

# 浙江大学

## 本科实验报告

课程名称：计算机逻辑设计基础

姓 名：刘晨

学 院：计算机科学与技术学院

系：图灵 1901

专 业：计算机科学与技术

学 号：3190104666

指导教师：董亚波

2020 年 10 月 15 日

# 浙江大学实验报告

课程名称： 计算机逻辑设计基础 实验类型： 综合

实验项目名称： 常用电子仪器的使用

学生姓名： 刘晨 专业： 图灵班 学号： 3190104666

同组学生姓名： 任翼 指导老师： 董亚波

实验地点： 东 4-509 实验日期： 2020 年 10 月 10 日

## 一、实验目的和要求

- 1.熟悉 Verilog HDL 语言并能用其建立基本的逻辑部件,在 Xilinx ISE 平台进行输入、编辑、调试、行为与仿真与综合后功能仿真
- 2.熟悉掌握 SWORD FPGA 开发平台,同时在 ISE 平台上进行时序约束、引脚约束及映射布线后时序仿真
- 3.运用 Xilinx ISE 具将设计验证后的代码下载到实验板上,并在实验板上验证

## 二、实验内容和原理

内容:

熟悉 ISE 工具软件的运行环境与安装过程

设计简单组合逻辑电路,采用图形输入逻辑功能描述,建立 FPGA 实现数字系统的 Xilinx ISE 设计管理工程,并进行编辑、调试、编译、行为仿真,时序约束、引脚指定(约束)、映射布线后时序仿真及 FPGA 编程代码下载与运行验证

设计简单时序逻辑电路,采用 Verilog 代码输入逻辑功能描述,建立 FPGA 实现数字系统的 ISE 设计管理工程,并进行编辑、调试、编译、行为仿真,时序约束、引脚约束、映射布线后时序仿真及 FPGA 编程代码下载与运行验证

以图形方式输入逻辑功能描述

(不考虑灯延时熄灭, 采用拨动开关)

用 Verilog 语言描述电路逻辑功能

(要考虑灯延时熄灭, 采用按钮开关)

原理:

问题 1: 某三层楼房的楼梯通道共用一盏灯, 每层楼都安装了一只开关并能独立控制该灯, 请设计楼道灯的控制电路。

问题 2: 增加控制要求, 灯打开后, 延时若干秒自动关闭, 请重新设计楼道灯的控制电路。

### 三、实验过程和数据记录

#### 1、图形方式输入逻辑功能描述

**建立楼道控制的工程: LampCtrl\_sch.ise**

1、依次点击菜单 File → New Project...

2、在对话框中设置:

Project Name: LampCtrl\_sch

Top-Level Source Type: Schematic

3、确认后, 点击 Next 到设备属性页, 设置:

Family: Kintex7

Device: XC7K160T

Package: FFG676

Speed: -1

4、确认后, 一直点击 Next 直到创建工程结束。

**创建原理图文件: LampCtrl.sch**

1、在 Sources 窗口 Sources 选项卡设备型号名处右键菜单选择 New Source

2、新建源文件向导中选择源文件类型为 Schematic, 输入文件名 LampCtrl, 勾选 Add to Project

3、连续点击 Next, 最后点击 Finish; 在 Sources 窗口中双击刚新建的文件图标, 进入电路原理图编辑窗口

**输入楼道灯控逻辑电路**

在 Sources 窗口中选择 Symbols 选项卡，配合 Schematic Editor 工具条输入原理图

### 查看输入电路的硬件描述代码

在 Sources 窗口中选择 Sources for: Synthesis / Implementation，选中 LampCtrl.sch 图标，在 Processes 窗口 Processes 选项卡中展开 Design Utilities 并双击 View HDL Functional Model，如图

### 建立基准测试波形文件：LampCtrl\_sim.tbw

- 1、在 Sources 窗口空白处的右键菜单中选择 New Source
- 2、在新建源文件向导中选择源类型为：Verilog Test Fixture，输入文件名 LampCtrl\_sim，并勾选 Add to Project
- 3、选择 LampCtrl 模块，点击 Next，在 Summary 窗口再点击 Finish，进入 LampCtrl\_sim.v 编辑窗口

### 仿真激励输入

对 LampCtrl 模块进行仿真

View 选择 Simulation 视图，Hierarchy 窗口中选择 LampCtrl\_LampCtrl\_sch\_tb，Process 窗口中选择 Simulate Behavioral Model

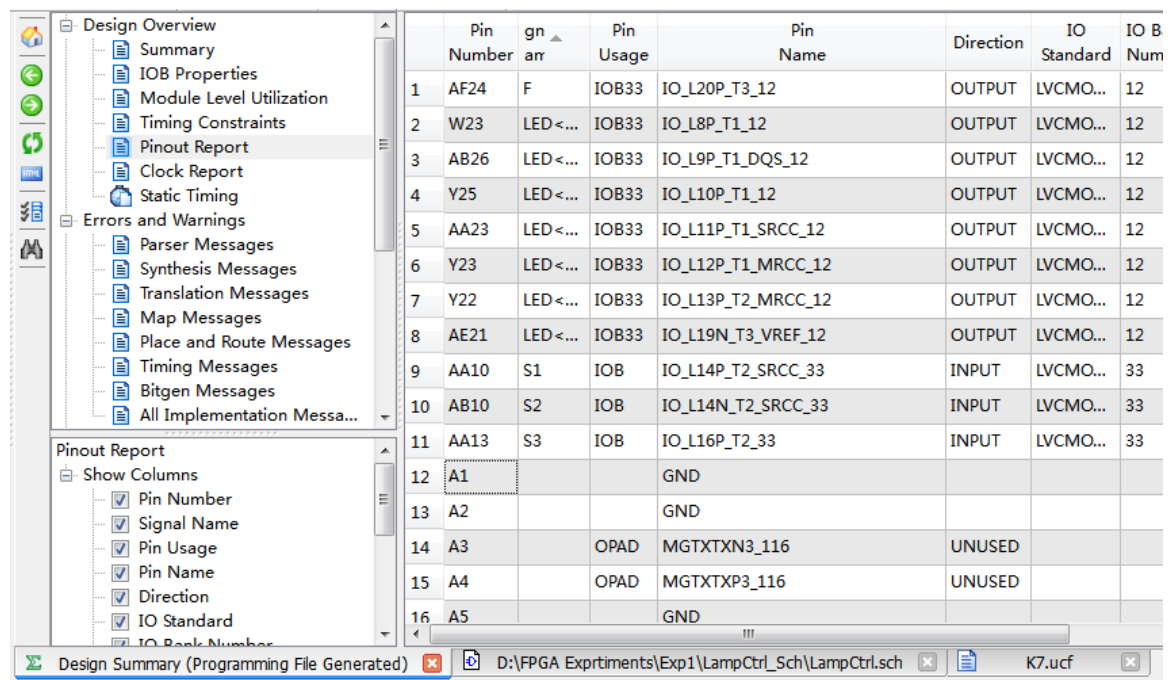
### 建立用户时序约束并为模块的端口指定引脚分配

- 1、在 Sources 窗口空白处的右键菜单中选择 New Source
- 2、在新建源文件向导中选择源类型为：Implementation Constraints File，输入文件名 K7，并勾选 Add to Project
- 3、点击 Finish 进入 K7.ucf 编辑窗口，输入以下代码：

```
NET "S1" LOC = AA10 | IOSTANDARD = LVCMOS15;
NET "S2" LOC = AB10 | IOSTANDARD = LVCMOS15;
NET "S3" LOC = AA13 | IOSTANDARD = LVCMOS15;
NET "F" LOC = AF24 | IOSTANDARD = LVCMOS33;
NET "LED[0]" LOC = W23 | IOSTANDARD = LVCMOS33;
NET "LED[1]" LOC = AB26 | IOSTANDARD = LVCMOS33;
NET "LED[2]" LOC = Y25 | IOSTANDARD = LVCMOS33;
NET "LED[3]" LOC = AA23 | IOSTANDARD = LVCMOS33;
NET "LED[4]" LOC = Y23 | IOSTANDARD = LVCMOS33;
NET "LED[5]" LOC = Y22 | IOSTANDARD = LVCMOS33;
NET "LED[6]" LOC = AE21 | IOSTANDARD = LVCMOS33;
```

设计实现并检查约束结果

在 Sources 窗口中选择 Synthesis / Implementation，选中 lamp\_ctrl；在 Processes 窗口下选择 Implement Design，进行物理转换、平面布图、映射、物理布线等 FPGA 目标格式实现文件生成。最后在设计摘要文档中有如下结果：



	Pin Number	Pin Name	Pin Usage	Direction	IO Standard	IO Bank
1	AF24	IO_L20P_T3_12	IOB33	OUTPUT	LVCMO...	12
2	W23	IO_L8P_T1_12	IOB33	OUTPUT	LVCMO...	12
3	AB26	IO_L9P_T1_DQS_12	IOB33	OUTPUT	LVCMO...	12
4	Y25	IO_L10P_T1_12	IOB33	OUTPUT	LVCMO...	12
5	AA23	IO_L11P_T1_SRCC_12	IOB33	OUTPUT	LVCMO...	12
6	Y23	IO_L12P_T1_MRCC_12	IOB33	OUTPUT	LVCMO...	12
7	Y22	IO_L13P_T2_MRCC_12	IOB33	OUTPUT	LVCMO...	12
8	AE21	IO_L19N_T3_VREF_12	IOB33	OUTPUT	LVCMO...	12
9	AA10	IO_L14P_T2_SRCC_33	IOB	INPUT	LVCMO...	33
10	AB10	IO_L14N_T2_SRCC_33	IOB	INPUT	LVCMO...	33
11	AA13	IO_L16P_T2_33	IOB	INPUT	LVCMO...	33
12	A1	GND				
13	A2	GND				
14	A3	MGTXTXN3_116	OPAD	UNUSED		
15	A4	MGTXTXP3_116	OPAD	UNUSED		
16	A5	GND				

- 1、在 Design 的 View 窗口中选择 Implementation
- 2、在 Sources 窗口中选择 LampCtrl.sch；在 Processes 窗口中，用鼠标点开 Config Target Device，双击 Manage Configuration Project(iMPACT) 选项，出现 IMPACT 窗口
- 3、双击 Boundary Scan 弹出下载编辑窗口
- 4、鼠标右键选择 Initialize Chain，系统自动查找已连接在电脑上的开发平台 JTAG 下载链
- 5、接下来出现 Assign Configuration Files 对话框。这时从文件列表中选择 LampCtrl.bit 文件，将会为 JTAG chain 上的 xc7k160t 设备指定配置文件；在弹出的 Attach SPI or PRI PROM 对话框弹出，点击 NO 按钮；在弹出的“Device Programming Property 对话框，选择 OK 按钮即可。
- 6、右键点击 xc7k160t 设备图标，选择菜单项 Program 后即可对硬件设备进行下载编程
- 7、下载后，验证是否满足设计要求

## 2、图形方式输入逻辑功能描述

建立楼道控制的工程：**LampCtrl\_HDL.isc**

1、依次点击菜单 File → New Project...

2、在对话框中设置如下：

Project Name:        LampCtrl\_HDL

Top-Level Source Type: HDL

3、确认后，点击 Next 到设备属性页，设置：

Family: Kintex7

Device:     XC7K160T

Package:    FFG676

Speed: -1

4、确认后，一直点击 Next 直到创建工程结束

创建 Verilog 输入源文件 **LampCtrl.v**

1、在 Sources 窗口空白处的右键菜单中选择 New Source

2、在新建源文件向导中选择源类型为 Verilog Module, 输入文件名 LampCtrl,

勾选 Add to Project

3、连续点击 Next，直到 Finish.

输入楼道灯控逻辑电路 Verilog HDL 代码

1、在源代码编辑器，输入代码

2、检查输入代码的语法规则，

并排除输入错误

3、延时时间修改

仿真时：

```
parameter C_NUM = 8;
```

```
parameter C_MAX = 8'hFF;
```

下载运行时：

```
parameter C_NUM = 28;
```

```
parameter C_MAX = 28'hFFFFFFFF;
```

楼道控制电路代码的综合

- 1、在 Sources 窗口选中文件 LampCtrl.v;
- 2、在 Processes 窗口运行 Synthesis XST → View RTL Schematic
- 3、检查综合的电路结构是否与设计目标一致

#### 建立基准测试波形文件：LampCtrl\_sim.tbw

- 1、在 Sources 窗口空白处的右键菜单中选择 New Source
- 2、在新建源文件向导中选择源类型为：Verilog Test Fixture，输入文件名 LampCtrl\_sim，并勾选 Add to Project
- 3、点击 Finish 进入 LampCtrl\_sim.v 编辑窗口

#### 仿真激励输入波形

为便于仿真，LampCtrl.v 代码中计数器位数采用的是 8 位长

代码如下：

```
module LampCtrl_sim;
    // Inputs
    reg clk;
    reg S1;
    reg S2;
    reg S3;
    // Outputs
    wire F;

    // Instantiate the Unit Under Test (UUT)
    LampCtrl uut (
        .clk(clk),
        .S1(S1),
        .S2(S2),
        .S3(S3),
        .F(F)
    );
    initial begin
        // Initialize Inputs
        clk = 0;
        S1 = 0; S2 = 0; S3 = 0;

        #600 S1 = 1;
        #20 S1 = 0;
        #6000 S2 = 1;
        #20 S2 = 0;
        #6000 S3 = 1;
        #20 S3 = 0;
    end
endmodule
```

```

end

always begin
    #10 clk = 0;
    #10 clk = 1;
end

endmodule

```

建立用户时序约束并为模块的端口指定引脚分配

1.建立引脚约束文件 k7.ucf，输入代码如下

```

NET "clk" LOC = AC18 | IOSTANDARD = LVCMOS18;
NET "S1" LOC = AA10 | IOSTANDARD = LVCMOS15;
NET "S2" LOC = AB10 | IOSTANDARD = LVCMOS15;
NET "S3" LOC = AA13 | IOSTANDARD = LVCMOS15;
NET "F" LOC = AF24 | IOSTANDARD = LVCMOS33;
NET "LED[0]" LOC = W23 | IOSTANDARD = LVCMOS33;
NET "LED[1]" LOC = AB26 | IOSTANDARD = LVCMOS33;
NET "LED[2]" LOC = Y25 | IOSTANDARD = LVCMOS33;
NET "LED[3]" LOC = AA23 | IOSTANDARD = LVCMOS33;
NET "LED[4]" LOC = Y23 | IOSTANDARD = LVCMOS33;
NET "LED[5]" LOC = Y22 | IOSTANDARD = LVCMOS33;
NET "LED[6]" LOC = AE21 | IOSTANDARD = LVCMOS33;

```

将 LampCtrl.v 代码中计数器位数改成 28 位，修改或增加以下红色部分代码：

```

module LampCtrl(
    input wire clk,
    .....
    output wire [6:0] LED,
    output wire F
);
    parameter C_NUM = 28;
    parameter C_MAX = 28'hFFFFFF;
    .....
assign LED = 7'b0000000; //需增加对应 output 端口

```

-> Synthesize - XST

-> Implement design

-> Generate Programming File

将生成 Bit 文件下载到 SWORD 实验板

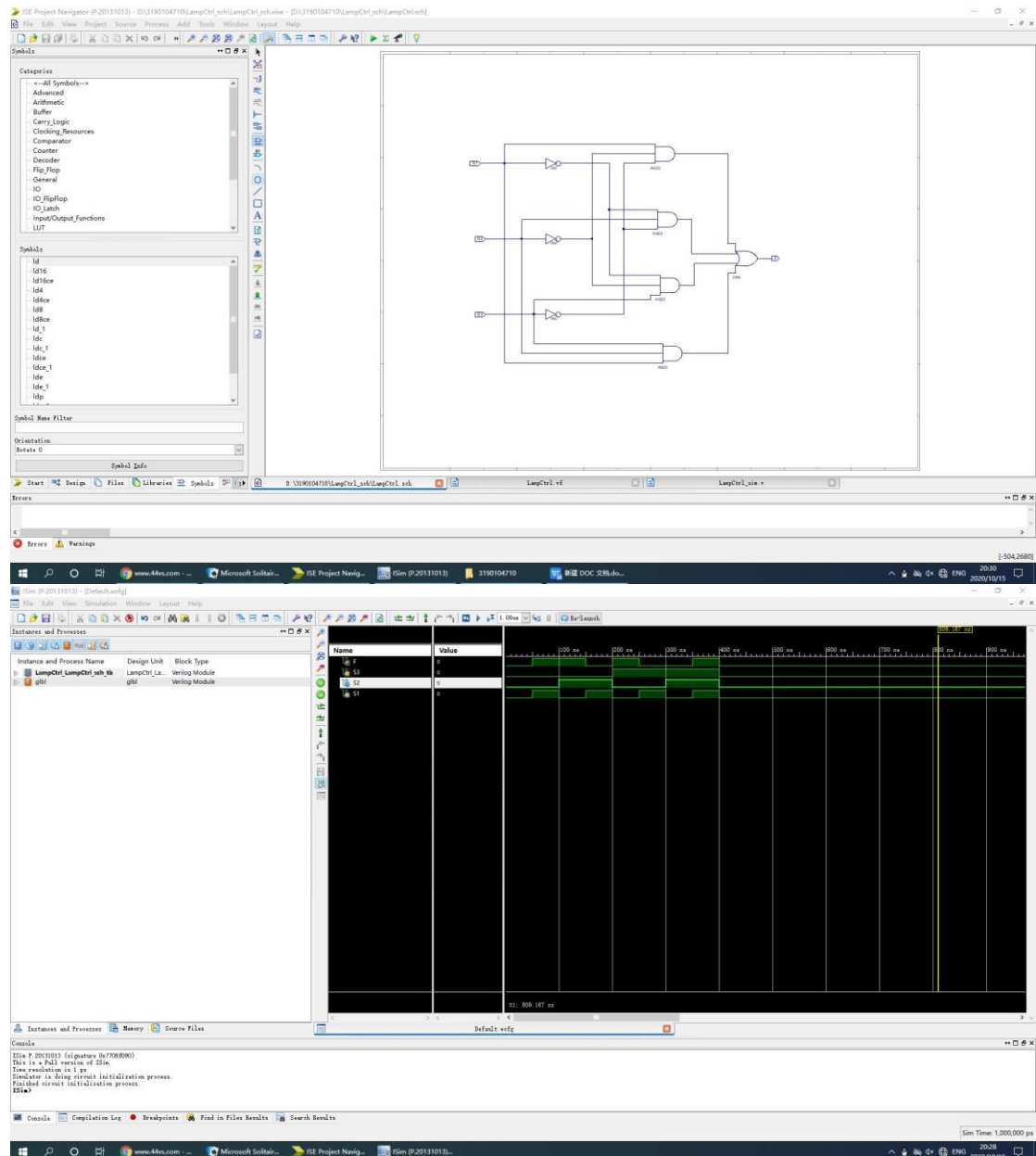
在 SWORD 板上物理运行

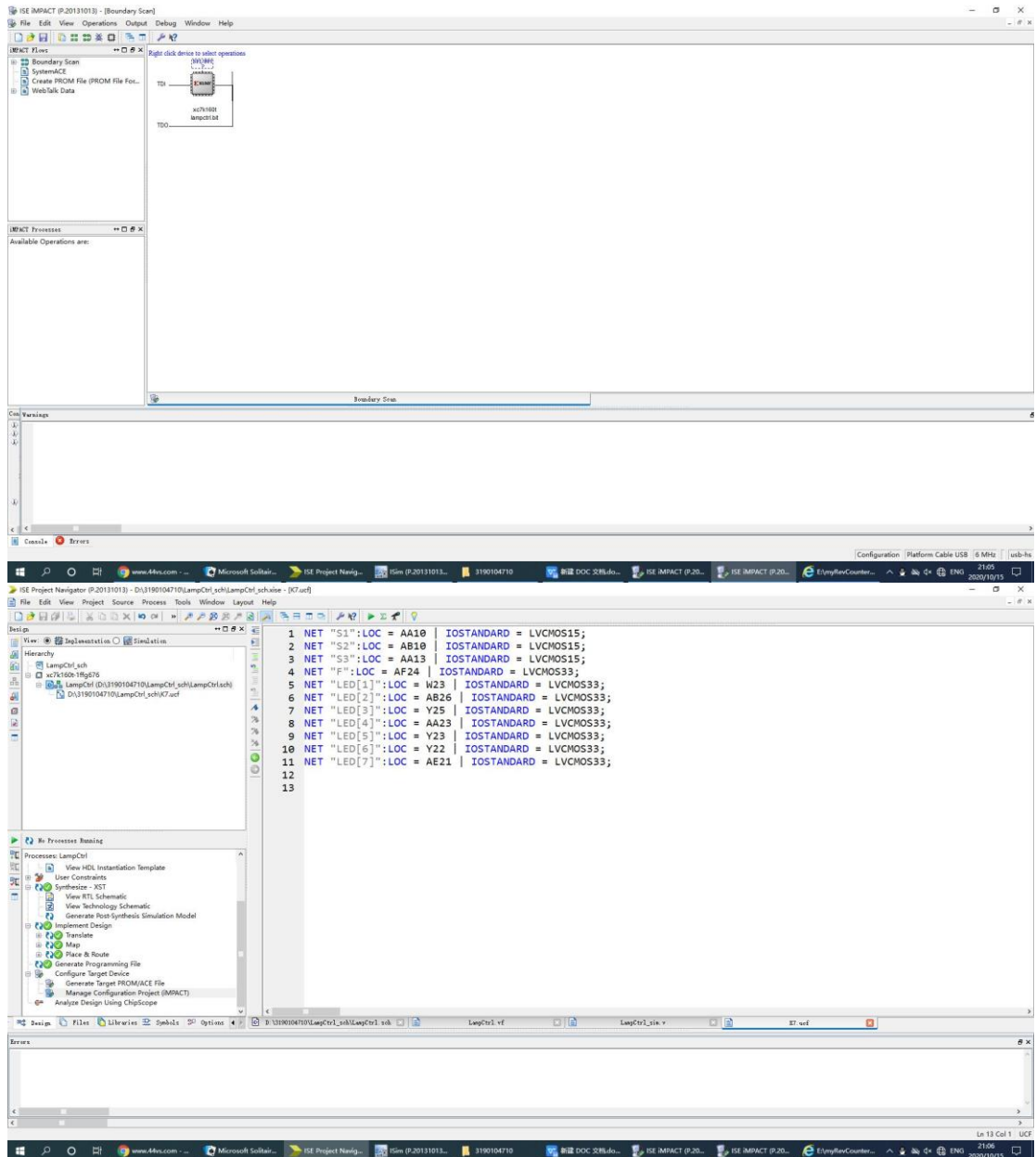
根据 I/O 约束定义和交互按钮操作和显示，板上通过按键开关，查看灯的变化



是否正确，验证设计是否成功。

#### 四、实验结果分析（以下为实验数据图片）







ISE Project Navigator (P.20131013) - D:\3190104710\LampCtrl\_HDL\LampCtrl\_HDL.vise - [lamp\_control (Tech2)]

File Edit View Project Source Process Tools Window Layout Help

Design View Implementation Simulation

Hierarchy

- LampCtrl\_HDL
- src\74162c-16g176
- lamp\_control (D:\3190104710\LampCtrl\_HDL\LampCtrl.v)

Processes lamp\_control

- Design Summary/Reports
- Design Utilities
- User Constraints
- Synthesizer - XST
- View RTL Schematic
- View Technology Schematic
- Check Syntax
- Generate Post-Synthesis Simulation Model
- Implement Design
- Generate Programming File
- Configure Target Device
- Analyze Design Using ChipScope

lamp\_control

clk

S1

S2

S3

LED(6:0)

F

lamp\_control

Errors

Messages View by Category

ISE Project Navigator (P.20131013) - D:\3190104710\LampCtrl\_HDL\LampCtrl\_HDL.vise - [Design Summary (Programming File Generated)]

File Edit View Project Source Process Tools Window Layout Help

Design View Implementation Simulation

Hierarchy

- LampCtrl\_HDL
- src\74162c-16g176
- lamp\_control (D:\3190104710\LampCtrl\_HDL\LampCtrl.v)

Processes lamp\_control

- View RTL Schematic
- View Technology Schematic
- Check Syntax
- Generate Post-Synthesis Simulation Model
- Implement Design
- Generate Programming File
- Configure Target Device
- Analyze Design Using ChipScope

Design Overview

- Summary
- IOB Properties
- Module Level Utilization
- Timing Constraints
- Pinout Report
- Clock Report
- Static Timing
- Errors and Warnings
- Errors Messages
- Synthesis Messages
- Translation Messages
- Map Messages
- Place and Route Messages
- Map Report
- Post-PAE Static Timing Rep...
- Power Report
- Elgen Report
- Secondary Reports
- WebTalk Report
- Design Properties
- Optional Design Summary Contents
- Show Clock Report
- Show Failing Constraints
- Show Warnings
- Show Errors

lamp\_control Project Status (10/16/2008 - 22:01:00)

Project File:	lamp_ctrl.vise	Parser Errors:	0 Errors
Module Name:	lamp_control	Implementation State:	Programming File Generated
Target Device:	xilinx162c-16g176	Errors:	0 Errors
Product Version:	10.1	Warnings:	0 Warnings
Revision:	1.0	Routing Results:	611 Signals Completely Routed
Design Strategy:	Default (unlocked)	Timing Constraints:	611 Constraints Met
Environment:	Default	Final Timing Score:	0 (Signal Report)

Device Utilization Summary

	Used	Available	Utilization	Note(s)
Number of Slice Registers	28	200,000	14	
Number used as Flip Flops	28			
Number used as Latches	0			
Number used as Look-throughs	0			
Number used as AND/OR logic	0			
Number of Slice LUTs	42	101,400	18	
Number used as logic	42	101,400	18	
Number using 06 output only	0			
Number using 06 and 06	7			
Number used as ROM	0	36,000	0%	
Number used as memory	1			
Number used exclusively as route-through	0			
Number with sum-of-products logic	1			
Number with sum-of-products carry logic	0			
Number with other logic	0			
Number of unused Slice LUTs	12	25,360	18	
Number of LUT Flip Flop pairs used	42			
Number with unused Flip Flop	14	42	33%	
Number with unused LUT	0	42	0%	
Number of fully used LUT-FF pairs	28	42	66%	
Number of unused control sets	2			
Number of slice register sites lost to control set restrictions	4	200,000	18	

Errors

Messages View by Category

ISE Project Navigator (P.20131013) - D:\3190104710\LampCtrl\_HDL\LampCtrl\_HDL.vise - [Design Summary (Programming File Generated)]

File Edit View Project Source Process Tools Window Layout Help

Design View Implementation Simulation

Hierarchy

- LampCtrl\_HDL
- src\74162c-16g176
- lamp\_control (D:\3190104710\LampCtrl\_HDL\LampCtrl.v)

Processes lamp\_control

- View RTL Schematic
- View Technology Schematic
- Check Syntax
- Generate Post-Synthesis Simulation Model
- Implement Design
- Generate Programming File
- Configure Target Device
- Analyze Design Using ChipScope

Design Overview

- Summary
- IOB Properties
- Module Level Utilization
- Timing Constraints
- Pinout Report
- Clock Report
- Static Timing
- Errors and Warnings
- Errors Messages
- Synthesis Messages
- Translation Messages
- Map Messages
- Place and Route Messages
- Map Report
- Post-PAE Static Timing Rep...
- Power Report
- Elgen Report
- Secondary Reports
- WebTalk Report
- Design Properties
- Optional Design Summary Contents
- Show Clock Report
- Show Failing Constraints
- Show Warnings
- Show Errors

lamp\_control Project Status (10/16/2008 - 22:01:00)

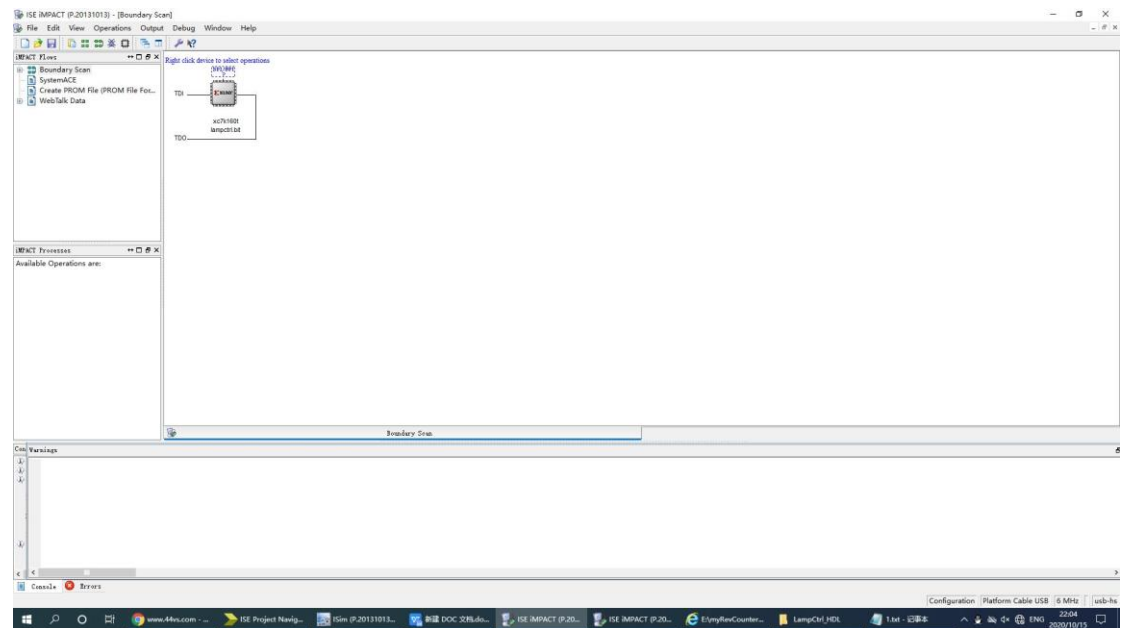
Project File:	lamp_ctrl.vise	Parser Errors:	0 Errors
Module Name: <td>lamp_control</td> <td>Implementation State:<td>Programming File Generated</td></td>	lamp_control	Implementation State: <td>Programming File Generated</td>	Programming File Generated
Target Device: <td>xilinx162c-16g176</td> <td>Errors:<td>0 Errors</td></td>	xilinx162c-16g176	Errors: <td>0 Errors</td>	0 Errors
Product Version: <td>10.1</td> <td>Warnings:<td>0 Warnings</td></td>	10.1	Warnings: <td>0 Warnings</td>	0 Warnings
Revision: <td>1.0</td> <td>Routing Results:<td>611 Signals Completely Routed</td></td>	1.0	Routing Results: <td>611 Signals Completely Routed</td>	611 Signals Completely Routed
Design Strategy: <td>Default (unlocked)</td> <td>Timing Constraints:<td>611 Constraints Met</td></td>	Default (unlocked)	Timing Constraints: <td>611 Constraints Met</td>	611 Constraints Met
Environment: <td>Default</td> <td>Final Timing Score:<td>0 (Signal Report)</td></td>	Default	Final Timing Score: <td>0 (Signal Report)</td>	0 (Signal Report)

Device Utilization Summary

	Used	Available	Utilization	Note(s)
Number of Slice Registers	28	200,000	14	
Number used as Flip Flops	28			
Number used as Latches	0			
Number used as Look-throughs	0			
Number used as AND/OR logic	0			
Number of Slice LUTs	42	101,400	18	
Number used as logic	42	101,400	18	
Number using 06 output only	0			
Number using 06 and 06	7			
Number used as ROM	0	36,000	0%	
Number used as memory	1			
Number used exclusively as route-through	0			
Number with sum-of-products logic	1			
Number with sum-of-products carry logic	0			
Number with other logic	0			
Number of unused Slice LUTs	12	25,360	18	
Number of LUT Flip Flop pairs used	42			
Number with unused Flip Flop	14	42	33%	
Number with unused LUT	0	42	0%	
Number of fully used LUT-FF pairs	28	42	66%	
Number of unused control sets	2			
Number of slice register sites lost to control set restrictions	4	200,000	18	

Errors

Messages View by Category



## 五、讨论与心得

1. ....