

# 计算机系统结构实验报告 实验 1

姓名：卞思沅

学号：518021910656

日期：2020/05/27

# 目录：

## 1. 实验概述

### 1.1 实验名称

### 1.2 实验目的

## 2. 实验描述

### 2.1 实验简述

### 2.2 实验代码及结果

## 3. 实验心得与总结

## 4. 参考资料

## 1. 实验概述

### 1.1 实验名称

FPGA 基础实验：LED Flow Water Light

### 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 I/O Planing 添加管脚约束（暂时无需做）
5. 生成 Bitstream 文件（暂时无需做）
6. 上板验证（暂时无需做）

## 2. 实验描述

### 2.1 实验简述

1. 按照要求创建工程
2. 熟悉 Vivado 整体界面
3. 添加文件并输入相关代码

```
1  reg [23 : 0] cnt_reg;  
2  reg [7 : 0] light_reg;  
3  always @ (posedge clock)  
4  begin  
5      if (reset)  
6          cnt_reg <= 0;  
7      else  
8          cnt_reg <= cnt_reg + 1;  
9      end  
10 always @ (posedge clock )  
11 begin  
12     if (reset)  
13         light_reg <= 8'h01;  
14     else if (cnt_reg == 24'hffffff)  
15         begin  
16             if (light_reg == 8'h80)  
17                 light_reg <= 8'h01;  
18             else  
19                 light_reg <= light_reg << 1;  
20         end  
21     end  
22 end  
23 assign led = light_reg;
```

### 4. 功能仿真

#### 4.1 创建激励文件，输入以下代码

```
1
2 reg clock;
3 reg reset ;
4 wire [7:0] led;
5
6 flowing_light u0(
7     .clock(clock),
8     .reset(reset),
9     .led(led));
10
11 parameter PERIOD= 10;
12
13 always #(PERIOD*2) clock = !clock;
14
15 initial begin
16     clock = 1'b0;
17     reset = 1'b0;
18     #(PERIOD*2) reset = 1'b1;
19     #(PERIOD*4) reset = 1'b0;
20
21     //580; reset = 1'b1;
22 end
```

#### 4.2 熟悉仿真页面，使用 Run Simulation

### 2.2 代码及结果

flowing\_light.v 代码如下:

```
23 module flowing_light(
24     input clock,
25     input reset,
26     output [7:0] led
27 );
28
29 reg [23 : 0] cnt_reg;
30 reg [7 : 0] light_reg;
31 always @ (posedge clock)
32 begin
33     if (reset)
34         cnt_reg <= 0;
35     else
36         cnt_reg <= cnt_reg + 1;
37 end
38 always @ (posedge clock)
39 begin
40     if (reset)
41         light_reg <= 8'h01;
42     else if (cnt_reg == 24'hffffff)
43     begin
44         if(light_reg == 8'h80)
45             light_reg <= 8'h01;
46         else
47             light_reg <= light_reg << 1;
48     end
49 end
50 assign led = light_reg;
51
52 endmodule
```

flowing\_light\_tb.v 代码如下:

```
module flowing_light_tb(
);
reg clock;
reg reset;
wire [7:0] led;

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

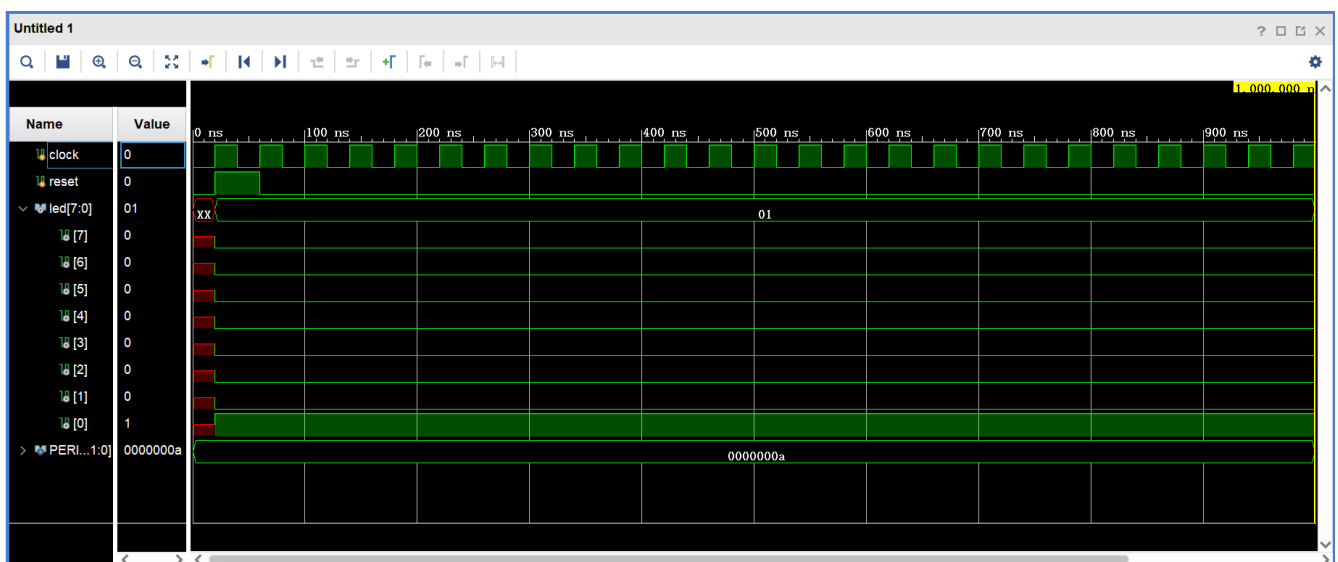
parameter PERIOD = 10;

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
```

仿真结果如下:



### 3. 实验心得与总结

本次实验是我第一次接触 Verilog 相关的编程。本次实验的目的是让我们熟

悉软件的使用以及熟悉语言。这次实验是验证实验，照着指导一步步做完就可以获得相关结果。在实验过程中，我逐步了解了 Vivado 的使用方法以及 Verilog 的基本语法。

在本次实验中，我也遇到了一些问题。例如课程指导书中由于篇幅原因，代码部分截图只显示了最重要的部分，其它不需要我们改动的部分便没有列出。由于我起初对 Verilog 语言比较陌生，对指导书中的截图有一些误解，写出了一些编译错误（比如我起初将 Module 的声明部分删去了）。之后，我通过上网查找资料等方式找到了错误原因，成功得到了准确的结果。这也让我对代码有了更深的理解。

由于课程是在线上课，没有能够上板验证，如果能够及时返校上课希望能补上上板验证的环节，完善整个实验。

#### **4、参考资料**

2020 计算机系统结构实验指导书-LAB01\_M