北京科技大学 计算机与通信工程学院

实 验 报 告

实验名称：Vivado 设计流程

一、实验目的与实验要求

1、实验目的

本实验的目的是熟悉实验环境，学习如何使用 Vivado 2015.4 创建、综合、实现、仿真等功能。

2、实验要求

（1） 学习Vivado-Design-Flow.pdf、约束文件.ppt，在实验报告中回答以下问题：

a) 描述Vivado 的设计流程

b) 什么是网表

c) 什么是约束文件？通过IO planning 完成的是什么方面的约束？

d) Vivado 设计流程中，Synthesis 的作用是什么？

e) Vivado 设计流程中，Implementation 的作用是什么？

（2） 在实验报告中提交上述实验内容的Verilog 代码、仿真结果图、RTL 详细设计图（ 参考Vivado-Design-Flow.pdf 第11 页 Open Elaborated Design—>Schematic）、综合实现图（参考Vivado-Design-Flow.pdf 第13 页 Open Synthesized Design—>Schematic）以及实验现象图（照片）；

（3） 提交实验报告和完整的工程文件。

二、实验设备（环境）及要求

（1） Xilinx Ego1 实验平台。

（2） OS：Win7 64 位

（3） Software：Vivado15.4 开发工具

三、实验内容与步骤

（1）实验内容

a) 学习视频，了解 Vivado 设计流程和功能：“EGO 五分钟快速上手.mp4”和“EGo 五分钟搭建你的数字积木.mp4”。

b) 按照“Ego 五分钟快速上手——流水灯.pdf”完成流水灯实验。

c) 学习 Vivado-Design-Flow.pdf、约束文件.ppt，回答问题。

（2）主要步骤

a) 在 Vivado 中创建 RTL 设计，新建一个名为 vivado_design_flow 的空白工程。

b) 模块代码如下：

```
module flowing_light(
    input clk,
    input rst,
    output [15:0] led
    );

    reg[23:0]cnt_reg;
    reg[15:0]light_reg;

    always@(posedge clk)
        begin
            if(rst)
            cnt_reg<=0;
            else
            cnt_reg<=cnt_reg+1;
        end
    always@(posedge clk)
  begin
      if(rst)
              light_reg<=16'h0001;
        else if (cnt_reg == 24'hffffff)
        begin
        if(light_reg == 16'h8000)
            light_reg<=16'h0001;
        else
            light_reg<=light_reg<<1;
          end
        end
    assign led = light_reg;
endmodule
```

c) RTL 门级结构图：

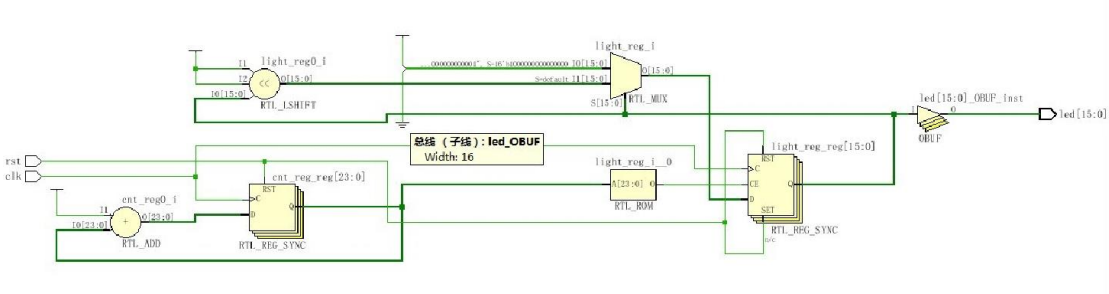

图 1: RTL 门级结构图

d) 综合 Run Synthesis 实现图

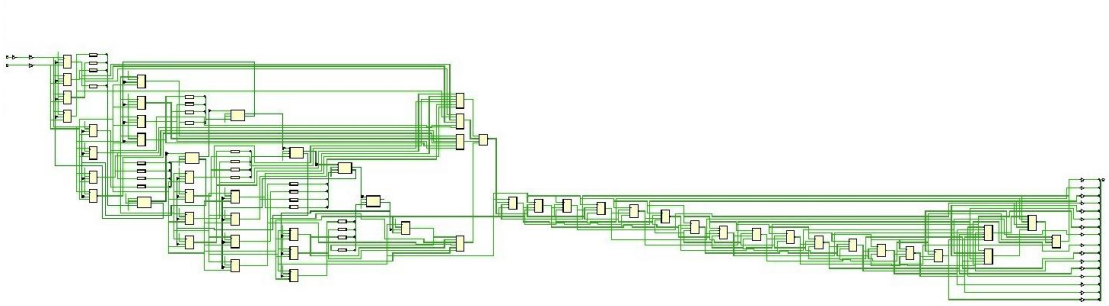

图 2：综合最终设计图

e) 添加约束文件，按照对应管脚，综合、实现进行管脚约束：

```
set_property PACKAGE_PIN F6 [get_ports {led[15]}]
set_property PACKAGE_PIN G4 [get_ports {led[14]}]
set_property PACKAGE_PIN G3 [get_ports {led[13]}]
set_property PACKAGE_PIN J4 [get_ports {led[12]}]
set_property PACKAGE_PIN H4 [get_ports {led[11]}]
set_property PACKAGE_PIN J3 [get_ports {led[10]}]
set_property PACKAGE_PIN J2 [get_ports {led[9]}]
set_property PACKAGE_PIN K2 [get_ports {led[8]}]
set_property PACKAGE_PIN K1 [get_ports {led[7]}]
set_property PACKAGE_PIN H6 [get_ports {led[6]}]
set_property PACKAGE_PIN H5 [get_ports {led[5]}]
set_property PACKAGE_PIN J5 [get_ports {led[4]}]
set_property PACKAGE_PIN K6 [get_ports {led[3]}]
set_property PACKAGE_PIN L1 [get_ports {led[2]}]
set_property PACKAGE_PIN M1 [get_ports {led[1]}]
set_property PACKAGE_PIN K3 [get_ports {led[0]}]
set_property PACKAGE_PIN P17 [get_ports clk]
set_property PACKAGE_PIN R15 [get_ports rst]
```

```
set_property IOSTANDARD LVCMOS33 [get_ports {led[15]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[14]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[13]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[12]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[11]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[10]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[9]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[8]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[7]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[6]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {led[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports clk]
set_property IOSTANDARD LVCMOS33 [get_ports rst]
```

f) 测试激励仿真代码：

```
module test( );

reg clk;

reg rst;

wire [3 : 0] led;

flowing_light u0(

.clk(clk),

.rst(rst),

.led(led) );

parameter PERIOD = 10;

always begin

clk = 1'b0;

#(PERIOD/2) clk = 1'b1;

#(PERIOD/2);

end

initial begin

clk = 1'b0;
```

```
rst = 1'b0;
#100;
 rst = 1'b1;
 #100;
rst = 1'b0;
end
```

endmodule 进行行为仿真，得到并验证波形图

g) 添加时序约束，设置 clock 的最大和最小的 delay time

h) 最后进行综合，生成比特流文件，加载到板子上进行调试分析。

四：实验结果与分析

（1） 通过设计后，利用 Vivado 自带工具，生成 RTL 门级结构如下：

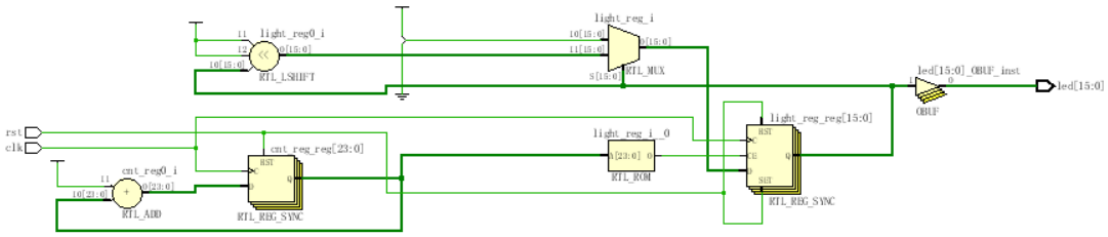

图 3： RTL 门级结构

（2） 仿真得到波形图如下：

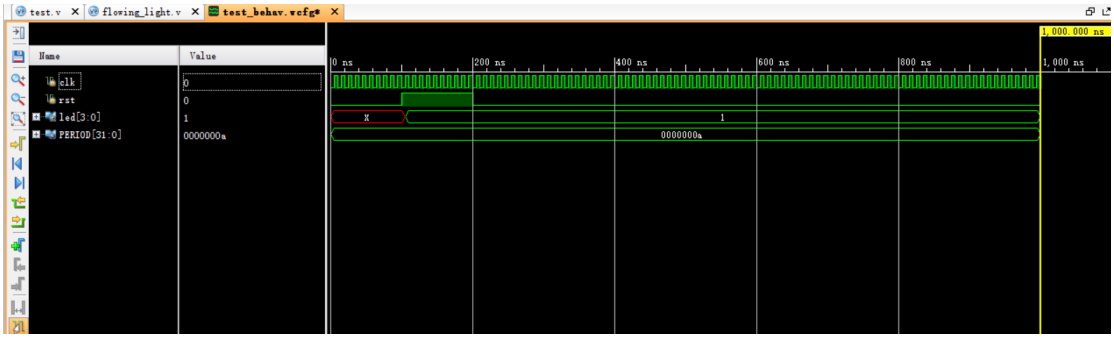

图 4：波型仿真图

（3） 实验板显示情况如下：

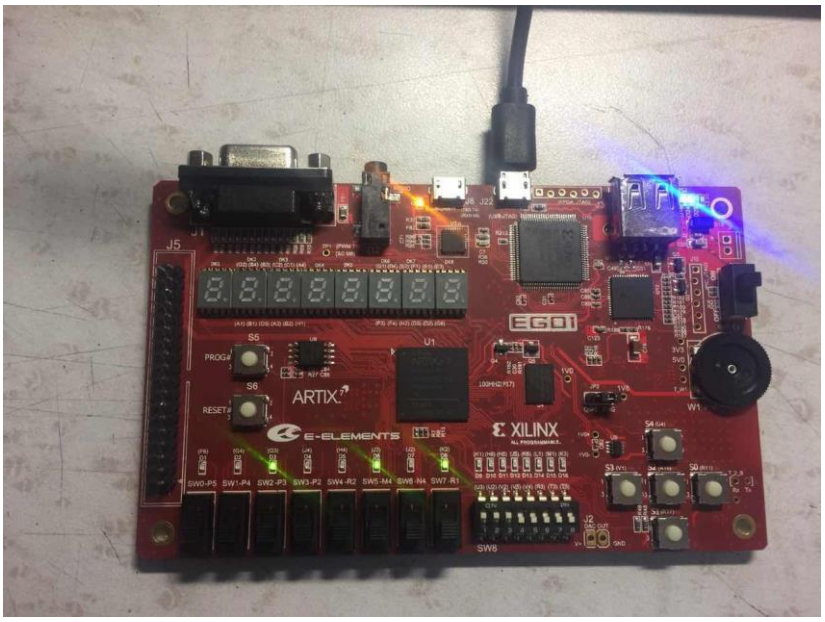

图 4：FPGA 板测试图样

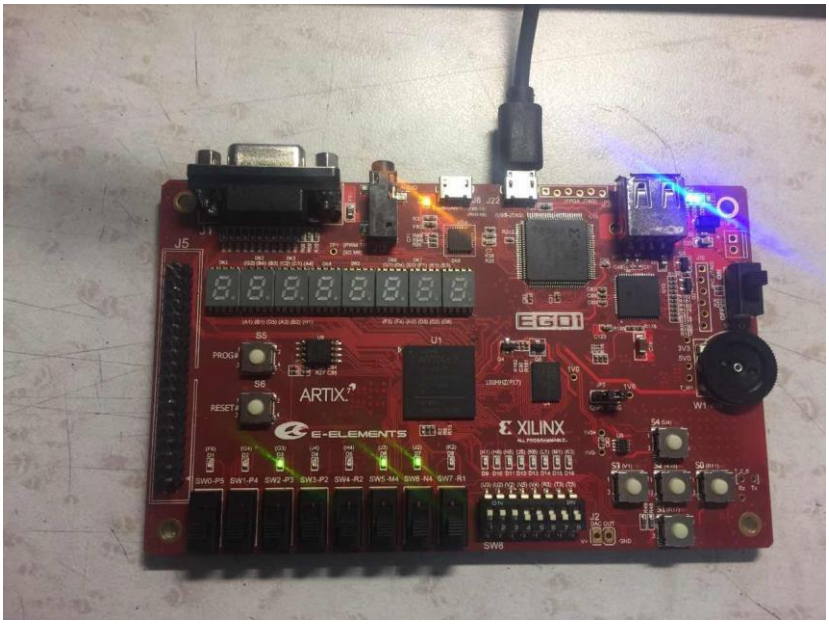

图 5：FPGA 板测试图样

五：结论（讨论）

1、实验结论

通过本次实验的流程和视频的观看，学会了如何使用 Xilinx Vivado 2015.4 创建、综合、实现、仿真等功能。通过编写一个流水灯实验来介绍使用 Xilinx Vivado

来进行基本的 FPGA 设计，简单易懂，并能很好的类比其他的设计。

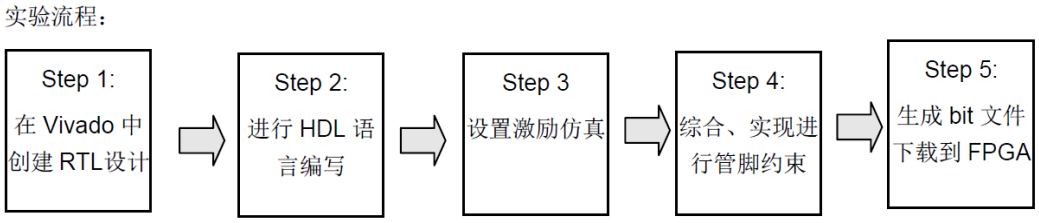

图 6 ：实验流程图

2、讨论

在流水灯实验中，由于频率过高，导致了显示频率过快，可以增加分频代码，将频率再进一步降低，使显示更加清楚明了。

现回答报告中所提出的问题：

a) 描述 Vivado 的设计流程：

实验流程主要有以下几步：

1、通过编写 HDL 文件的方式创建 Vivado 设计。

2、进行 Verilog HDL 语言编写

3、写出测试激励，在测试激励中进行仿真

4、综合、实现进行管脚约束

5、生成 bitstream 文件，下载到 FPGA 开发板里

b) 什么是网表：

xilinx 的步骤 synphysize（综合） translate（注译），map（映射），和 place and route（布局布线）

综合以后生成的就为网表文件即一个电路的雏形，之后可以看到 RTL（寄存器传输级）电路。也可以看到 technology 电路。两者区别等你看到就很快明白。这两张图片是对网表的一种直观的显示。也就是综合器最后综合出了你的逻辑电路。放在网表文件中。

c) 什么是约束文件：

约束文件是用来指定选用 FPGA 器件的型号及定义引脚属性的文件。其中包括管脚约束（将模块的端口和 FPGA 的管脚对应）、区域约束（将模块放置在 FPGA 的特点位置）、时序约束（对数据建立、保持时间进行约束，保证设计在高速时钟下的工作可靠性等）。通过 IO planing 完成的是**管脚约束**。

d) Vivado 设计流程中，Synthesis 的作用：

即综合，作用是将所写的 verilog 代码通过编译器处理，生成一系列文件，向操作者报告综合的结果。并且会自动生成门级逻辑结构网表，比源文件更加具体，可以用测试模块调用它做仿真。

e) Vivado 设计流程中，Implementation 的作用：

即实现，作用是在 Synthesis（综合）完成后，配合约束文件，实现设计。它比综合更加接近实际结果。

六、教师评审

教师评语	实验成绩
签名： 日期：	