


1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42



# 计算机组成与结构实验指导书-LAB2

 上海交通大学 计算机科学与工程系 Dept. of Computer Science & Engineering	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1 作者 WnSN Lab	CSE-COA-LAB-001 修改日期 9/21/2012	0.3	1 of 17  公开

# 1. OVERVIEW

## 1.1 实验名称

### FPGA 基础实验 2: 4 bits binary counter with time divider

## 1.2 实验目的

1. 掌握 Xilinx 逻辑设计工具 ISE 的设计流程
2. 初步掌握使用 VerilogHDL 硬件描述语言进行简单的逻辑设计
3. 掌握 UCF（用户约束文件）的用法和作用
4. 熟悉 Xilinx Spartan 3E 开发板。
5. 学习分频器的编写，了解同步信号的工作过程。

## 1.3 实验范围

本次实验将覆盖以下范围

1. ISE13.4 的使用
2. 使用 VerilogHDL 进行逻辑设计
3. 编辑 UCF
4. iMPACT 的使用
5. Spartan 3E 实验板的使用

## 1.4 实验预计时间


120 分钟

## 1.5 实验报告与验收办法

本实验不需提交纸质报告，实验完毕需验收登记作为平时成绩

## 1.6 注意事项

1. 本实验的逻辑设计工具为 Xilinx ISE13.4，但不仅如此，学生可以使用自己喜欢的逻辑设计工具，如 Snyplify 等。

 上海交通大学 计算机科学与工程系 Dept. of Computer Science & Engineering	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	2 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		

## 2. 四位二进制计数器

### 2.1 编号

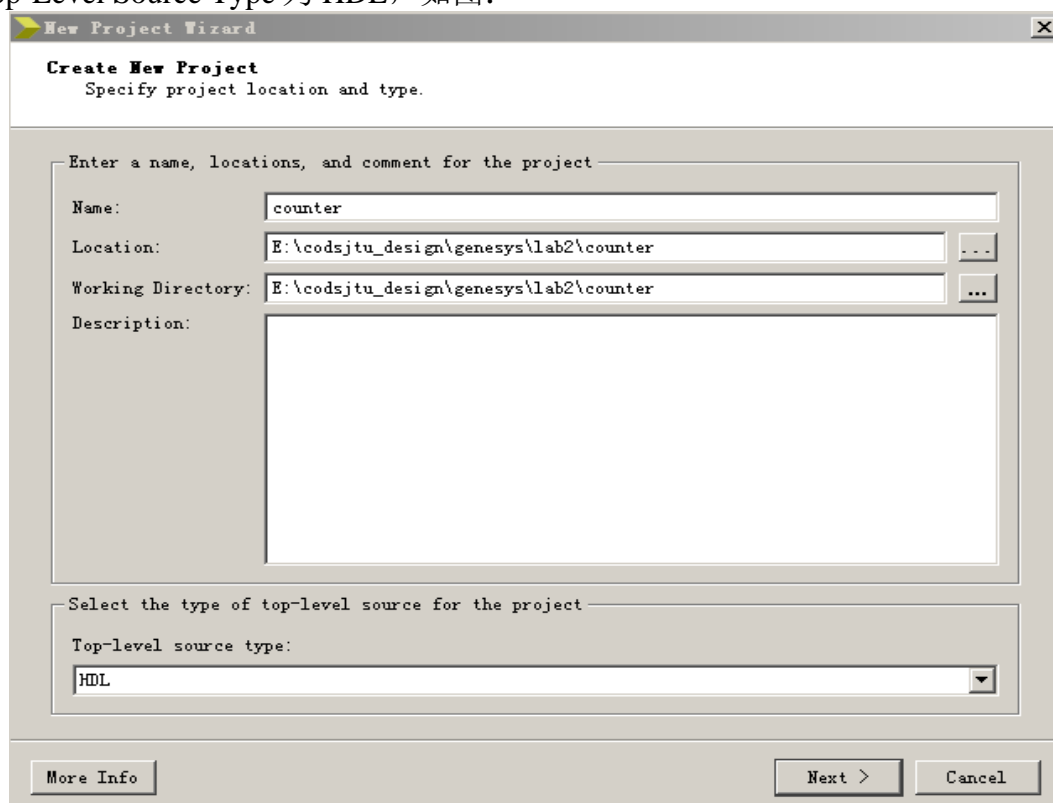
### 2.2 实验描述


#### 2.2.1 实验要求:

1. 用 Verilog 实现一个简单的带同步复位的 4 位 2 进制计数器
2. 使用某种仿真器对该计数器进行仿真，本例中采用 Xilinx ISE Simulator，也可使用 ModelSim
3. 对本实验板 FPGA 提供的高速时钟进行分频，为计数器提供慢速时钟。
4. 将最终得到的解决方案使用 Xilinx ISE13.4 进行综合，并在实验板上得到预期结果。

#### 2.2.2 新建工程

1. 运行 Xilinx ISE13.4。
2. 菜单栏 File->New Project 弹出以下对话框，指定要创建的如下工程目录及名称，确保 Top-Level Source Type 为 HDL，如图：




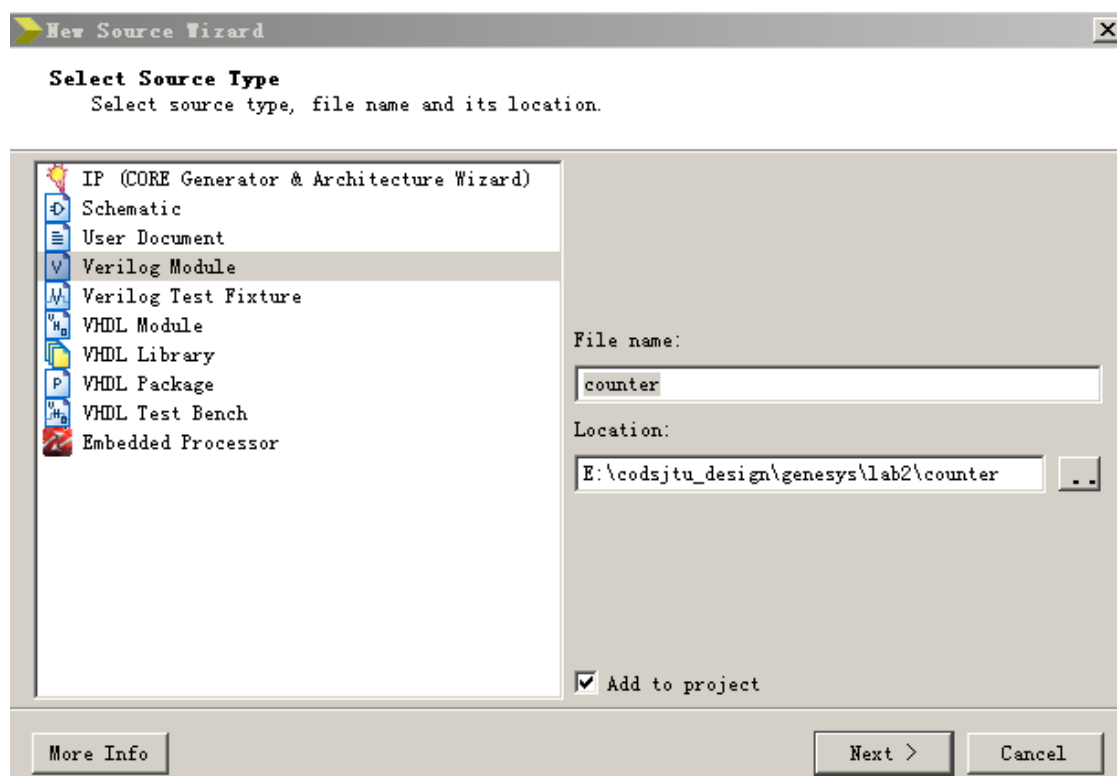
 上海交通大学 计算机科学与工程系 Dept. of Computer Science & Engineering	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	3 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		

点击 Next，查找 FPGA 版上芯片类型，并在下拉菜单中选择相应的型号（本例中使用 Spartan-3E, XC3S500E, FG320），仿真器 simulator 可选用自己喜欢的，本例中使用 ISE 自带仿真器。Preferred Language 选用 Verilog，如图：

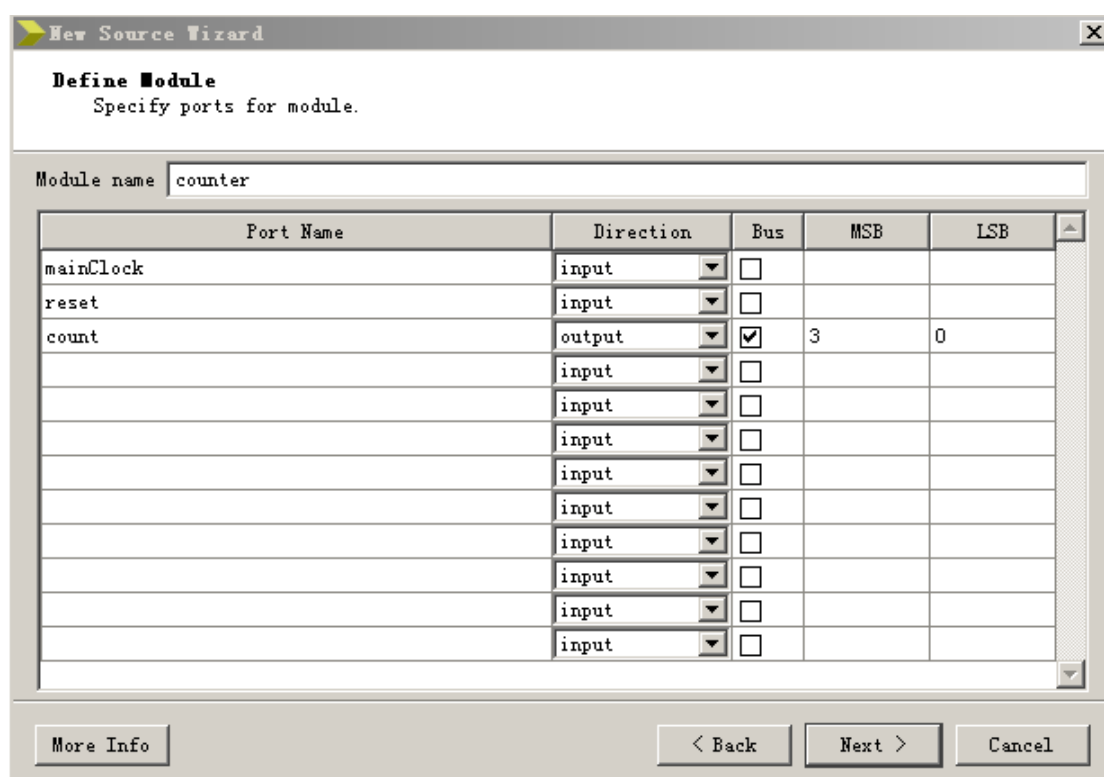
Property Name	Value
Evaluation Development Board	None Specified
Product Category	All
Family	Spartan3E
Device	XC3S500E
Package	FG320
Speed	-4
Top-Level Source Type	HDL
Synthesis Tool	XST (VHDL/Verilog)
Simulator	ISim (VHDL/Verilog)
Preferred Language	Verilog
Property Specification in Project File	Store all values
Manual Compile Order	<input type="checkbox"/>
VHDL Source Analysis Standard	VHDL-93
Enable Message Filtering	<input type="checkbox"/>


3. 点击 Next，核查信息并 Finish。
4. 选中 counter 右键点击 NewSource...：在对话框中选择 Verilog Moudle 类型文件，命名为 counter，并勾选 Add to Project，如图：

 上海交通大学 计算机科学与工程系 Dept. of Computer Science & Engineering	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	4 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		

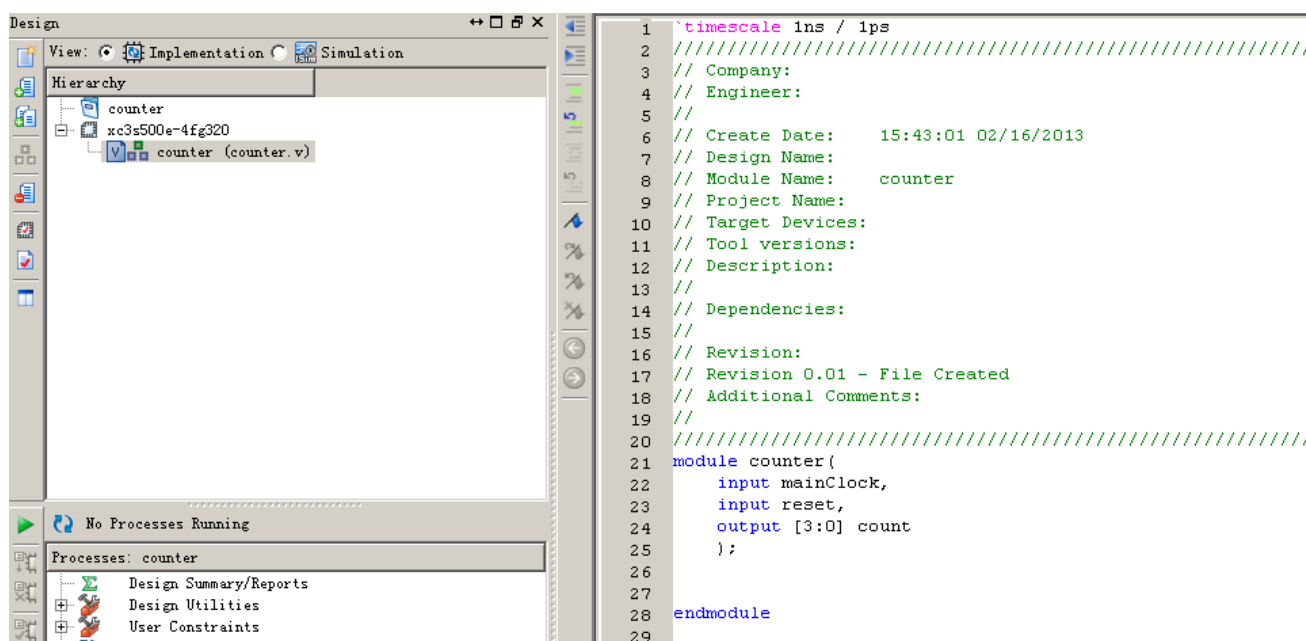


5. 点击 Next，添加输入输出端口（注意要设置 Direction），如图：



	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	5 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		

6. 点击 Next。
7. 查看信息无误后，点击 Finish。
8. 完成新代码文件的创建。如图：




## 2.2.3 编写 Verilog 代码

修改模块 counter.v 的代码并保存，实现带同步复位的 4 Bits 计数器，如图：

```

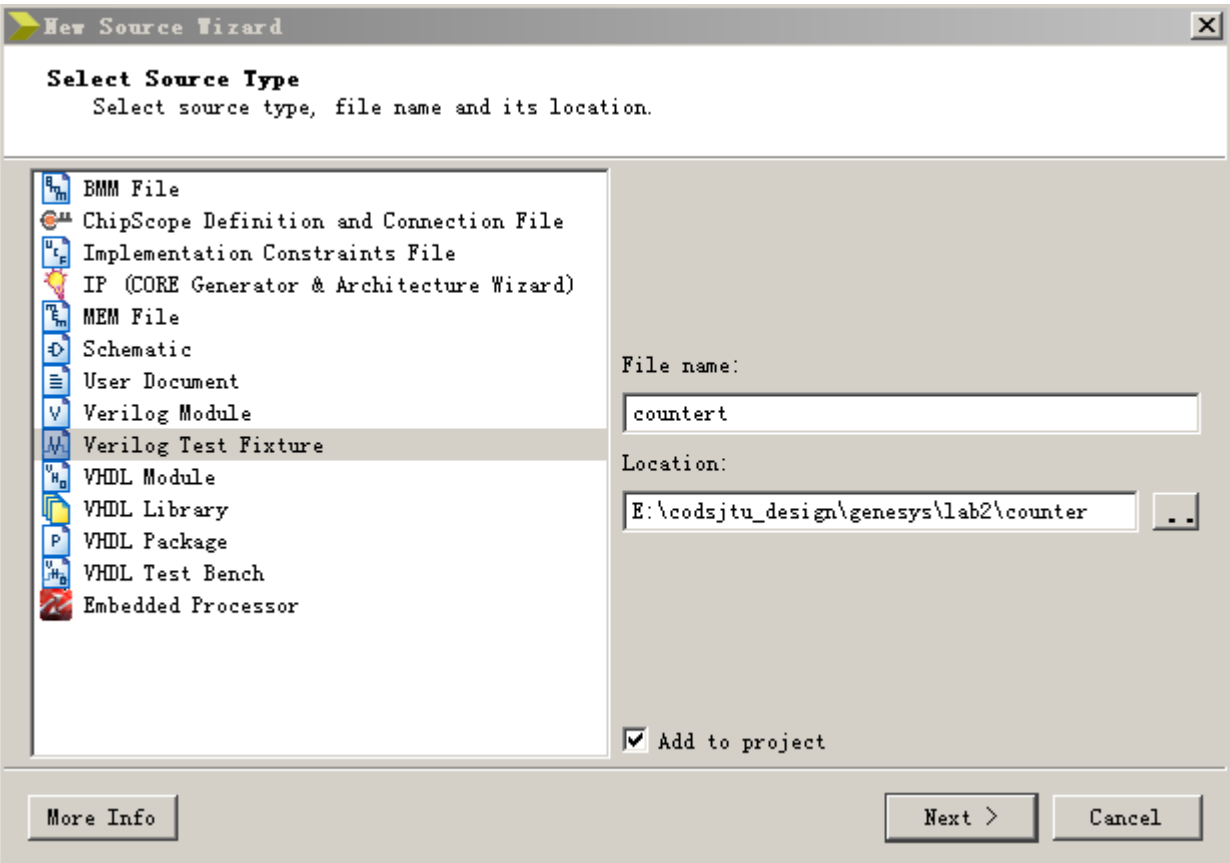
20 //////////////////////////////////////////////////
21 module counter(
22     mainClock,
23     reset,
24     count,
25 );
26     input mainClock;
27     input reset;
28     output [3:0] count;
29     reg [3:0] count;
30
31     always @(posedge mainClock)
32     begin
33         if (!reset)
34             count <= 4'b1111;
35         else
36             count <= count-1;
37     end
38
39 endmodule

```

	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	6 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		

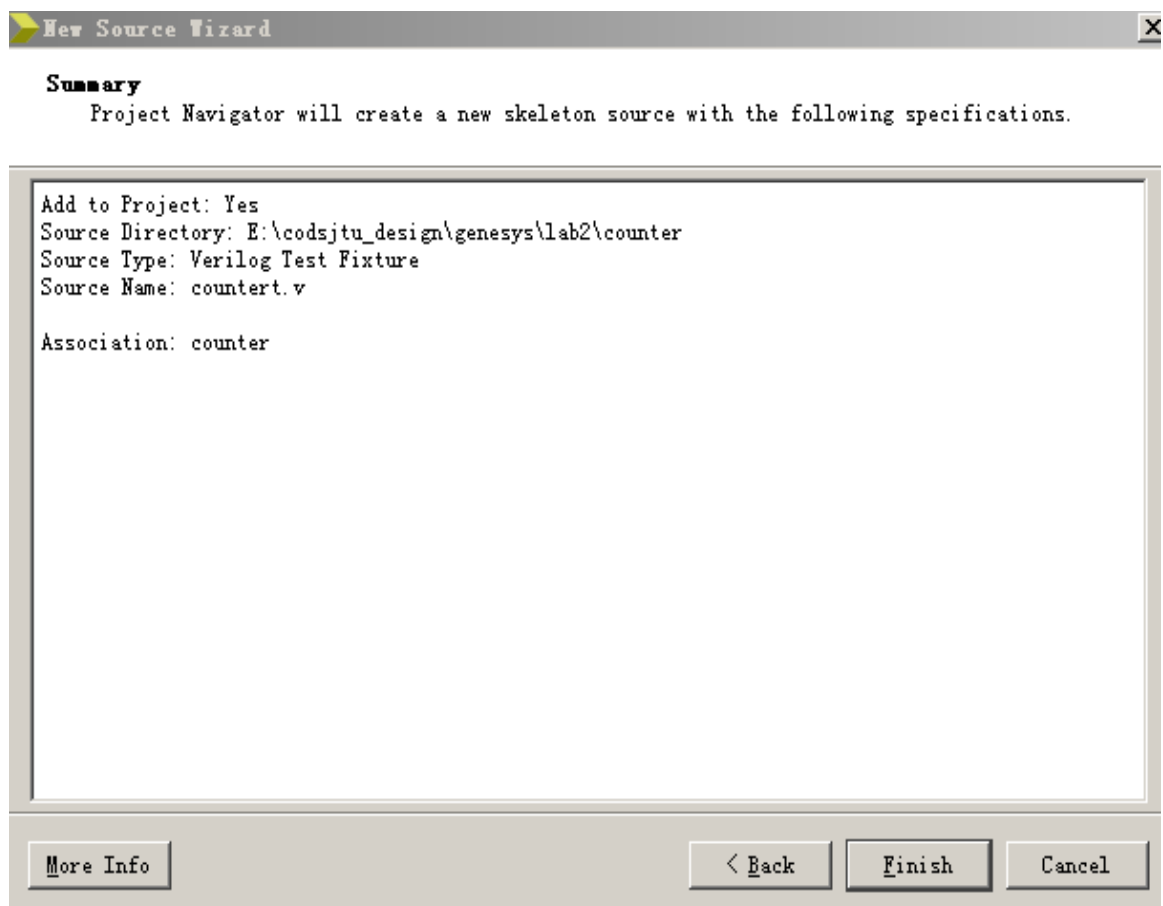
2.2.4 添加 Testbench 仿真文件进行行为级仿真

1. 在 Hierarchy 中选中 counter.v，菜单栏中选择 Project-> New Source，选择 Verilog Test Fixture，输入文件名 countert，点击 Next，如图：




2. 选择 counter 模块，点击 Next直到点击 Finish关闭 Summary 如下对话框，ISE 将生成 countert.v 模块的代码。

	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	7 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		



3. 在 View 栏中选中 Simulation，添加修改新生成的 countert.v 模块，使其如下图所示：

 上海交通大学 计算机科学与工程系 Dept. of Computer Science & Engineering	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	8 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		

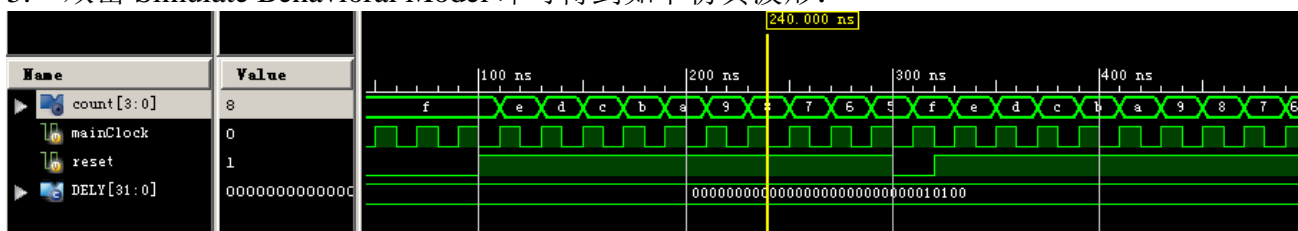


```

34     parameter DELY=20;
35     // Instantiate the Unit Under Test (UUT)
36     counter uut (
37         .mainClock(mainClock),
38         .reset(reset),
39         .count(count)
40     );
41
42     always #(DELY/2) mainClock = ~mainClock;
43     initial begin
44         // Initialize Inputs
45         mainClock = 0;
46         reset = 0;
47
48         // Wait 100 ns for global reset to finish
49         #100;
50
51         // Add stimulus here
52         reset=1'b1;
53
54         #(DELY*10) reset=1'b0;
55         #(DELY) reset=1'b1;
56
57     end

```

4. 在 Hierarchy 中选中 countert.v, 在 Processes 中右键点击 Simulate Behavioral Model 然后选中 Process Properties..., 更改 Simulation Run Time 值为 500ns, 点击 OK 保存。
5. 双击 Simulate Behavioral Model 即可得到如下仿真波形:




仿真波形

## 2.2.5 分频器的编写与添加

由于板子晶振提供的时钟频率速度较高, 需要对时钟进行分频, 可通过 TimerDivider 模块来实现。

1. 在 View 栏中选中 Implementation, 双击打开 counter.v。
2. 添加修改 counter.v 模块代码如图所示并保存:

 上海交通大学 计算机科学与工程系 Dept. of Computer Science & Engineering	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	9 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		

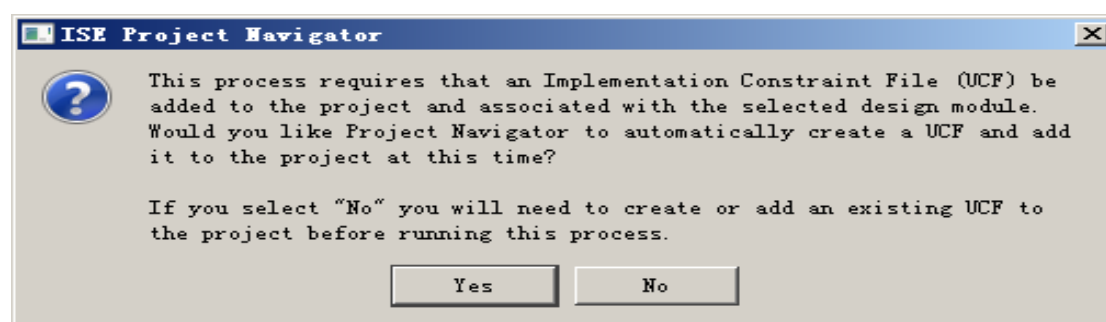
```

21 module timerDivider(
22     clockIn,
23     clockOut
24 );
25     input clockIn;
26     output clockOut;
27     reg clockOut;
28
29     reg[23:0] buffer;
30
31     always@(posedge clockIn)
32     begin
33         buffer <= buffer+1;
34         clockOut <= $buffer;
35     end
36
37 endmodule
38
39 module counter(
40     mainClock,
41     reset,
42     count
43 );
44     input mainClock;
45     input reset;
46     output count;
47     reg[3:0] count;
48     wire slowClock;
49
50     timerDivider td(.clockIn(mainClock), .clockOut(slowClock));
51
52     always@(posedge slowClock)
53     begin
54         if(!reset)
55             count <= 4'b1111;
56         else
57             count <= count-1;
58     end
59
60 endmodule
61


```

## 2.2.6 指定管脚位置约束

1. 在 Hierarchy 窗口中选 counter.v，在 Processes 窗口选择 User Constraints 的展开项，双击 Creat Timing Constraints，弹出如下窗口提示创建 ucf 文件，点 Yes。



2. 关闭 Timing Constraints 窗口，选中 Hierarchy 窗口创建的 counter.ucf，在 Processes 窗口双击 Edit Constraints (Text) 打开文本编辑器，在该 UCF 文件中输入端口映射的配置，将输出映射至 LED，如图：

 上海交通大学 计算机科学与工程系 Dept. of Computer Science & Engineering	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	10 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		

```

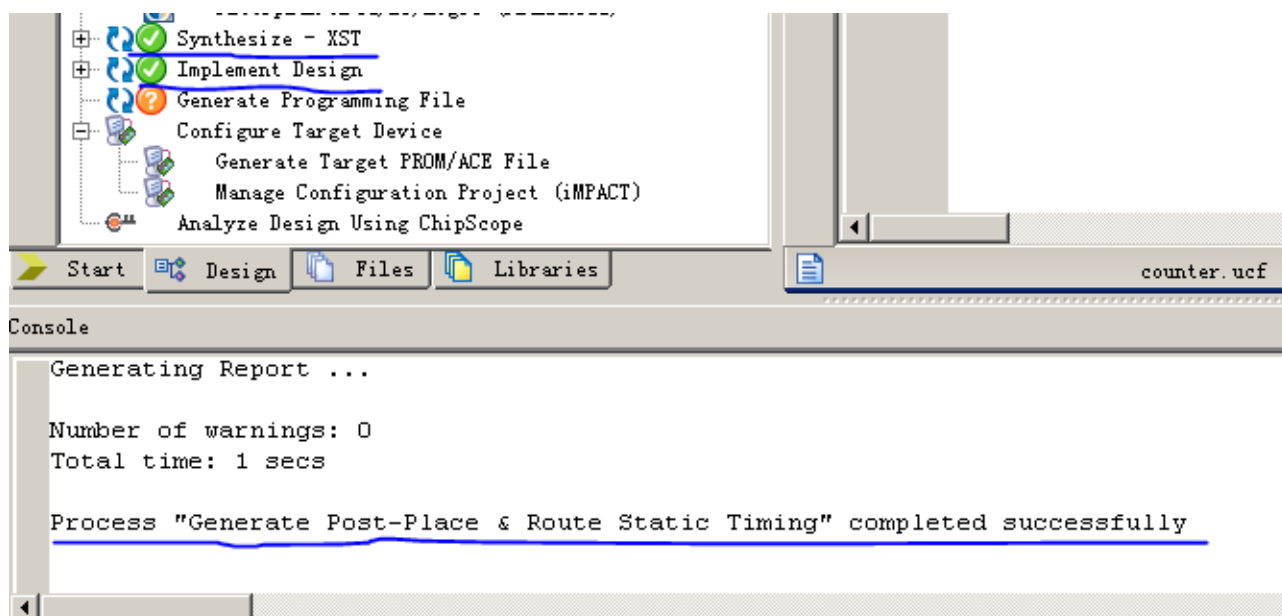
1 NET "mainClock" LOC = "C9" | IOSTANDARD = LVCMOS33;
2 NET "count<3>" LOC = "F11" | IOSTANDARD = LVTTTL | SLEW = SLOW | DRIVE = 8 ;
3 NET "count<2>" LOC = "E11" | IOSTANDARD = LVTTTL | SLEW = SLOW | DRIVE = 8 ;
4 NET "count<1>" LOC = "E12" | IOSTANDARD = LVTTTL | SLEW = SLOW | DRIVE = 8 ;
5 NET "count<0>" LOC = "F12" | IOSTANDARD = LVTTTL | SLEW = SLOW | DRIVE = 8 ;
6 NET "reset" LOC = "L13" | IOSTANDARD = LVTTTL | PULLUP ;

```


3. 点击工具栏保存。

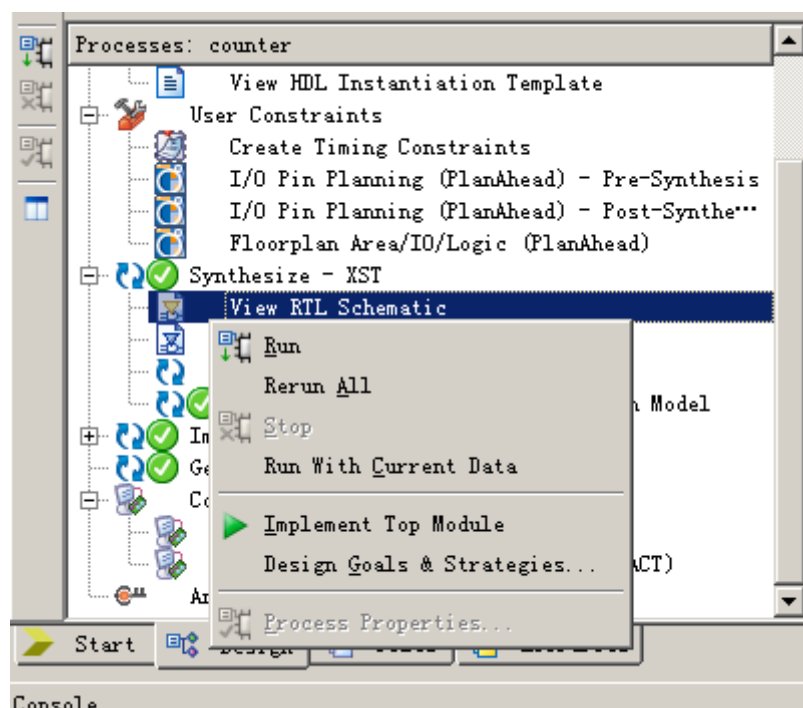
## 2.2.7 综合编译硬件设计

1. 在菜单栏 Process 下点击 Implement Top Module 或其它方法运行之，ISE 会调用后台工具对工程进行综合和布线。此过程较慢，请注意下方输出窗口中的返回信息。如果成功完成，会有如下图提示：

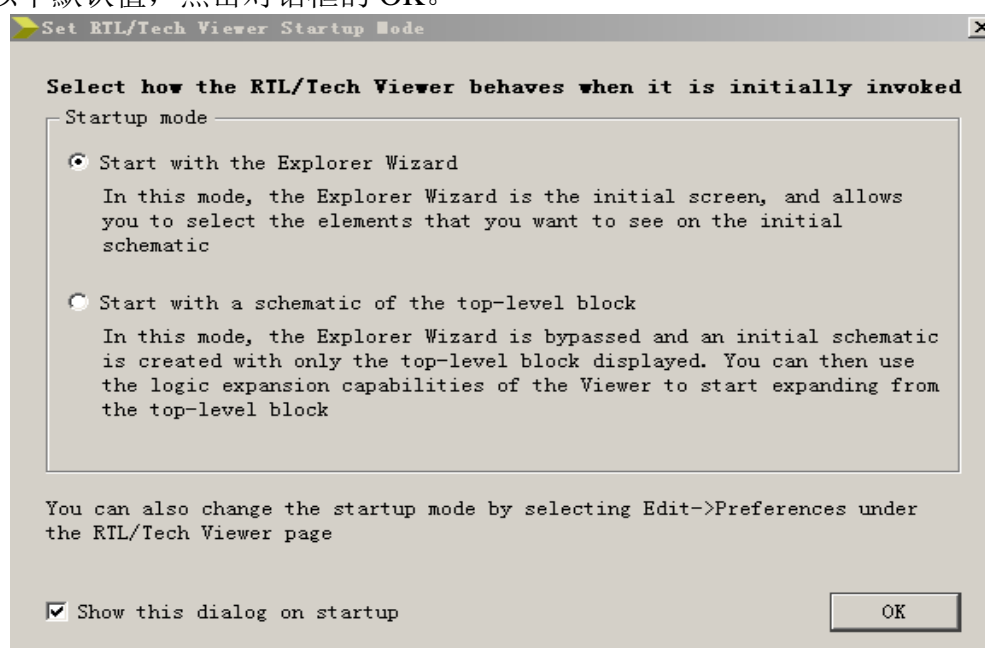


2. 选中 Hierarchy 窗口的 counter.v，在 Processes 窗口选择 Synthesize – XST 展开项的 View RTL Schematic，右键或双击运行。


 上海交通大学 计算机科学与工程系 Dept. of Computer Science & Engineering	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	11 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		

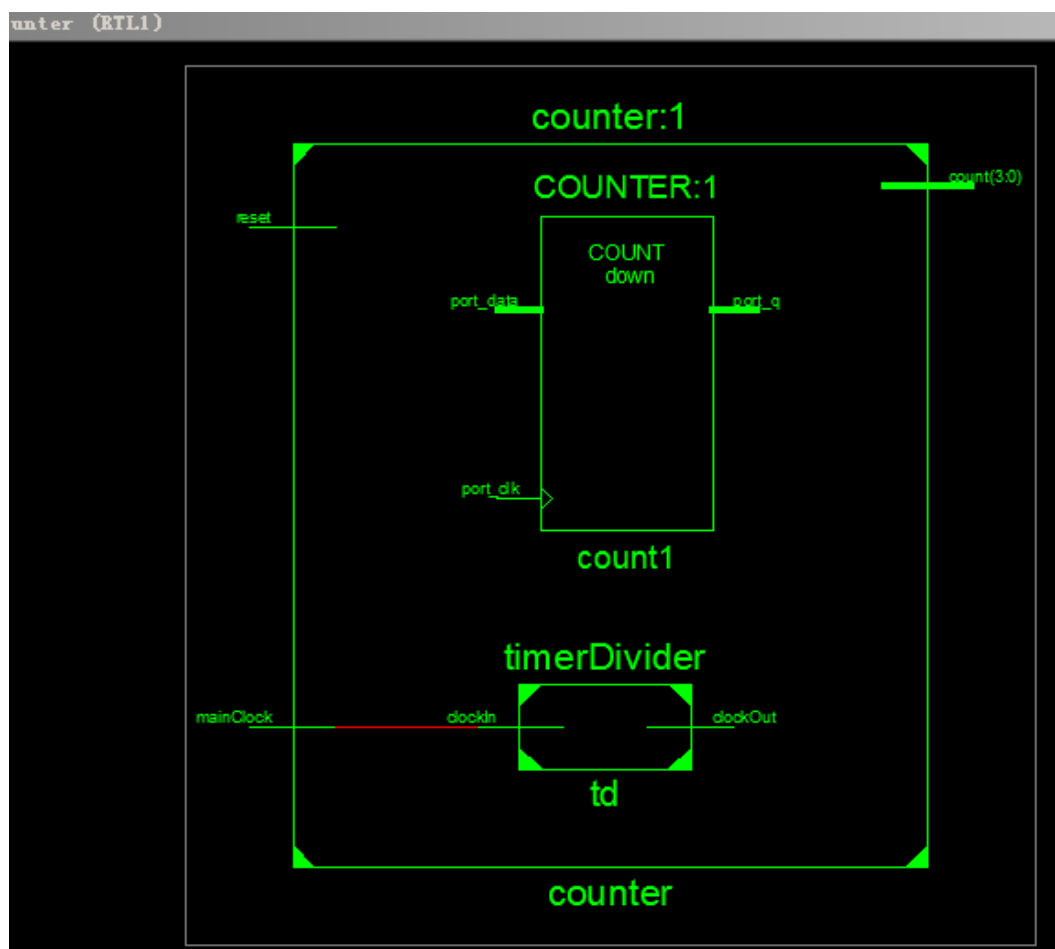


3. 保留以下默认值，点击对话框的 OK。



4. 在 Create RTL Schematic 中，选择有效元素，综合后生成 RTL 原理分析图可供查看，可用以检查代码对应的电路逻辑是否符合设计所需：

 上海交通大学 计算机科学与工程系 Dept. of Computer Science & Engineering	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	12 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		



5. 双击 Processes 中 Generate Programming File, 系统会自动生成二进制 bit 流文件 counter.bit, 用于 FPGA 的配置 :

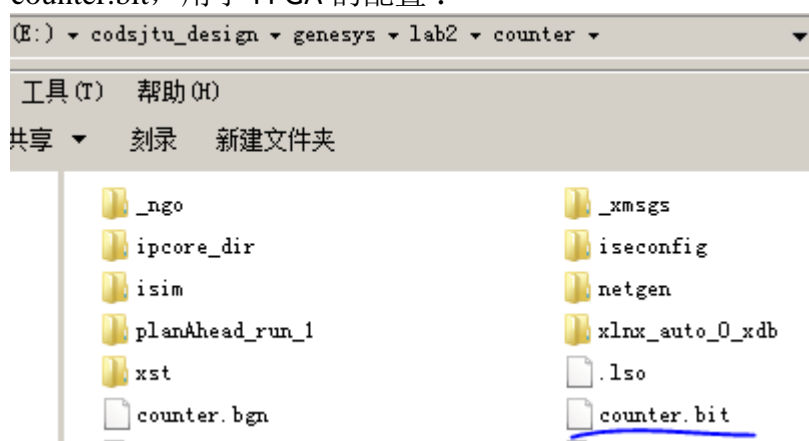


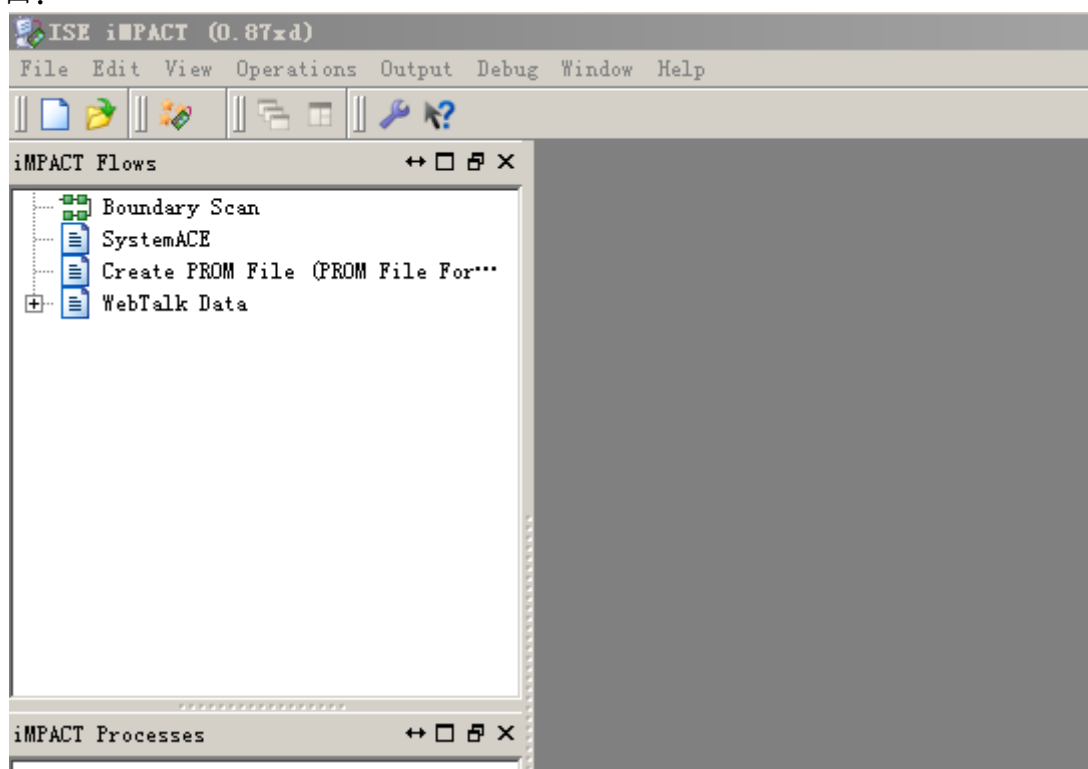
图 2-17

## 2.2.6 下载到开发板验证

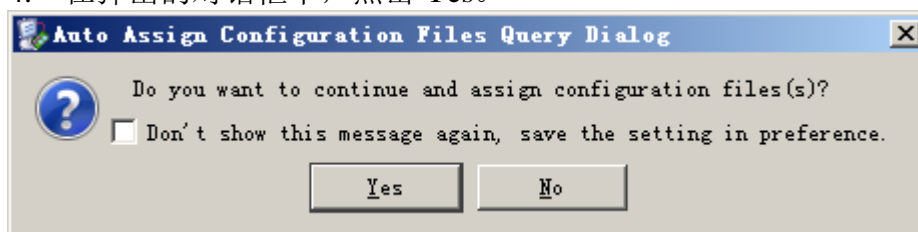
1. 将 FPGA 实验板接入计算机。


	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	13 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		

2. 双击 Configure Target Device 下的 Manage Configuration Project(iMPACT)。会弹出如下图窗口：

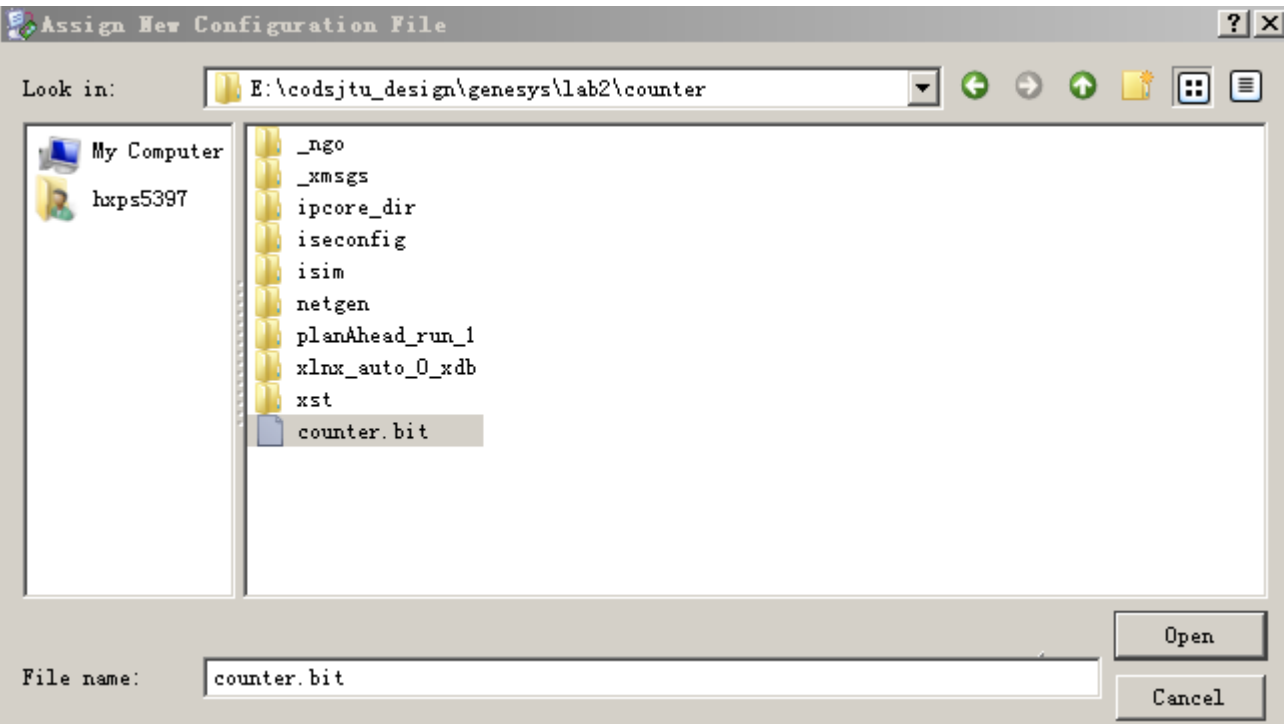


3. 双击 Boundary Scan ， 在 iMPACT右侧窗口右键选择： Initialize Chain。
4. 在弹出的对话框中， 点击 Yes。



	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	14 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		

5. 在弹出的对话框中，选择刚刚生成的 counter.bit 文件，点击 Open。




6. 弹出对话框，点击 No。

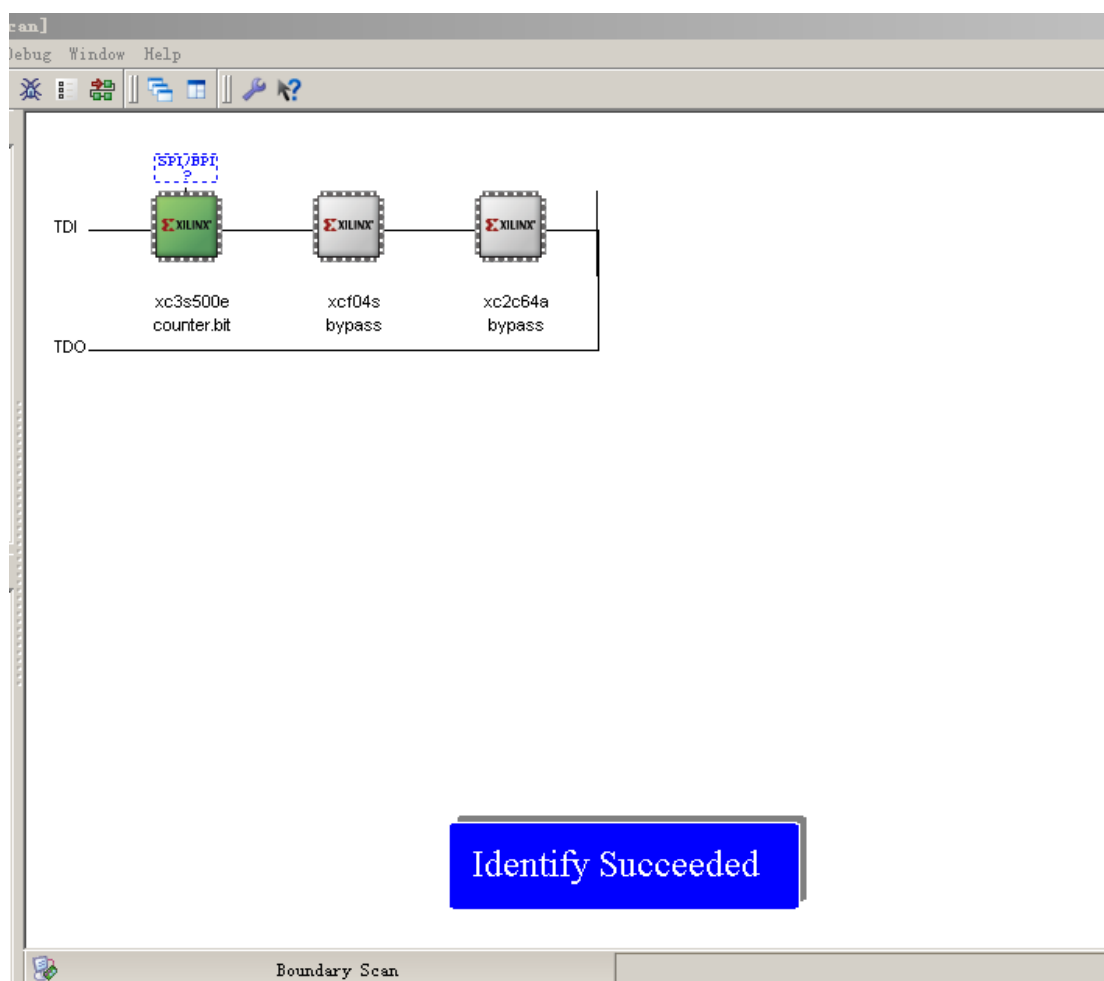
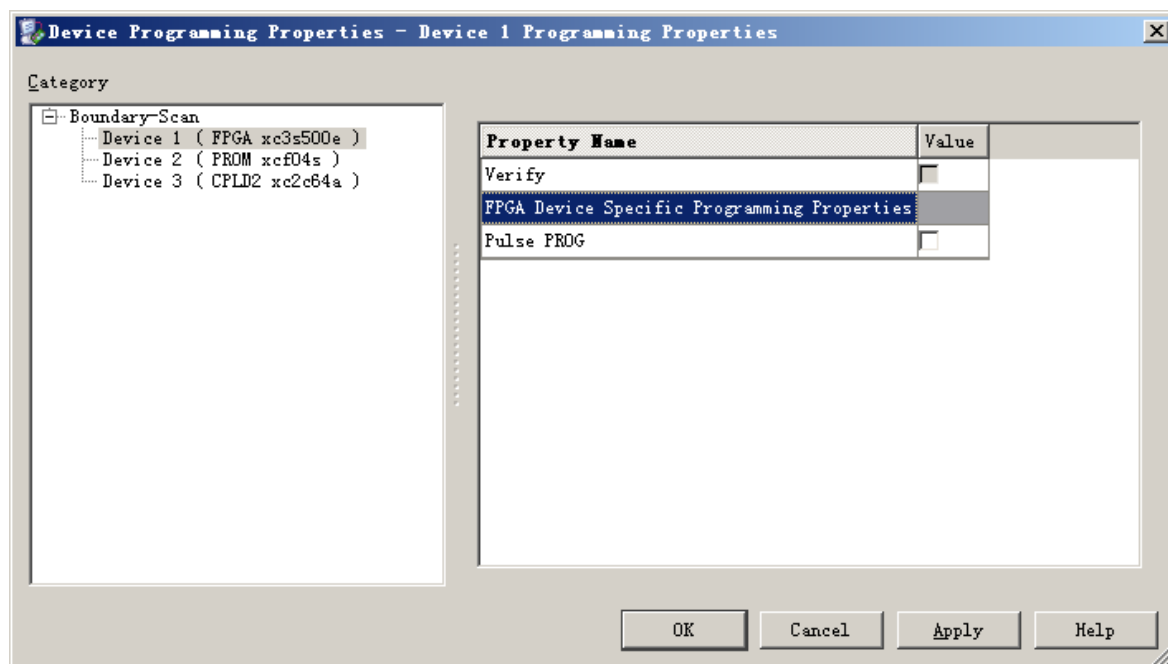



7. 点击 Bypass/Cancel。

8. 再点击 Bypass/Cancel。

9. 下图中点击 Ok。

 上海交通大学 计算机科学与工程系 Dept. of Computer Science & Engineering	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	15 of 17
	作者	修改日期		
	WnSN Lab	9/21/2012	公开	



 上海交通大学 计算机科学与工程系 Dept. of Computer Science & Engineering	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	16 of 17
作者	修改日期	公开		
WnSN Lab	9/21/2012			



10. 在 iMPACT 窗口中，右键选中待烧写 counter.bit 的在用的 FPGA 芯片，运行 Program，屏幕将出现蓝色提示：



说明烧写成功。

11. 实验板上 4 个 LED，亮起代表该位为 1，灭掉代表该位为 0。4 个 LED 表示了 4 位二进制数字，通过它们的组合变化观察计数过程。拨动拨码开关 Switch0（L13）可以控制该计数器同步复位。

## 2.3 实验报告

<div>上海交通大学</div> <div>计算机科学与工程系</div> <div>Dept. of Computer Science &amp; Engineering</div>	标题	文档编号	版本	页
	计算机组成与结构实验指导书 LAB1	CSE-COA-LAB-001	0.3	17 of 17
	作者	修改日期	公开	
	WnSN Lab	9/21/2012		