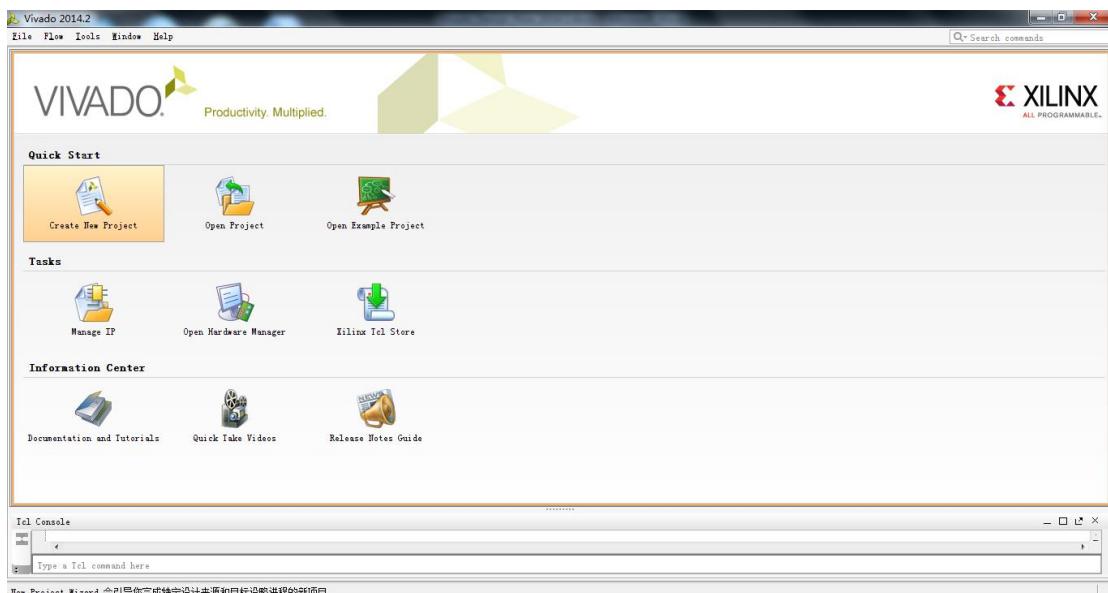


## Vivado 设计流程指导手册——2014.2

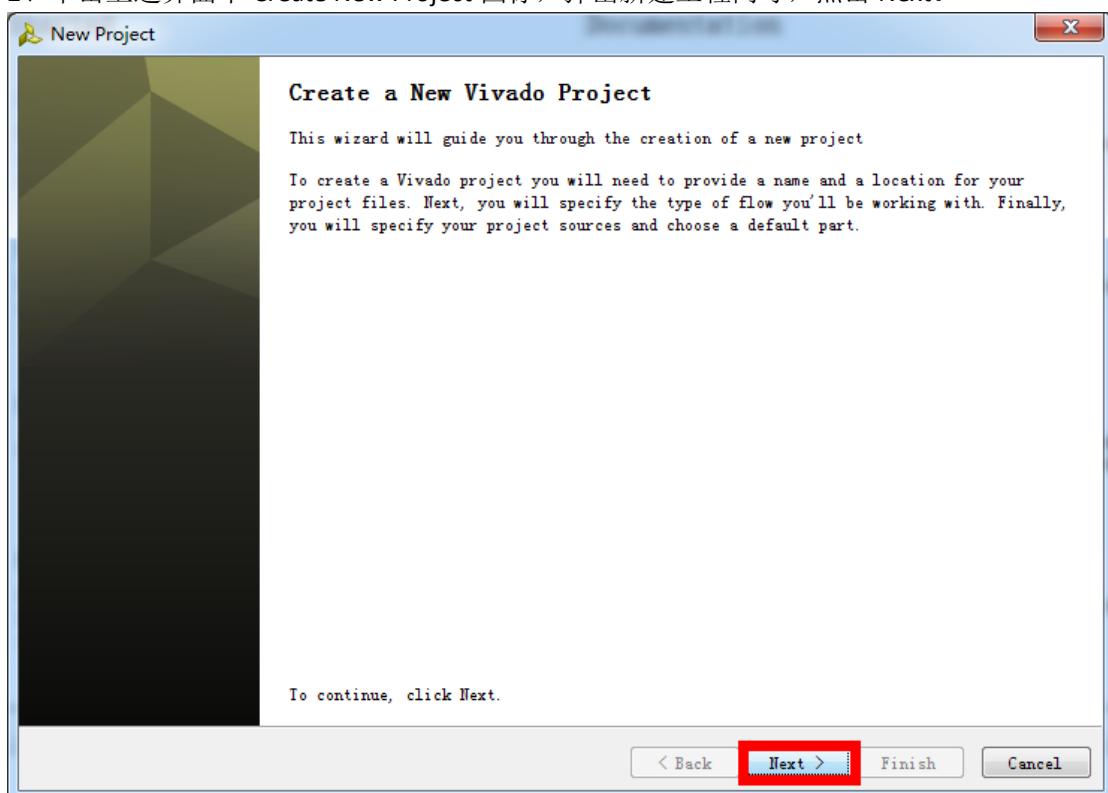
Vivado 设计分为 Project Mode 和 Non-project Mode 两种模式，一般简单设计中，我们常用的是 **Project Mode**。在本手册中，我们采用 **Digilent Basys3** 开发板，将以一个简单的实验案例，一步一步的完成 Vivado 的整个设计流程。

### 一、新建工程

1、打开 Vivado 2014.2 开发工具，可通过桌面快捷方式或开始菜单中 Xilinx Design Tools->Vivado 2014.2 下的 Vivado 2014.2 打开软件，开启后，软件如下所示：

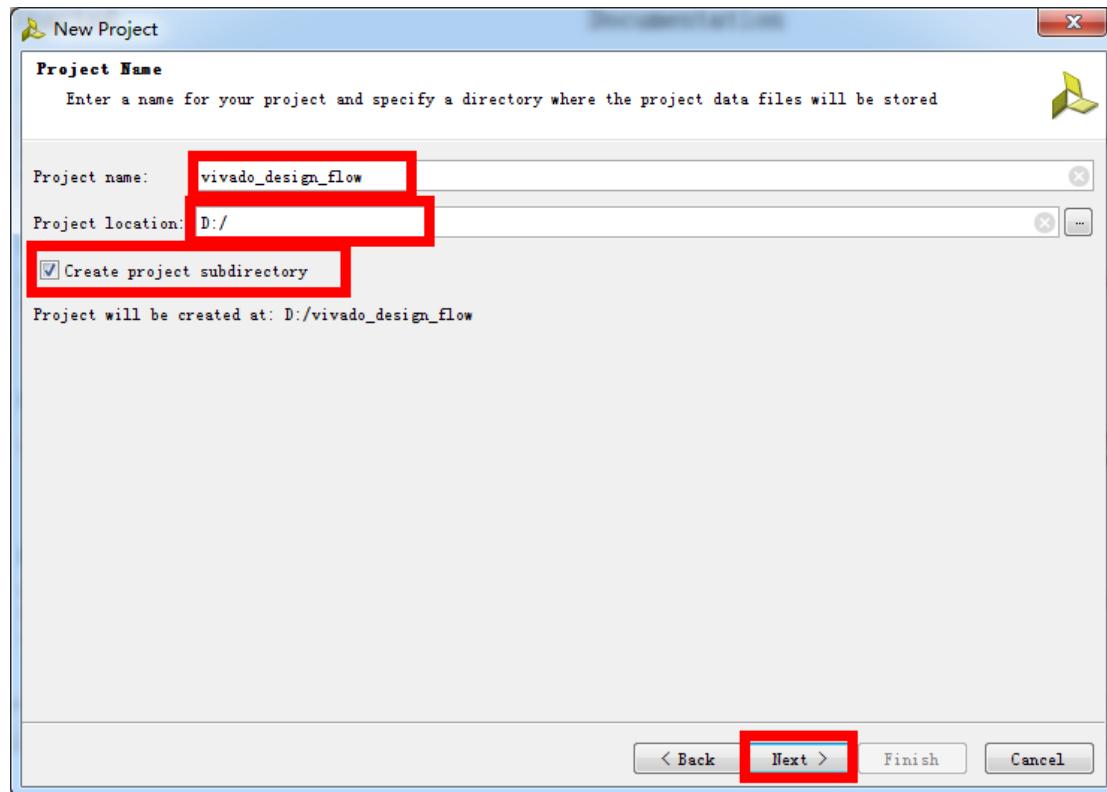


2、单击上述界面中 Create New Project 图标，弹出新建工程向导，点击 Next。

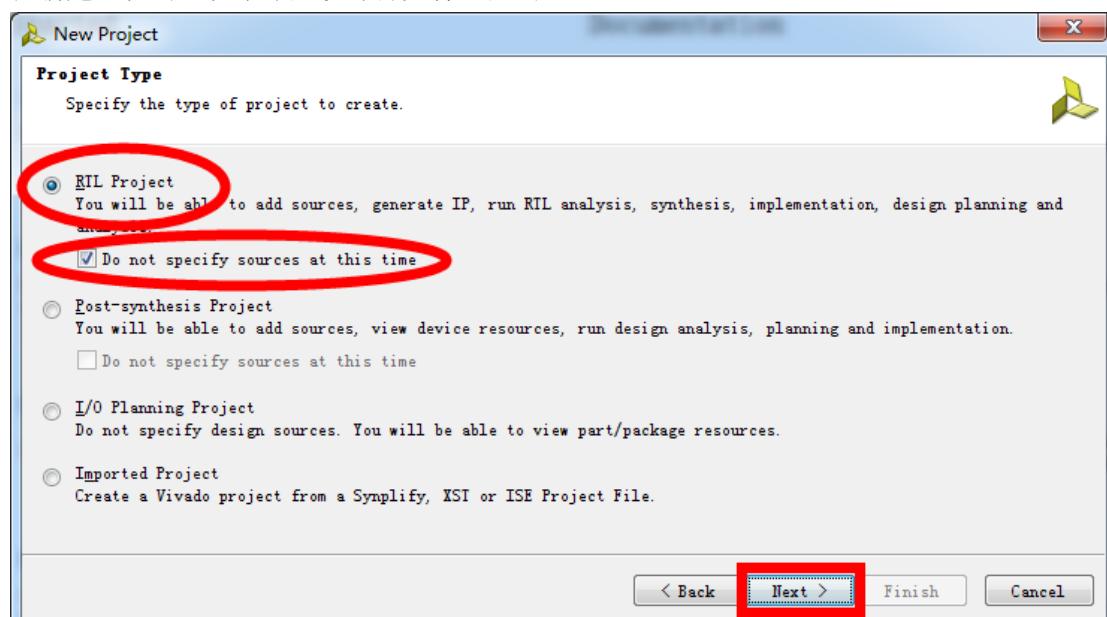


3、输入工程名称、选择工程存储路径，并勾选 **Create project subdirectory** 选项，为工程在指定存储路径下建立独立的文件夹。设置完成后，点击 **Next**。

**注意：**工程名称和存储路径中不能出现中文和空格，建议工程名称以字母、数字、下划线来组成。

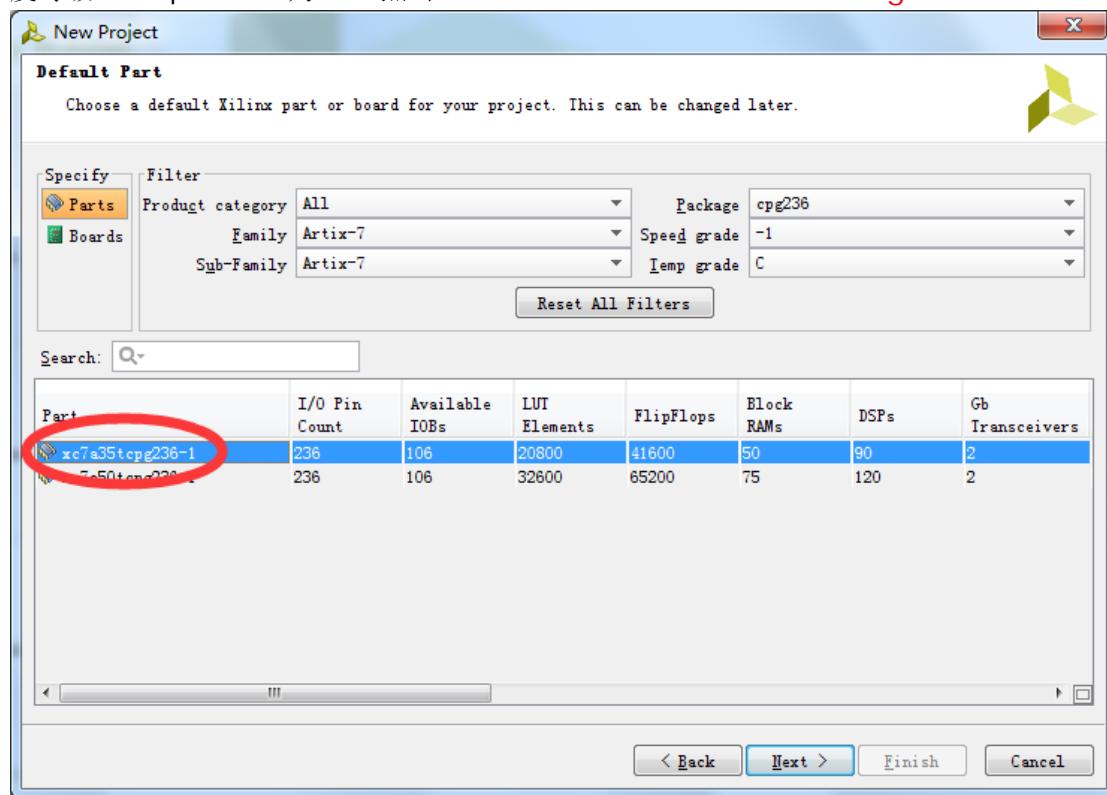


4、选择 **RTL Project** 一项，并勾选 **Do not specify sources at this time**，勾选该选项是为了跳过在新建工程的过程中添加设计源文件。点击 **Next**。

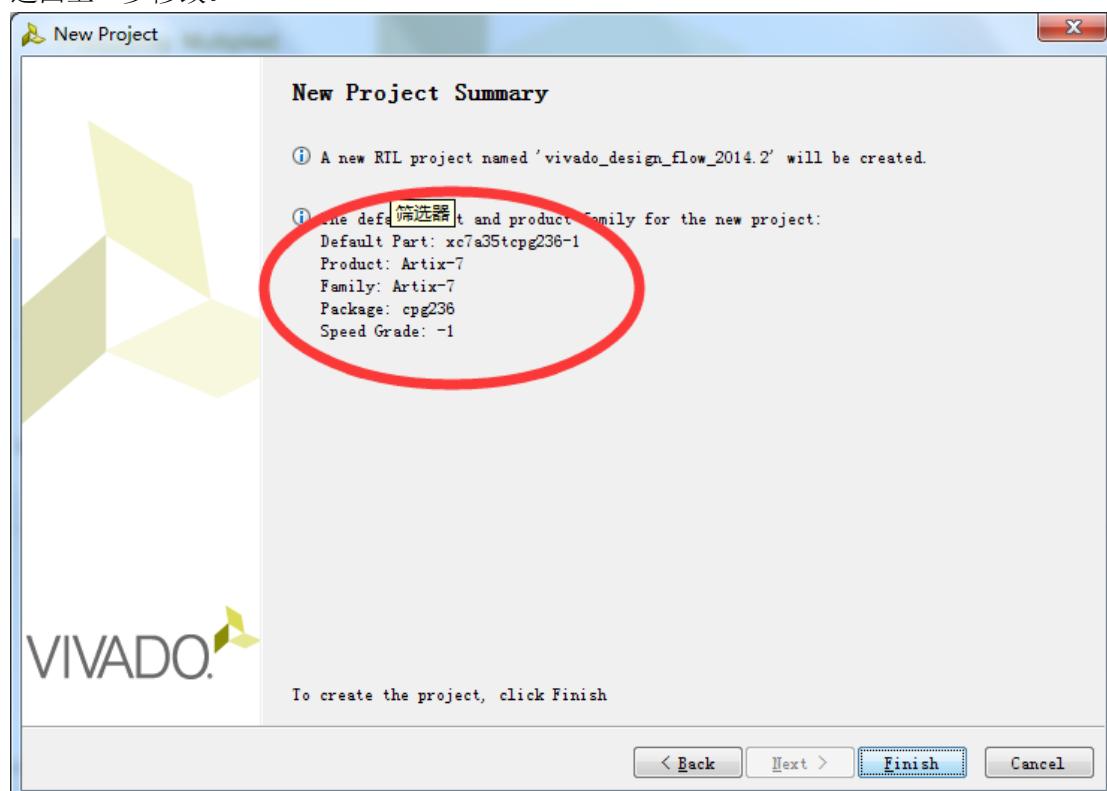


5、根据使用的 **FPGA** 开发平台，选择对应的 **FPGA 目标器件**。(在本手册中，以 Xilinx 大学计

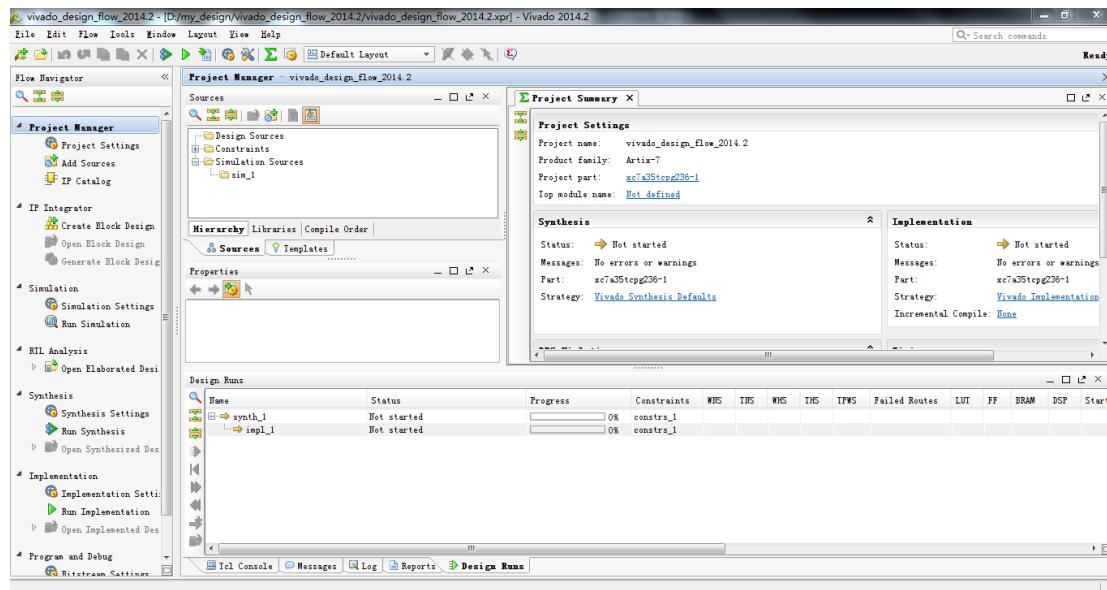
以开发板 **Diligent Basys3** 为例，FPGA 采用 Artix-7 XC7A35T-1CPG236-C 的器件，即 Family 和 Subfamily 均为 Artix-7，封装形式（Package）为 CPG236，速度等级（Speed grade）为 -1，温度等级（Temp Grade）为 C。点击 Next。 **Artix-7 xc7a35t-1ftg256**



6、确认相关信息与设计所用的的 FPGA 器件信息是否一致，一致请点击 Finish，不一致，请返回上一步修改。

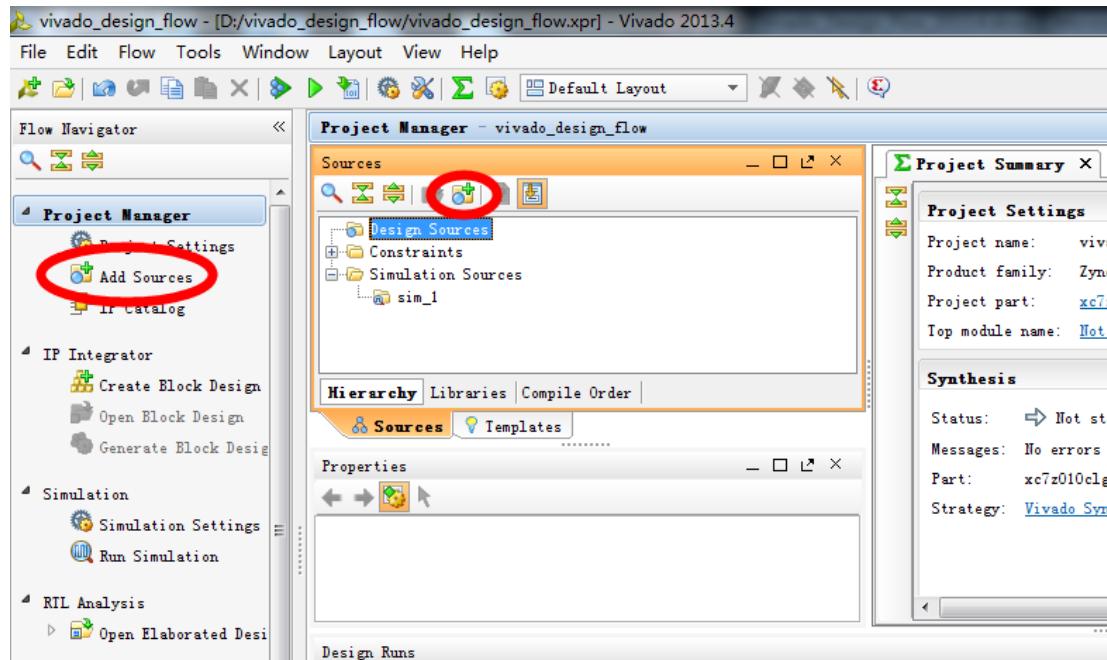


7、得到如下的空白 Vivado 工程界面，完成空白工程新建。

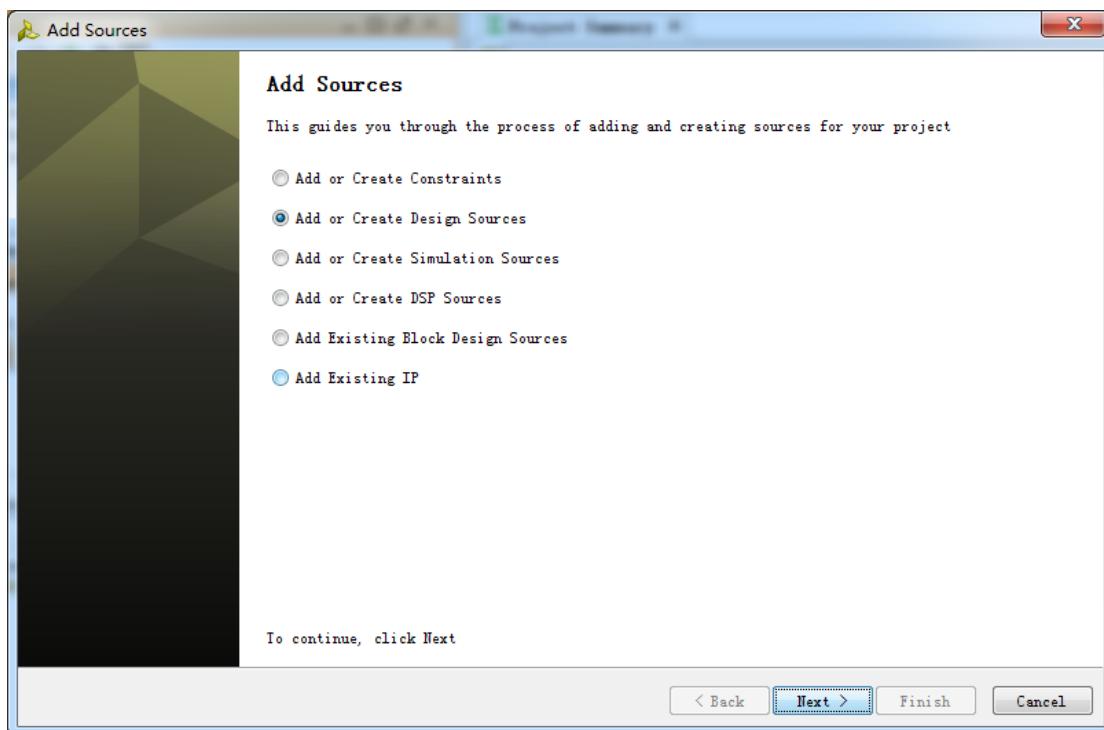


## 二、设计文件输入

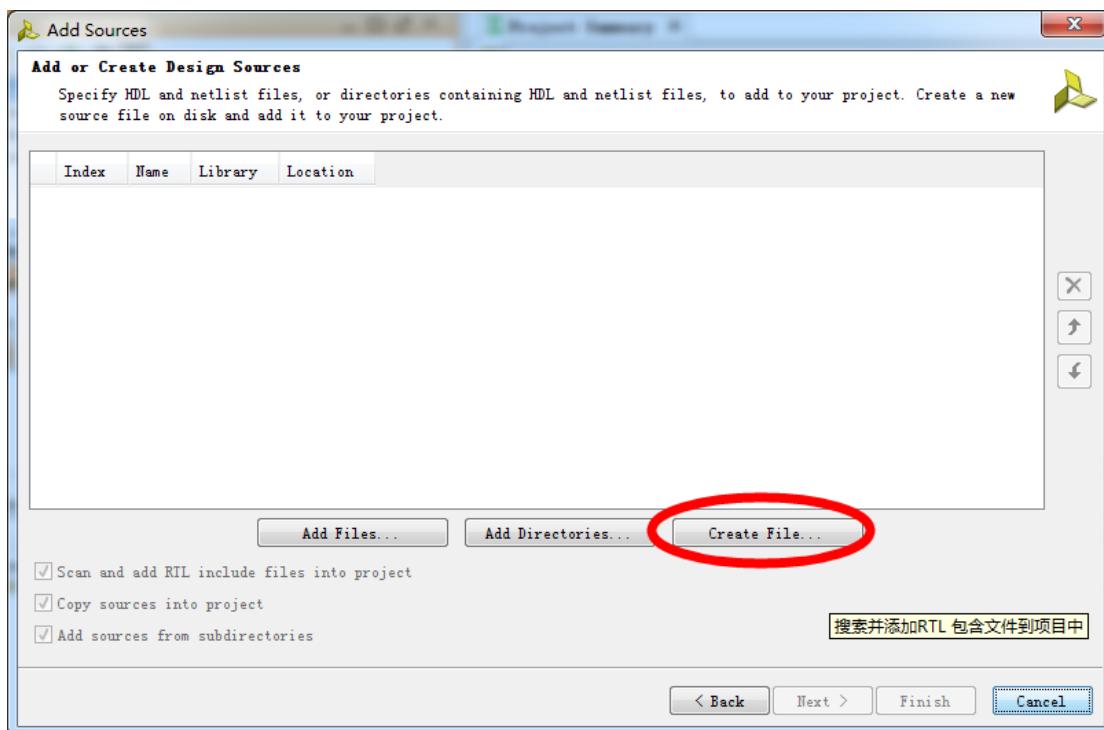
1、如下图所示，点击 Flow Navigator 下的 Project Manager->Add Sources 或中间 Sources 中的对话框打开设计文件导入添加对话框。



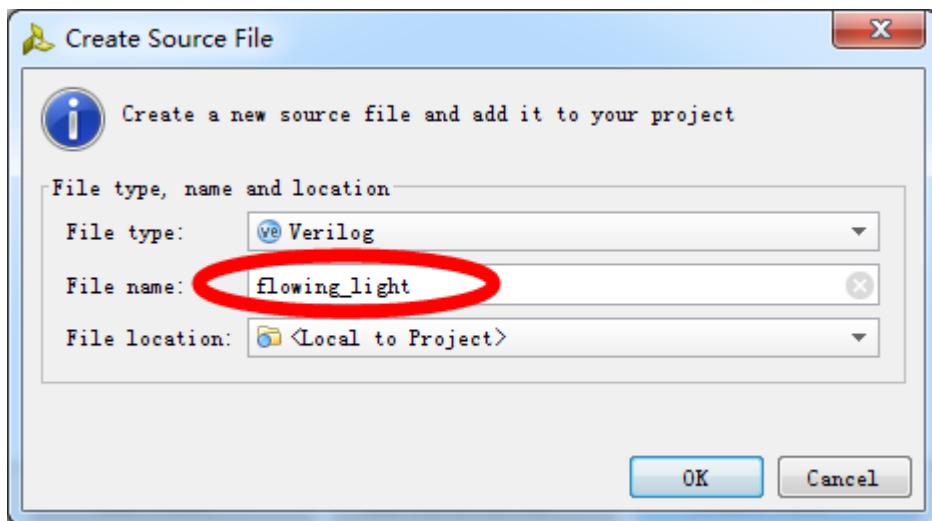
2、选择第二项 Add or Create Design Sources，用来添加或新建 Verilog 或 VHDL 源文件，点击 Next。



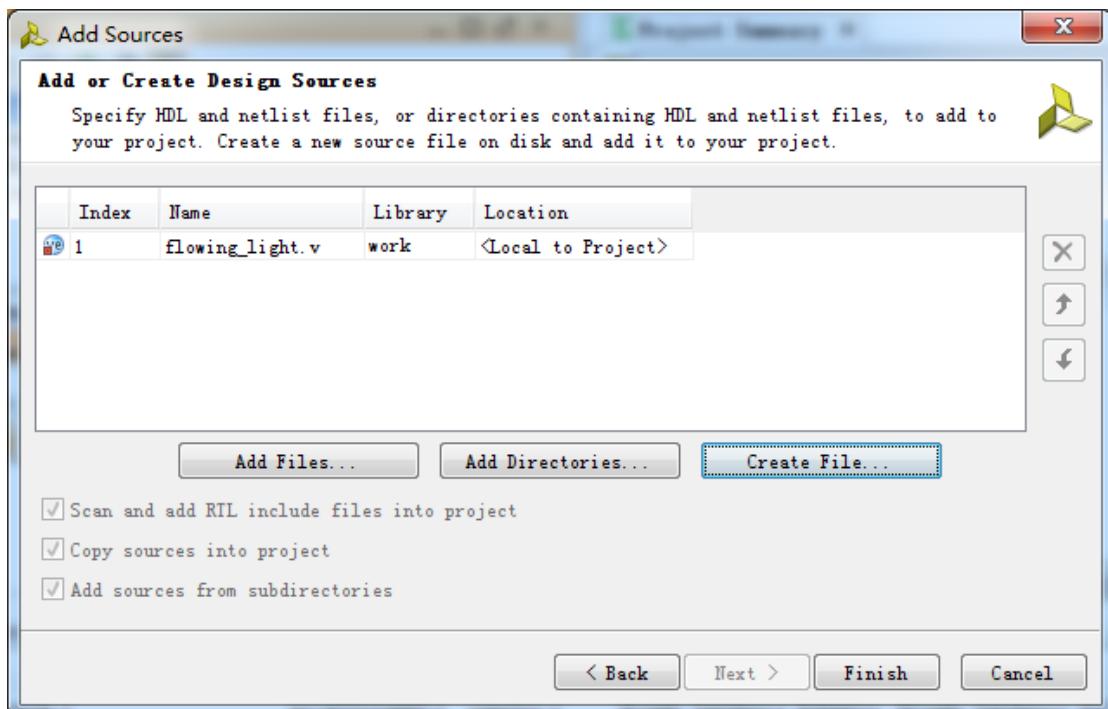
3、如果有现有的 V/VHD 文件，可以通过 Add Files 一项添加。在这里，我们要新建文件，所以选择 Create File 一项。



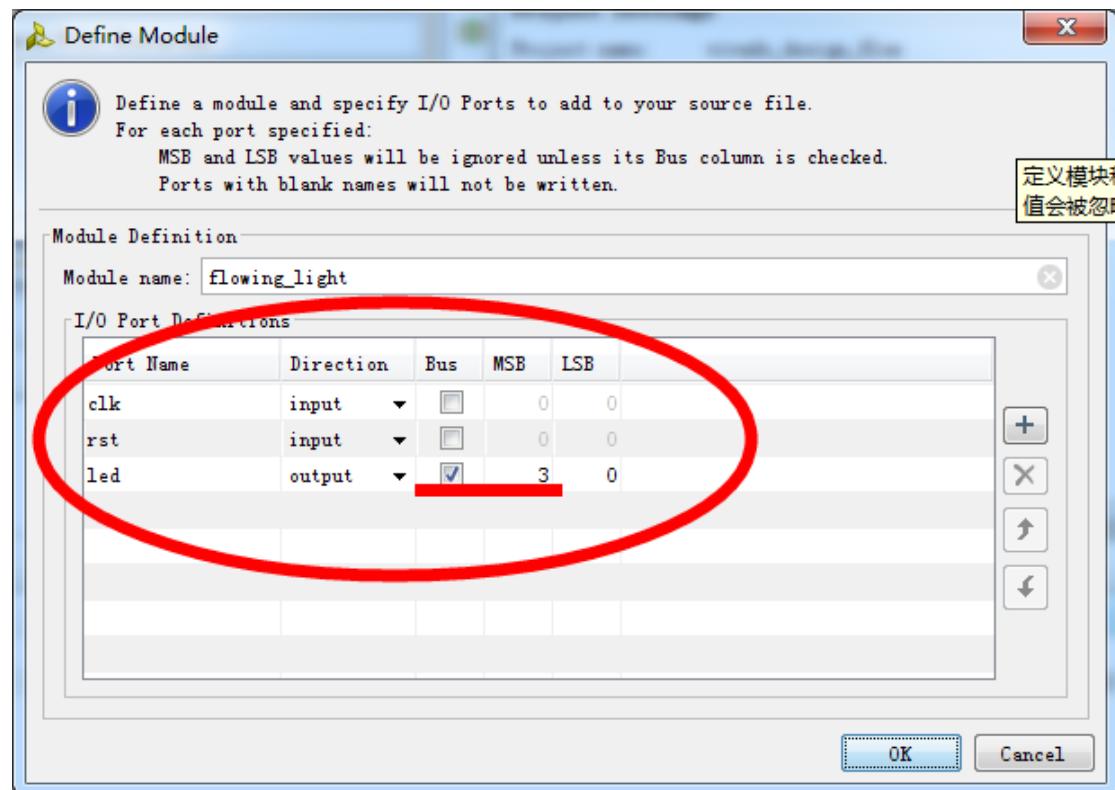
4、在 Create Source File 中输入 File Name，点击 OK。注：名称中不可出现中文和空格。



5、点击 Finish。

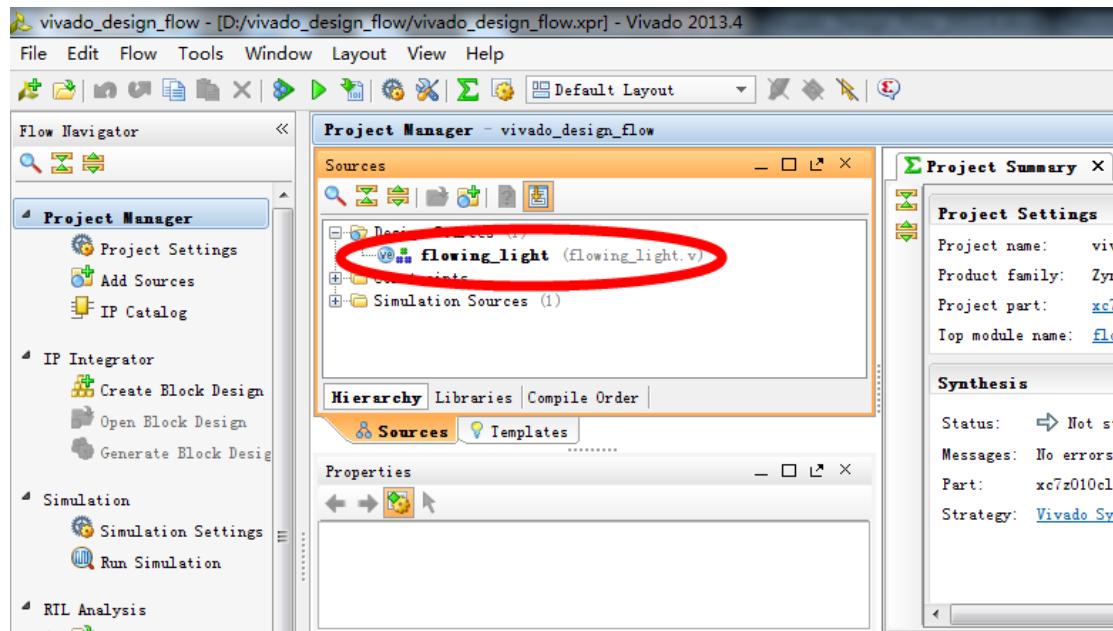


6、在弹出的 Define Module 中的 I/O Port Definition，输入设计模块所需的端口，并设置端口方向，如果端口为总线型，勾选 Bus 选项，并通过 MSB 和 LSB 确定总线宽度。完成后点击 OK。



注: led 实际宽度与代码中一致, 也可在代码中修改。

7、新建的设计文件（此处为 flowing\_light.v）即存在于 Sources 中的 Design Sources 中。双击打开该文件，输入相应的设计代码。



```
'timescale 1ns / 1ps

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

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

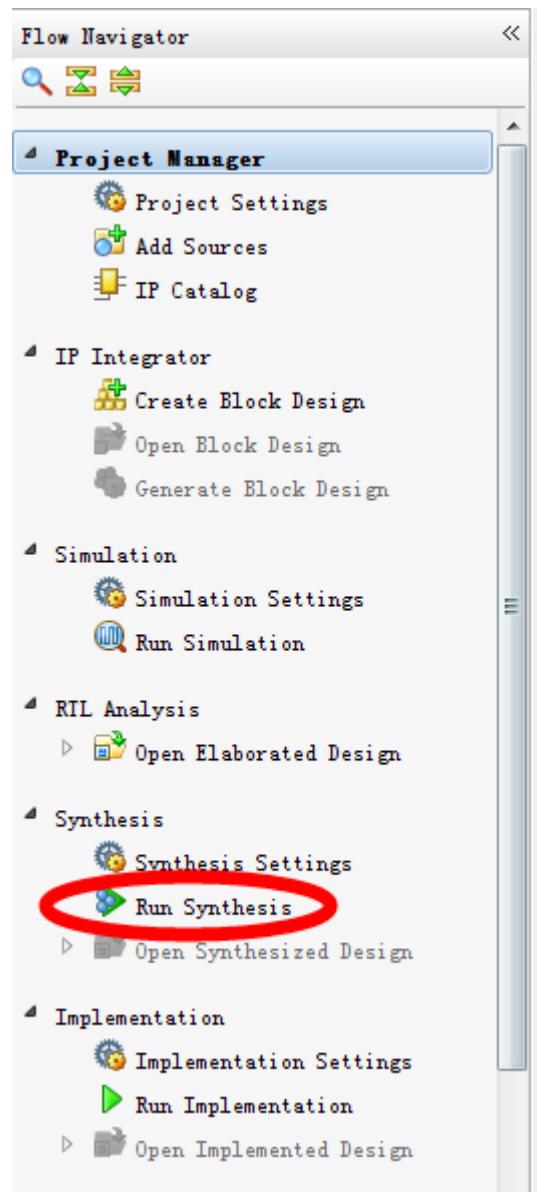
always @ (posedge clk)
begin
    if (rst)
        cnt_reg <= 0;
    else
        cnt_reg <= cnt_reg + 1;
end

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

8、添加约束文件，有两种方法可以添加约束文件，一是可利用 Vivado 中 IO planning 功能，二是可以直接新建 XDC 的约束文件，手动输入约束命令。

a、先来看第一种方法，利用 IO planning。

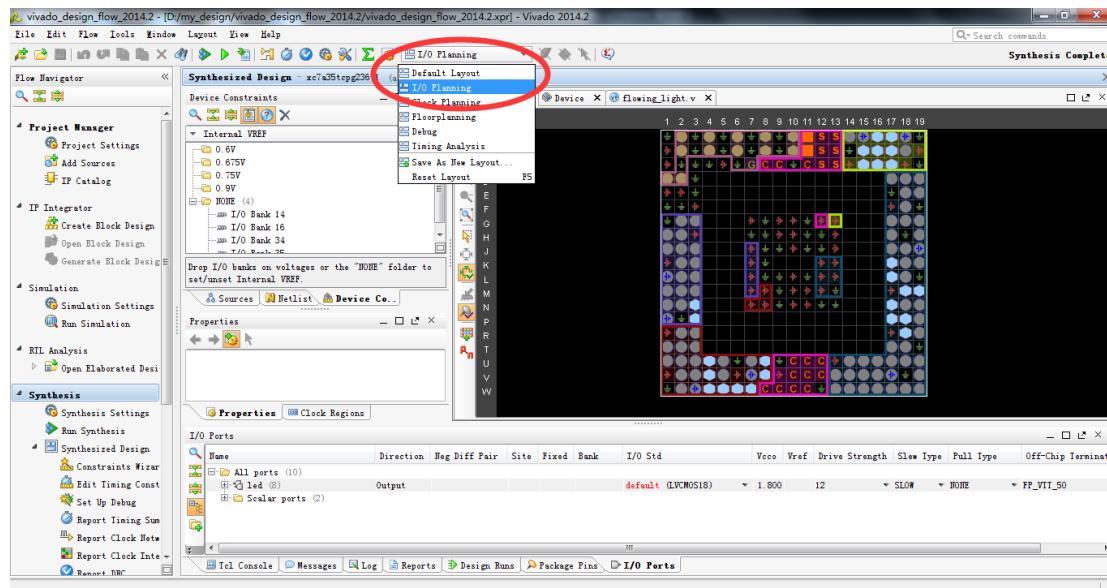
a.8.1、点击 Flow Navigator 中 Synthesis 中的 Run Synthesis，先对工程进行综合。



a.8.2、综合完成之后，选择 **Open Synthesized Design**，打开综合结果。



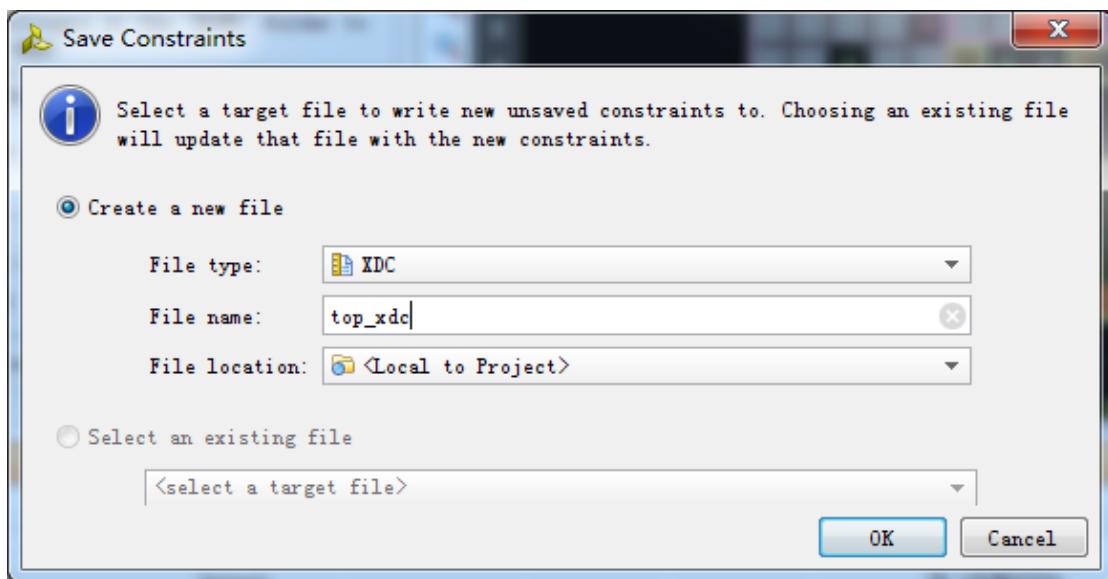
a.8.3、此时应看到如下界面，如果没出现如下界面，在图示位置的 layout 中选择 IO planning 一项。



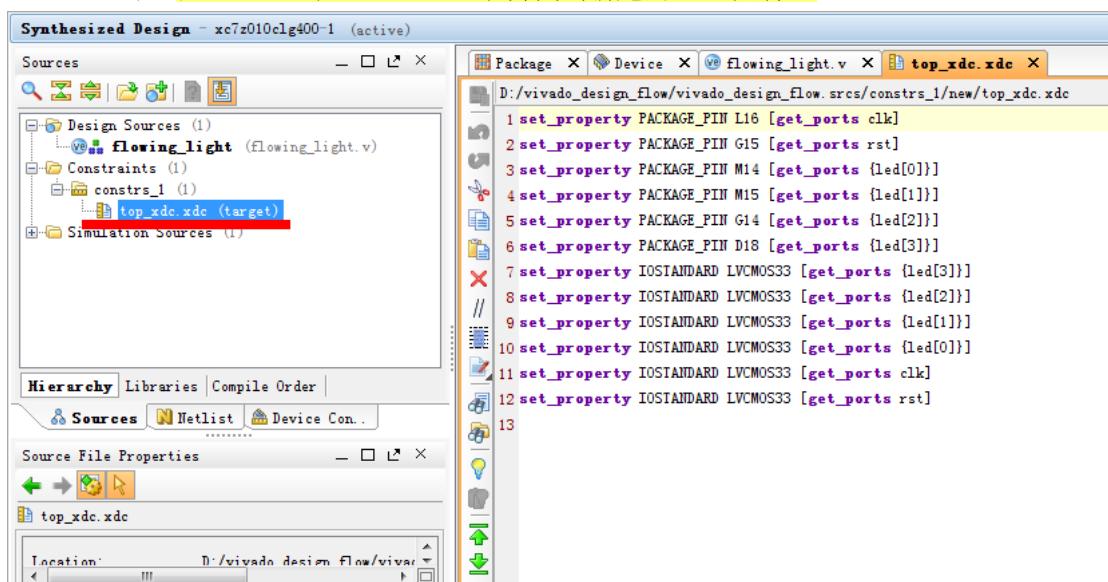
a.8.4、在右下方的选项卡中切换到 I/O ports 一栏，并在对应的信号后，输入对应的 FPGA 管脚标号（或将信号拖拽到右上方 Package 图中对应的管脚上），并指定 I/O std。（具体的 FPGA 约束管脚和 IO 电平标准，可参考对应板卡的用户手册或原理图）。

Name	Direction	Neg Diff Pair	Site	Fixed	Bank	I/O Std
All ports (18)						
led (16)	Output					
led[15]	Output		L1	<input checked="" type="checkbox"/>	(Multiple) LVCMS33*	35 LVCMS33*
led[14]	Output		P1	<input checked="" type="checkbox"/>	35 LVCMS33*	35 LVCMS33*
led[13]	Output		N3	<input checked="" type="checkbox"/>	35 LVCMS33*	35 LVCMS33*
led[12]	Output		P3	<input checked="" type="checkbox"/>	35 LVCMS33*	34 LVCMS33*
led[11]	Output		U3	<input checked="" type="checkbox"/>	34 LVCMS33*	34 LVCMS33*
led[10]	Output		W3	<input checked="" type="checkbox"/>	34 LVCMS33*	34 LVCMS33*
led[9]	Output		V3	<input checked="" type="checkbox"/>	34 LVCMS33*	14 LVCMS33*
led[8]	Output		V13	<input checked="" type="checkbox"/>	14 LVCMS33*	14 LVCMS33*
led[7]	Output		V14	<input checked="" type="checkbox"/>	14 LVCMS33*	14 LVCMS33*
led[6]	Output		U14	<input checked="" type="checkbox"/>	14 LVCMS33*	14 LVCMS33*
led[5]	Output		U15	<input checked="" type="checkbox"/>	14 LVCMS33*	14 LVCMS33*
led[4]	Output		W18	<input checked="" type="checkbox"/>	14 LVCMS33*	14 LVCMS33*
led[3]	Output		V19	<input checked="" type="checkbox"/>	14 LVCMS33*	14 LVCMS33*
led[2]	Output		U19	<input checked="" type="checkbox"/>	14 LVCMS33*	14 LVCMS33*
led[1]	Output		E19	<input checked="" type="checkbox"/>	14 LVCMS33*	14 LVCMS33*
led[0]	Output		U16	<input checked="" type="checkbox"/>	14 LVCMS33*	14 LVCMS33*
Scalar ports (2)						
clk	Input		W5	<input checked="" type="checkbox"/>	34 LVCMS33*	
rst	Input		V17	<input checked="" type="checkbox"/>	14 LVCMS33*	

a.8.5、完成之后，点击左上方工具栏中的保存按钮，工程提示新建 XDC 文件或选择工程中已有的 XDC 文件。在这里，我们要 Create a new file，输入 File name，点击 OK 完成约束过程。

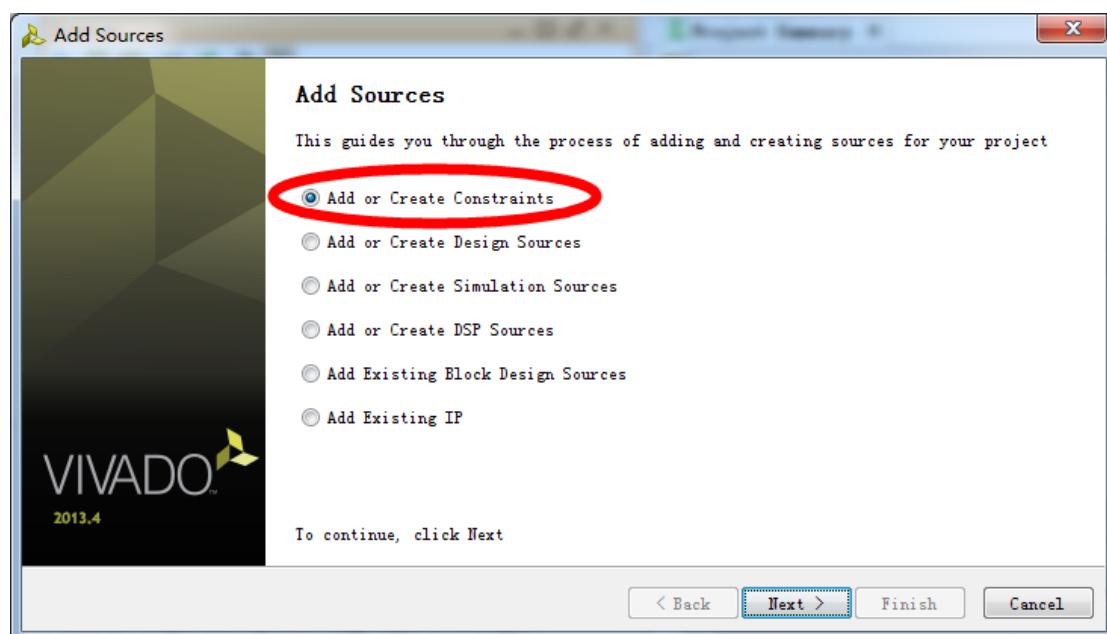


a.8.6、此时，在 Sources 下 Constraints 中会找到新建的 XDC 文件。

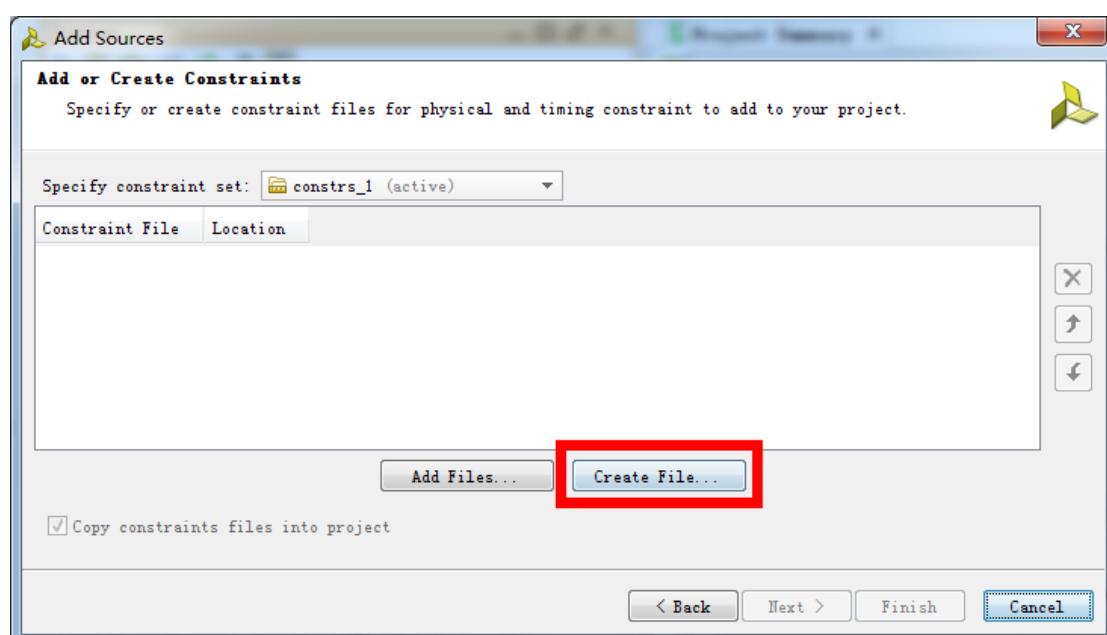


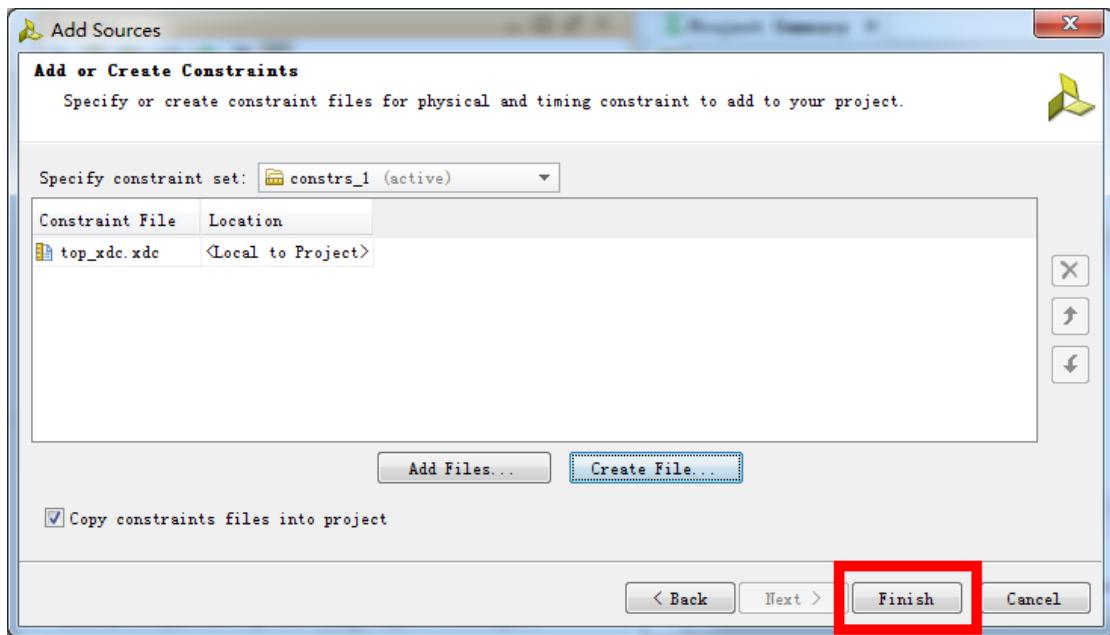
b、如何利用第二种方法添加约束文件。

b.8.1、点击 Add Sources，选择第一项 Add or Create Constraints 一项，点击 Next。

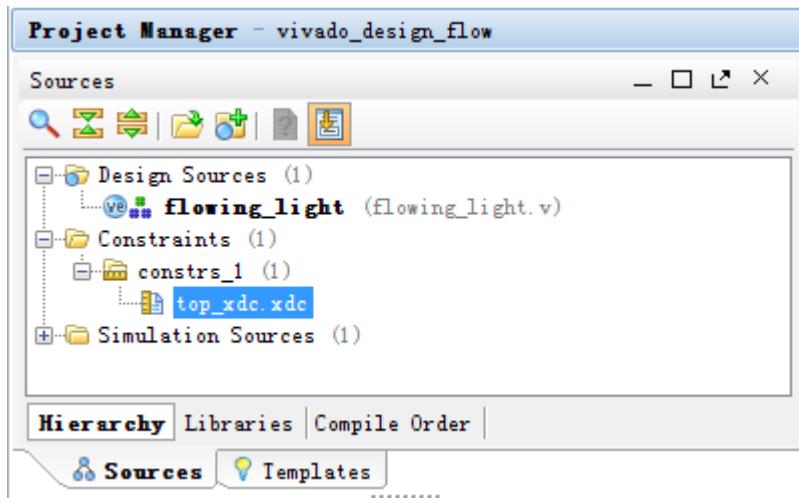


b.8.2、点击 **Create File**，新建一个 XDC 文件，输入 XDC 文件名，点击 OK。点击 Finish。





b.8.3、双击打开新建好的 XDC 文件，并按照如下规则，输入相应的 FPGA 管脚约束信息和电平标准。



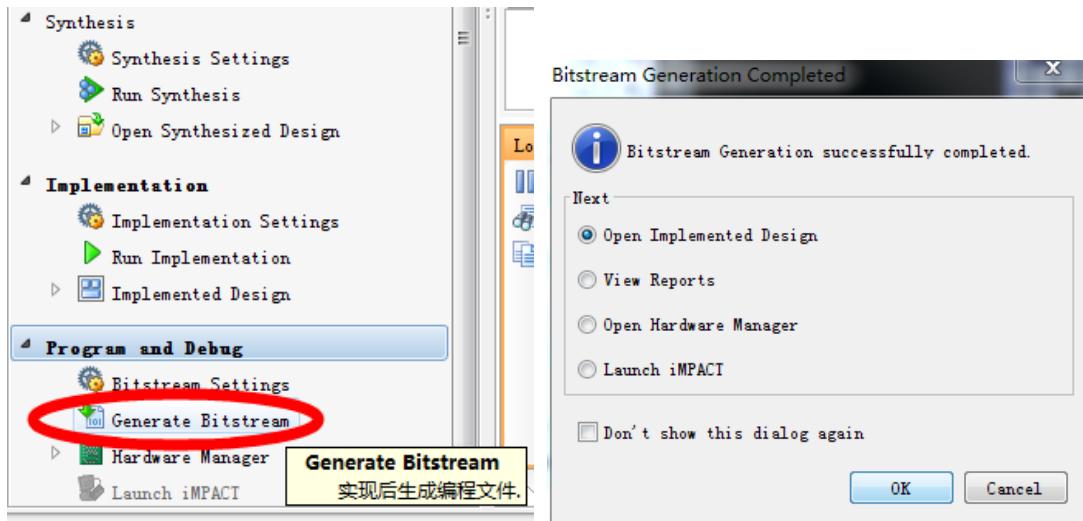
```

set_property PACKAGE_PIN L16 [get_ports clk]
set_property PACKAGE_PIN G15 [get_ports rst]
set_property PACKAGE_PIN M14 [get_ports {led[0]}]
set_property PACKAGE_PIN M15 [get_ports {led[1]}]
set_property PACKAGE_PIN G14 [get_ports {led[2]}]
set_property PACKAGE_PIN D18 [get_ports {led[3]}]
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]

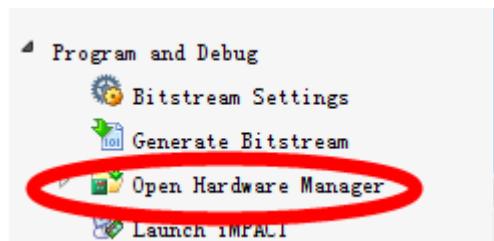
```

### 三、工程实现

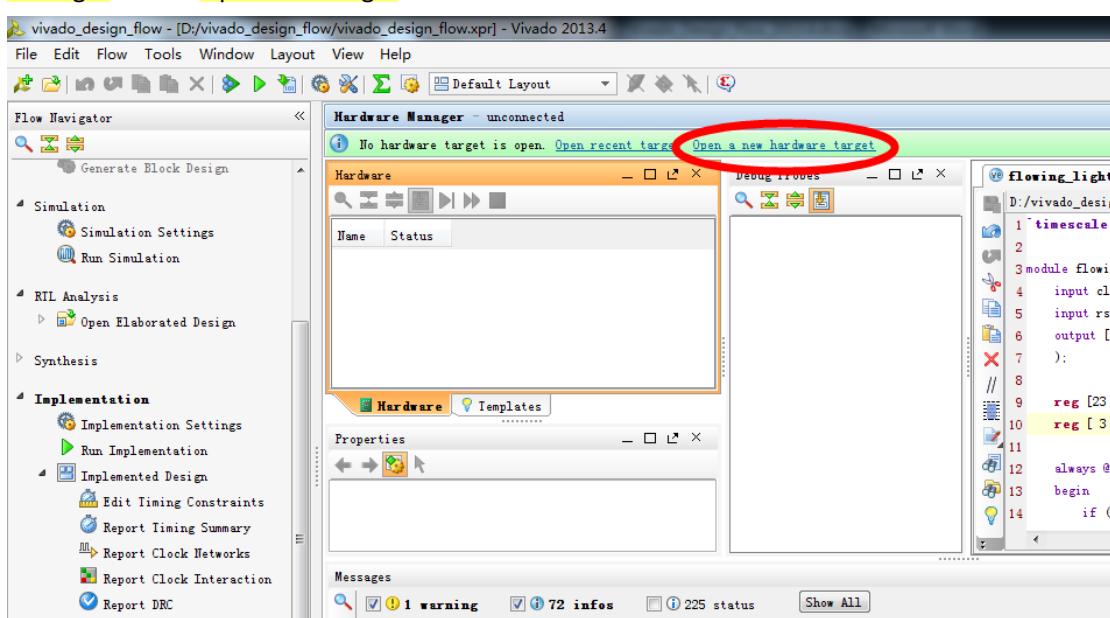
1、在 **Flow Navigator** 中点击 **Program and Debug** 下的 **Generate Bitstream** 选项，工程会自动完成综合、实现、Bit 文件生成过程，完成之后，可点击 **Open Implemented Design** 来查看工程实现结果。

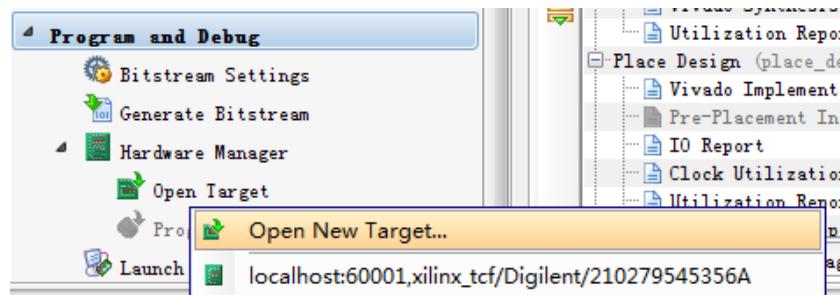


2、点击 **Flow Navigator** 中 **Open Hardware Manager** 一项，进入硬件编程管理界面。

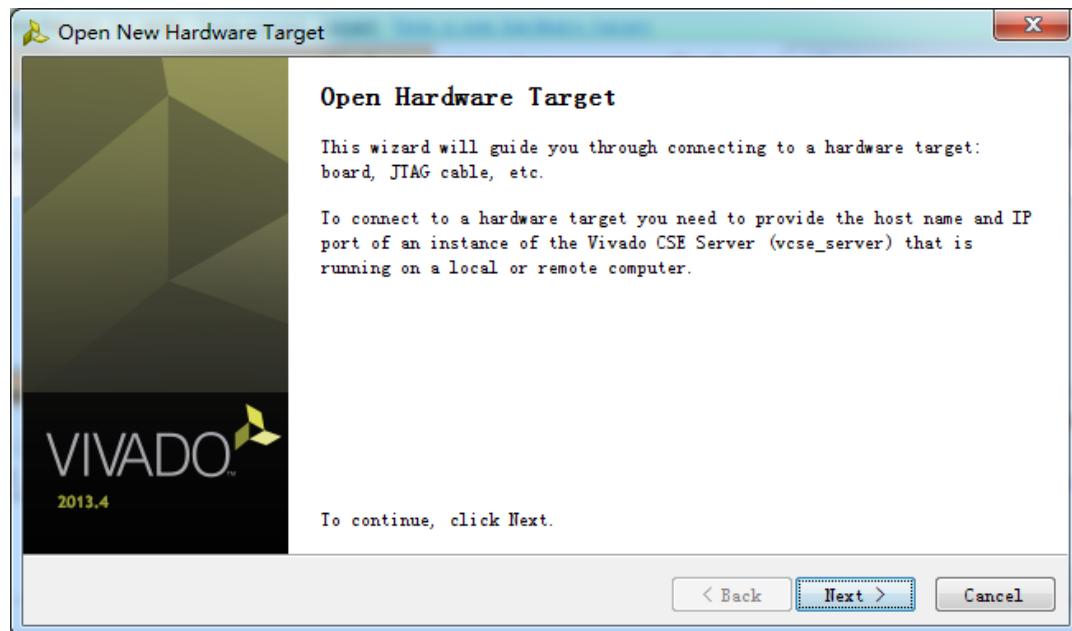


3、在提示的信息中，选择 **Open a new hardware target**（或在 **Flow Navigator** 中展开 **Hardware Manager**，点击 **Open New Target**），

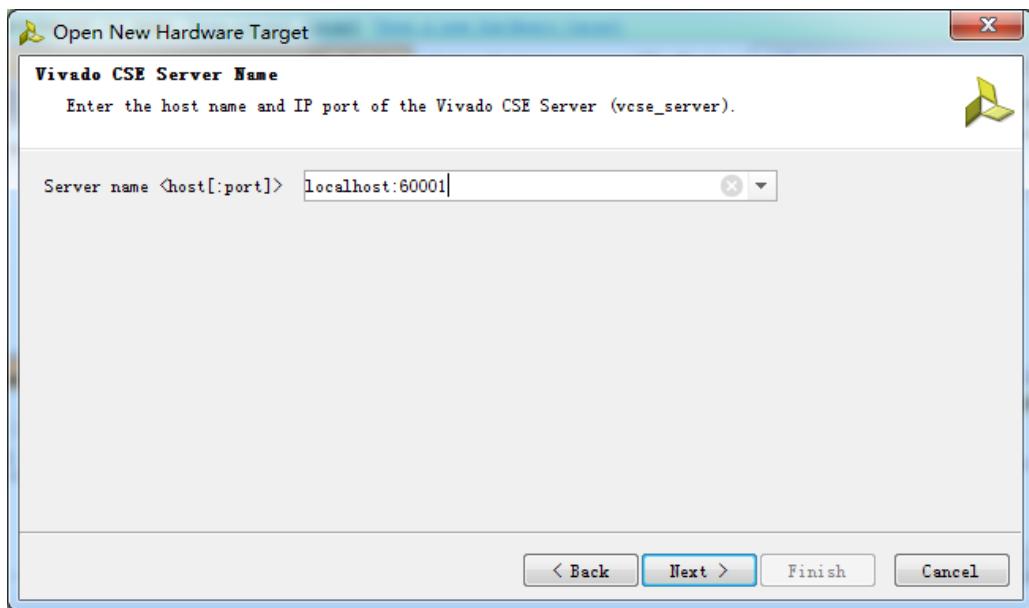




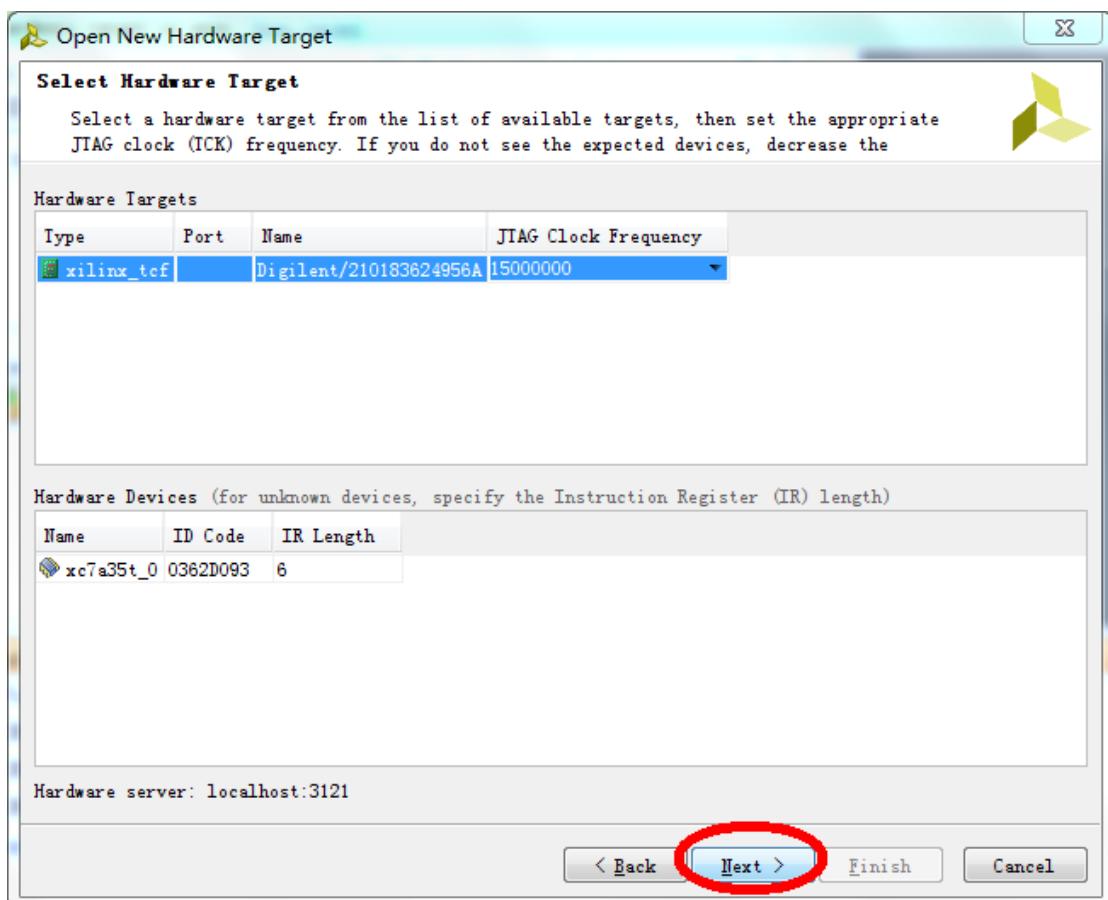
4、在弹出的 Open hardware target 向导中，先点击 Next，进入 Server 选择向导。

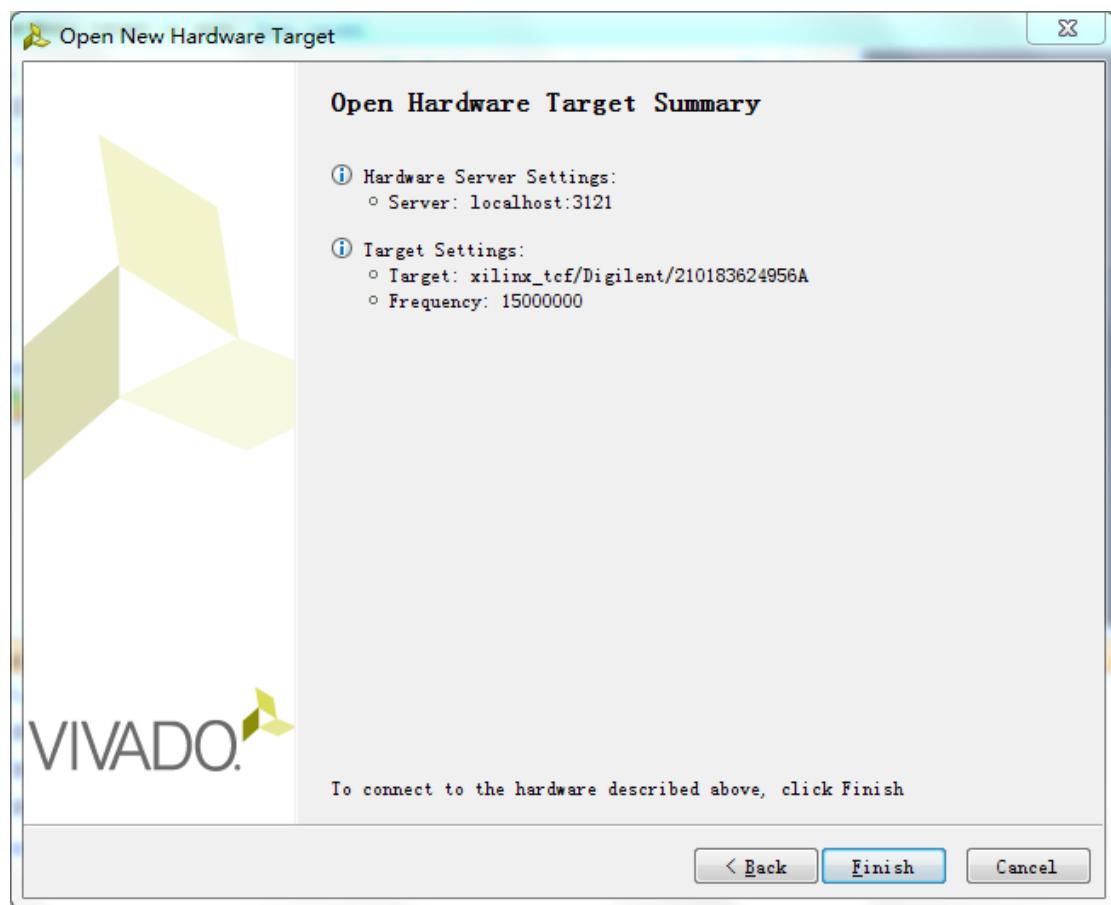


5、保持默认的 Server name 为 localhost: 60001，如无默认，在下拉框中选择即可。连接好板卡的 PROG 端口，并上电。点击 Next。

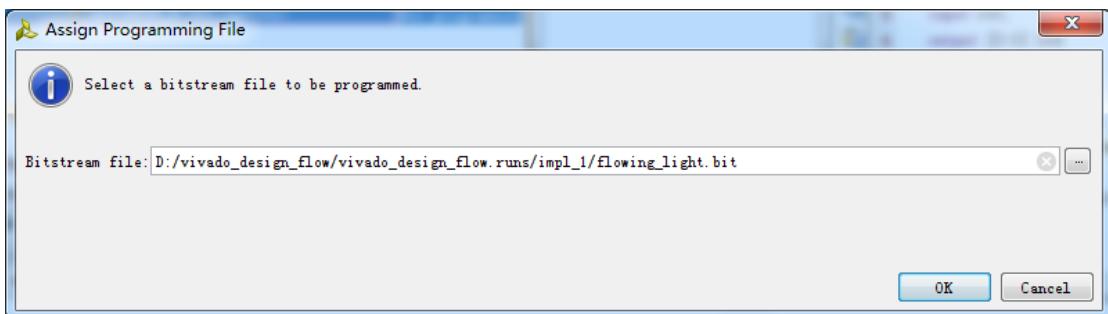
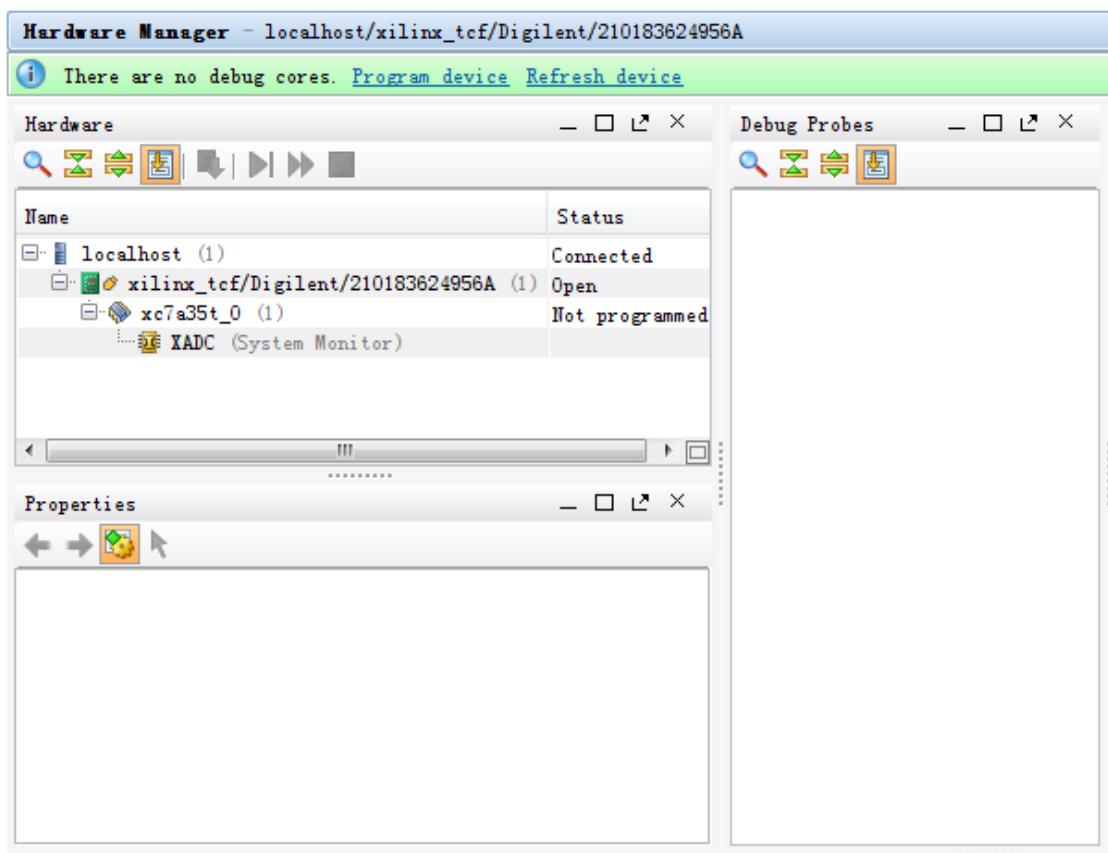


6、依次点击 Next、Next、Finish 完成新建 Hardware Target。

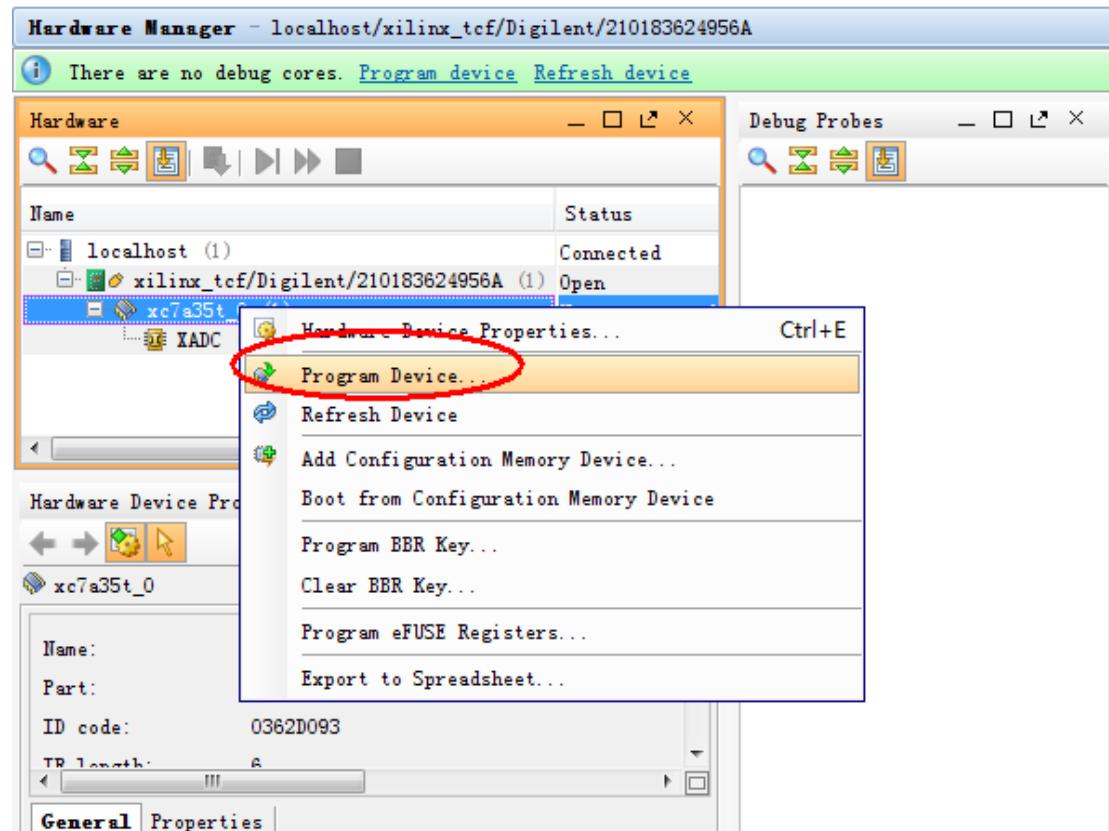
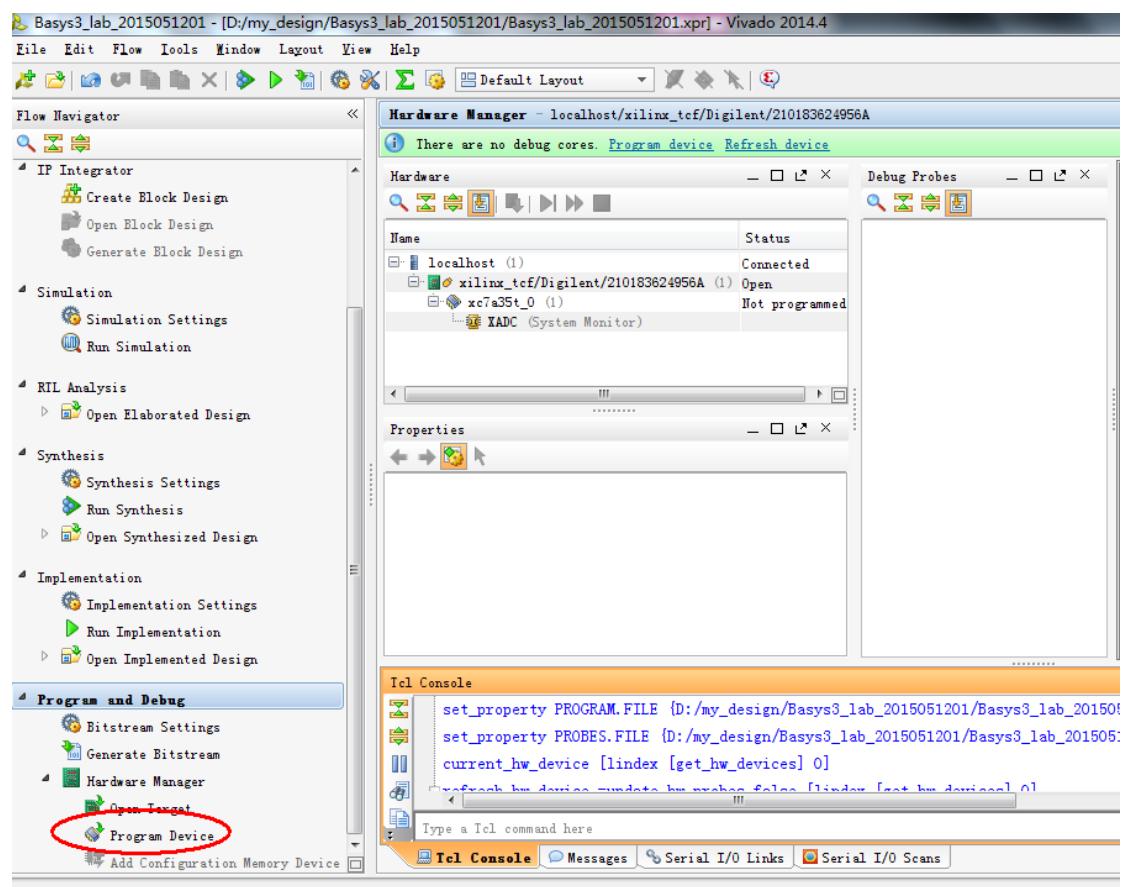




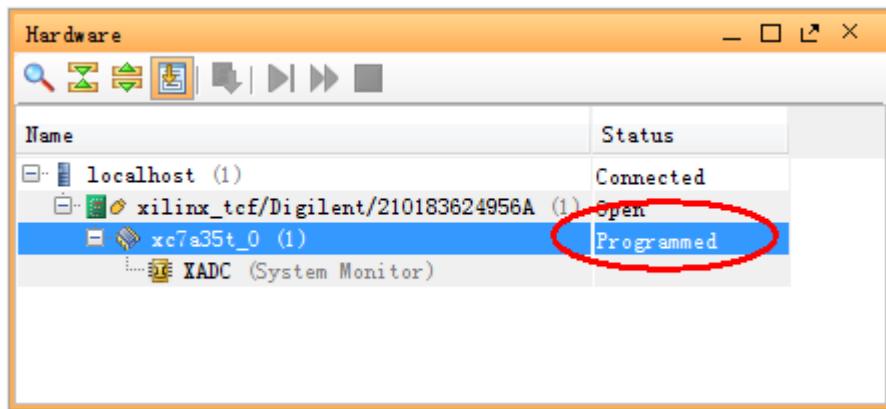
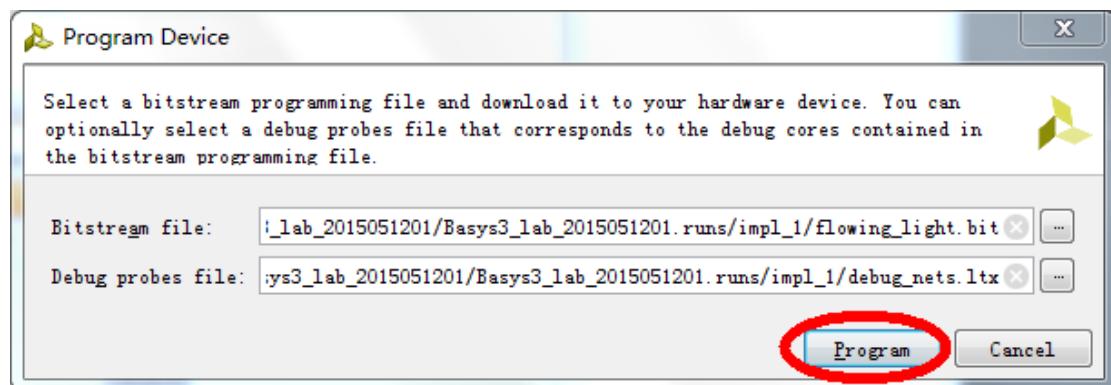
7、此时，**Hardware**一栏中出现硬件平台上可编程的器件。在对应的**FPGA**器件上右击，选择**Assign Programming File**，指定所需的**bit**文件。（系统默认已存在该工程的**bit**文件，如不需更改，可跳过该步骤）



8、在 FPGA 器件上右击选择 **Program Device** 或在 **Flow Navigator** 下的 **Hardware Manager** 中点击 **Program Device** 一项。



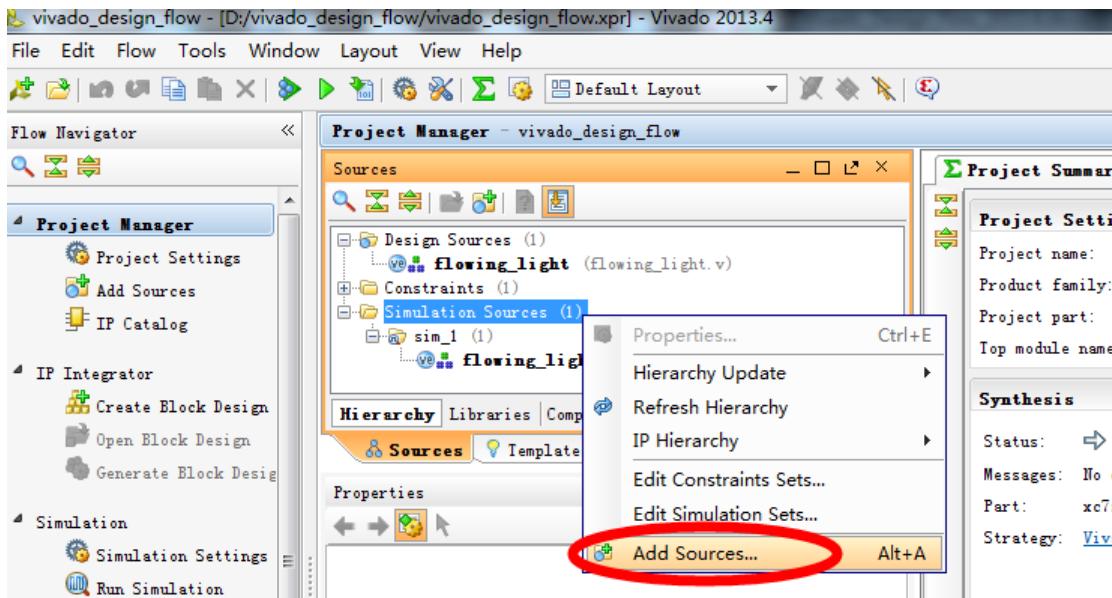
9、点击 **Program**, 将 bit 文件下载到板卡上的 FPGA 中, 此时 Hardware 对应的 FPGA 状态即变为 **Programmed**。



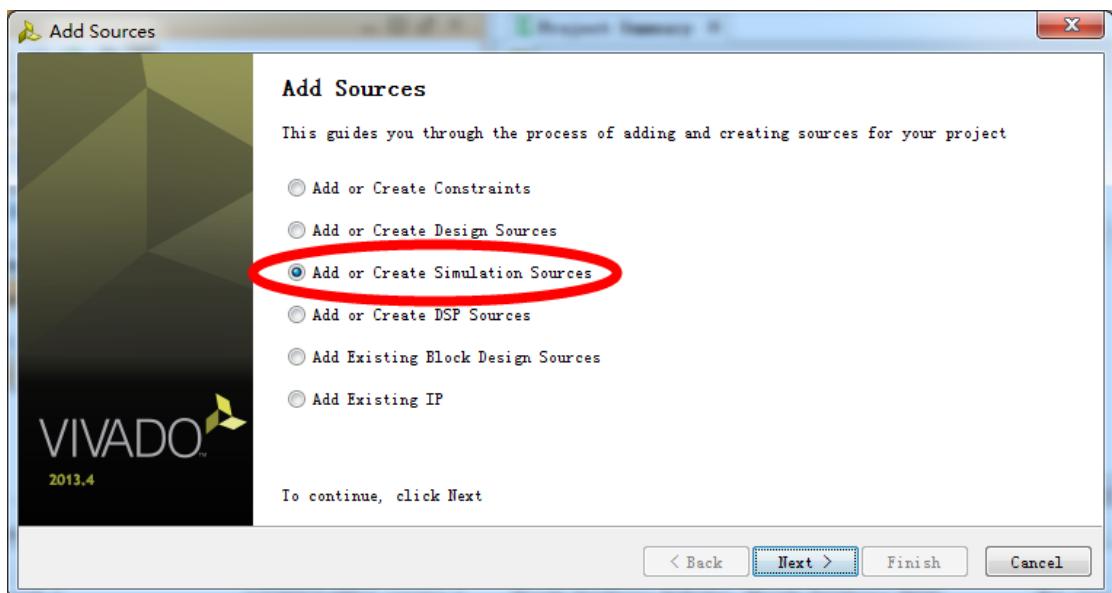
10、设计完成。

## 附一：利用 Vivado 进行功能仿真

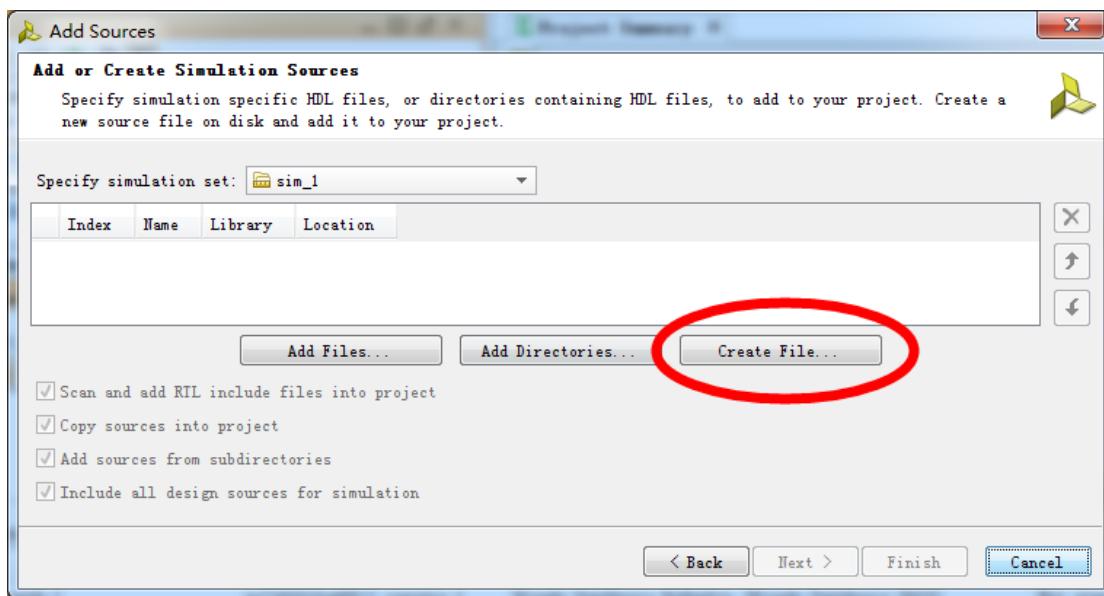
1、创建激励测试文件，在 Source 中右击选择 Add source



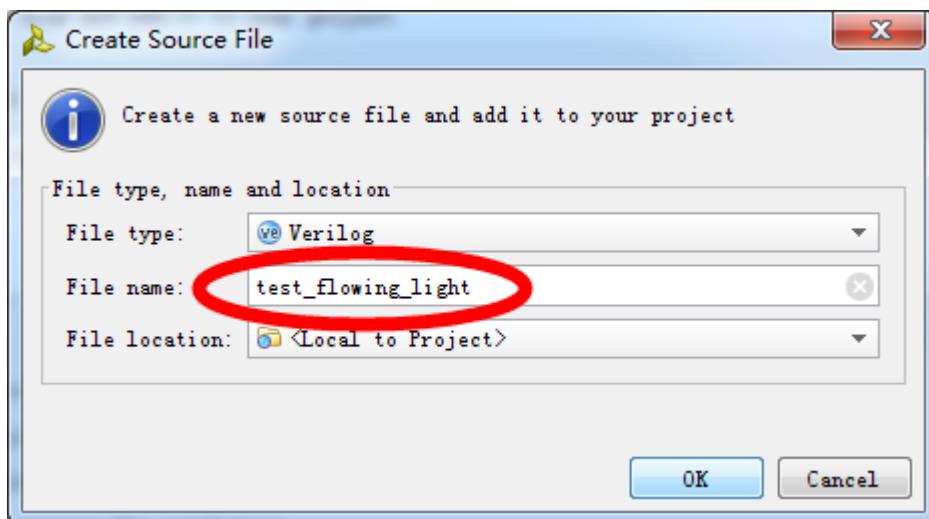
2、在 Add Source 界面中选择第三项 Add or Create Simulation Sources，点击 Next。

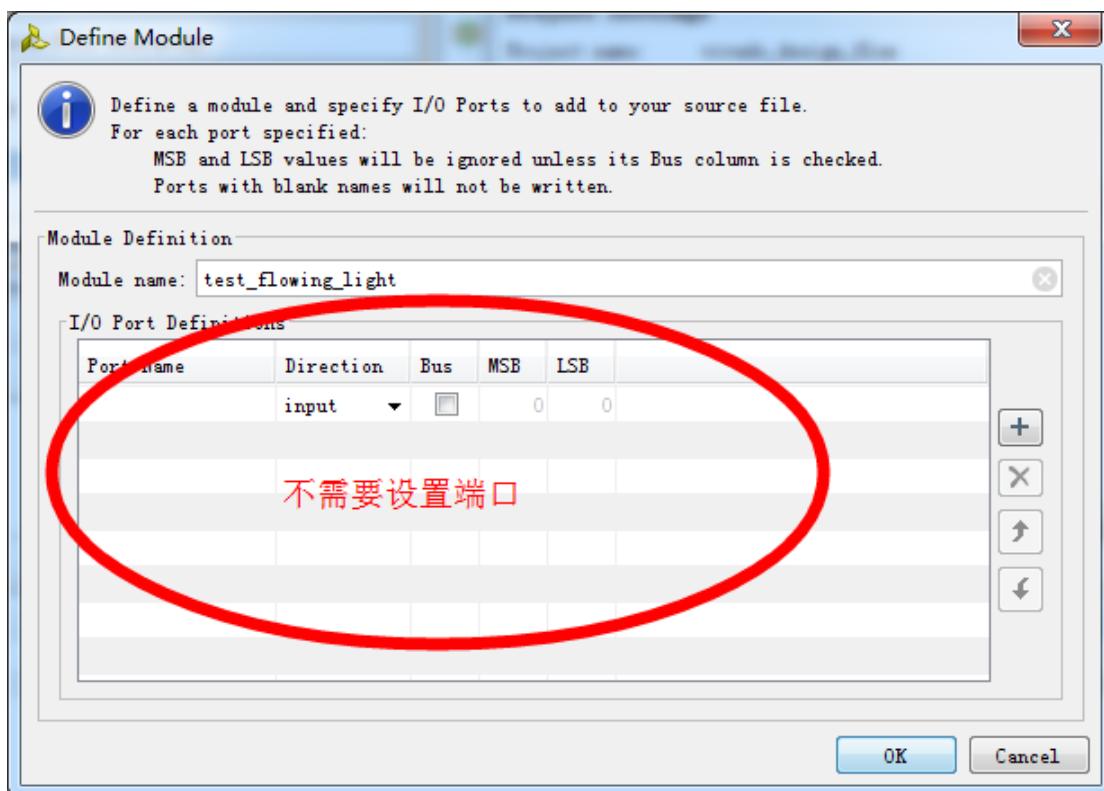
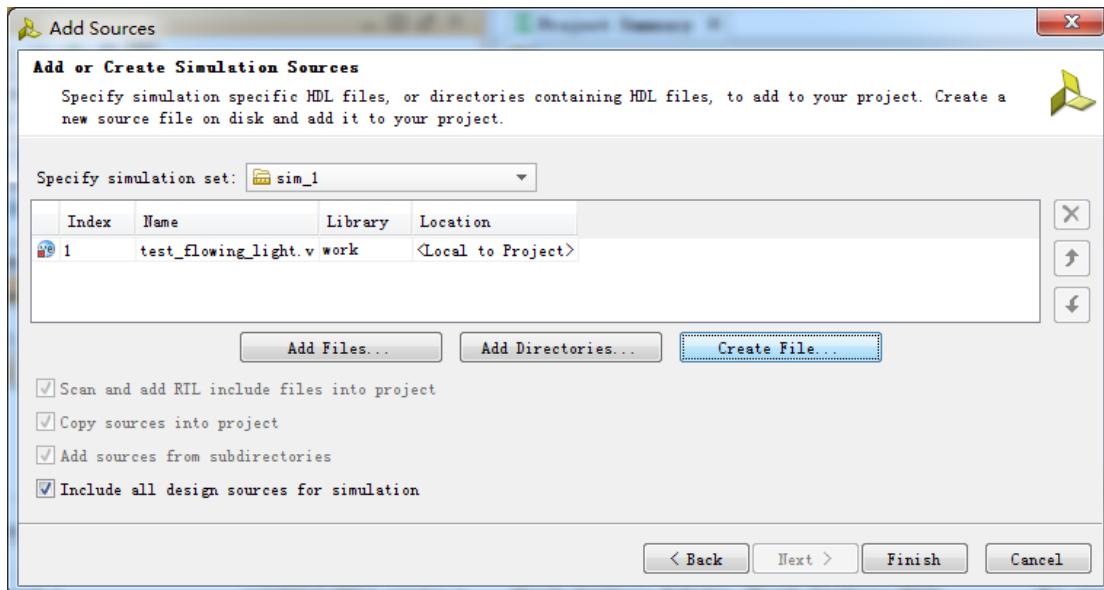


3、选择 Create File，创建一个新的激励测试文件

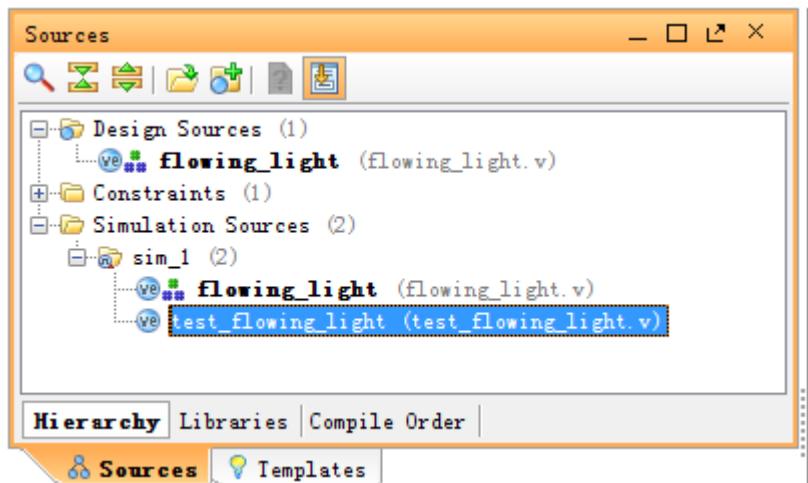


4、输入激励测试文件名，点击 OK，然后点击 Finish，弹出 module 端口定义对话框，由于此处是激励测试文件，不需要有对外的接口，所以，此处为空。点击 OK，空白的激励测试文件就建好了。





5、在 Source 下双击打开空白的激励测试文件，完成对将要仿真的 module 的实例化和激励代码的编写，如下图和下述代码所示。



```
'timescale 1ns / 1ps
module test_flow_light();
    reg        clk;
    reg        rst;
    wire [3 : 0] led;

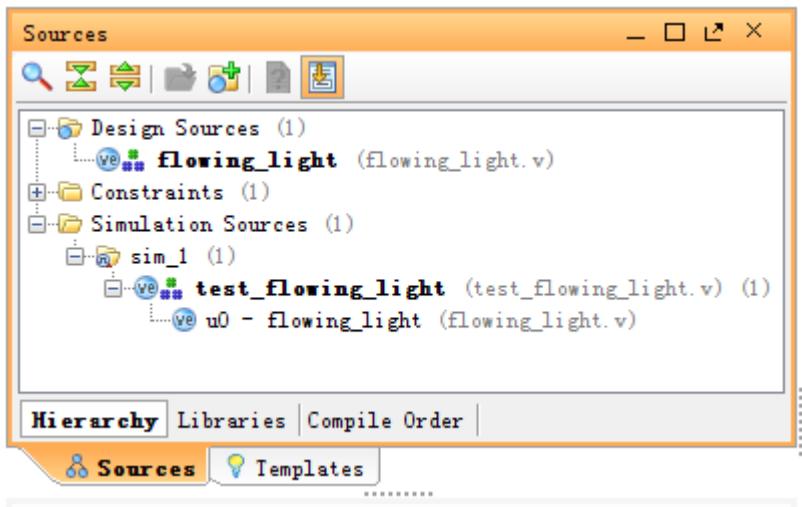
    flowing_light u0(
        .clk(clk),
        .rst(rst),
        .led(led) );

    parameter PERIOD = 10;

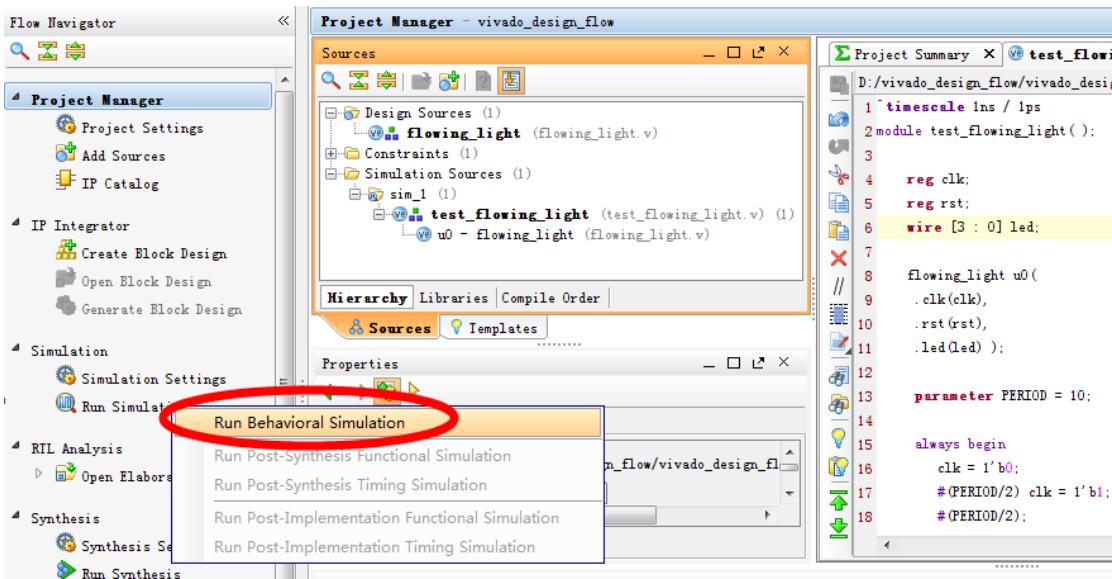
    always begin
        clk = 1'b0;
        #(PERIOD/2) clk = 1'b1;
        #(PERIOD/2);
    end

    initial begin
        clk = 1'b0;
        rst = 1'b0;
        #100;
        rst = 1'b1;
        #100;
        rst = 1'b0;
    end
endmodule
```

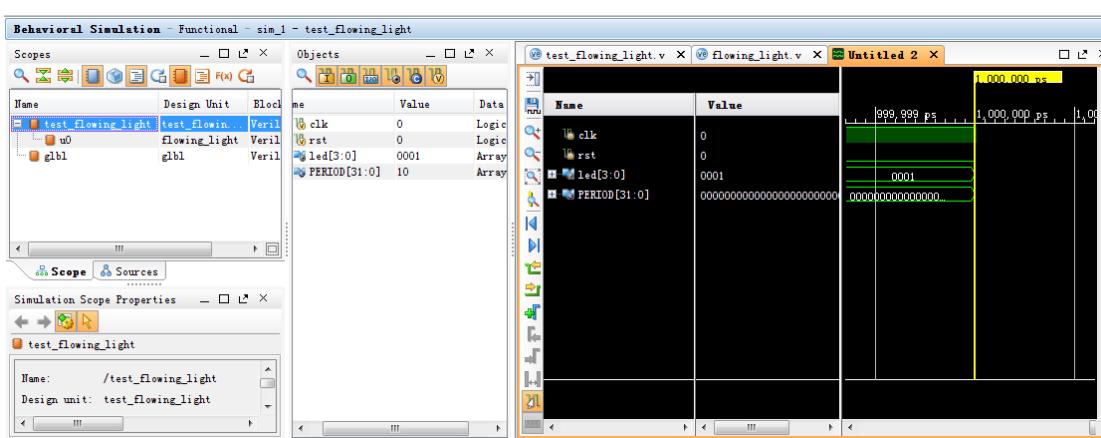
激励文件完成之后，工程目录如下图所示。



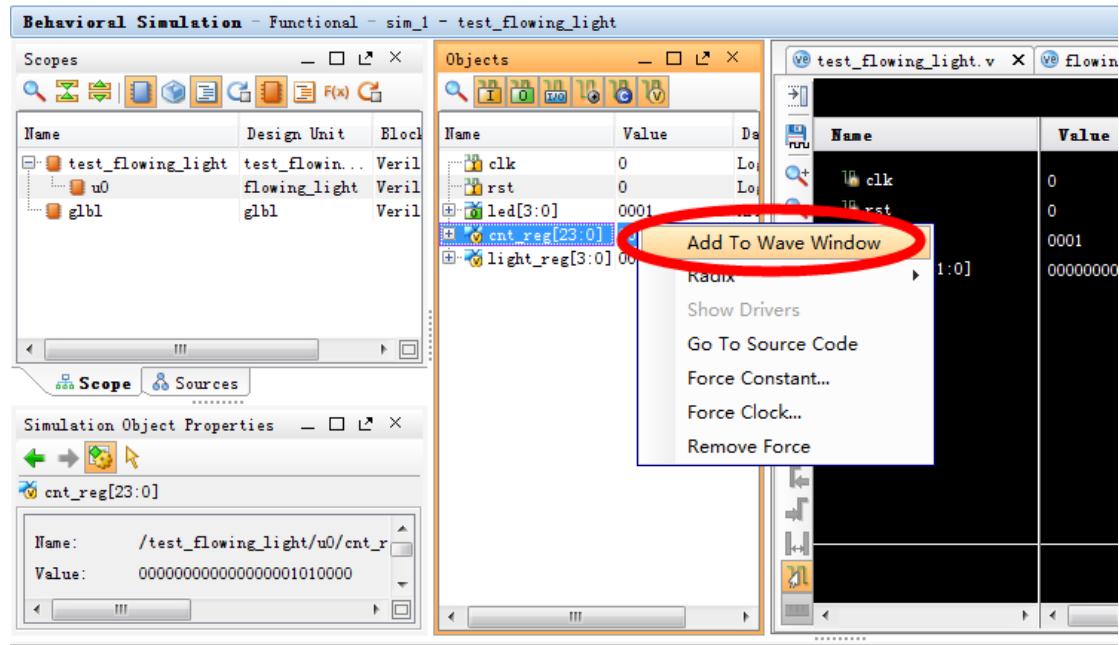
6、此时，进入仿真。在左侧 Flow Navigator 中点击 Simulation 下的 Run Simulation 选项，并选择 Run Behavioral Simulation 一项，进入仿真界面。



7、下图所示为仿真界面。



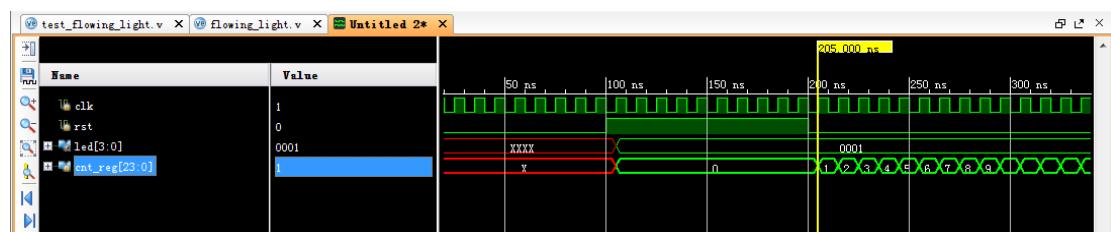
可通过左侧 Scope 一栏中的目录结构定位到设计者想要查看的 module 内部寄存器，在 Objects 对应的信号名称上右击选择 Add To Wave Window，将信号加入波形图中。



可通过选择工具栏中的如下选项来进行波形的仿真时间控制。如下工具条，分别是复位波形（即清空现有波形）、运行仿真、运行特定时长的仿真、仿真时长设置、仿真时长单位、单步运行、暂停……



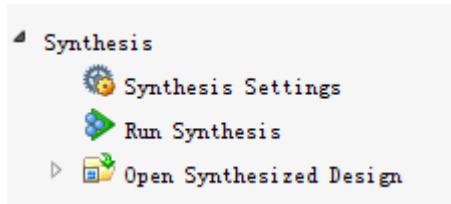
8、最终得到的仿真效果图如下。核对波形与预设的逻辑功能是否一致。仿真完成。



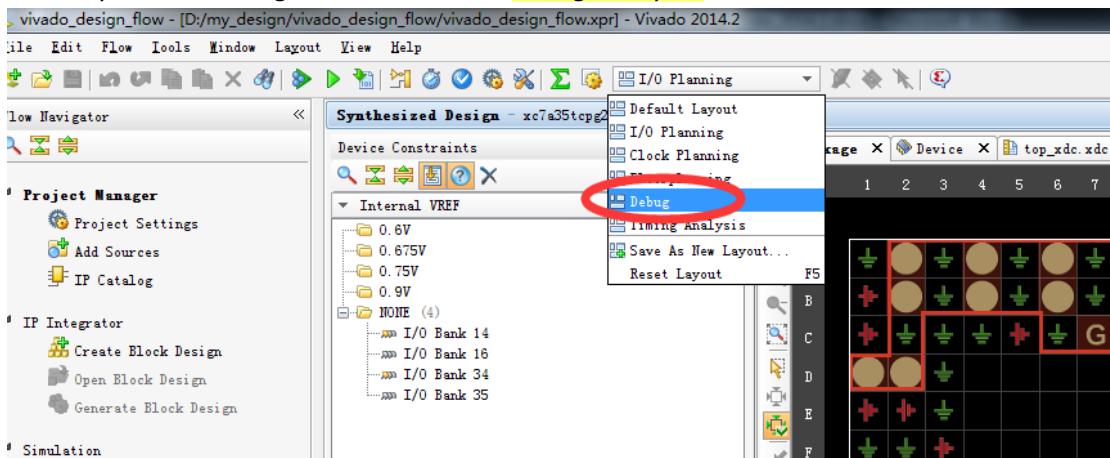
## 附二：利用 Vivado 进行硬件 Debug 流程

利用 Vivado 进行硬件 Debug 的过程有两种：一是基于综合生成的网表，利用 Set Debug 的向导，完成 Debug 设置；二是在源代码中利用 ILA 的 IP 核，直接提取要查看的信号。此处，以第一种 Debug 形式为例。

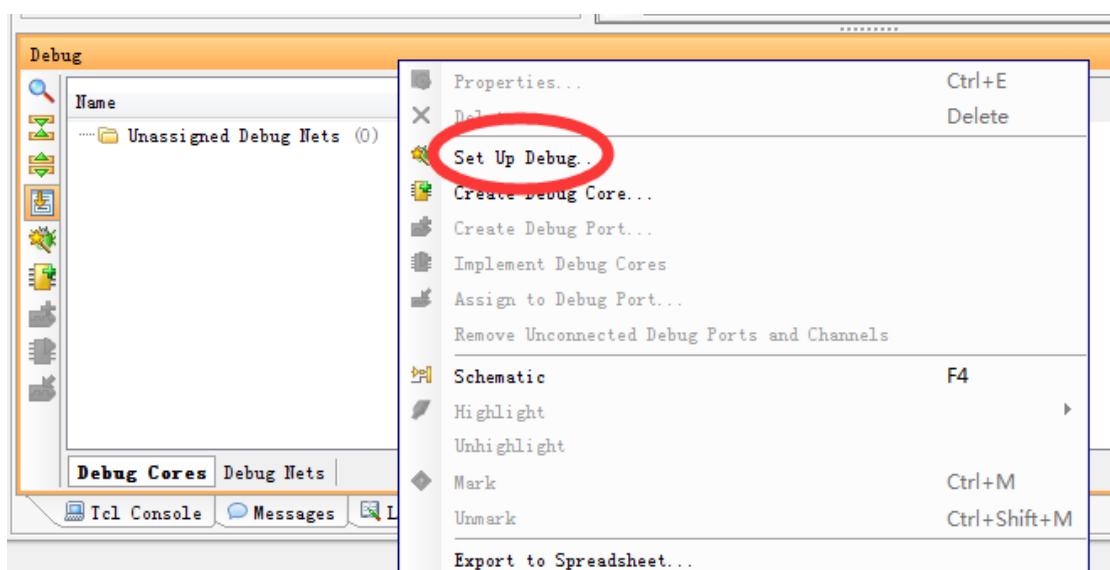
### 1、工程综合后，选择 Open Synthesized Design

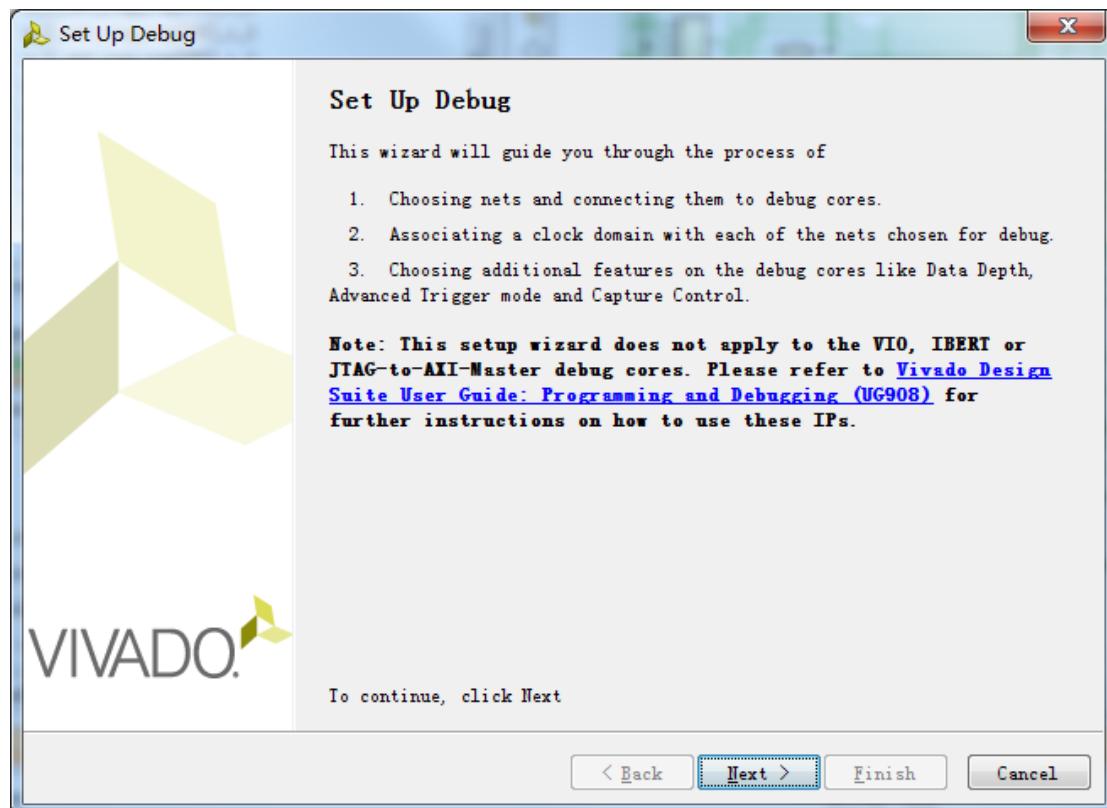


### 2、将 Synthesized Design 查看界面切换至 Debug 的 Layout，如下图所示

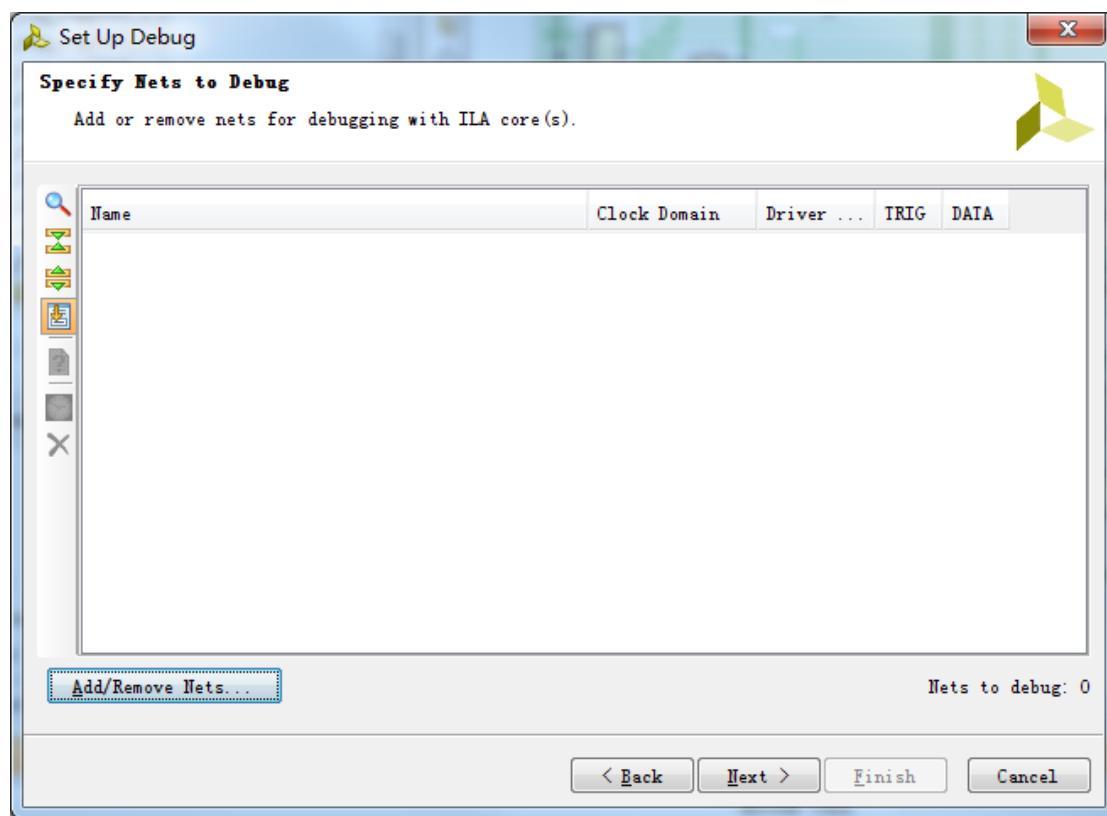


### 3、在工作区域下方的 Debug 选项卡空白处右击选择 Set Up Debug 选项，进入 Debug 设置向导，点击 Next。

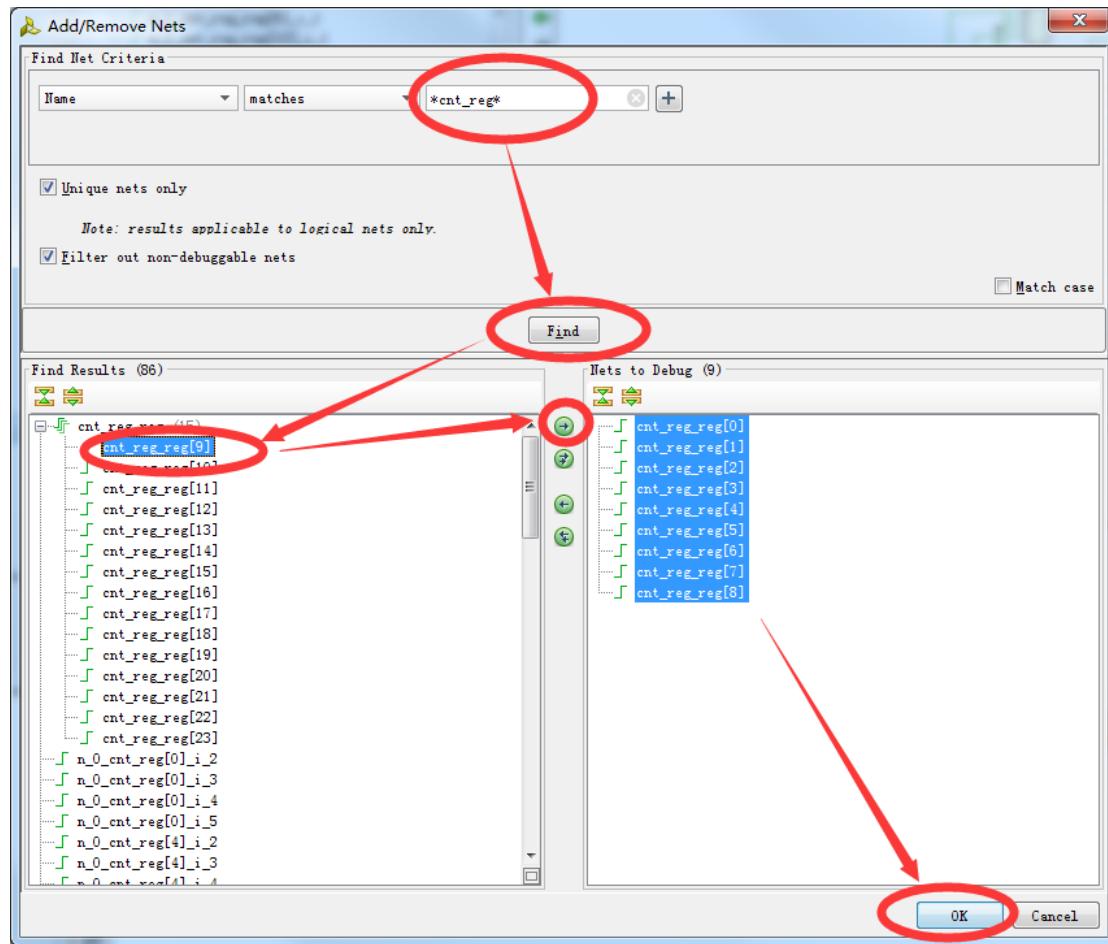




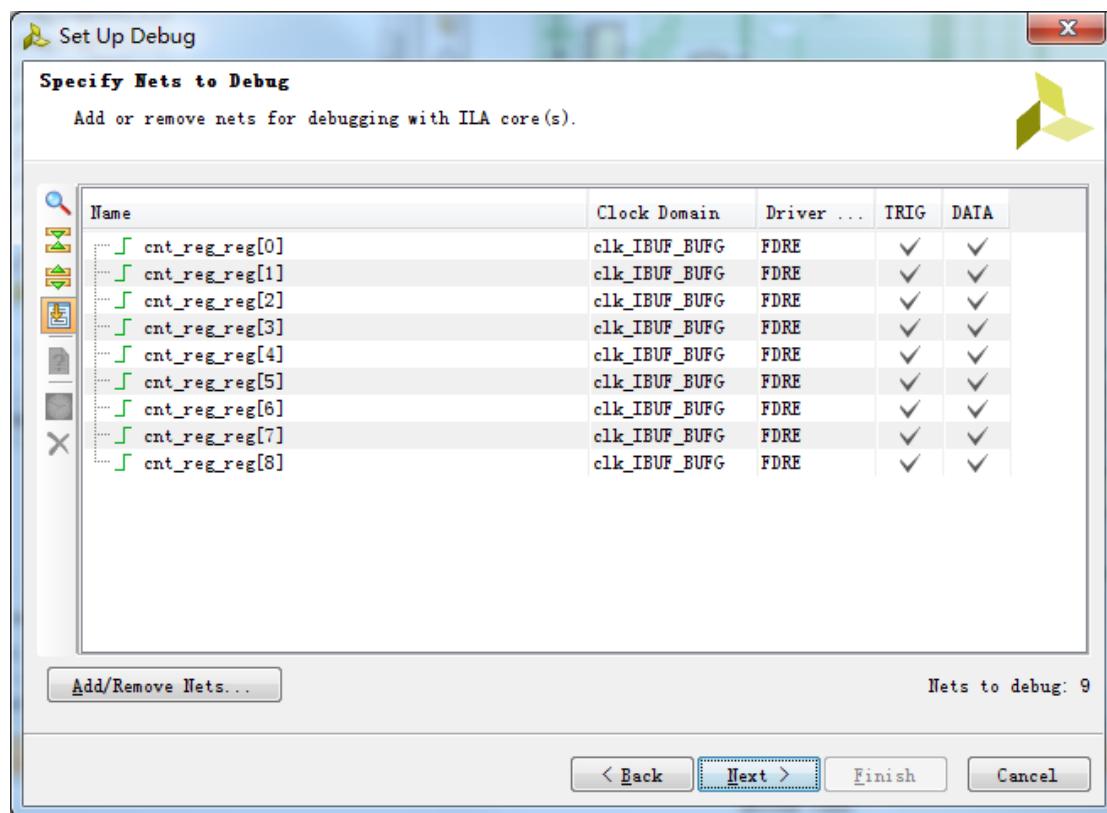
4、添加需要查看的信号，在下述的 **Specify Nets to Debug** 界面左下方点击 **Add/Remove Nets** 选项，进入侦测信号添加向导。



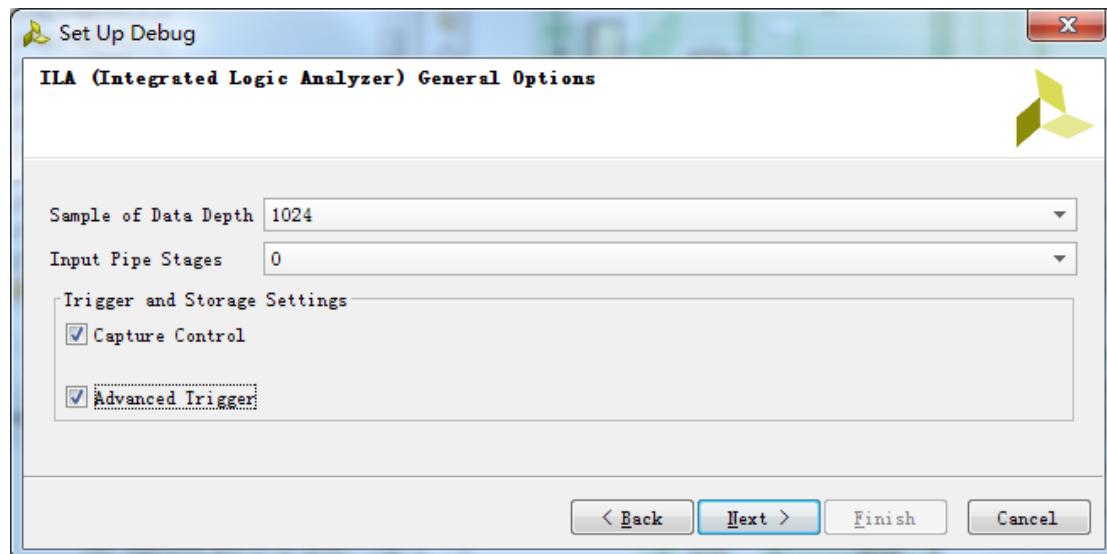
5、在 Add/Remove Nets 界面中，可通过 Fine Net Criteria 选项查找预侦测的信号，或直接点击 Find 按钮列出所有可侦测的信号，从中选择预侦测信号，并点击绿色箭头，添加至 Nets to Debug 列表中。点击 Finish。

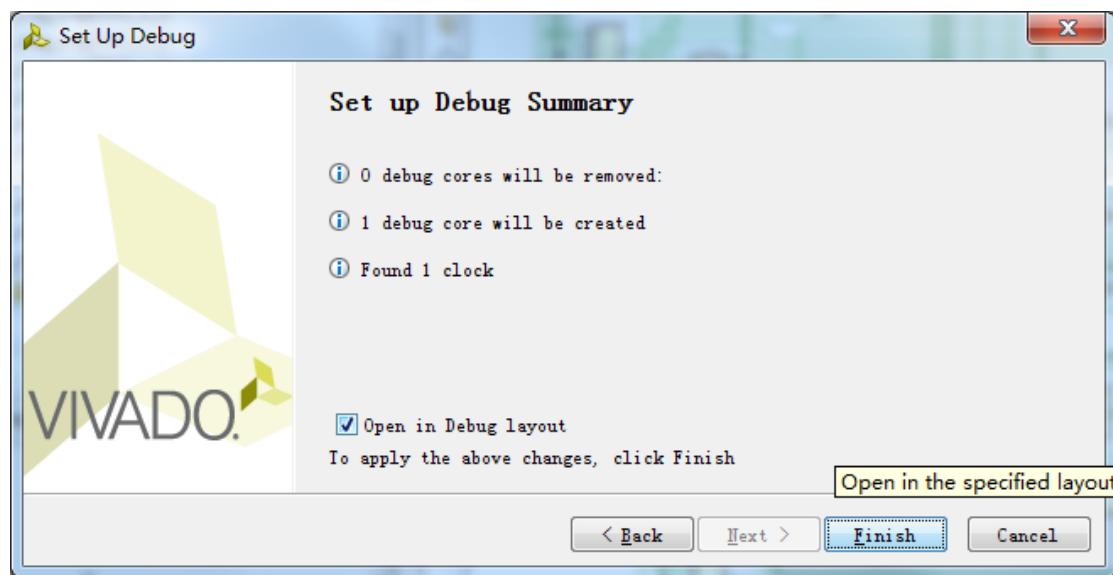


6、此时添加的预侦测信号已存在于 Debug 列表中，点击 Next。

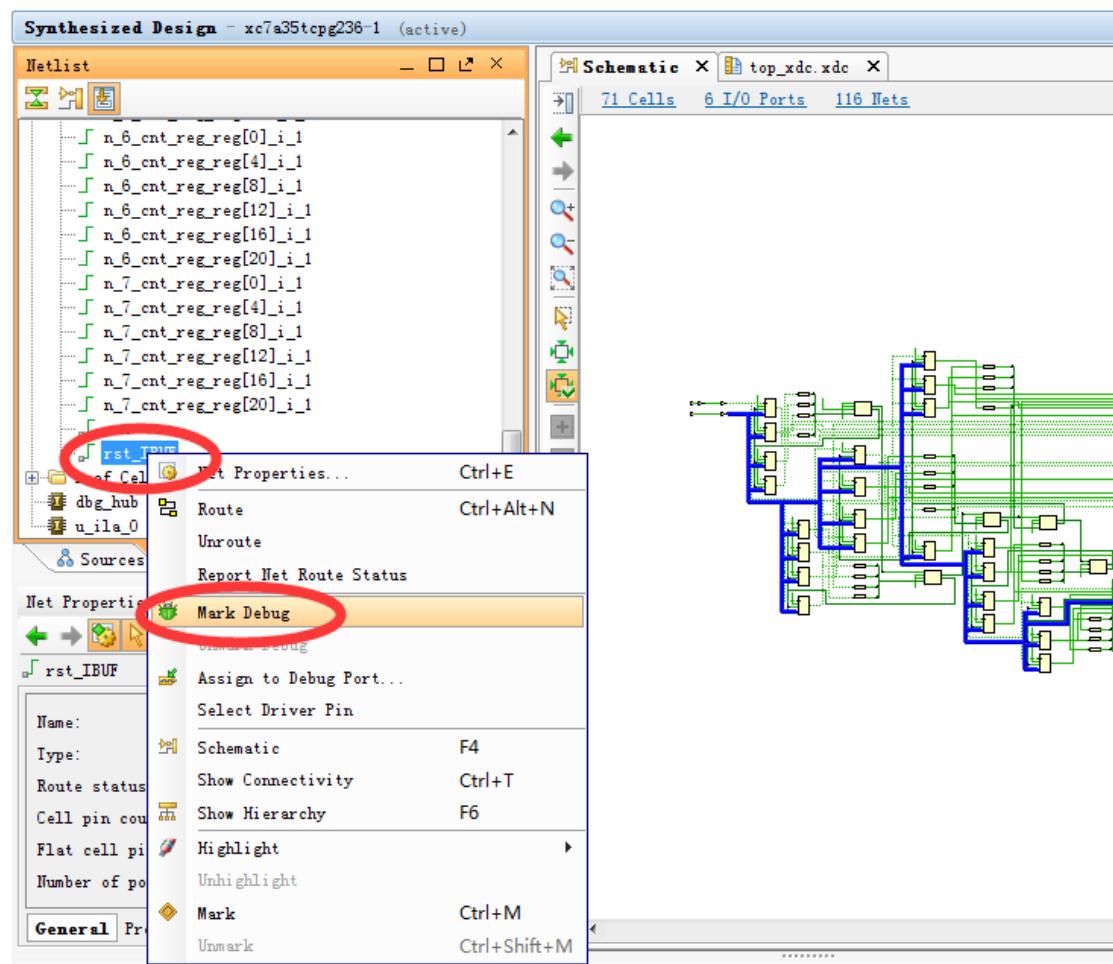


7、设置采样深度 Sample of Data Depth 为 1024 (其他值亦可, 根据),

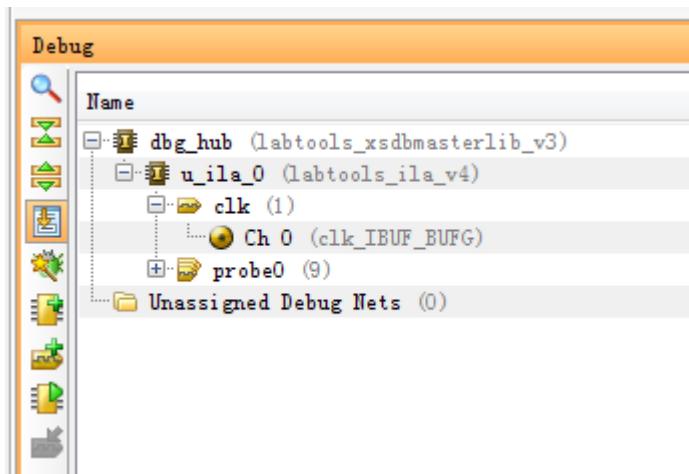




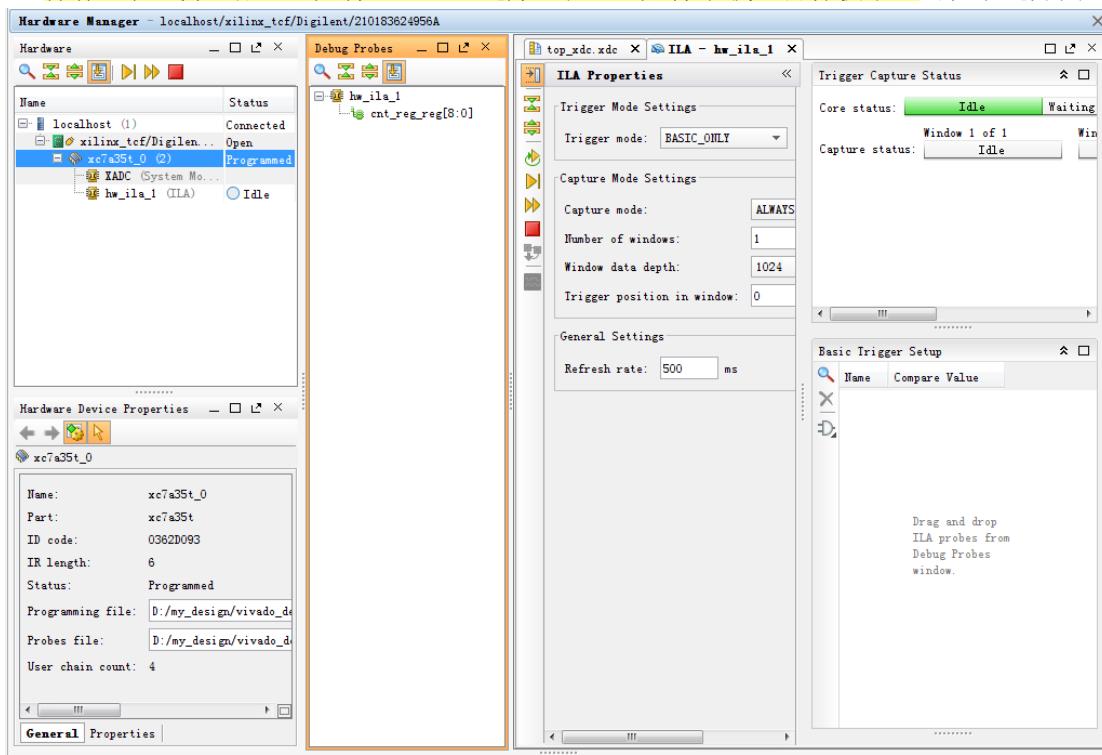
(补充，便捷的方法)



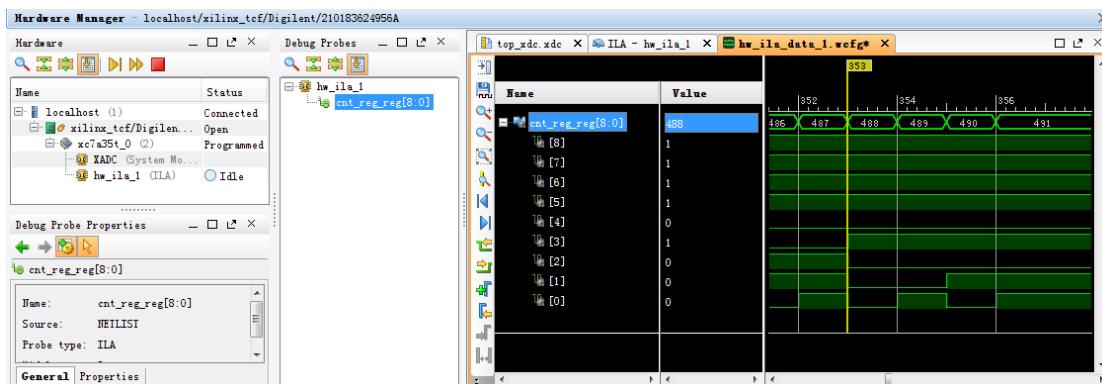
8、完成 Debug 信号设置后，Debug 窗口下出现下述界面。



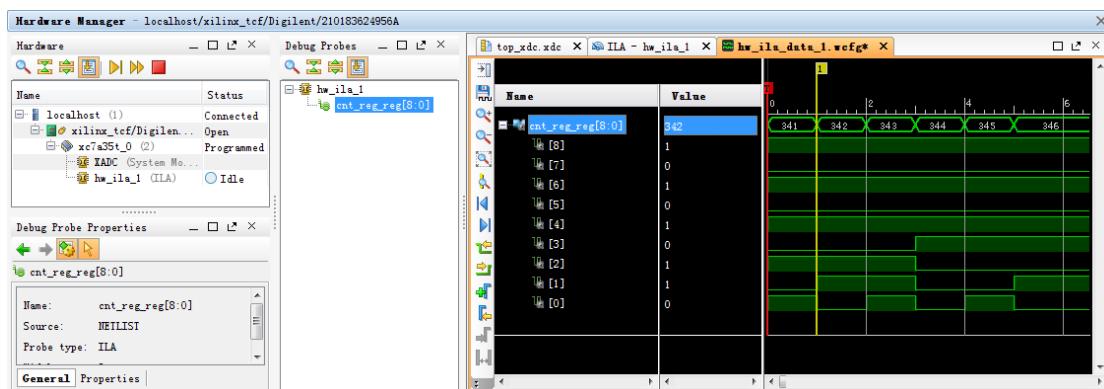
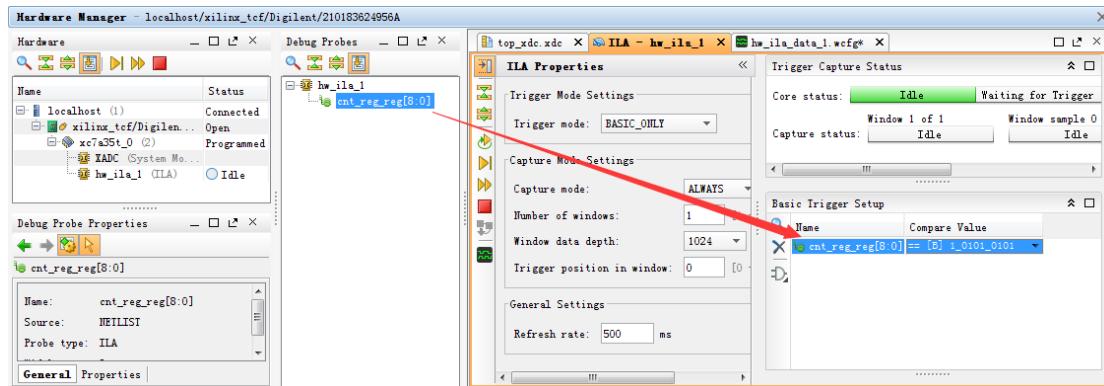
9、保存工程，并生成 BIT 文件。通过上述办法把 BIT 文件下载至目标板卡。出现下述界面。



10、点击触发采样，观察采样波形。



11、亦可添加触发条件设置，来准确获取采样点。



## 附三：Vivado 安装简介（以 Vivado 2014.2 为例）

Vivado 软件支持的操作系统：

### Microsoft Windows Support

- Windows XP Professional (32-bit and 64-bit), English/Japanese
- Windows 7 and 7 SP1 Professional (32-bit and 64-bit), English/Japanese
- Windows 8.1 Professional (64-bit), English/Japanese

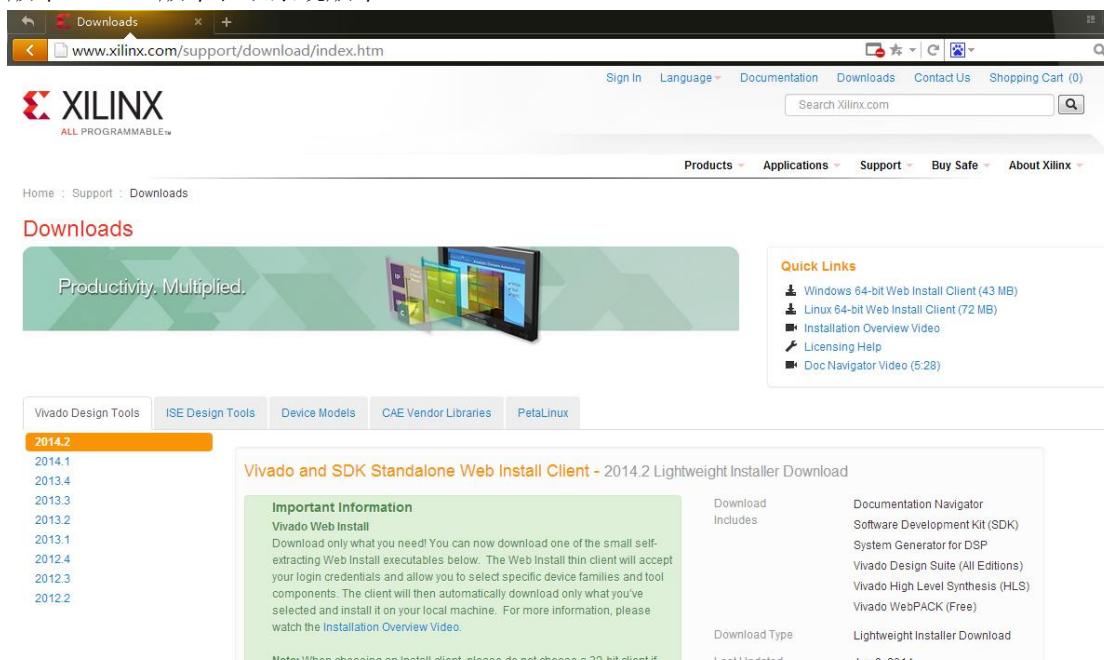
### Linux Support

- Red Hat Enterprise Workstation 5.8 - 5.10 (32-bit and 64-bit)
- Red Hat Enterprise Workstation 6.4 - 6.5 (32-bit and 64-bit)
- SUSE Linux Enterprise 11.1 - 11.2 (32-bit and 64-bit)
- Cent OS 6.4 and 6.5 (64-bit)

建议安装内存：大于 4GB；建议硬盘空间：大于 20GB。

注意：由于 Vivado 版本不同，安装时界面可能不一致，具体参考相应版本的 user guide。

1、在 <http://www.xilinx.com/support/download/index.htm> 下载所需的 Vivado 版本，Windows 版本、Linux 版本和双系统版本。

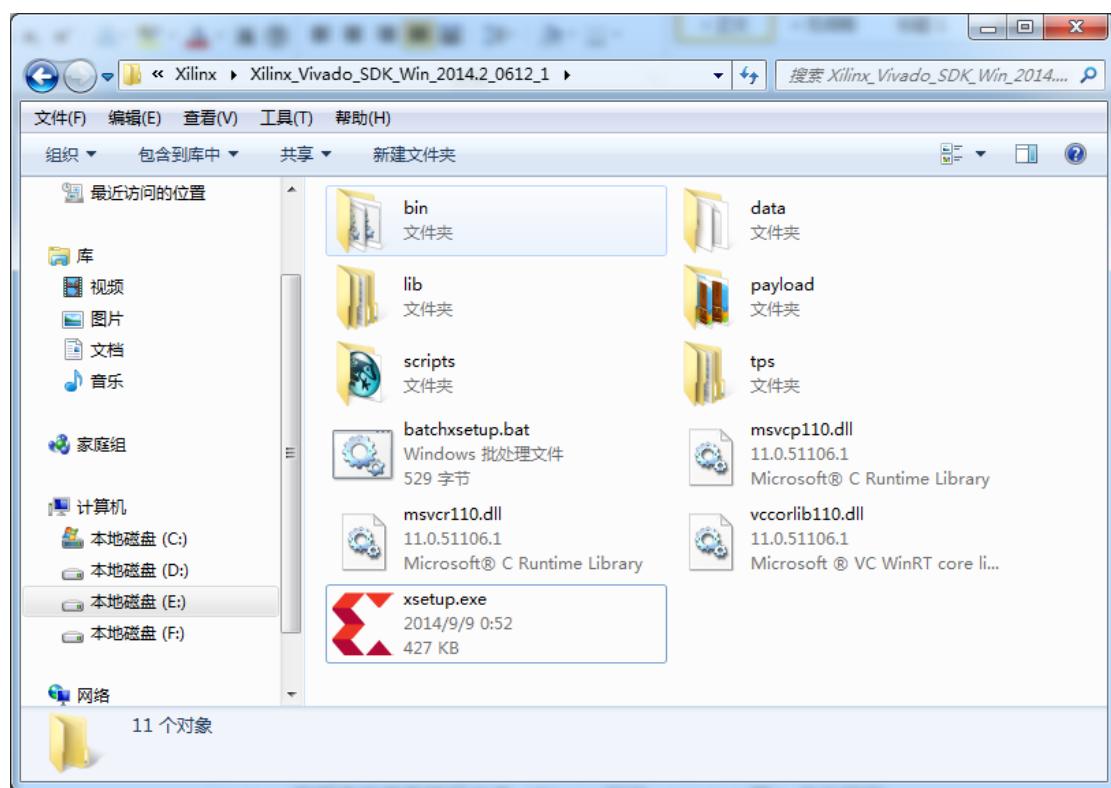


The screenshot shows the Xilinx support website's download section. The URL in the address bar is [www.xilinx.com/support/download/index.htm](http://www.xilinx.com/support/download/index.htm). The page title is "Downloads". A banner at the top says "Productivity. Multiplied." and features an image of a tablet displaying various software interfaces. On the left, there's a sidebar with a navigation menu for "Downloads" and a dropdown for "Vivado Design Tools" which is currently set to "2014.2". The main content area displays a "Vivado and SDK Standalone Web Install Client - 2014.2 Lightweight Installer Download" section. This section includes an "Important Information" box about the Vivado Web Install client, a "Download Includes" table listing the software components (Documentation Navigator, Software Development Kit (SDK), System Generator for DSP, Vivado Design Suite (All Editions), Vivado High Level Synthesis (HLS), Vivado WebPACK (Free)), and a "Download Type" table indicating it's a "Lightweight Installer Download" last updated on Jun 9, 2014.

下载后解压到不含中文和空格的路径中。

Windows 下可直接使用解压工具（如 360 压缩、WinRAR 等）进行解压。

Linux 下可使用 `tar -xvf Xilinx_Vivado_SDK_Win_2014.2_0612_1.tar.gz` 命令解压。



2、Windows 下直接双击 xsetup.exe 安装。

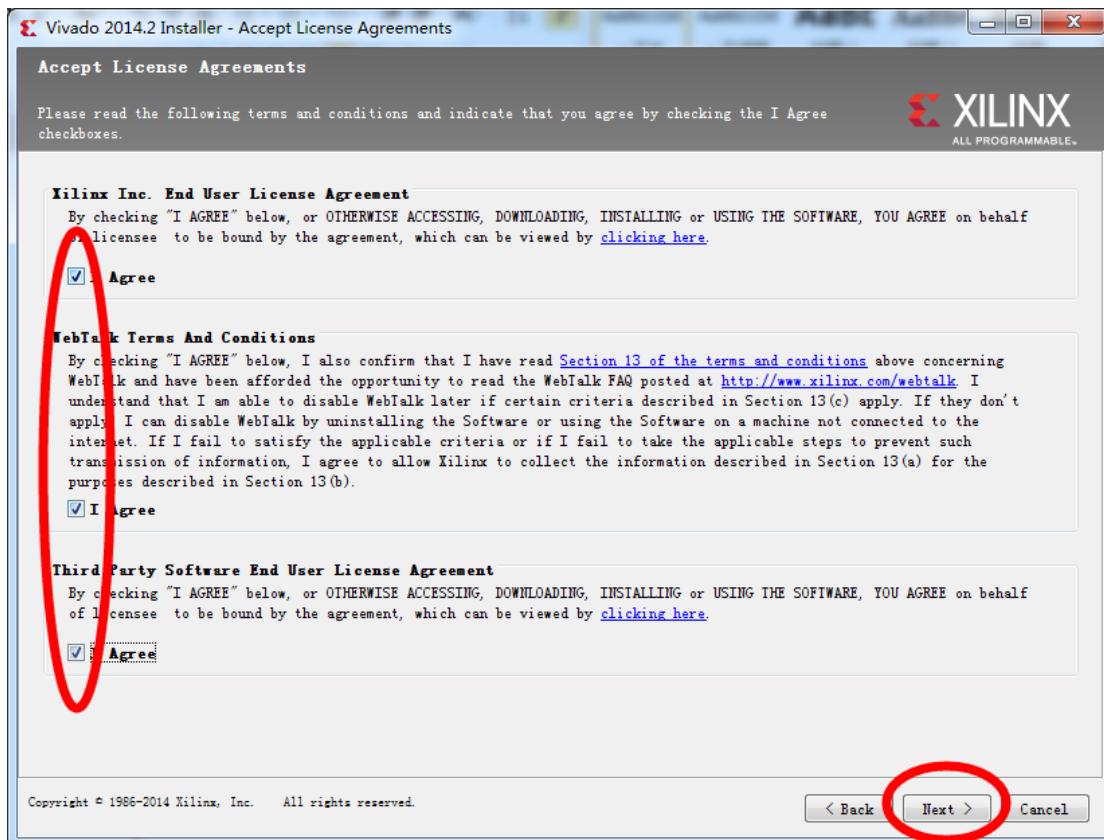
Linux 下需要先打开安装包文件夹的读写权限，然后安装，如下：

```
sudo chmod 777 /path to the software folder/Xilinx_Vivado_SDK_Win_2014.2_0612_1  
sudo cd /path to the software folder/Xilinx_Vivado_SDK_Win_2014.2_0612_1  
.xsetup
```

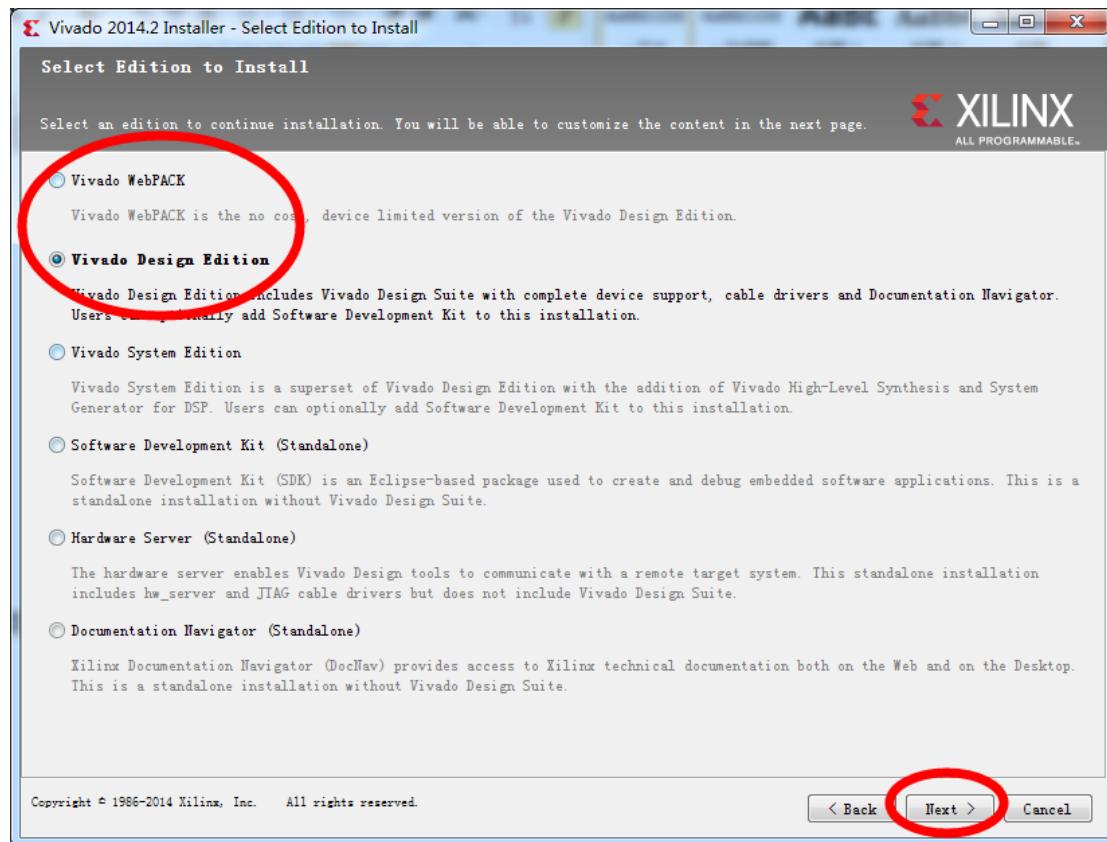
执行后进入下述安装向导，点击 Next。



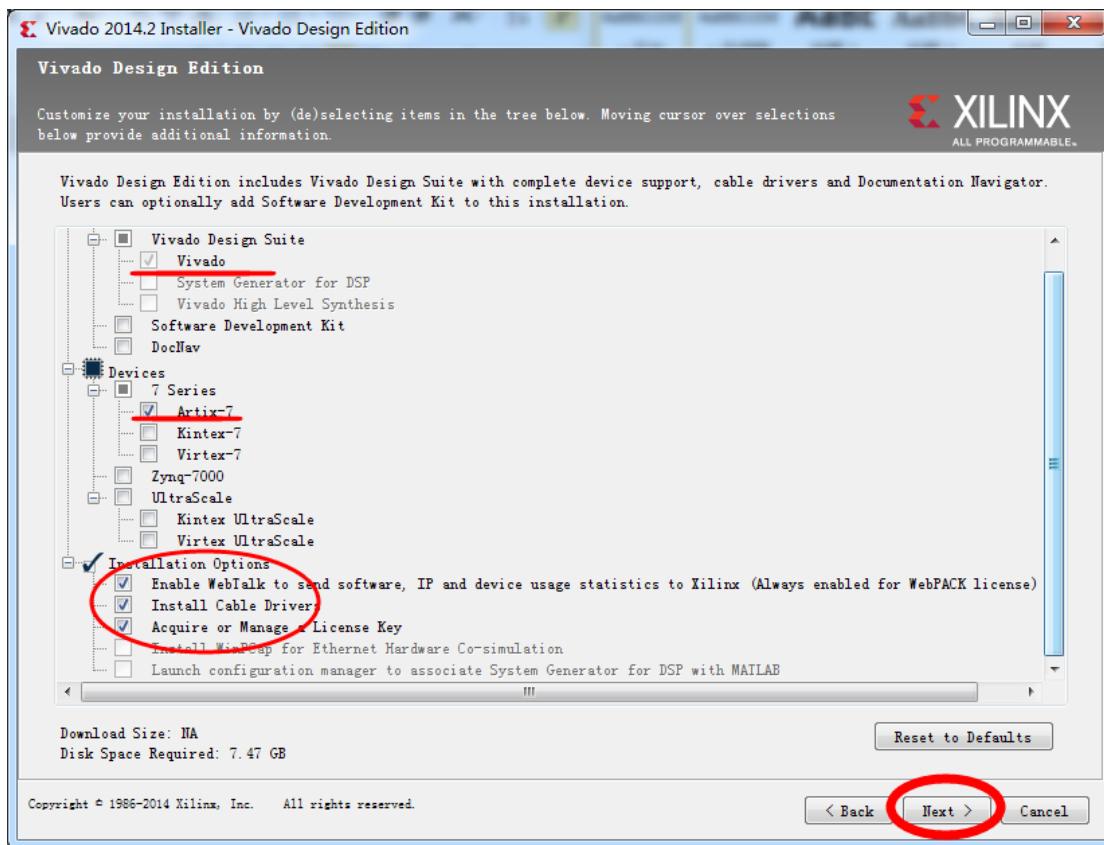
2、勾选所有 I Agree 选项。点击 Next。



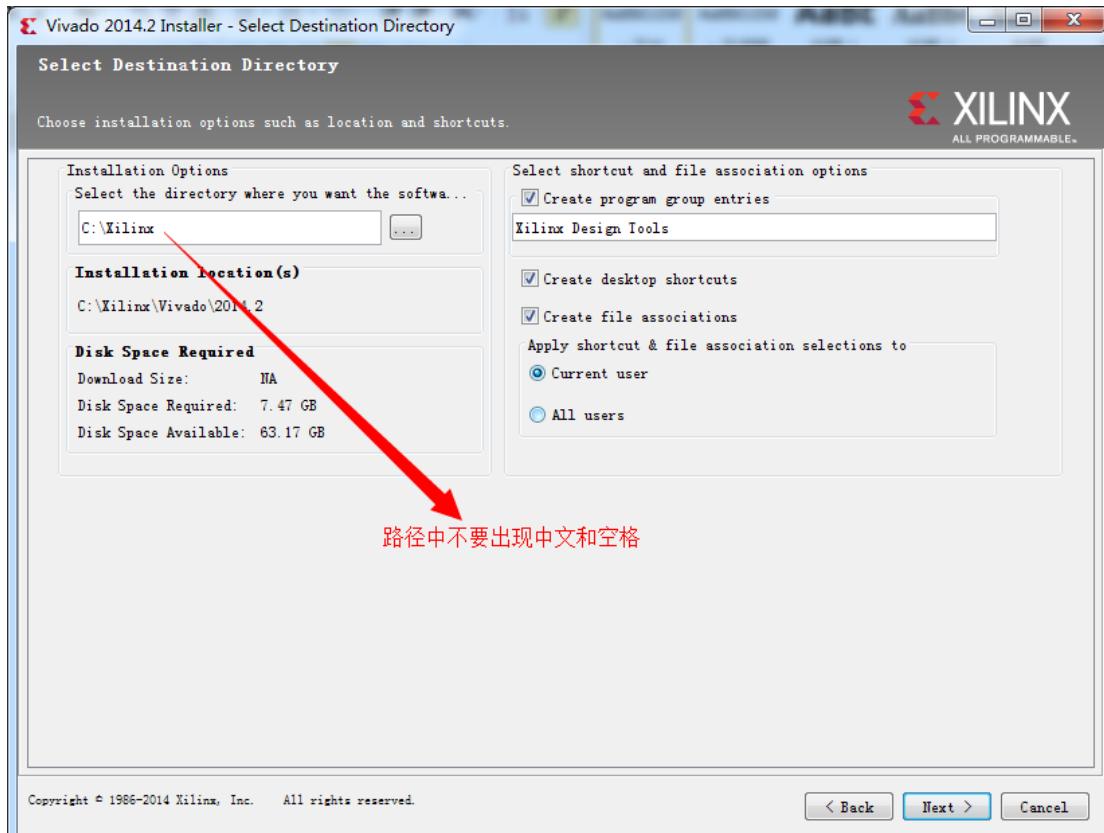
3、选择 Vivado 安装版本，对于初学者，可以选择第一项或第二项，第一项 Vivado Webpack 支持器件和功能受限，但不需要再行安装 License，第二项 Vivado Design Edition，安装完成后还需要安装 License 后才可使用。点击 Next。



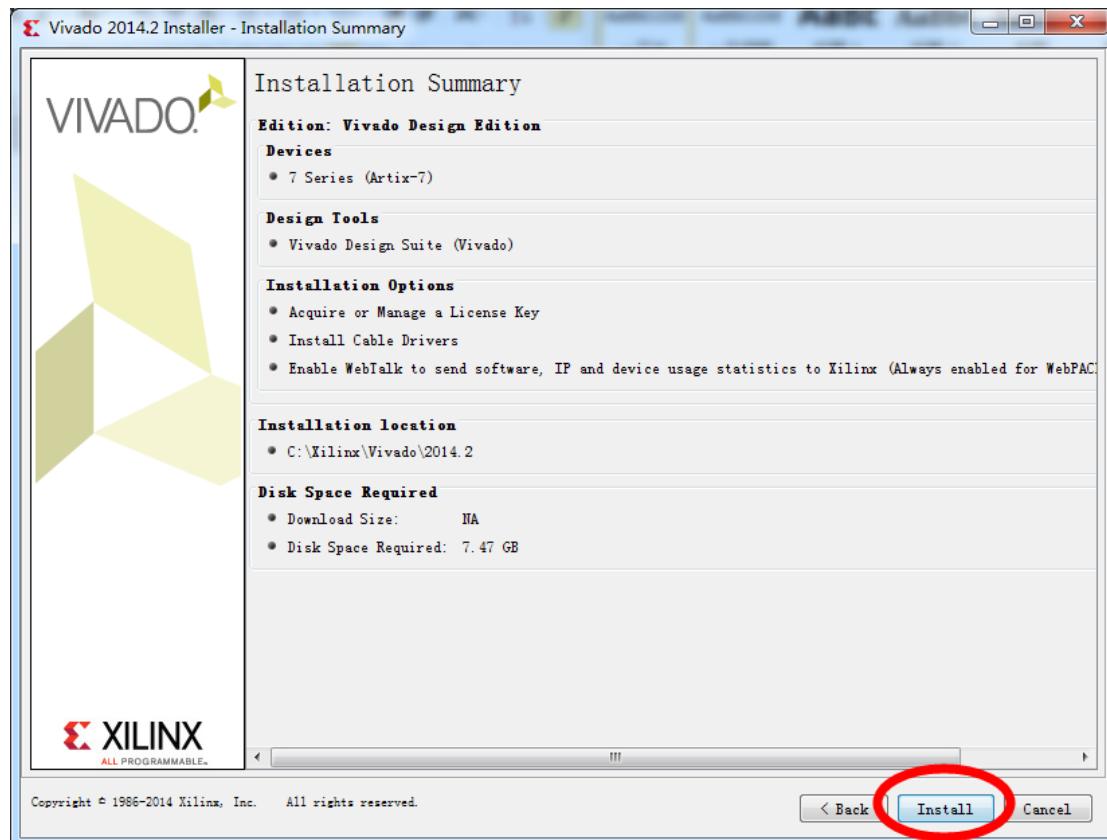
4、选择 Vivado 工具组件和器件库。对于初学者，如下几项是必须安装的。点击 Next。



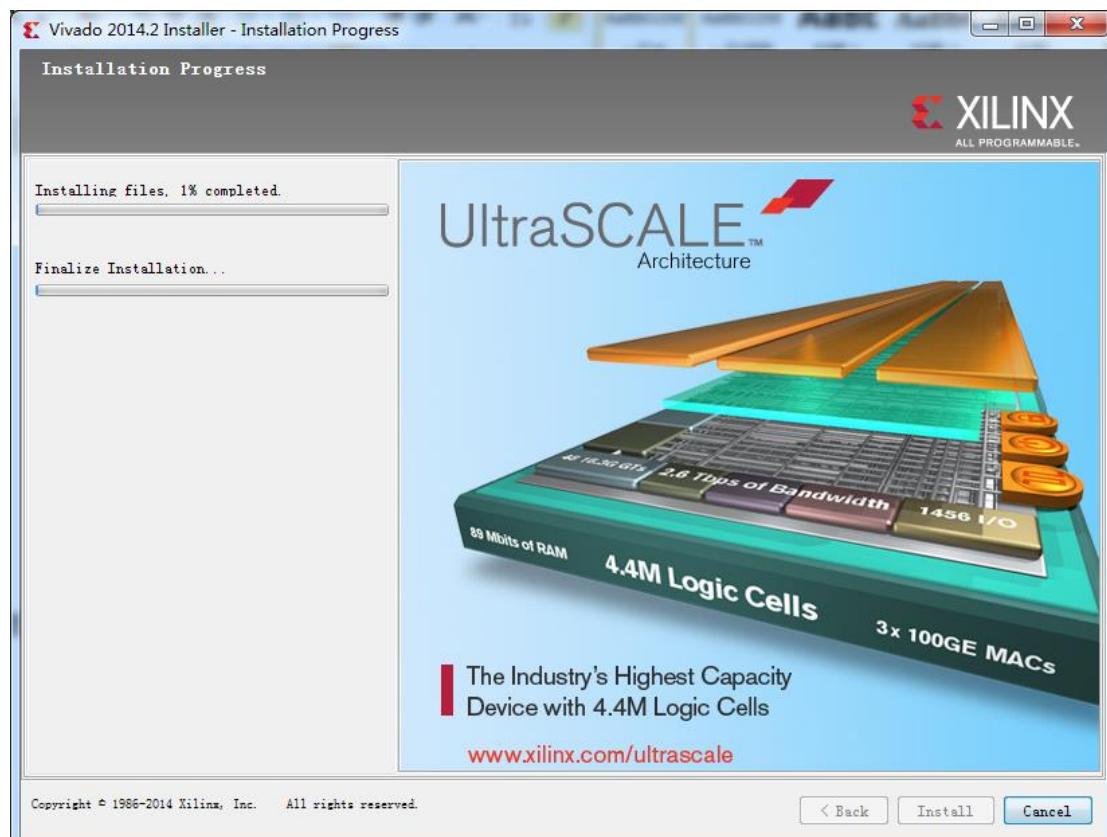
5、设置安装路径。系统默认安装路径为 C:\Xilinx (Linux 下为 /opt/Xilinx)，如需更改，点击浏览，但是安装路径中不要出现中文和空格。点击 Next。



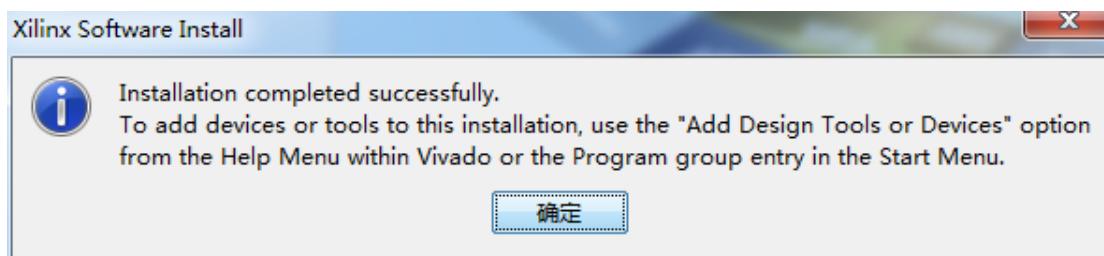
6、点击 Install 进入工具安装过程。



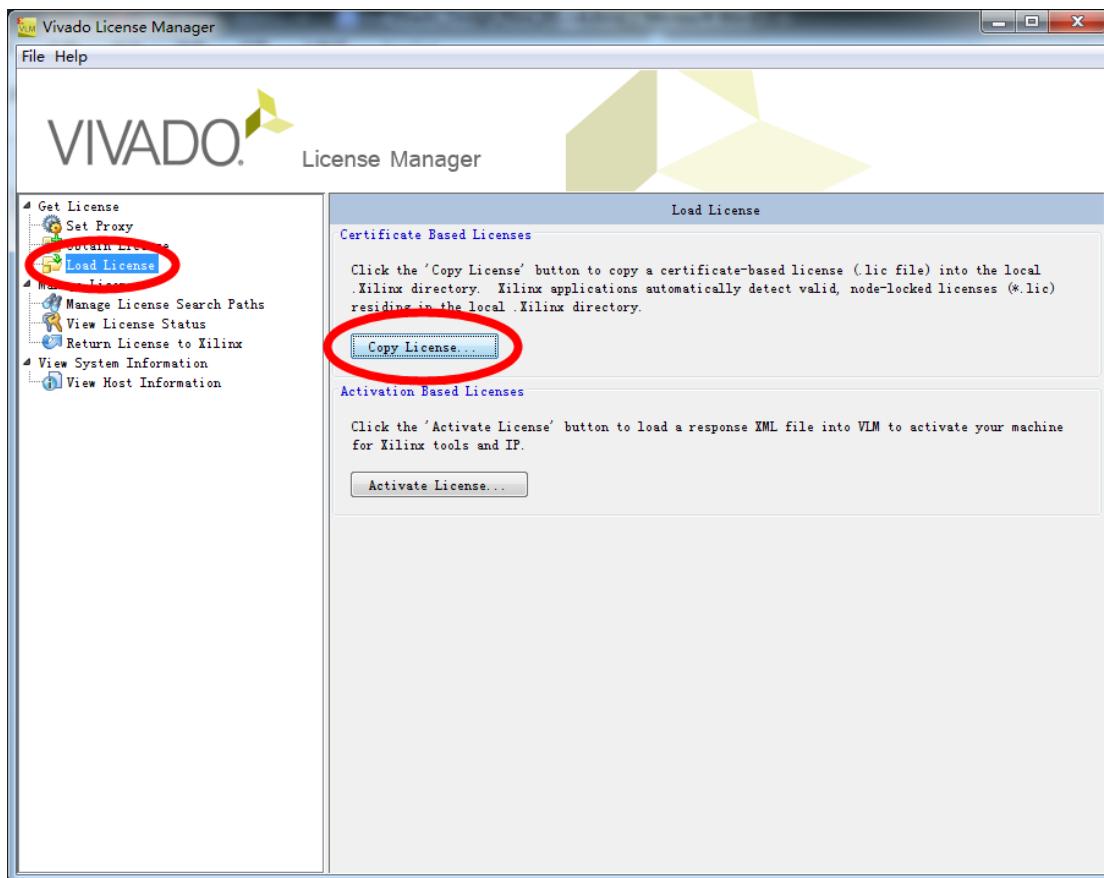
7、安装时间视 PC 性能而定，10min~30min 不等，**安装过程中，弹出的附属工具/软件一律允许安装。**



8、等待安装完成，弹出完成提示。点击“确定”。



9、弹出 License 管理软件，在左侧选择 Load License，并在右侧点击 Copy License 一项，选择到准备好的 license 文件。（Vivado Webpack 无此步）



10、Windows 系统下安装到此全部完成，Vivado 软件可正常使用。

Linux 下需要添加环境变量，打开 root 根目录的.bashrc 添加如下代码（建议不在 user 用户下添加，否者有些 linux 会有图形化界面冲突，桌面无法正常加载）。(或者每次使用 Vivado 软件的时候打开 terminal，先运行一下下面的命令)

```
source /opt/Xilinx/Vivado/2014.2/settings32.sh 或 source /opt/Xilinx/Vivado/2014.2/settings64.sh
```

依元素科技有限公司  
技术支持工程师  
全信 letter.tung@e-elements.com