变为 1 时, 各信号回到了初始状态, reset 信号从 1 变为 0 时, 各信号继续随时钟变化而变化。其中最后一个信号为编码后的自然数数列, 对应没有译码的数码管使用, 在本次仿真中其数值没有具体含义。²

2.2 JieLabs 测试结果

见录屏文件 JieLabsTest.mp4。在视频中展示了十六进制自然数数列、偶数列和奇数列。可看到大约每秒各数列都会发生变化。按下 reset 后可以看到各数列回到初始状态, 不按 reset 时各信号随时钟变化而变化。

3 思考与总结

3.1 遇到的问题与解决方法

在我第一次仿真时, 我遇到了奇怪的问题, 即我在代码中设置为在 clock 信号上升沿时, 更新数列和输出的值, 但是得到的波形图却是在每次 clk 信号下降沿时发生变化, 这令我十分不解。在参考了 StackOverflow 的解答并且与吕志远助教交流后, 我学习到一个 process 中 signal 赋值的过程, 只会在 process 结束时一起执行, 被赋予的值是所有 signal 在 process 进行前的值, 即所有信号在本轮 process 中, 值仍为上一轮的值。

因此, 在刚才发现的下降沿信号发生变化的问题中, 出现上升沿时, 该程序会进行数列临时变量的更新和输出信号的更新, 但在上升沿结束时输出信号仍然保持为上一轮的结果。在下降沿时, 不会有数列临时变量的更新, 有输出信号的更新, 这时输出信号更新为了上升沿本意要更新的数值, 因此

² 为什么图中第一次遇到 2 个上升沿时, 波形没有变化? 详见 3.1 中的说明。

出现了在下降沿时信号变化而上升沿中信号不便的问题。出现该问题的代码和波形图如下，注意第 27 行的代码和上面的注释。

```
1  entity DigitalLife is
2  port (
3      reset:in std_logic := '0';
4      clock:in std_logic := '0';
5      natural_out_hex:
6          out std_logic_vector(3 downto 0) := "0000"
7  );
8  end entity DigitalLife;
9  architecture bhv of DigitalLife is
10     signal natural_seq_hex :
11         std_logic_vector(3 downto 0) := "0000";
12 begin
13     process (clock, reset) begin
14         if (reset = '1') then
15             natural_seq_hex <= "0000";
16             natural_out_hex <= "0000";
17         elsif (rising_edge(clock)) then
18             if (natural_seq_hex = 15) then
19                 natural_seq_hex <= (others => '0');
20             else
21                 natural_seq_hex <= natural_seq_hex +1;
22             end if;
23         end if;
24         -- When falling edge, this also be activated and
25         -- the value of the last rising edge
26         -- is updated at falling edge.
27         natural_out_hex <= natural_seq_hex;
28     end process;
29 end architecture bhv;
```

这个问题我通过将输出信号的赋值加到了对于上升沿的判断分支中解决了，这样做不会在下降沿时触发赋值，只会在下一次上升沿时对输出变

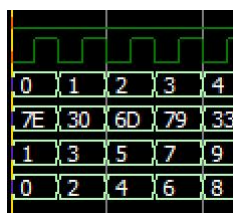


图 3: 下降沿时数据变化而上升沿时数据不变

量为上一次上升沿时的值。不过这样做也有一定的小问题，即在第一次遇到上升沿时，输出信号并不会变化，只会在第二次上升沿及其之后发生规律的变化。这也解释了 2.1.1 和 2.1.2 中脚注提出的两个小疑问。

3.2 一些感想与建议

非常感谢老师和助教在本次实验中给予我的帮助，无论是在群聊的答疑还是助教细心地与我私聊解决问题，都让我感受到了老师和助教的用心！

不过在 VHDL 入门、testbench 入门时，我感觉还是遇到了很大的困难。可能一方面我在面对生疏知识时接受没有那么迅速，另一方面可能也是关于 VHDL 的网络资源不是特别丰富。有可能的话，建议老师和助教为新入门的同学准备更多的参考资料或者学习路线，以帮助同学们更好地掌握这门课的知识，享受数字逻辑实验的乐趣。

4 参考资料

1. RISING EDGE AND FALLING EDGE PROBLEM

<https://stackoverflow.com/questions/50461404>

2. MODELSIM SIMULATION

<https://blog.csdn.net/u013273161/article/details/82454134>

3. TESTBENCH TUTORIAL

<https://vhdlguide.readthedocs.io/en/latest/vhdl/testbench.html#>

4. TESTBENCH CREATION

https://www.doulos.com/knowhow/perl/testbench_creation/

5. HOW SIGNAL ASSIGNMENT WORK IN PROCESS

<https://stackoverflow.com/questions/5060635/how-does-signal-assignment-work-in-a-proce>