

实验 4：数码管控制器

一、数码管控制器设计实现

1.1 本实验你的实现中有几个 Verilog 模块，列出具体的模块名和功能，含顶层模块
有 7 个 Verilog 模块，含 1 个顶层模块和 6 个子功能模块。

一、顶层模块（1 个）

top.v（顶层模块）

功能：作为整个系统的入口，例化所有子模块，连接各模块信号，处理全局控制逻辑。

关键职责：接入全局时钟、异步复位（S1）、数码管使能（SW0）、按键（S2/S3）等外部 IO。

将学号后两位固定数据、两个 S3 计数结果、0-30 循环计数结果拼接为 32 位 display 信号，送入 led_ctrl_unit。传递复位信号、使能信号至所有子模块，确保全局同步工作。

二、子功能模块（6 个）

1. debounce.v（消抖模块）

功能：处理按键机械抖动，输出稳定的按键信号。

适配场景：为 S3 按键的消抖计数功能提供稳定输入（DK3-DK2 显示对应的计数）。

2. edge_detector.v（边沿检测模块）

功能：捕捉按键信号的上升沿（或指定边沿），输出单周期脉冲。

适配场景：配合计数器模块，实现“按一次按键计数一次”的边沿触发逻辑（S3 按键计数）。

3. counter_0to99.v（两位十进制计数器模块）

功能：接收边沿信号触发计数，计数范围 00-99，支持复位。

适配场景：分为两个实例，分别实现 DK5-DK4（无消抖）和 DK3-DK2（带消抖）的按键计数显示。

4. decimal_counter.v（0-30 十进制计数器模块）

功能：以 0.1s 为计数间隔，循环计数 0-30，支持 S2 按键控制启停、异步复位。

适配场景：驱动 DK1-DK0 的显示，复位后自动启动，计数到 30 归零。

5. toggle_ff.v（触发翻转模块）

功能：接收 S2 按键的边沿信号，实现输出信号的“翻转”（0→1 或 1→0）。

适配场景：为 decimal_counter 提供启停控制信号，实现 S2“按一下暂停、再按一下继续”的逻辑。

6. led_ctrl_unit.v（数码管控制核心模块）

功能：接收 31:0 位待显示数据，生成位选信号（led_en）和段选信号（led_cx），实现 8 个数码管动态扫描显示。

适配场景：核心显示驱动模块，支持共阳极数码管，刷新频率 1-2ms（500-1000Hz）。

1.2 消抖的实现

贴出对应的代码，并简要说明实现方法和思路

```

16 always @(posedge clk or posedge rst) begin
17     if (rst) begin
18         counter <= 21'd0;
19         key_reg <= 1'b0;
20         key_stable <= 1'b0;
21         key_out <= 1'b0;
22     end else begin
23         key_reg <= key_in;
24
25         if (key_reg != key_stable) begin
26
27             counter <= 21'd0;
28             key_stable <= key_reg;
29         end else if (counter < DEBOUNCE_TIME) begin
30
31             counter <= counter + 21'd1;
32         end else begin
33
34             key_out <= key_stable;
35         end
36     end
37 end
38
39 endmodule

```

本模块采用“计数器计时法”实现按键消抖，思路是通过检测按键状态的持续稳定时间过滤机械抖动。

①信号同步：通过 `key_reg <= key_in` 将异步的按键输入信号同步到系统时钟域，避免亚稳态对后续逻辑的干扰。

②状态变化检测：比较同步后的输入（`key_reg`）与上一次记录的待验证状态（`key_stable`），若二者不同，说明按键可能在抖动，则立即重置计数器并更新待验证状态，重新开始计时。

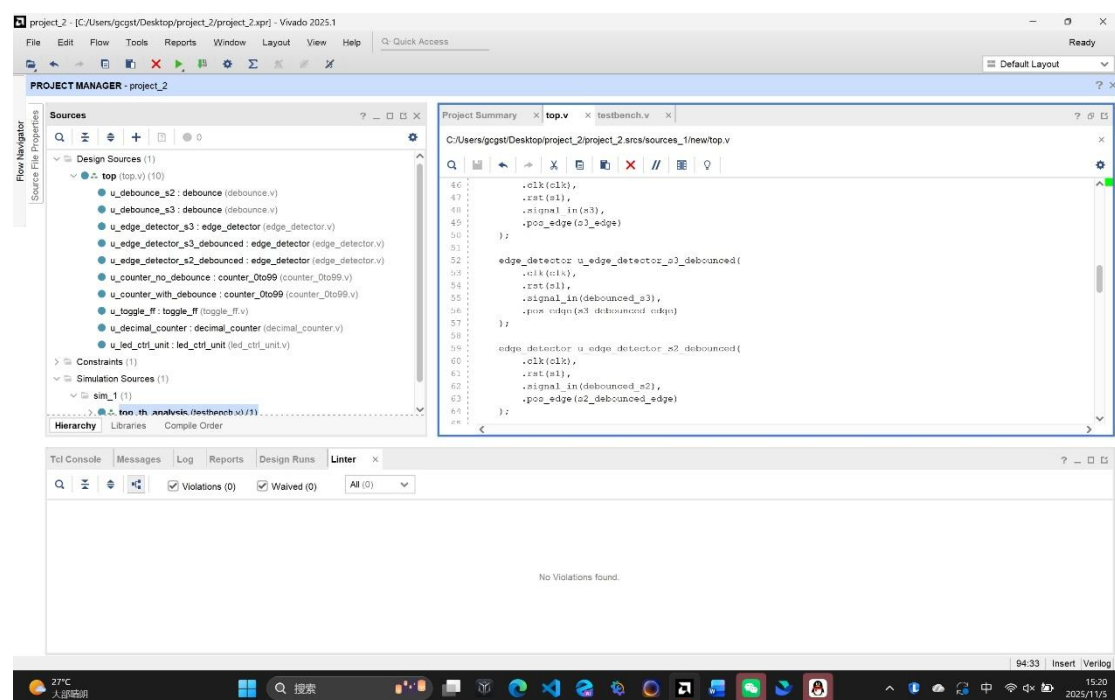
③稳定时长累计：若状态未变化（`key_reg == key_stable`），则计数器开始累加，记录状态持续稳定的时钟周期数。

④输出稳定信号：当计数器累加到预设阈值 `DEBOUNCE_TIME`（20ms）时，确认当前状态已稳定（非抖动），将待验证状态（`key_stable`）作为结果输出（`key_out`）。

通过上述逻辑，模块可过滤掉按键机械抖动，仅输出持续稳定 20ms 以上的有效状态，确保后续电路接收可靠信号。

1.3 Linter 报告截图

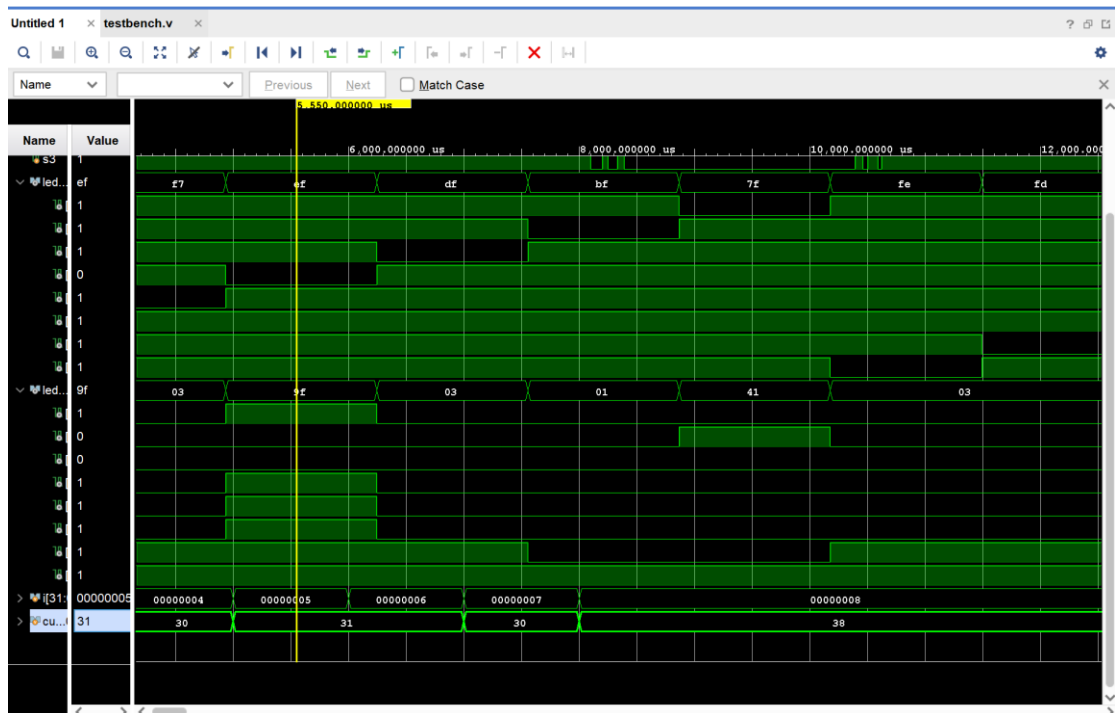
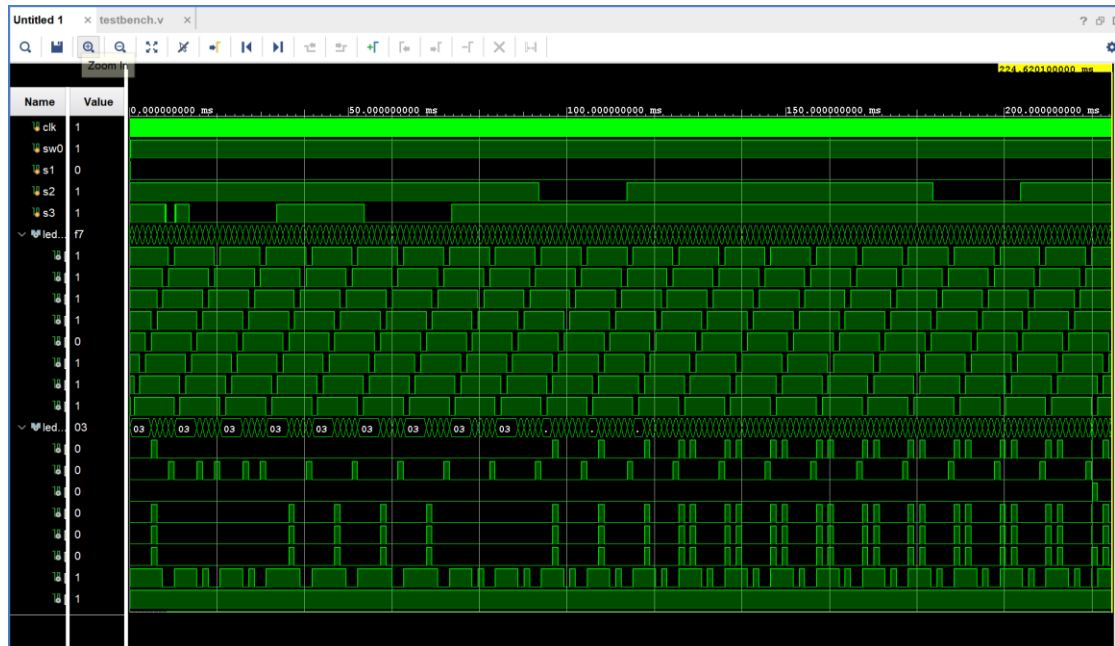
截图需体现是实验 4 的工程或代码

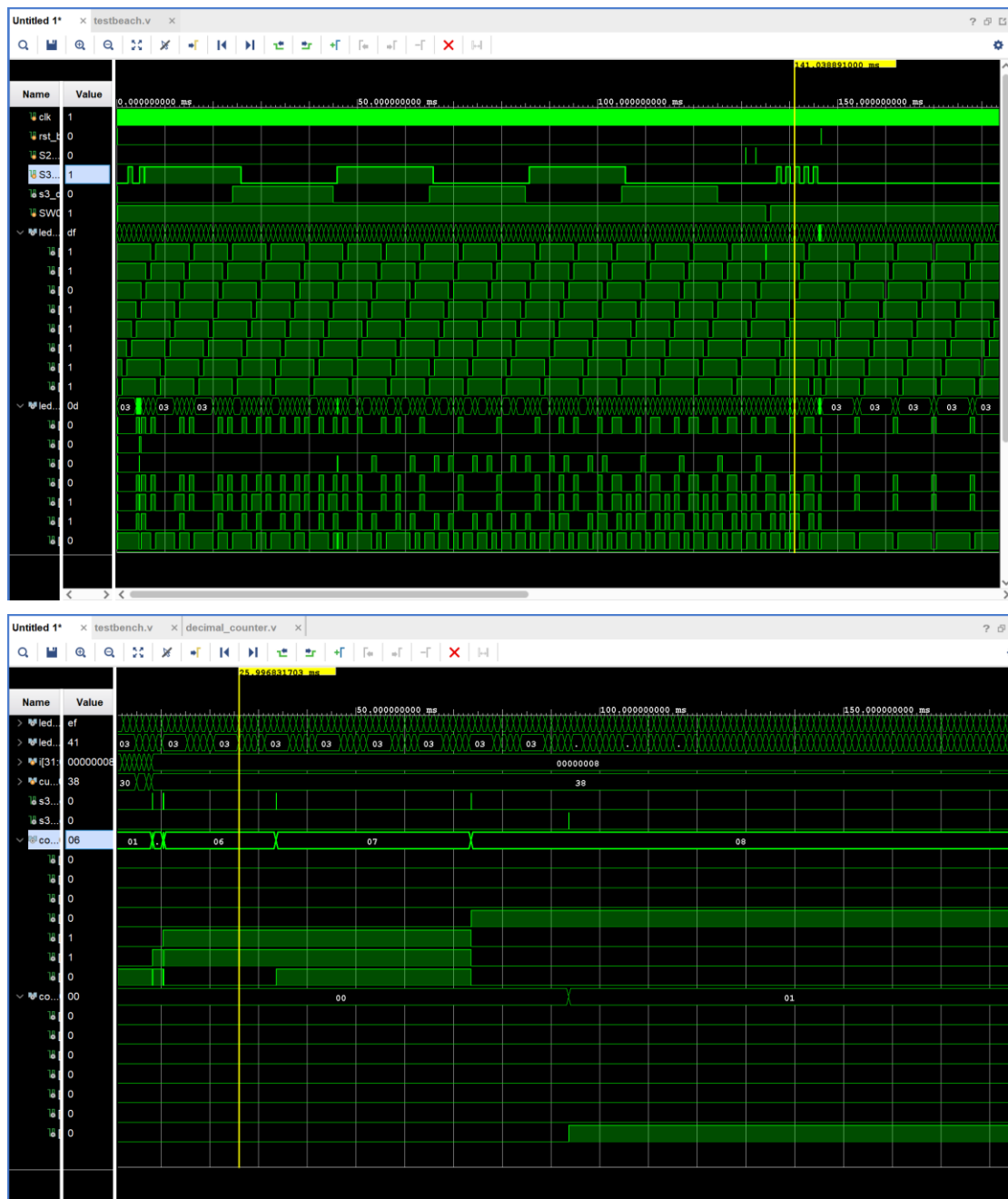


二、数码管控制器仿真分析

2.1 仿真波形截图

- 能正确清晰体现所要求的功能，根据需要可以截多张仿真波形图
- 请在相关波形图下做对应的分析说明，或者明确指出分析说明对应哪张仿真截图





2.2 波形分析

- 按要求完成分析，数码管轮询、消抖、按键计数
- 数码管轮询，需体现一轮 8 个数据使能信号和段选信号的变化；
- 消抖，模拟一个带有抖动的按键波形输入，分析输出；
- 按键计数，只需体现 2 次消抖的计数即可；

图 1-2：数码管轮询

位选信号 led_en 呈现 11111110→11111101→11111011→11110111→11101111→11011111→10111111→01111111 的规律性变化，依次对应 DK0 至 DK7 八个数码管的选通控制。每个数码管的激活时间间隔精确维持在 1ms 左右，完整的轮询周期约为 8ms，这一刷新频率有效避免了视觉闪烁现象。

图 3：消抖。

S3_btn 快速抖动多次时，S3_db 保持稳定，不随抖动变化。只有当按键稳定按压，释放足够

长时间后，S3_db 才变化。且 S3_db 的变化明显滞后于 S3_btn 的初始变化。

图 4：按键计数：

前几次按键持续时间较短，模拟抖动。

上方波形（未消抖计数器）：对每次按键抖动均产生响应，计数数值快速递增。

下方波形（消抖后计数器）：仅对稳定按键动作进行计数，有效过滤抖动干扰。

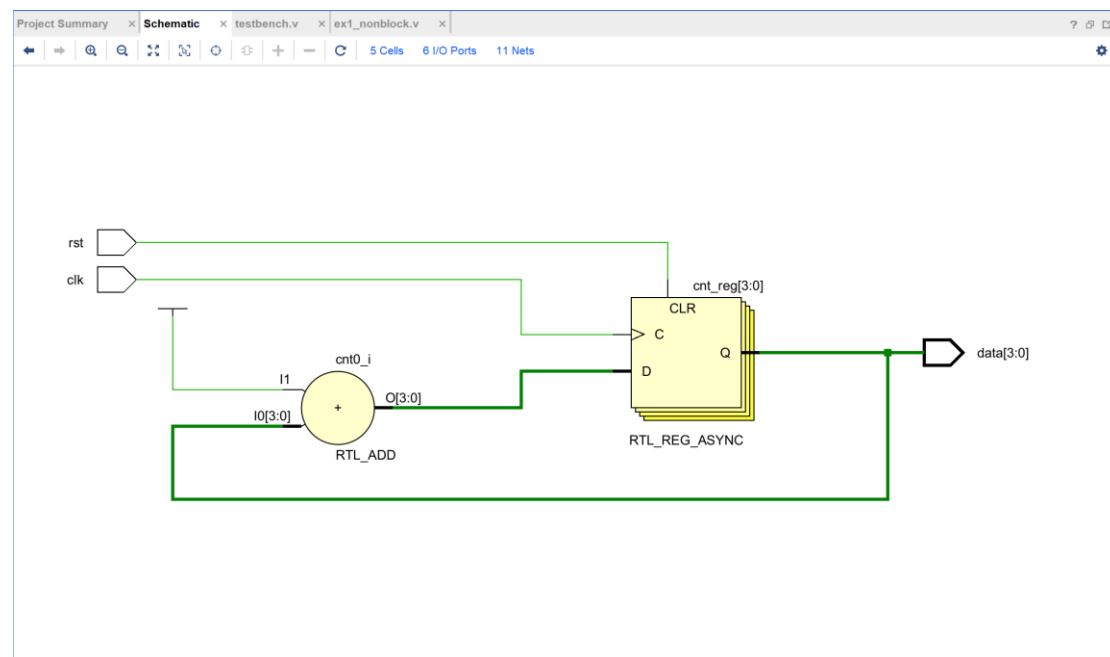
不消抖计数一直在增加，增加到 8 时消抖计数才为 1。

三、课后作业内容

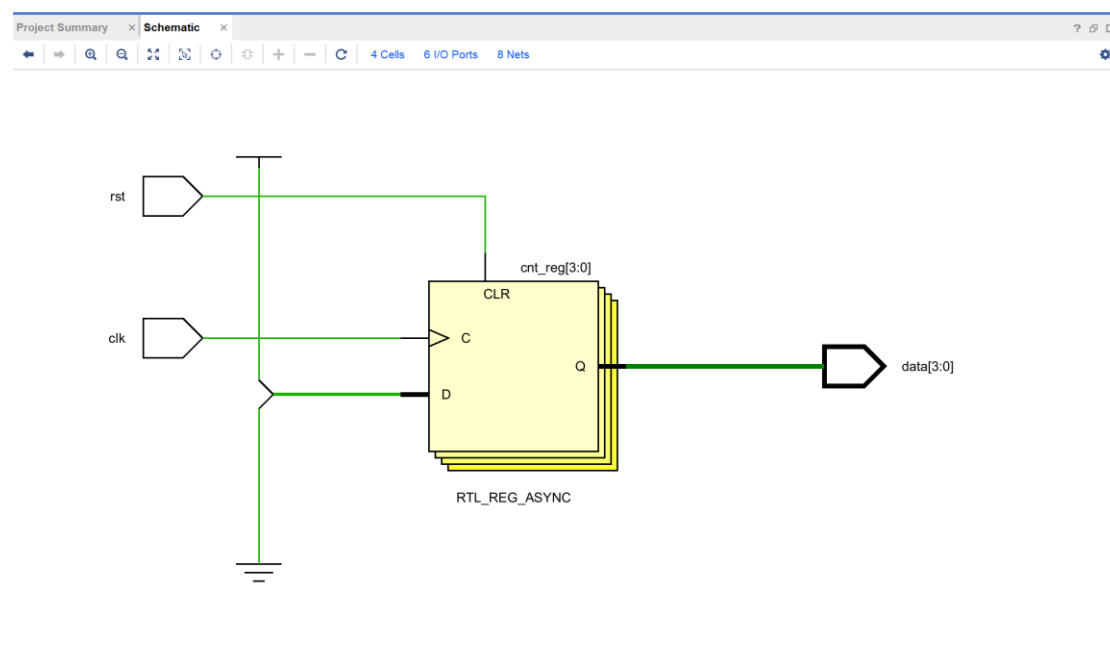
3.1 请参照以下表格中的内容要求，对比“课后作业”中所给电路阻塞和非阻塞赋值的区别，所用截图要清晰。格式可以不用表格，要能够方便对比查看要求的内容。

(1) RTL 分析：

非阻塞赋值：

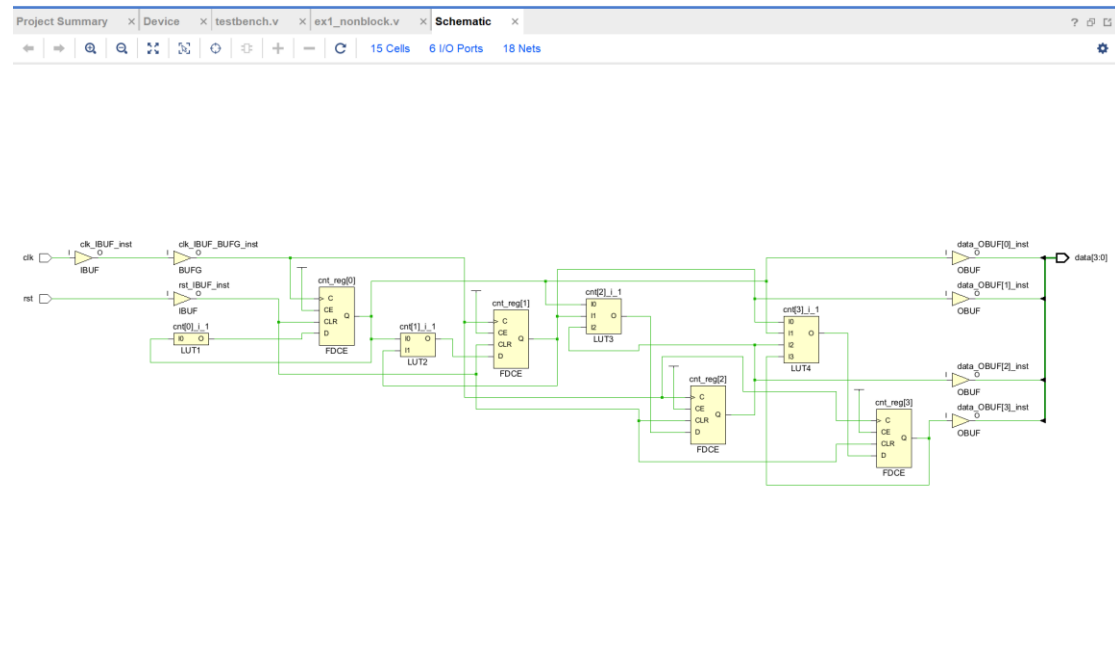


阻塞赋值：

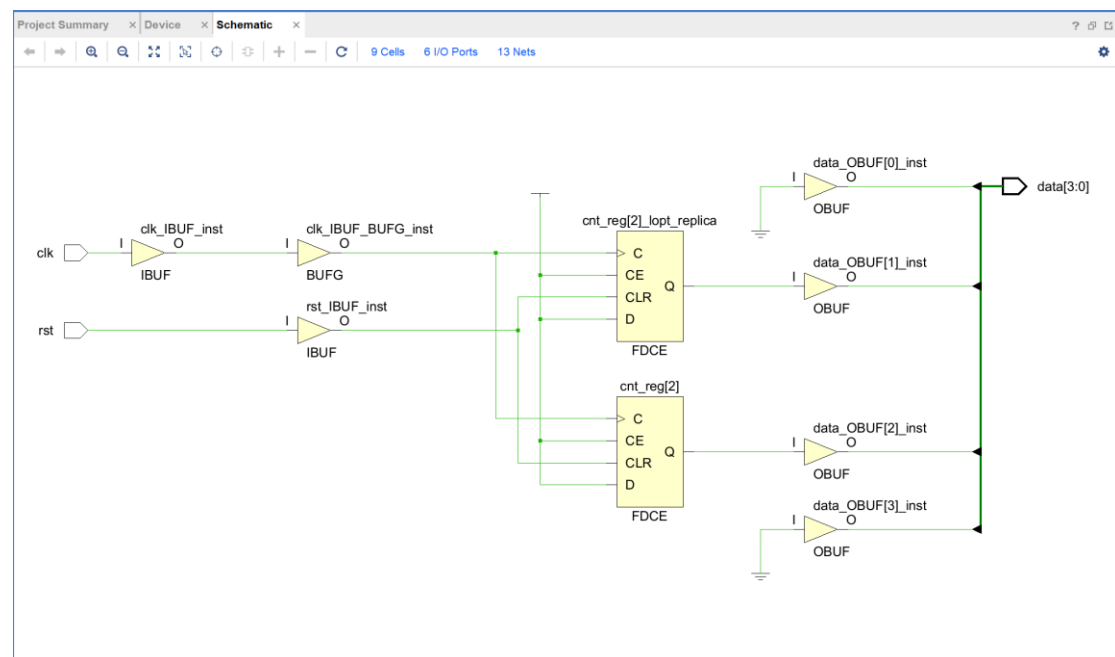


(2) 综合后电路图：

非阻塞赋值：

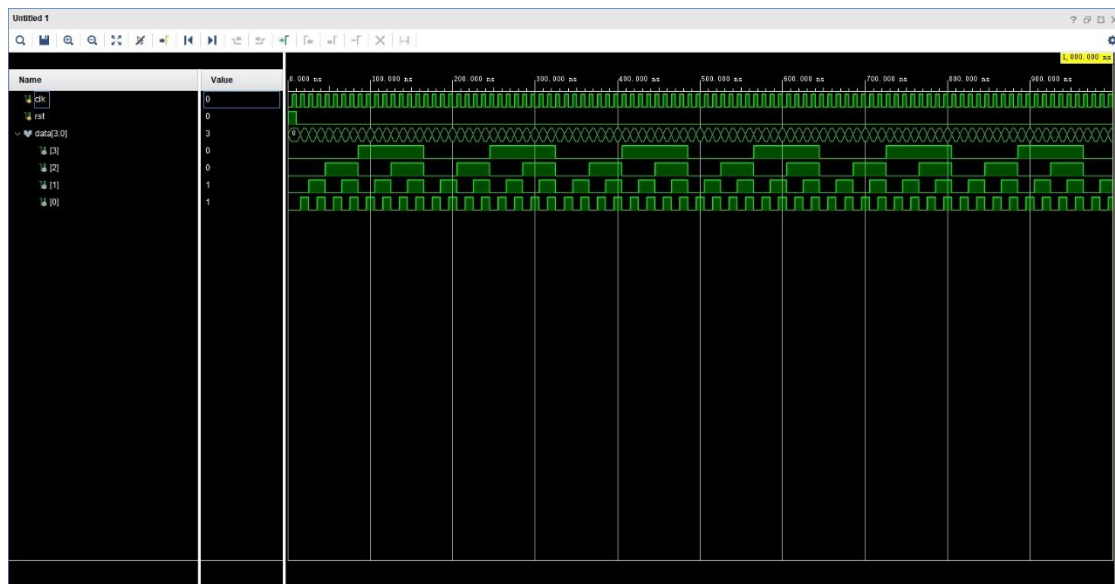


阻塞赋值：

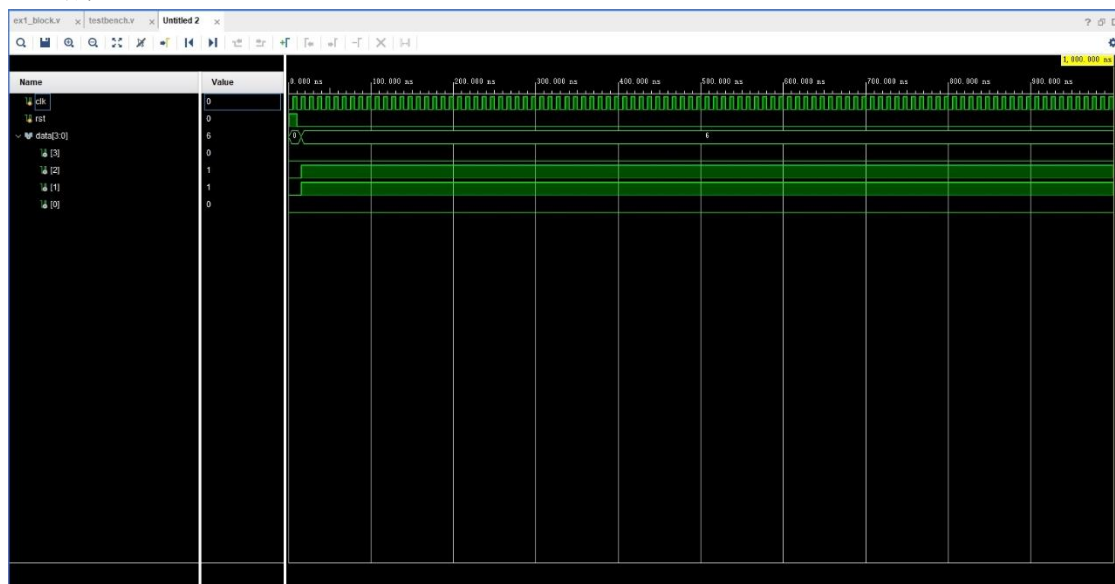


(3) 仿真波形截图

非阻塞赋值：



阻塞赋值：



3.2 针对 ex1_block 和 ex1_nonblock 两种实现，分别回答

一个时钟后 cnt 是多少，是 1 还是 6 还是 5? 2 个时钟后的 cnt 是多少?

	1 个时钟后 cnt	2 个时钟后 cnt
ex1_block	6	6
ex1_nonblock	1	2