Robei 集成电路实战

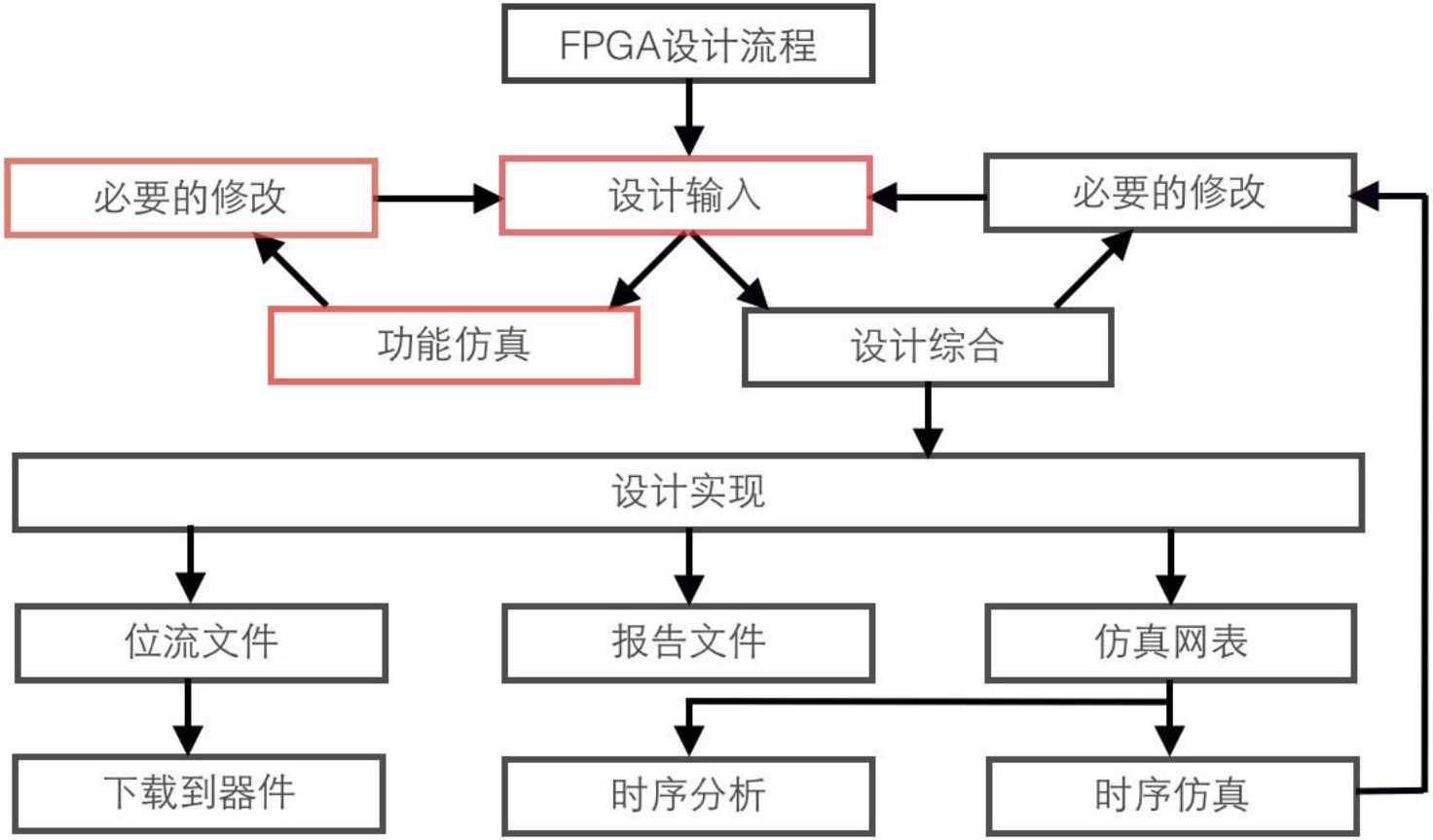
Robei-Vivado联合设计

流水灯设计

Robei实验：流水灯设计

1. FPGA设计流程
2. 综合-实现-比特流生成
3. 实验设计：前端
4. 实验设计：后端

Robei实验：流水灯设计



Robei实验：流水灯设计

综合（Synthesis）：

综合是为了完成从代码到门级电路的转换，也是把我们的设计转换为FPGA可以读懂的配置文件的第一个步骤。

常用的综合工具：

Design Compiler
Xilinx ISE
Altera Quartus II
Xilinx Vivado

Robei实验：流水灯设计

实现（Implementation）：

结合约束文件，把综合得到的逻辑门级电路制作成为可以在目标FPGA上运行的实际电路。

转换→映射→布局布线→时序提取→配置

把多个设计源文件转换合并为一个设计库文件

把逻辑门级电路中的元件映射为物理元素

将映射后的物理元素分配到FPGA结构中

产生反标文件，供给后续时序仿真使用

产生FPGA配置的位流文件

Robei实验：流水灯设计

位流（比特流）（BitStream）：

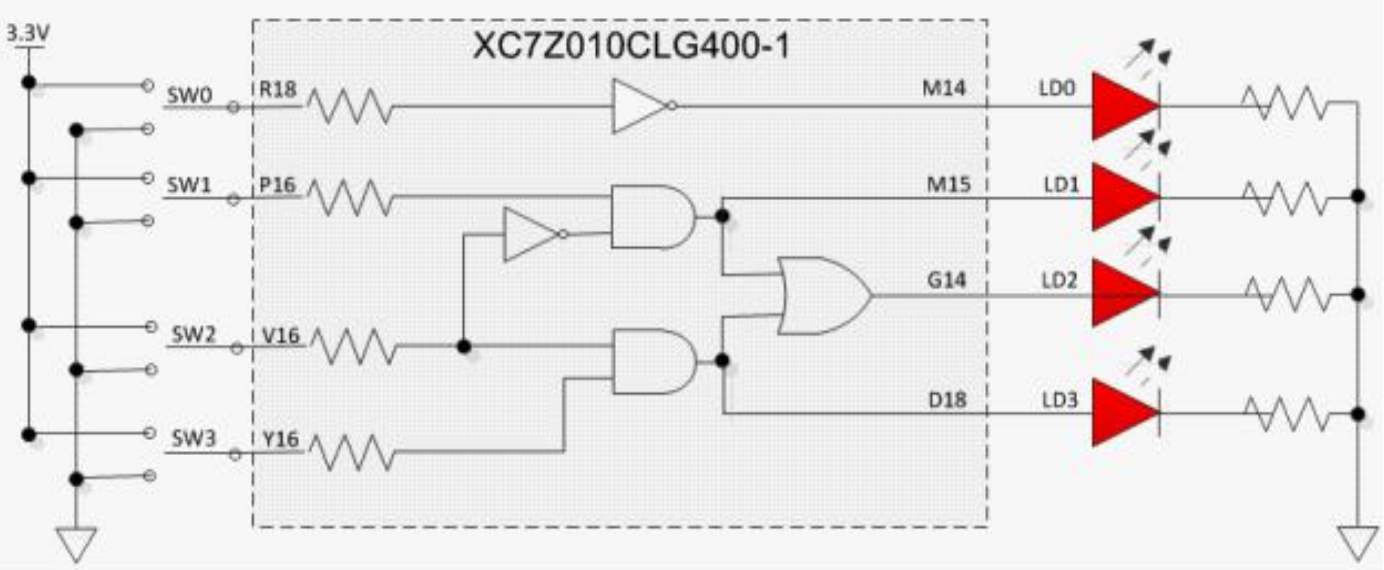
是一个连续的位序列，实现软件与器件间的直接通讯。

位流文件成功生成后，即可将其下载到FPGA中来实现设计的功能。

Robei实验：流水灯设计

实验设计：前端

设计包括直接连接到对应的输出LED的一些输入。其它的输入在进行一些逻辑操作之后输出到剩余的LED。

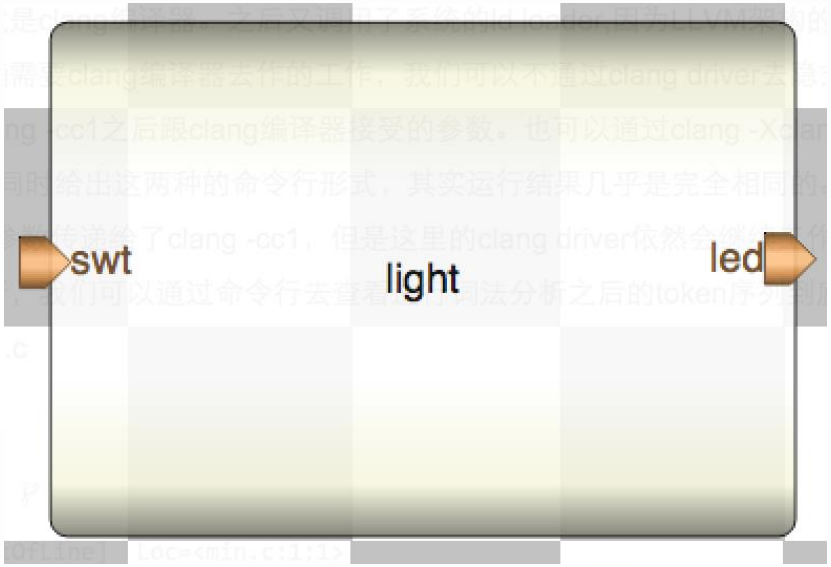


Robei实验：流水灯设计

light Datasheet

Ports:

Name	Inout	DataType	Datasize	Function
swt	input	wire	4	
led	output	wire	4	



Robei实验：流水灯设计

light

```
12 assign led[0] = ~swt[0];
13 assign led[1] = swt[1] & ~swt[2];
14 assign led[2] = (swt[1] & ~swt[2]) | (swt[2] & swt[3]);
15 assign led[3] = swt[2] & swt[3];
16
```

```
1
2  module lighter(swt,led);
3
4     //---Ports declearation---
5     input [3:0] swt;
6     output [3:0] led;
7
8     wire [3:0] swt;
9     wire [3:0] led;
10
11    //----Code starts here-----
12 assign led[0] = ~swt[0];
13 assign led[1] = swt[1] & ~swt[2];
14 assign led[2] = (swt[1] & ~swt[2]) | (swt[2] & swt[3]);
15 assign led[3] = swt[2] & swt[3];
16 endmodule    //lighter
17
```

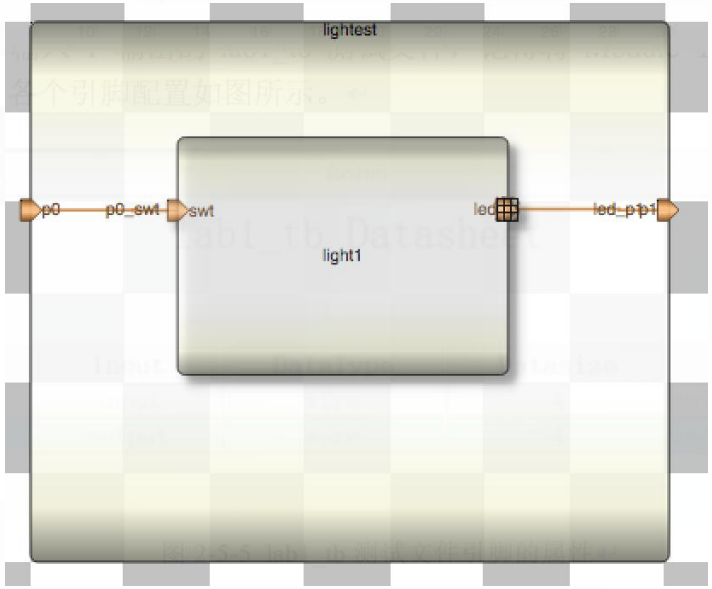
assign led[0] = ~swt[0];
assign led[1] = swt[1] & ~swt[2];
assign led[2] = (swt[1] & ~swt[2]) | (swt[2] & swt[3]);
assign led[3] = swt[2] & swt[3];

Robei实验：流水灯设计

测试模块设计：

lightest

```
11 initial
12 begin
13 #5 switches=4'b1111;
14 #5 switches=4'b1110;
15 #5 switches=4'b1101;
16 #5 switches=4'b1010;
17 #5 switches=4'b1011;
18 #5 switches=4'b0110;
19 #5 switches=4'b0101;
20 #5 switches=4'b0110;
21 #10 $finish;
22 end
23
```



Robei实验：流水灯设计

SWT[3:0]=A led[3:0]=7

SWT[3]=1 led[3]=0

SWT[2]=0 led[2]=1

SWT[1]=1 led[1]=1

SWT[0]=0 led[0]=1

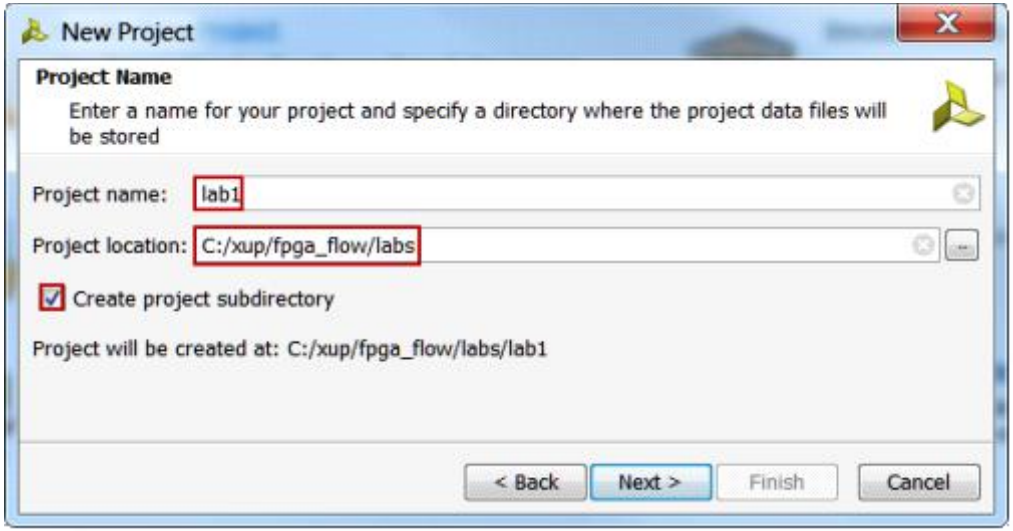
```
assign led[0] = ~swt[0];
assign led[1] = swt[1] & ~swt[2];
assign led[2] = (swt[1] & ~swt[2]) | (swt[2] & swt[3]);
assign led[3] = swt[2] & swt[3];
```

Robei实验：流水灯设计

实验设计：后端

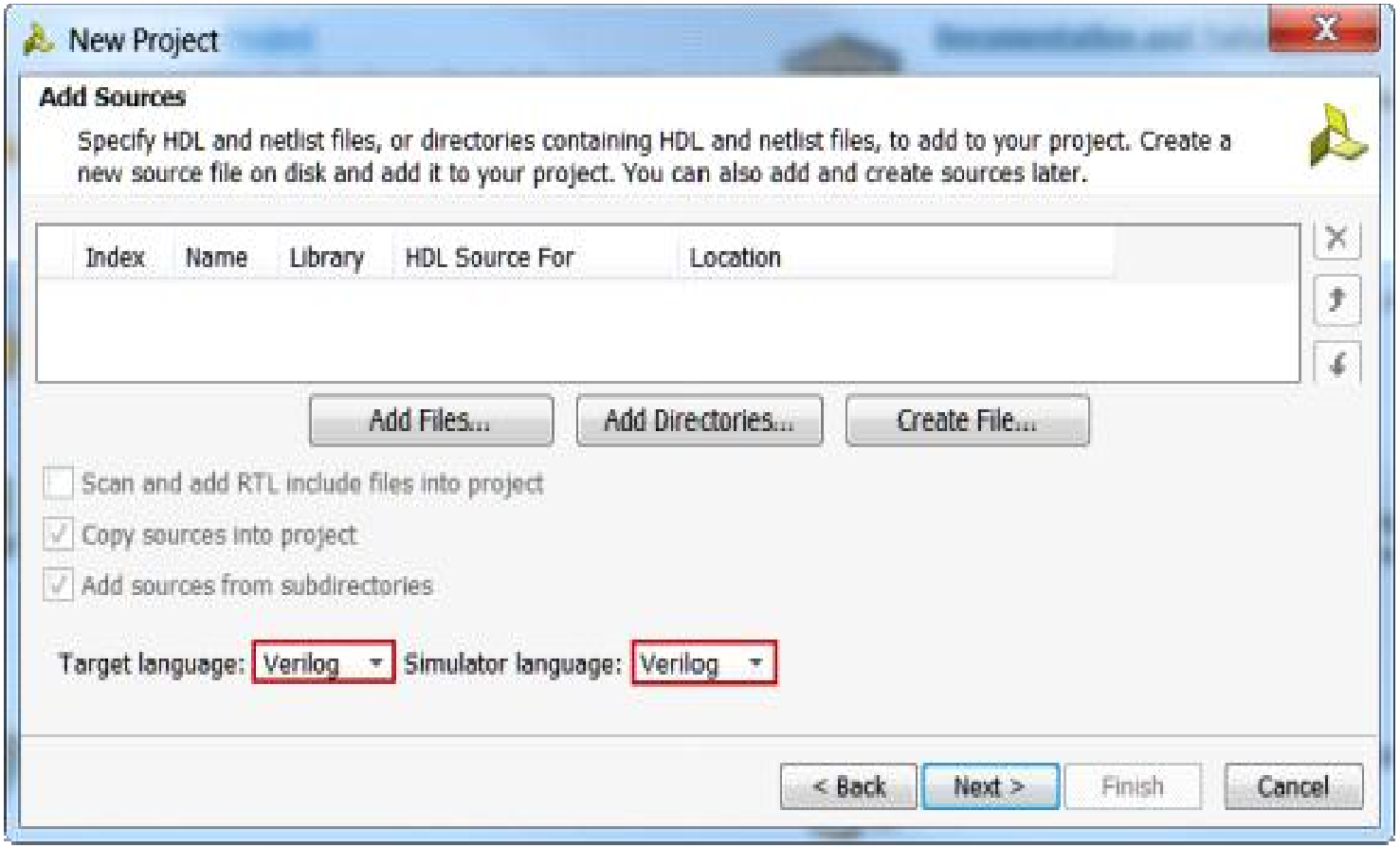
Robei实验：流水灯设计

单击创建新项目**Create New Project**启动向导。你将看到创建一个新的Vivado项目对话框。
单击**Next**；

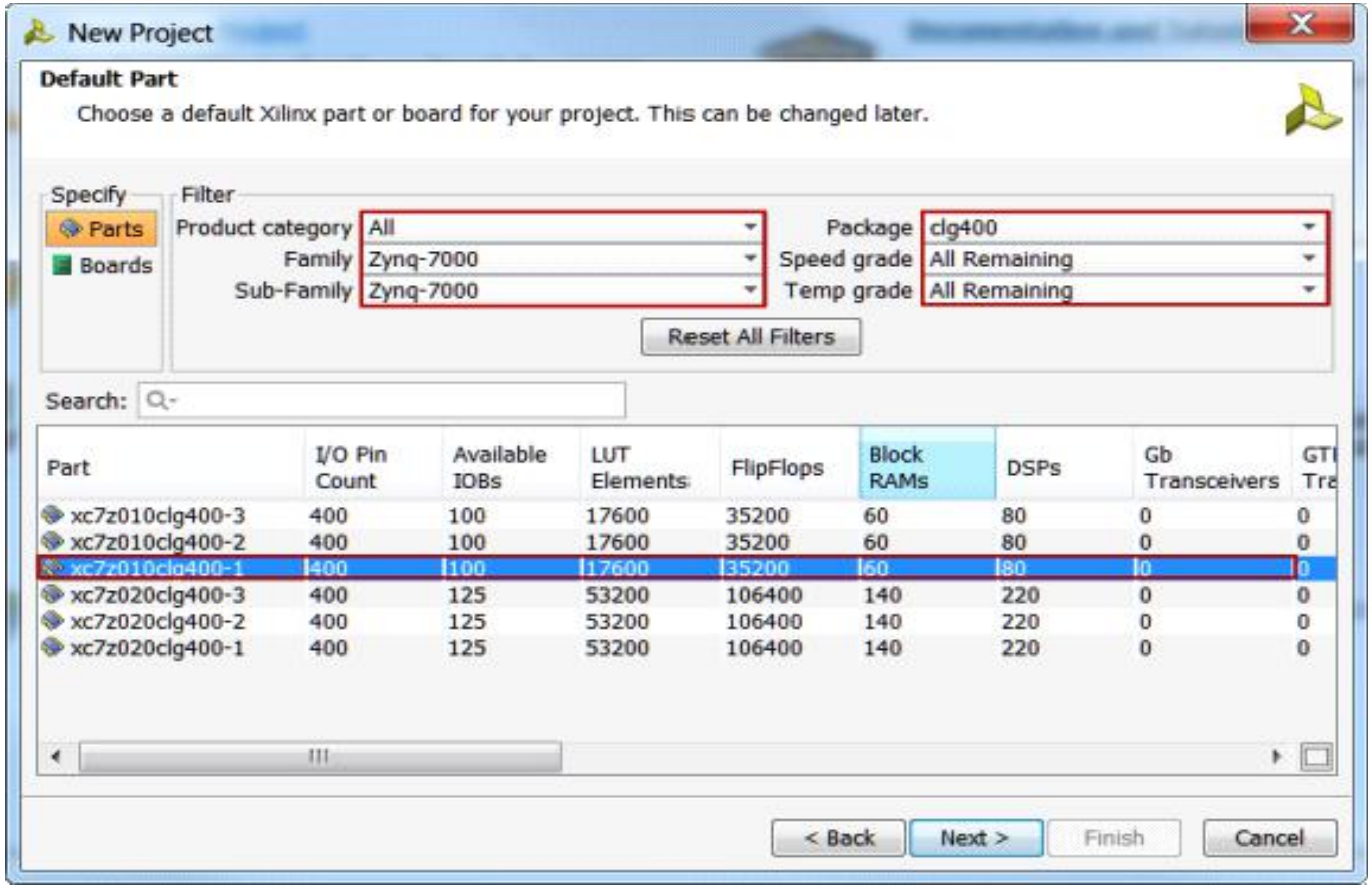


Robei实验：流水灯设计

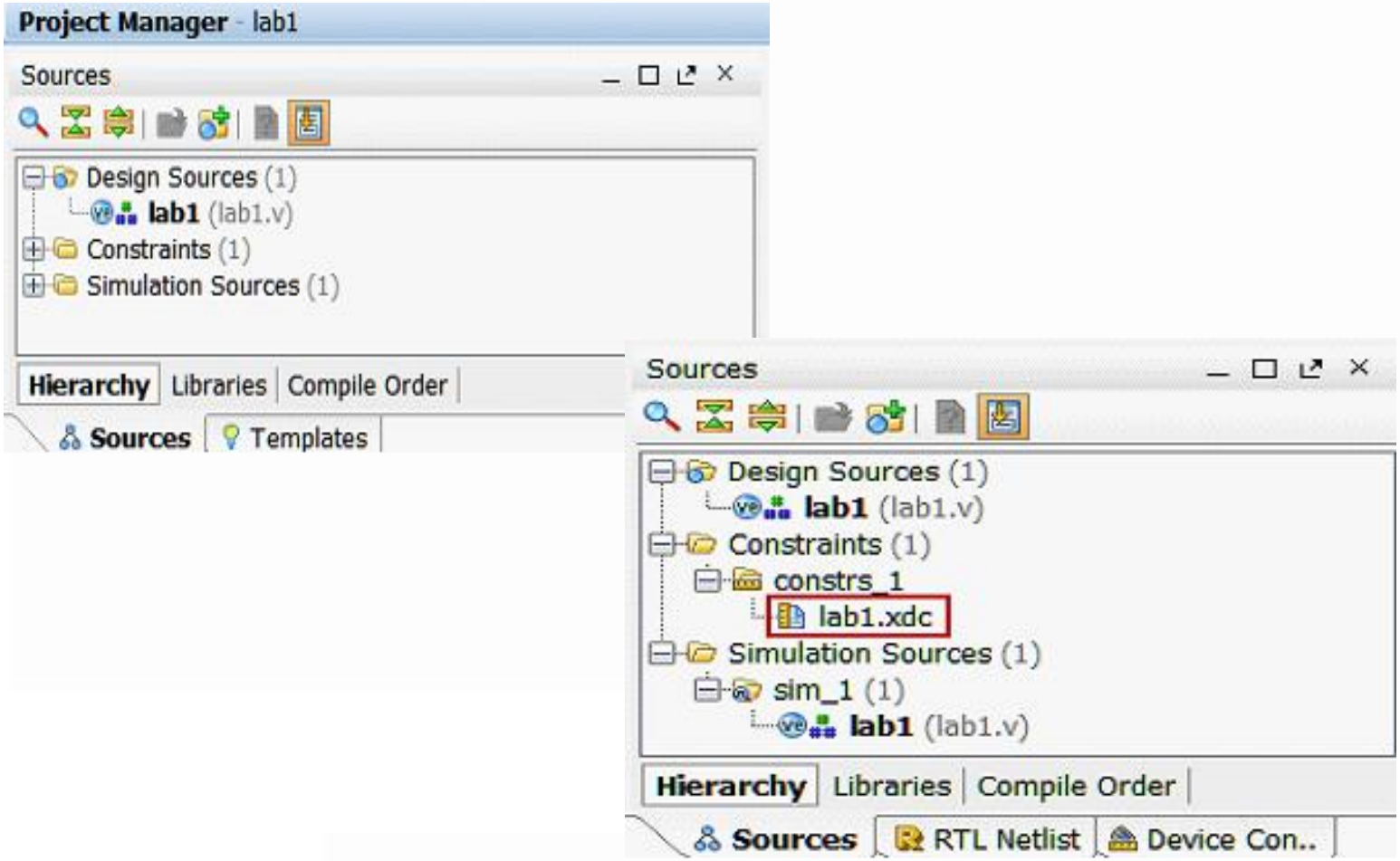
点击Add Files添加设计好的.v文件：



Robei实验：流水灯设计



Robei实验：流水灯设计

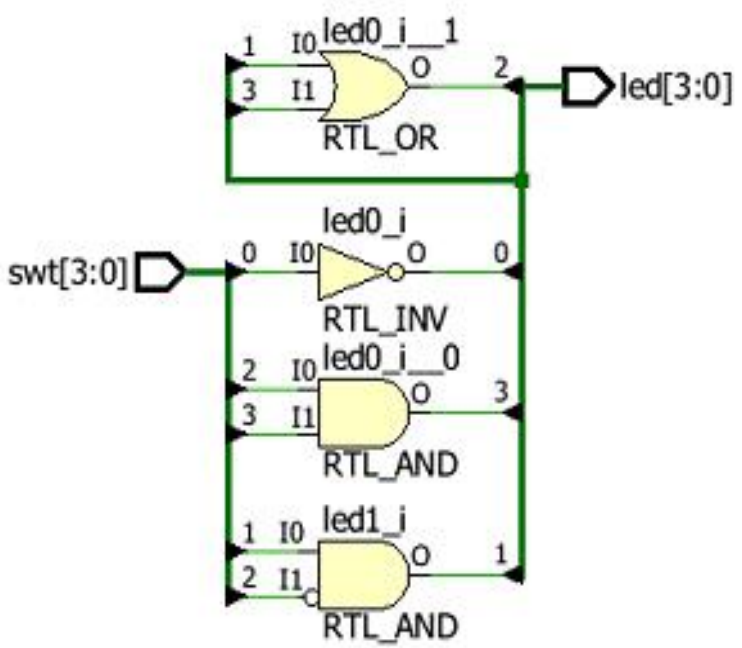


Robei实验：流水灯设计

```
# ZYBO xdc
# LED [3:0]
##################################
# On-board led            #
##################################

set_property PACKAGE_PIN M14 [get_ports led[0]]
set_property IOSTANDARD LVCMOS33 [get_ports led[0]]

set_property PACKAGE_PIN M15 [get_ports led[1]]
set_property IOSTANDARD LVCMOS33 [get_ports led[1]]

set_property PACKAGE_PIN G14 [get_ports led[2]]
set_property IOSTANDARD LVCMOS33 [get_ports led[2]]

set_property PACKAGE_PIN D18 [get_ports led[3]]
set_property IOSTANDARD LVCMOS33 [get_ports led[3]]
```

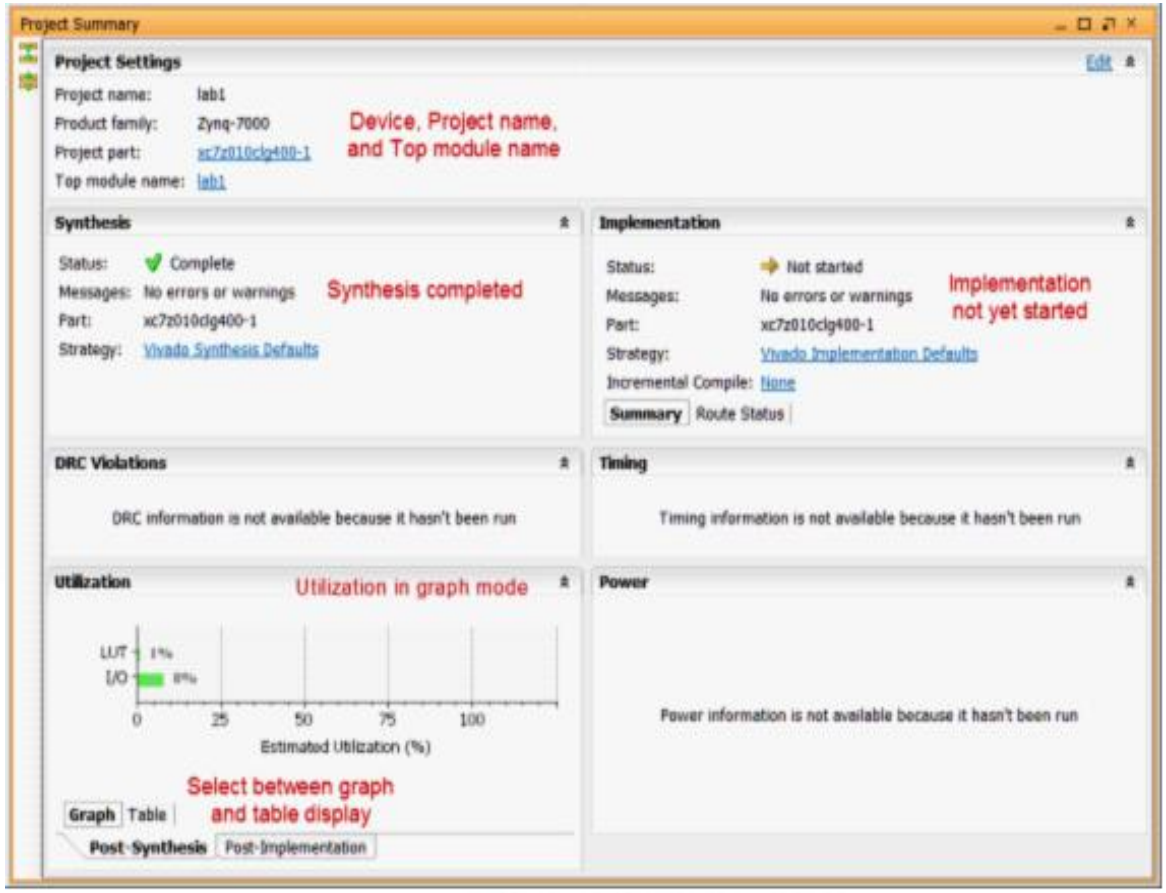
Robei实验：流水灯设计

对RTL源文件进行分析。展开**the Open Elaborated Design**的下拉菜单中的RTL分析任务并且单击**Schematic**查看原理图。



Robei实验：流水灯设计

单击综合任务下拉菜单中的**Run Synthesis**。在综合过程将在lab1.v文件运行以及所有分层文件。



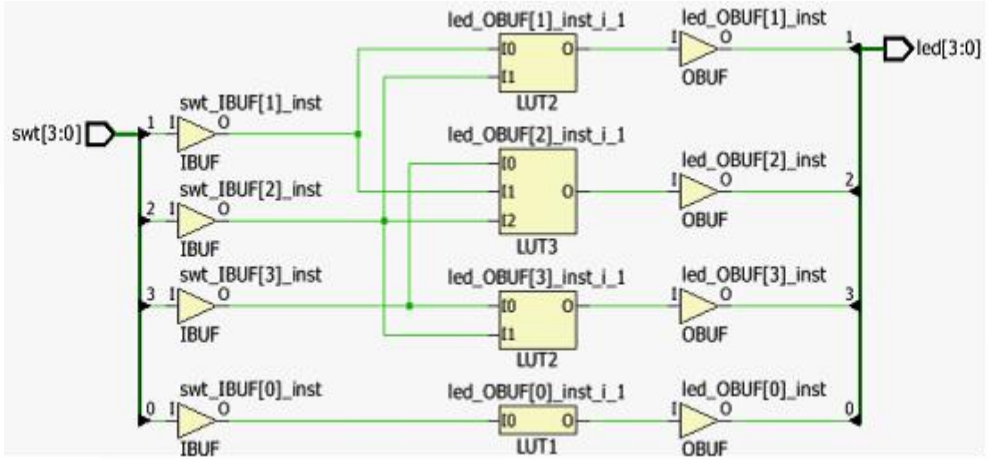
Robei实验：流水灯设计

单击项目摘要选项卡中的**Table**。请注意，估计有3个LUT和8个IO（4输入和4输出）被使用。

Utilization

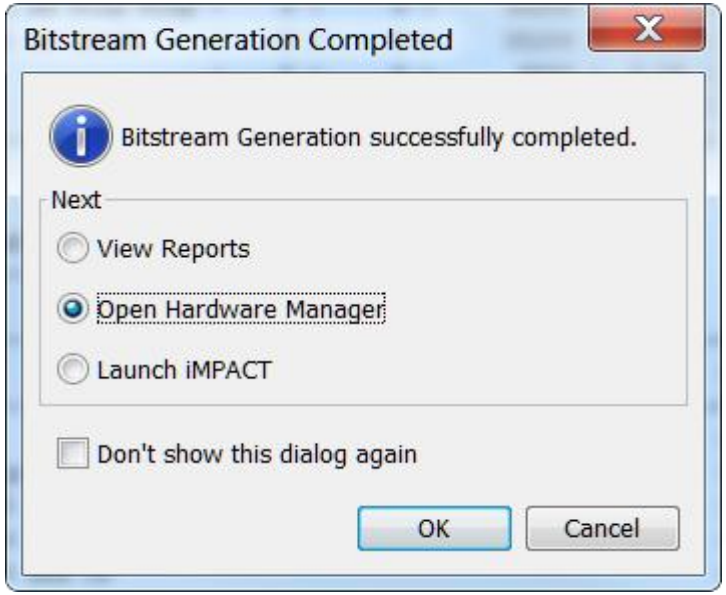
Resource	Estimation	Available	Utilization %
LUT	3	17600	1
I/O	8	100	8

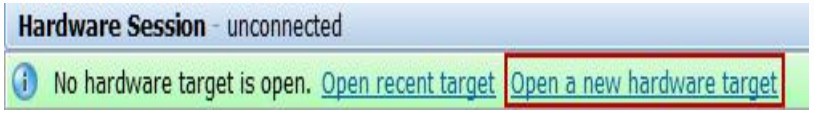
在综合下拉菜单下，单击**Schematic**，查看综合设计的示意图。



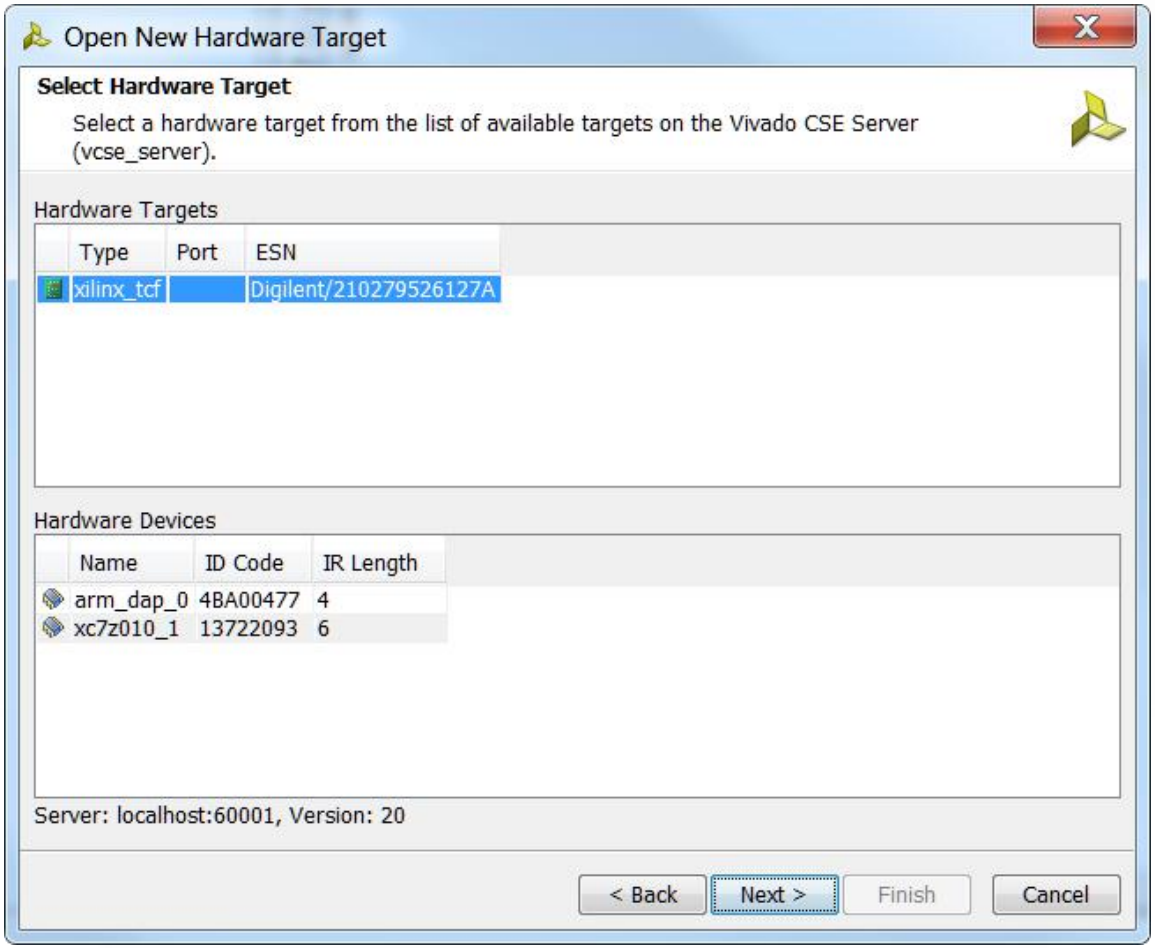
Robei实验：流水灯设计

点击 Generate Bitstream 生成比特流文件：



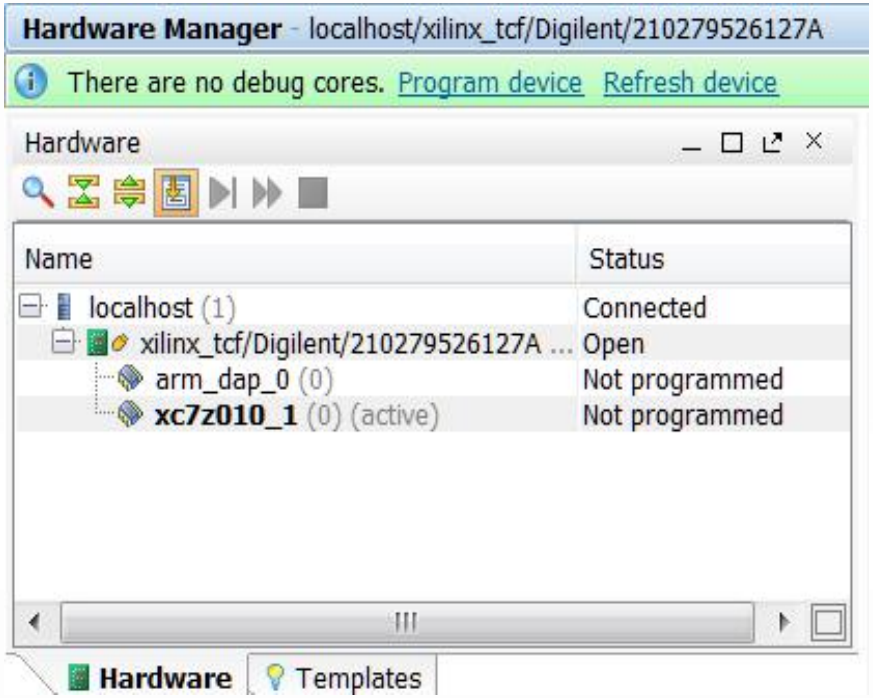


Robei实验：流水灯设计



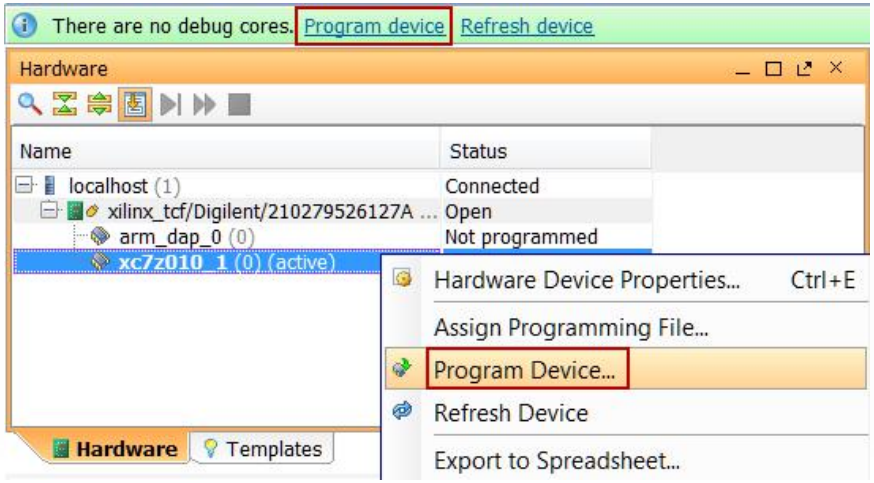
Robei实验：流水灯设计

单击两次**Next**，然后单击**Finish**。未连接硬件会话状态更改为服务器名称并且器件被高亮显示。



Robei实验：流水灯设计

在器件上单击鼠标右键，选择Program device或单击窗口上方弹出的Program device—> XC7z010_1链接到目标FPGA器件进行编程；



单击确定对FPGA进行编程。开发板上Done指示灯亮时，器件编程结束。

www.robei.com

Thank You!