**计算机系统结构实验lab01**

**Flowing light**

徐阳 521021910363

2023年3月15日

目录

[目录 2](#_Toc130156391)

[1. 摘要 2](#_Toc130156392)

[2. 实验目的 2](#_Toc130156393)

[3. 原理分析 2](#_Toc130156394)

[4. 功能实现 2](#_Toc130156395)

[5. 结果验证 3](#_Toc130156396)

[5.1仿真模拟 3](#_Toc130156397)

[5.2管脚约束与上板验证 5](#_Toc130156398)

[6. 总结与反思 7](#_Toc130156399)

1. 摘要

本实验中，我们实现了FPGA基础实验中LED流水灯的设计，初步了解了Vivado软件的使用和Verilog语言的基础语法，学习了如何通过Vivado软件平台仿真，并在FPGA器件上进行了验证。

1. 实验目的
2. 熟悉 Xilinx 逻辑设计工具 Vivado 软件开发环境；
3. 了解硬件描述语言Verilog，进行简单的逻辑设计；
4. 理解流水灯实现原理，实现flowing light，使用软件仿真验证正确性;
5. 学习使用I/0 Planning添加管脚约束，上板验证正确性；
6. 熟悉系统硬件开发的基本实验流程。
7. 原理分析

实验要求实现LED 流水灯器件，该器件在每个周期的时钟上升沿时计数器加 1 ，当计数器达到最大值时，通过位移操作使8个LED灯依次轮流亮灭；该器件还支持接收 reset 信号对 LED灯进行初始化与复位。

Vivado软件中的工程文件由源文件(“\*.v”）、激励文件(“\*.v”)、管脚约束文件(“\*.xdc”)构成.

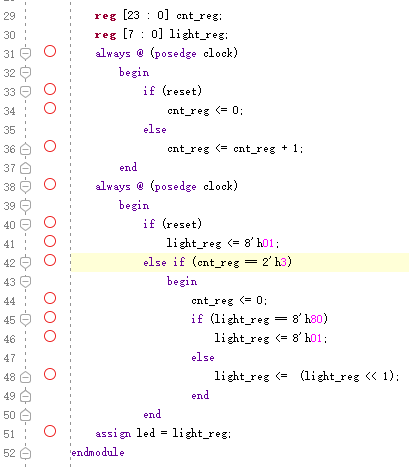
1. 功能实现

flowing\_light.v的关键代码如下，定义了两个寄存器数据类型，cnt\_reg用来计数，light\_reg用来点灯。

从仿真0时刻开始循环执行always行为语句模块，posedge指在clock上升沿的时候执行该模块，always模块并行执行。

在第一个always模块中，当reset信号为1时，清零重置cnt\_reg计数器，否则正常更新cnt\_reg。

在第二个always模块中，当 reset 信号为 1 时，将 LED 灯状态重置，计数器达到特定值后，重置计数器，对light\_reg进行左移操作，实现灯的轮流亮灭。



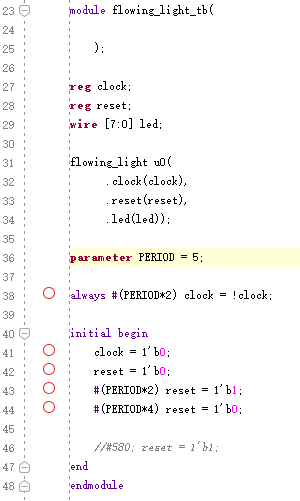
在仿真中，为使实验结果明显，我们将寄存器最大值减小为2’h3，而在上板验证中为24’hffffff。

实现flowing\_light.v后，添加激励文件flowing\_light\_tb.v用于仿真测试，添加管脚约束文件lab01\_xdc.xdc用于上板验证

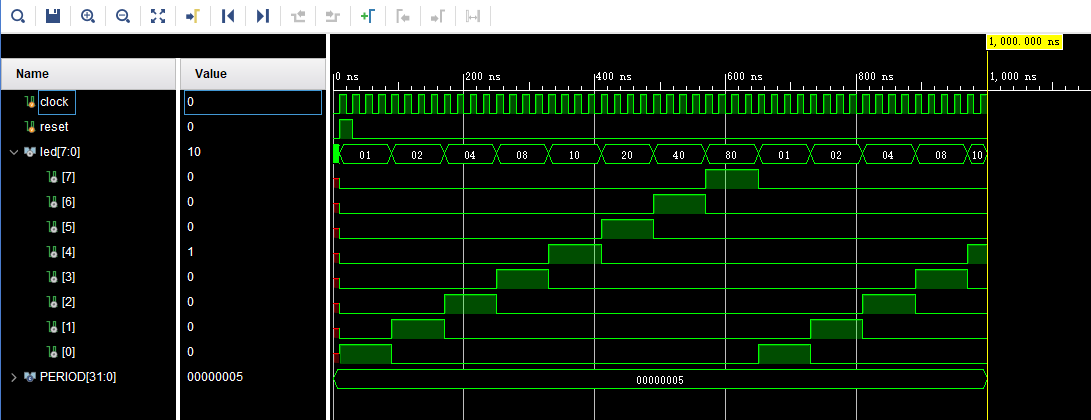
1. 结果验证

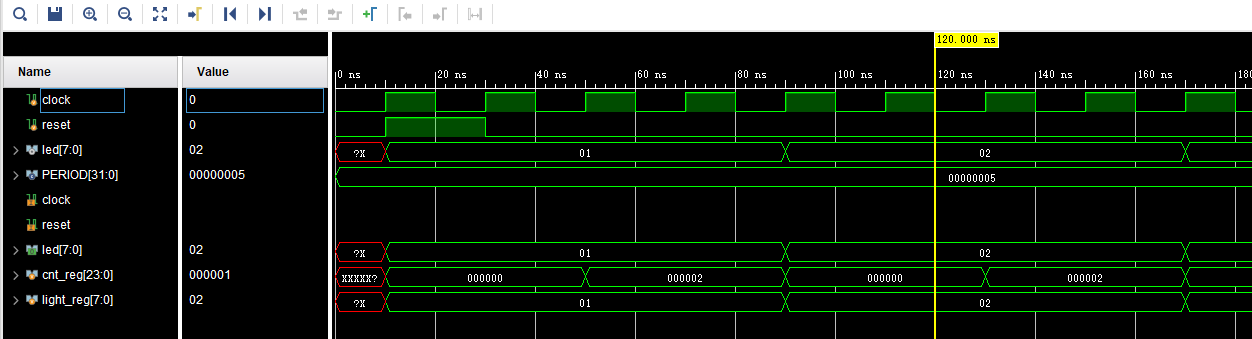
## 5.1仿真模拟

我们按实验指导书编写激励文件如下，可以看出时钟周期是20个时间单位。



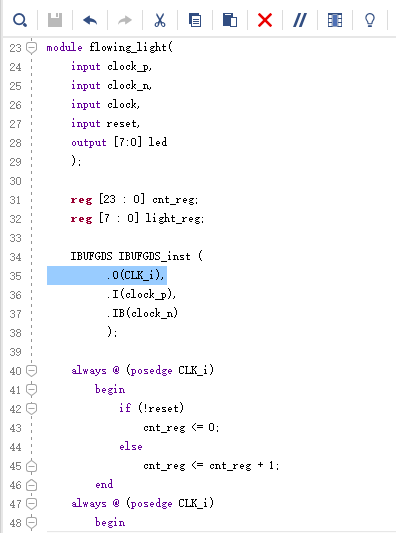
仿真结果如下，可见我们完成了LED流水灯功能的实现，仿真结果正确。

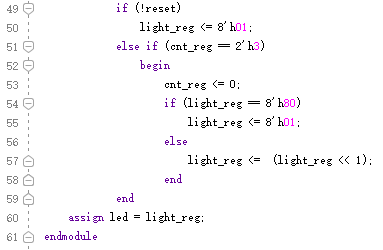




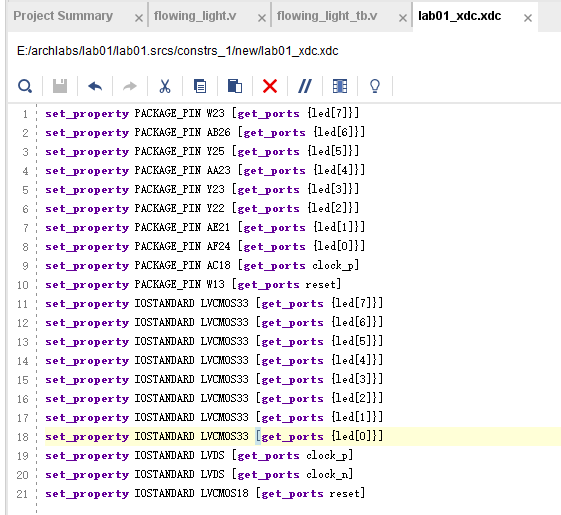
## 5.2管脚约束与上板验证

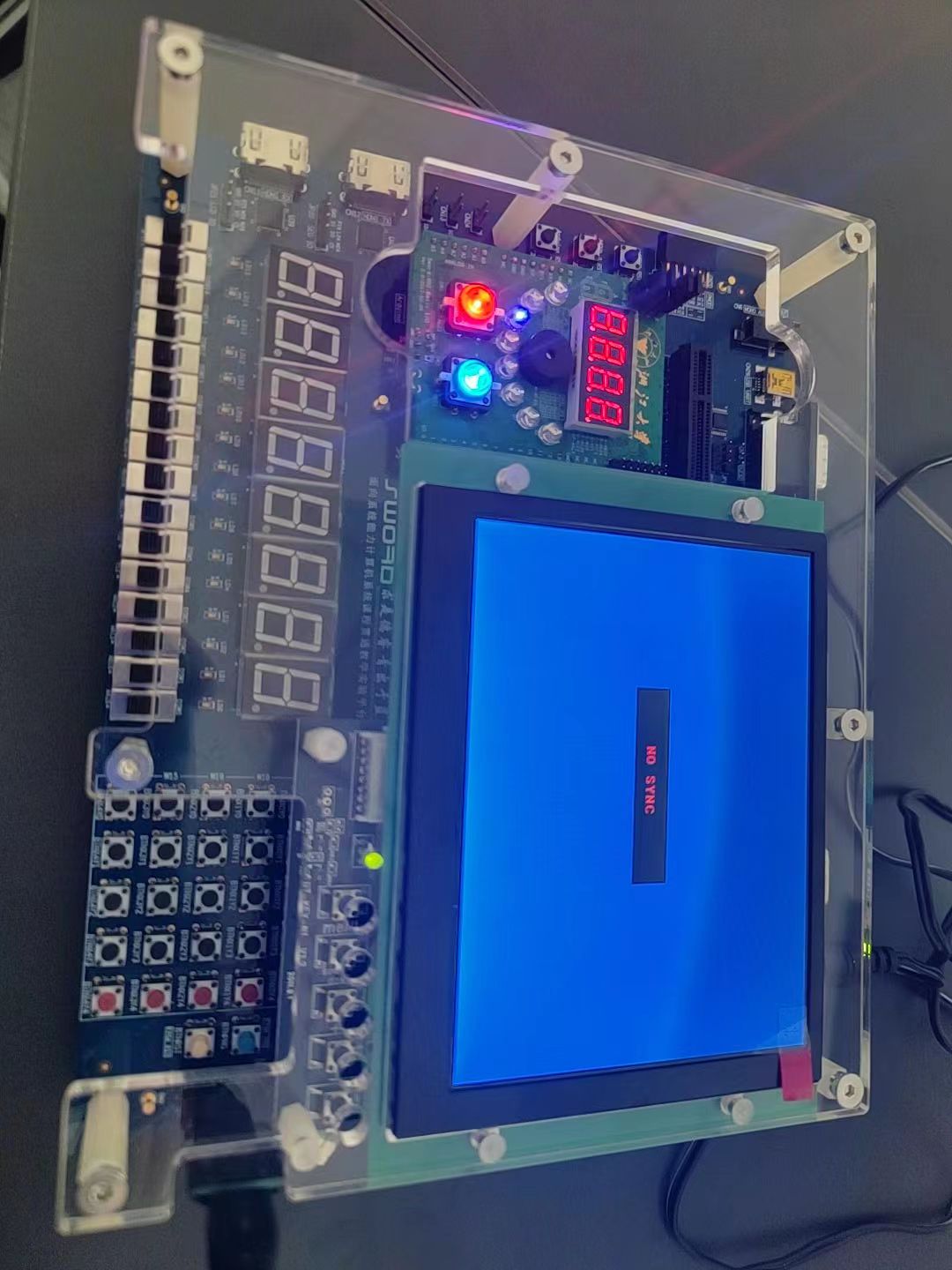
上板验证时，按实验指导书，代码需要做细节修改如下，添加了拆分时钟





接着添加管脚约束文件，可以使用Vivado的I/O Planning功能，也可手动添加管脚约束xdc文件，本次实验中，手动添加管脚约束文件如下，



下载代码至FPGA烧写，成功实现LED流水灯功能。

1. 总结与反思

本次实验是我接触FPGA、Vivado的第一节课，作为一个初学者，我在这次实验中对Xilinx 逻辑设计工具 Vivado 软件开发环境和Verilog语言有了一个初步的认识，基本掌握了Verilog语言的基础语法。同时我还学习了在Vivado平台仿真验证并上板实现的过程。感谢老师与助教提供的教学、指导答疑以及实验资源，感谢各位同学在实验过程中提供的帮助。