

FPGA 实验报告

姓名： 薛宇翔

学号： 20049200057

班级： 2003052

熟悉 Xilinx 开发工具

一、实验介绍

本实验主要介绍 Xilinx 的开发工具——ISE 的使用。

二、实验目标

- 了解 FPGA 的开发流程；
- 熟悉 Spartan-3E 开发套件的功能特点；
- 清楚 PicoBlaze8 位控制器的特性。

三、实验过程

本实验包含五个主要的部分：

1. 使用 ISE 集成环境创建一个新的工程；
2. 添加设计文件到工程中；
3. 编译设计；
4. 仿真设计；
5. 实现设计。

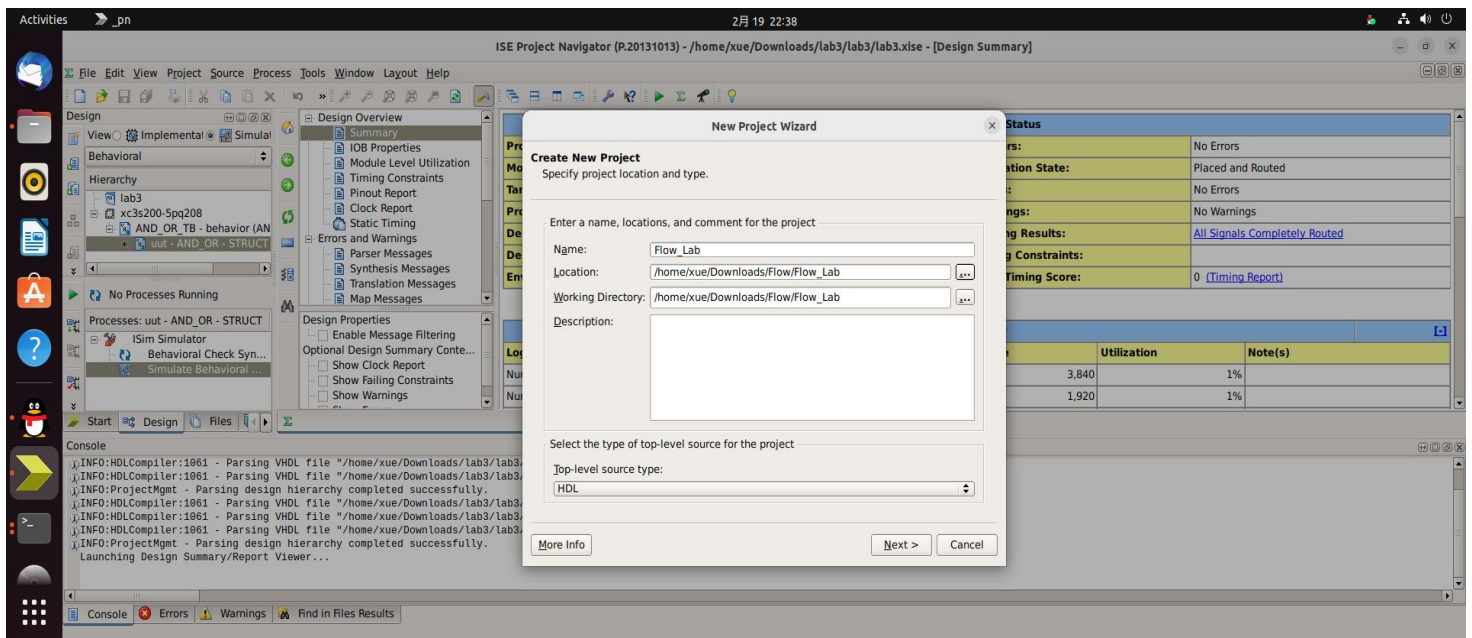
四、实验步骤

首先，阅读 PicoBlaze 的说明文档，以便熟悉 8 位微控制器的体系结构和编译系统，可参考 “.JKCPSM3 docs\” 目录下的 KCPMS3_manual.pdf 文档。本系列实验将使用基于 PicoBlaze 处理器的设计，举例说明 ISE 设计流程及其相关的各个重要工具。

1. 启动 ISE 创建一个新的工程

(1) 选择 “开始→程序→Xilinx ISE 14.7.1i” 或直接在桌面双击 Xilinx ISE 14.7.1i 图标，打开 ISE14.7 集成环境。

(2) 在 ISE 中，选择菜单栏中的 File→New Project，打开创建新工程界面，如图所示。



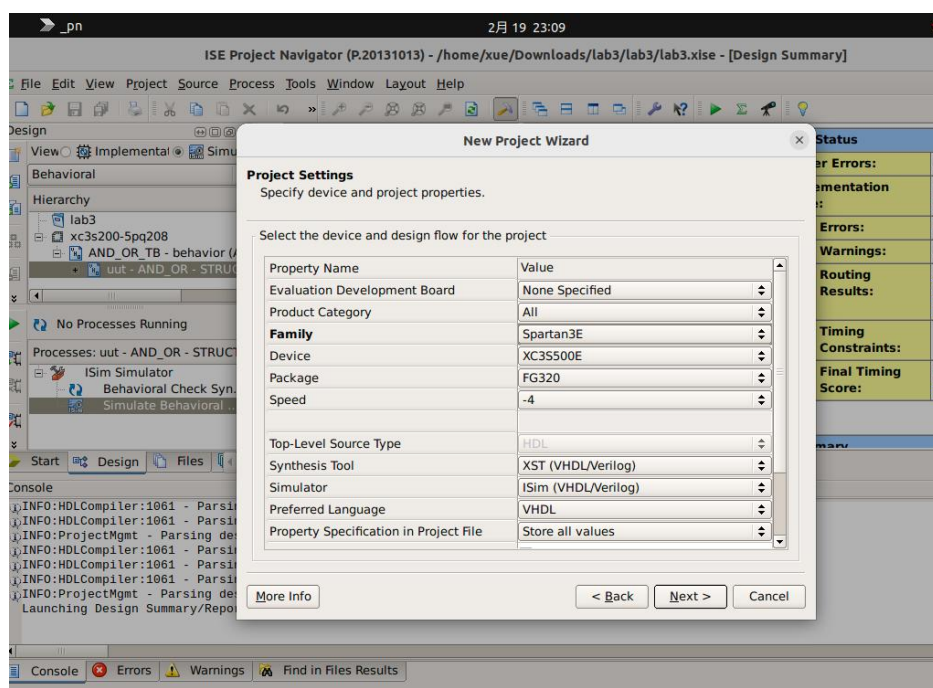
(3) 在工程名称填写栏里输入 Flow Lab; 工程存放位置选择“...”按钮指定到下列目录之一, 然后单击 OK 按钮。

- Verilog \xup\fpgaflow\labs\verilog\lab1;
- VHDL \xup\fpgaflow\labs\vhdl\lab1。

(4) 单击 Next 按钮, 将出现器件和设计工具选择对话框。

(5) 选择下列参数, 并单击 Next 按钮。

- Family (器件类别): Spartan3E;
- Device (器件型号): XC3S500E;
- Package (封装): FG320;
- Speed (速度等级): -4;
- Synthesis Tool (综合工具): XST (VHDL/Verilog);
- Simulator (仿真工具): ISE Simulator (VHDL/Verilog);
- Preferred Language (偏好语言): Verilog or VHDL (select your preference)。



单击 Next 按钮，将会出现创建新文件的对话框，如图 7.3 所示，在这个对话框里可以通过单击 New Source..按钮，按照用户定义的模块名称、端口属性等信息来创建一个新的 HDL 源文件，对文件的全部后续修改只能在 HDL 编辑器中完成。本实验所用到的 HDL 源文件已被创建好了，故此处不用单击 New Source..按钮创建新文件。

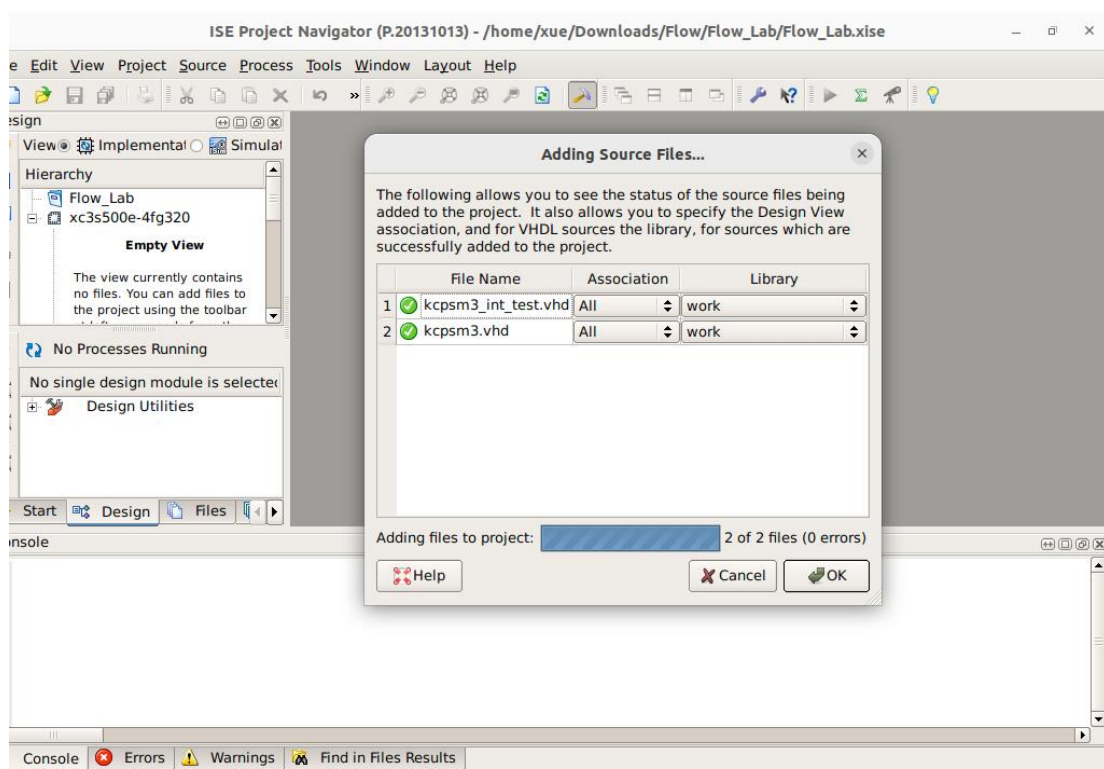
(6) 单击 Next 按钮，将会出现添加已有文件的对话框。

2.添加 HDL 源文件到工程

(1)单击 Add Source, 浏览 \xup\fpgaflow\KCPSM3\VHDL 目录。

(2) 选择 VHDL/Verilog 文件 kcpsm3 int test 和 kcpsm3, 并单击 Open 按钮。

(3) 单击 Next 按钮，然后单击 Finish 按钮，将出现选择源文件类型对话框，如图所示。



(4) 单击 OK 按钮，接受默认设置。

注意：在分层显示列表中，会出现带有红色问号标记名为 `int_test` 的模块。这个模块是个 BlockRAM，它包含 PicoBlaze 控制器的指令信息，这些指令信息将在随后的步骤中被添加。

3. 编译设计

编写 MY_AND2 实体的 VHDL 代码。

(1) 打开 Windows 搜索，在 KCPSM3 子目录里找到编译器。

(2) 使用诸如写字板这样的文本编辑器打开文件 `int_test.psm`，查看代码，可参考 PicoBlaze 8-bit Embedded Microcontroller User Guide 或 KCPSM3 manual 得到技术帮助。这些文档存放在 Docs 子目录中。

(3) 打开命令窗口（开始→程序→附件→命令提示符）。

(4) 使用 `cd` 命令切换到 Assembler 目录。

(5) 在命令提示符下输入以下命令，立即编译代码输出程序 ROM 文件。

```
>kcpsm3 int_test.psm
```

注意：在编译器子目录里，现在应该能看到几个以 `INIT_TEST*` 开头的文件，包括 VHDL (`INT_TEST.VHD`) 和 Verilog (`INT_TEST.V`) 程序 ROM 文件。

(6) 在 ISE 集成环境里，打开 Project→Add Copy of Source 并找到 `INT_TEST.VHD` 或 `INT_TEST.V` 文件

(7) 单击 Open 按钮，然后单击 OK 按钮把 INIT_TEST 当作设计文件添加到工程。

4. 仿真设计

添加测试文件 testbench.vhd/.v 并阅读其代码。使用 Xilinx iSIM 仿真工具完成行为级仿真并检查仿真结果。

(1) 在 ISE 的 Sources 窗口中，选择 Project→Add Copy of Source 并找到 vhd1 目录。

(2) 选择文件 test bench.vhd (或 testbench.v) 并单击 Open 按钮。

(3) 选择 Simulation Only 并单击 OK 按钮，添加测试文件到工程。

(4)选中 testbench 测试文件,在 Processes 窗口中展开 Xilinx ISE Simulator 仿真工具，右键单击 Simulate Behavioral Model 并选择属性 Properties。

(5) 在 Simulation Run Time 选项下输入值 25000ns 并单击 OK 按钮。

(6) 双击 Simulate Behavioral Model 仿真设计。仿真结束后会出现两个窗口。一个窗口以波形方式显示仿真结果，另一个以 HDL 格式显示 testbench 文件。

(7)单击 waveform 查看仿真结果，缩放并仔细观察波形确认仿真结果准确性。

(8) 关闭窗口，退出仿真。

5. 实现设计

- (1) 在 Sources 窗口中，选择 Synthesis/Implementation，并选中顶层设计文件 kcpsm3_int_test.v。
- (2) 在 Processes 窗口中双击 Implement Design，ISE 工具会运行所有需要的过程来实现设计，在实现之前首先运行综合。
- (3) 在实现过程中，单击 Implement Design 旁边的“+”展开实现步骤，查看进展。每个阶段完成后，都会出现一个对应符号：
- 对号表示正确；
 - 感叹号表示警告；
 - X 表示错误。
- 对于这个设计，在某些步骤中可能会出现一个感叹号（警告），这里的警告是可以忽略的。
- (4) 在 ISE 集成环境的底部消息窗口阅读消息。
- (5) 实现结束后，在 Design Summary 窗口中查看设计的资源利用情况。

| | | | |
|------------------|-----------------|----------------|------------------------------|
| Project File: | Flow_Labise | Current State: | Placed and Routed |
| Module Name: | kcp3m3_int_test | • Errors: | No Errors |
| Target Device: | xc3e500e-4fg320 | • Warnings: | 346 Warnings |
| Product Version: | ISE, 9.1.0.31 | • Updated: | Mon May 15 09:43:03 2006 |

| Device Utilization Summary | | | | |
|--|--------|-----------|-------------|---------|
| Logic Utilization | Used | Available | Utilization | Note(s) |
| Number of Slice Flip Flops | 76 | 9,312 | 1% | |
| Number of 4 input LUTs | 107 | 9,312 | 1% | |
| Logic Distribution | | | | |
| Number of occupied Slices | 99 | 4,656 | 2% | |
| Number of Slices containing only related logic | 99 | 99 | 100% | |
| Number of Slices containing unrelated logic | 0 | 99 | 0% | |
| Total Number 4 input LUTs | 177 | 9,312 | 1% | |
| Number used as logic | 107 | | | |
| Number used as a route-thru | 2 | | | |
| Number used for Dual Port RAMs | 16 | | | |
| Number used for 32x1 RAMs | 52 | | | |
| Number of bonded I/Os | 18 | 232 | 7% | |
| IDB Flip Flops | 16 | | | |
| Number of Block RAMs | 1 | 20 | 5% | |
| Number of GDLKs | 1 | 24 | 4% | |
| Total equivalent gate count for design | 74,954 | | | |
| Additional TAG gate count for I/Os | 864 | | | |

| Performance Summary | | | |
|---------------------|---|--------------|-------------------------------|
| Final Timing Score: | 0 | Pinout Data: | Pinout Report |
| Routing Results: | All Signals Completely Routed | Clock Data: | Clock Report |
| Timing Constraints: | All Constraints Met | | |

| Detailed Reports | | | | | |
|--|---------|--------------------------|--------|------------------------------|-------------------------|
| Report Name | Status | Generated | Errors | Warnings | Infos |
| Synthesis Report | Current | Mon May 15 09:42:22 2006 | 0 | 346 Warnings | 0 |
| Translation Report | Current | Mon May 15 09:42:28 2006 | 0 | 1 Warning | 0 |
| Map Report | Current | Mon May 15 09:42:38 2006 | 0 | 0 | 2 Infos |
| Place and Route Report | Current | Mon May 15 09:43:01 2006 | 0 | 0 | 2 Infos |

五、实验总结与反思

本次实验中，创建了一个简单的逻辑门模块，并且在上层的模块中调用这个模块，使用信号将模块之间相连。通过这个实验，接触了解了一些 VHDL 的基本技术和方法，在之后的应用中将会有更多的使用。