

Lab1 FPGA基础实验：LED Flow Water Light

Lab1 FPGA基础实验：LED Flow Water Light

实验目的

实验原理

实现细节

基本设计

工程实现

实验结果

总结与反思

实验目的

- 掌握 Xilinx 逻辑设计工具 Vivado的基本操作
- 掌握使用 Verilog HDL进行简单的逻辑设计
- 掌握功能仿真
- 使用 I/O Planing 添加管脚约束
- 生成 Bitstream 文件（属于下载验证，暂不做）
- 上板验证（暂不做）

实验原理

- LED流水灯功能：每隔一段时间点亮下一个LED并熄灭当前的LED灯。
- 用一个计数器记录时间，当计数器达到某个值时，进行LED灯的切换，并重置计数器。
- 用一个8位二进制数来记录8个LED灯的状态，1表示点亮，0表示熄灭。

实现细节

基本设计

1、编写设计文件flowing_light.v

- 模块入口
 - input
 - clock: 时钟
 - reset: 重置信号
 - output
 - led: 8个LED灯
- 变量
 - cnt_reg: 24位寄存器，作为计数器。
 - light_reg: 8位寄存器，记录8个LED灯的状态。
- 行为逻辑
 - 在每次时钟信号上升沿的时候执行一次，即每个时钟周期执行一次。
 - 若reset为1，重置计数器，否则将其加1。

- 若reset为1，重置LED灯状态（00000001）。若此时计数器达到最大值时，检查LED灯状态：如果第8个LED被点亮（10000000），那么下一次要点亮第1个，故重置LED灯状态；如果还没有遍历所有的LED灯，则将状态数左移一位，即点亮下一个LED灯。
 - 将LED灯状态数作为led输出。
- 模块出口

2、编写仿真激励文件flowing_light.tb

- 模块入口：无输入输出
- 变量
 - clock：1位寄存器，时钟信号。
 - reset：1位寄存器，重置信号。
 - led：输出总线，内有8条线路。
- 对要仿真的模块作实例化
 - u0是模块flowing_light的一个实例化对象
 - 用clock等变量来作为u0的输入输出。
- 行为逻辑
 - 每20个单位时间（PERIOD = 10），clock取反，所以时钟周期为40个单位时间。
 - 在仿真开始时初始化clock为0，reset为0。
 - 在仿真一开始时生成激励波形作为电路的测试仿真信号。仿真开始半个时钟周期后，reset置1，再过1个时钟周期，reset置0。
- 模块出口

3、运行仿真

4、修改计数器位数，再次运行仿真

```

31 |      |      reg [1 : 0] cnt_reg; //24 -> 2
49 |  ○ |      always @ (posedge clock) //CLK_i
50 |  ○ |      begin
51 |  ○ |          if (reset) //!reset: The button on the board actually is Low when being pushed
52 |  ○ |              light_reg <= 8'h01;
53 |  ○ |          else if (cnt_reg == 2'b11) //24'hfffffff -> 2'b11
54 |  ○ |              begin
55 |  ○ |                  if (light_reg == 8'h80)
56 |  ○ |                      light_reg <= 8'h01;
57 |  ○ |                  else
58 |  ○ |                      light_reg <= light_reg << 1;
59 |  ○ |              end
60 |  ○ |      end

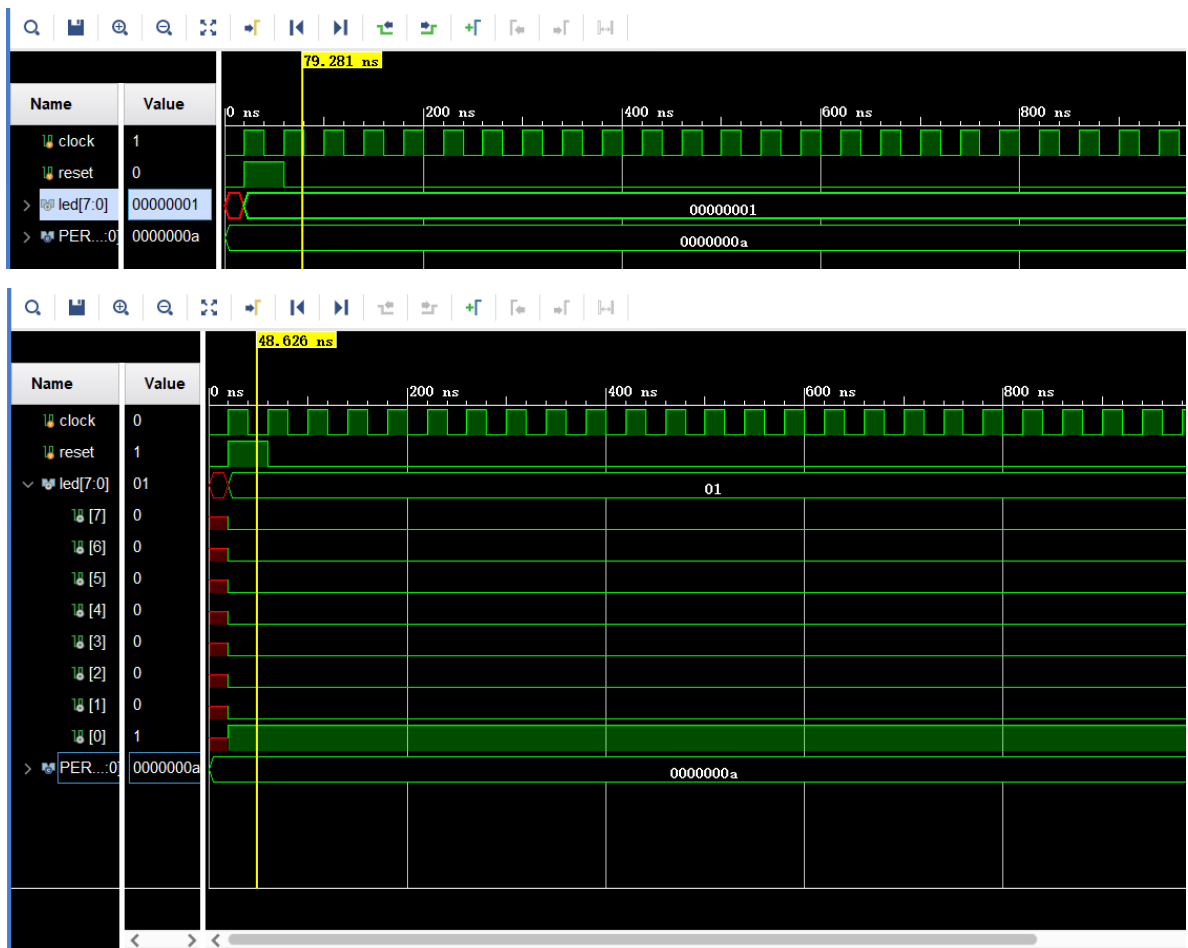
```

工程实现

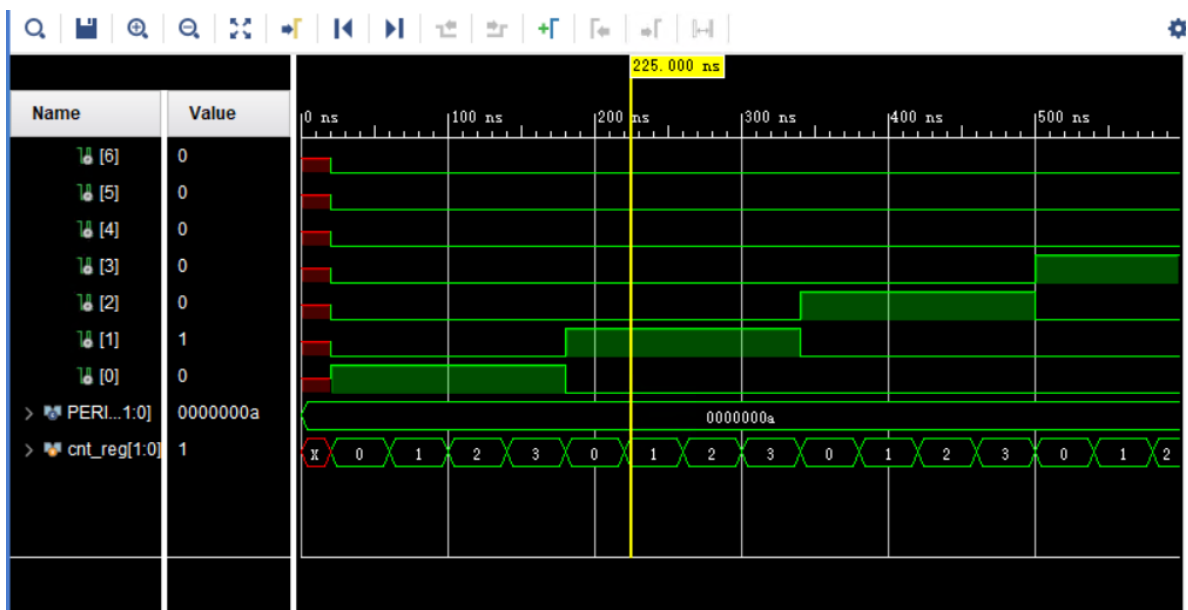
- 1、修改flowing_light.v
- 2、添加管脚约束文件：利用Vivado的IO planning功能。

实验结果

- 1、第一次运行：始终只有第一个LED灯点亮。



2、修改了计数器位数：LED灯依次点亮。



总结与反思

- 学习了如何使用Vivado创建新项目。
- 学习了简单的Verilog语法。
- 学习了Verilog程序的编写逻辑。
- 学习了如何使用Vivado进行仿真。
- 学习了如何通过观察仿真波形判断实验结果是否正确。