

计算机系统结构实验报告

Lab01

FPGA 基础实验：LED Flow Water Light

姓 名： 徐薪

学 号： 519021910726

日 期： 2021 年 6 月 09 日

## 摘 要

本实验实现了 FPGA 基础实验中的 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，LED 灯左移一位点亮；该器件还支持接收 reset 信号对 LED 灯进行初始化与复位。本实验通过软件仿真与上板验证的形式进行实验结果的验证。

## 目录

|     |                                   |   |
|-----|-----------------------------------|---|
| 1.  | 实验概述 .....                        | 2 |
| 1.1 | 实验内容 .....                        | 2 |
| 1.2 | 实验目的 .....                        | 2 |
| 2.  | 原理分析 .....                        | 2 |
| 2.1 | 基础 LED Flow Water Light 的原理 ..... | 2 |
| 2.2 | 工程 LED Flow Water Light 的原理 ..... | 3 |
| 2.3 | 上板验证的原理 .....                     | 3 |
| 3.  | 功能实现 .....                        | 3 |
| 3.1 | flowing_light Module .....        | 3 |
| 3.2 | flowing_light_tb Module .....     | 4 |
| 4.  | 仿真测试 .....                        | 4 |
| 5.  | 实验总结 .....                        | 5 |
| 5.1 | 实验评价 .....                        | 5 |
| 5.2 | 实验心得 .....                        | 5 |

## 1. 实验概述

### 1.1 实验内容

本实验实现了 FPGA 基础实验中的 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，LED 灯左移一位点亮；该器件还支持接收 reset 信号对 LED 灯进行初始化与复位。本实验通过软件仿真与上板验证的形式进行实验结果的验证。

### 1.2 实验目的

- (1) 熟悉 Xilinx 逻辑设计工具 Vivado 的基本操作；
- (2) 掌握使用 VerilogHDL 进行简单的逻辑设计；
- (3) 使用功能仿真；
- (4) 使用 1/0 Planing 添加管脚约束；
- (5) 生成 Bitstream 文件；
- (6) 上板验证。

## 2. 原理分析

### 2.1 基础 LED Flow Water Light 的原理

本次实验需要实现 LED Flow Water Light。顾名思义，就是让 LED 灯像流水一样点亮。我们用移位操作来表示点亮的 LED 灯的传递。同时，我们用软件方法来进行延迟，从而控制每个 LED 被点亮的时间。另外，需要注意的是，在最后一位 LED 灯被点亮后，下一次被点亮的 LED 灯又变回了第一个，从而实现了 LED 流水灯的循环。最后，我们

还需要设置一个 `reset` 信号，以便对 LED 灯进行重置。

## 2.2 工程 LED Flow Water Light 的原理

实验原理与基础 LED Water Light 的实验原理相似，这里不再赘述。但是由于我们的实验板板载了 200MHz 的时钟振荡器，属于高频时钟，所以做下载验证是需要用到差分时钟，从而原 `flowing_light` 代码模块需做时钟方面的修改。

## 2.3 上板验证的原理

因为 Flow Navigator 可以自动完成综合、实现、生成 FPGA 配置文件，所以上板验证的实验原理与工程 LED Flow Water Light 的实验原理相同，这里不再赘述。

# 3. 功能实现

## 3.1 `flowing_light` Module

```
module flowing_light(  
    input clock,  
    input reset,  
    output [7:0] led  
);  
  
    reg [23 : 0] cnt_reg;  
    reg [7 : 0] light_reg;  
    /*  
        IBUFGDS IBUFGDS_inst (  
            .O(CLK_i),  
            .I(clock_p),  
            .IB(clock_n)  
        );  
    */
```

```

always @ (posedge clock)
begin
    if (!reset)
        cnt_reg <= 0;
    else
        cnt_reg <= cnt_reg + 1;
    end
always @ (posedge clock)
begin
    if (!reset)
        light_reg <= 8'h01;
    else if (cnt_reg == 24'hffffff)
        begin
            if (light_reg == 8'h80)
                light_reg <= 8'h01;
            else
                light_reg <= light_reg << 1;
            end
        end
    end
    assign led = light_reg;
endmodule

```

### 3.2 flowing\_light\_tb Module

```

module flowing_light_tb(

);
    reg clock;
    reg reset;
    wire [7:0] led;

    flowing_light u0 (
        .clock(clock),
        .reset(reset),
        .led(led) );

    parameter PERIOD= 10;

    always #(PERIOD*2) clock = !clock;

    initial begin
        clock = 1'b0;
        reset = 1'b0;
        #(PERIOD*2) reset = 1'b1;
        #(PERIOD*4) reset = 1'b0;

        //580; reset = 1'b1;
    end
endmodule

```

## 4. 仿真测试

我们使用 Verilog 编写激励文件，采用软件仿真的形式对于 LED 流水灯进行测试，测试结果如图 1 所示：

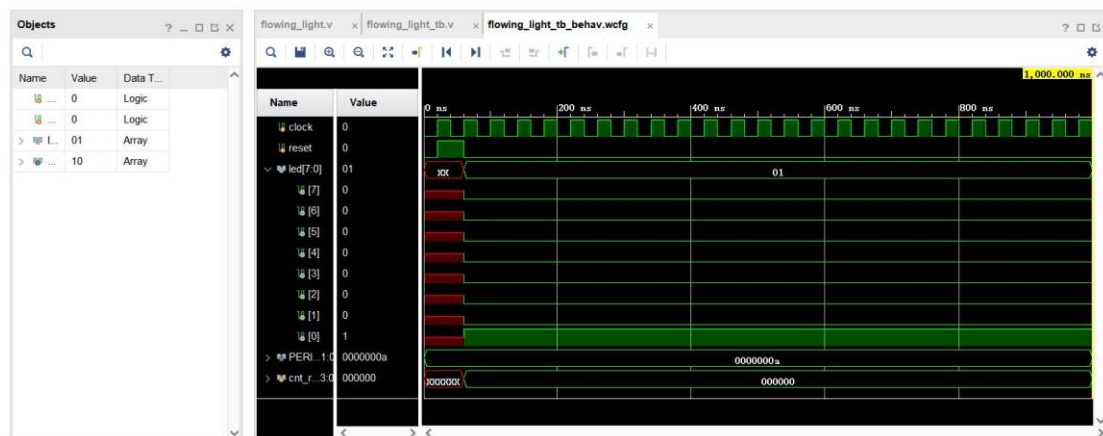


图 1. Flowing Water Light 仿真结果

从图 1 可以看出，我们完成了 LED 流水灯的功能实现，并且仿真结果正确。

## 5. 实验总结

### 5.1 实验评价

通过图 1 的结果验证以及后续的工程实现、上板验证，LED 流水灯都能正确工作，因此本次实验是成功的，LED Flowing Water Light 成功实现。

### 5.2 实验心得

这个实验总体来讲是简单的，只要按照实验操作指导书一步一步来，就能得到正确的实验结果。

在这一次实验中，我最大的收获就是逐渐认识和了解了 verilog 语言。对着指导书逐行敲代码，让我明白了 module 的定义是什么样子的，对 wire 和 reg 的使用也有了初步的了解。尤其是明白了 assign 语句的用法，这对于一个模块的输出是很重要的。最后，我还了解了

阻塞式赋值与非阻塞式赋值的区别（也就是=与<=的区别），时序电路和逻辑电路的区别。第一个实验就让我接触到了之后实验经常会犯错的内容，实在是让我受益匪浅。