



# Lab 1 MIPSfpga 基础



These materials produced in association with Imagination.  
Join our University community for more resources.

[community.imgtec.com/university](https://community.imgtec.com/university)

# Lab 1 MIPSfpga 基础

---

## 1. 概述

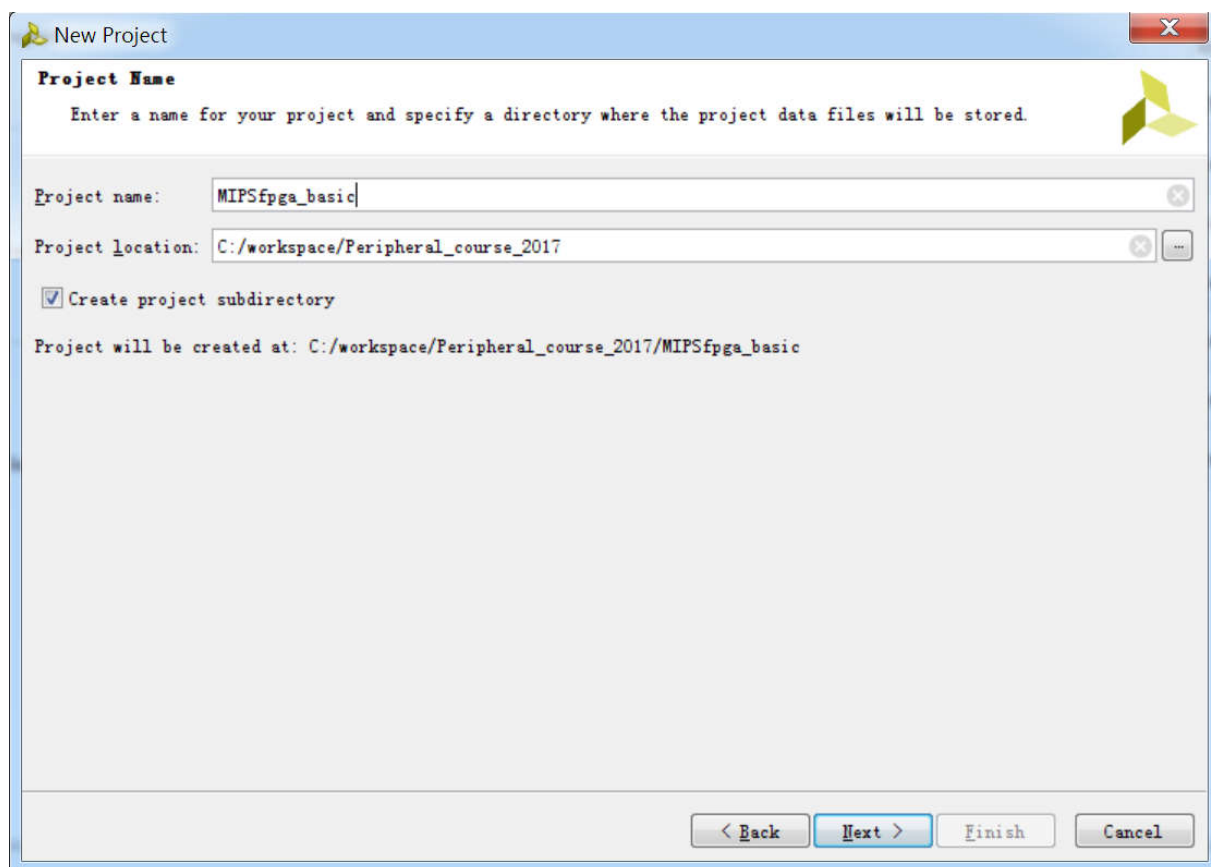
这是 MIPSfpga 系列实验中的第一个，在这个实验室里你将学会建立一个 Vivado 工程，该工程能够对 MIPSfpga 系统进行仿真、综合、实现、并下载到 Digilent Nexys4 DDR FPGA 开发板上进行实际验证。

本实验使用 Vivado 2015.2 作为开发工具，如果是 Vivado 更新的版本，也可以基本按照这个操作流程来进行，只是可能有些局部地方稍有不同。

## 2. MIPSfpga 处理器系统搭建

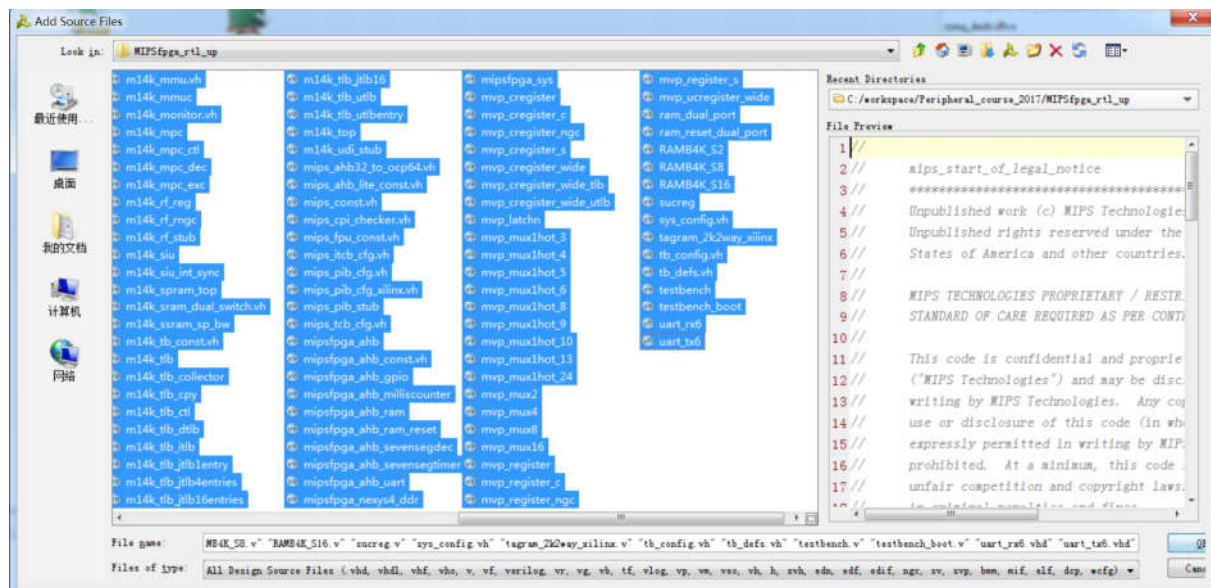
### 2.1 新建一个 Vivado 工程

打开 Vivado，新建一个工程，将其命名为 MIPSfpga\_basic。

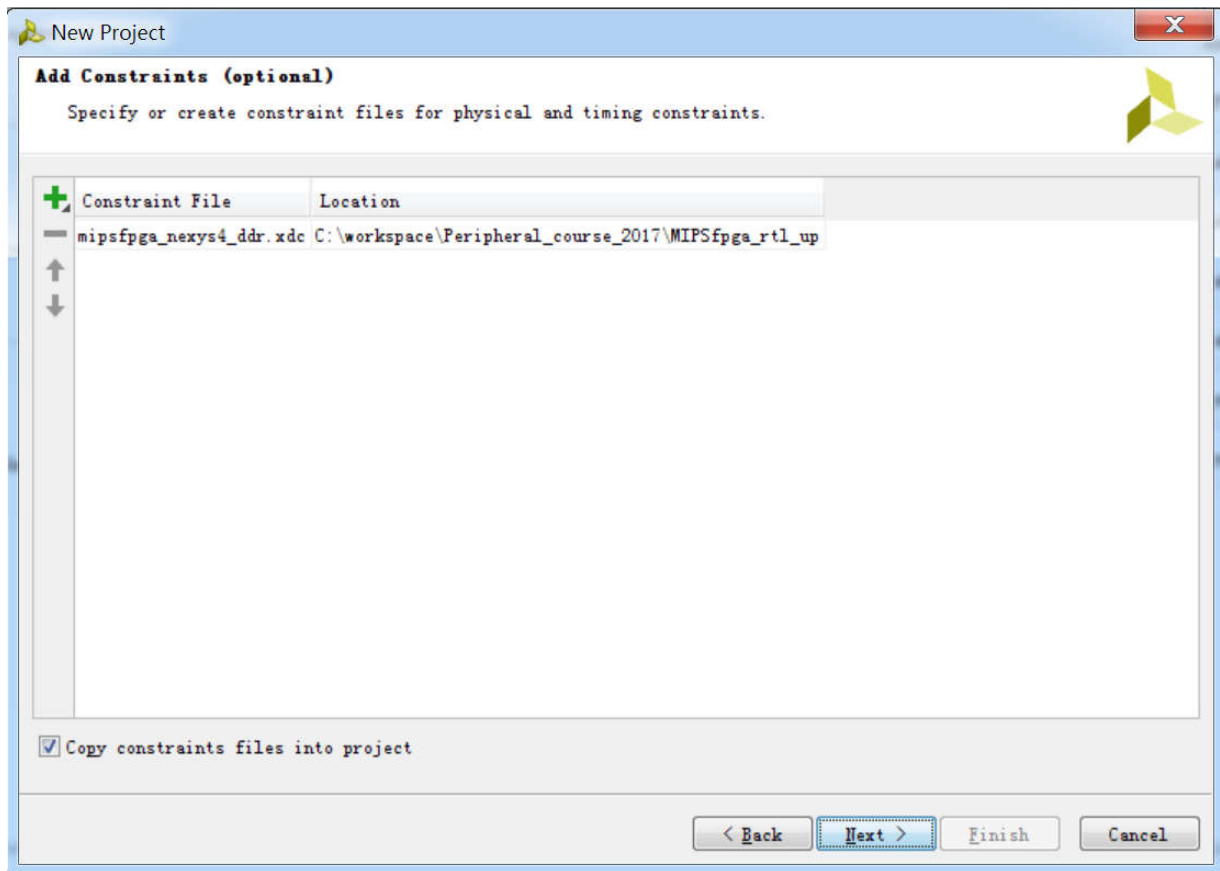


## 2.2 导入 MIPSfpga 设计文件

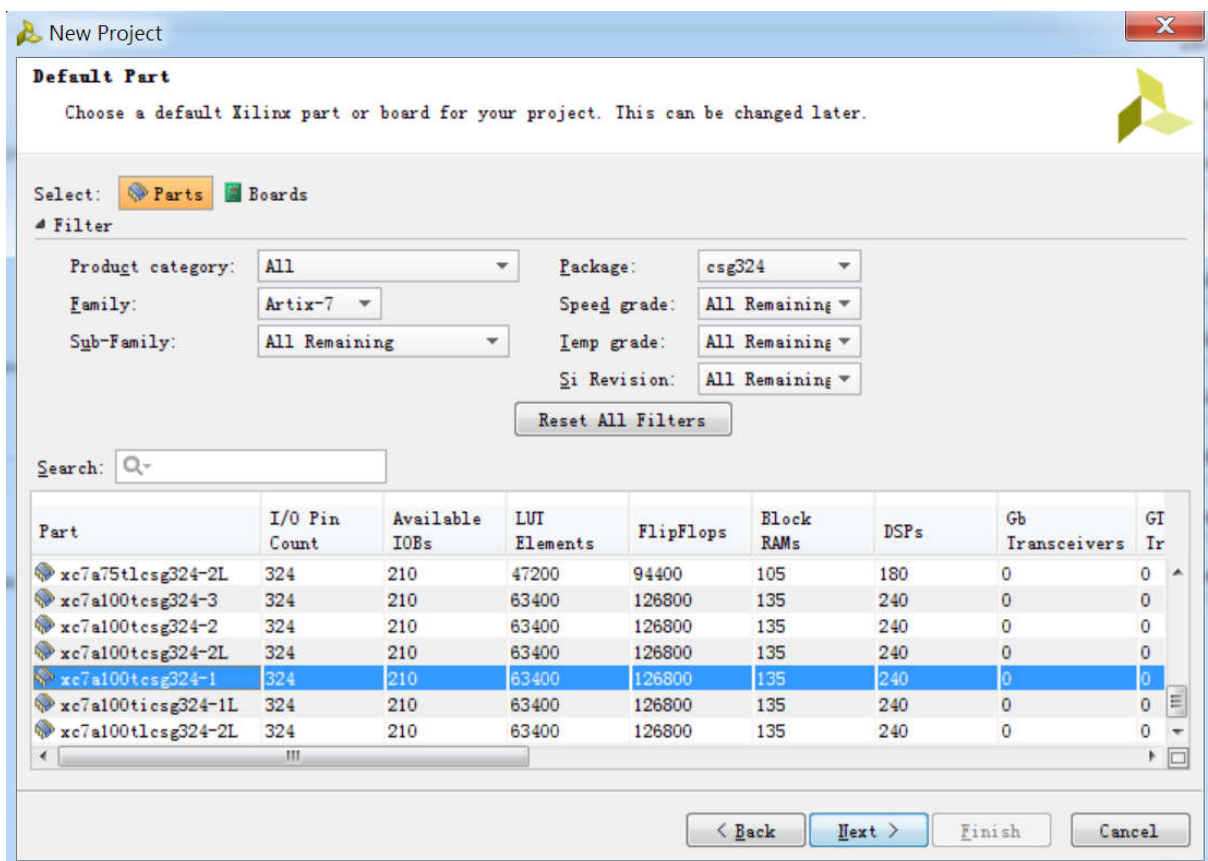
找到 MIPSfpga\_rtl\_up 目录，选择将目录中的全部设计文件加入工程。



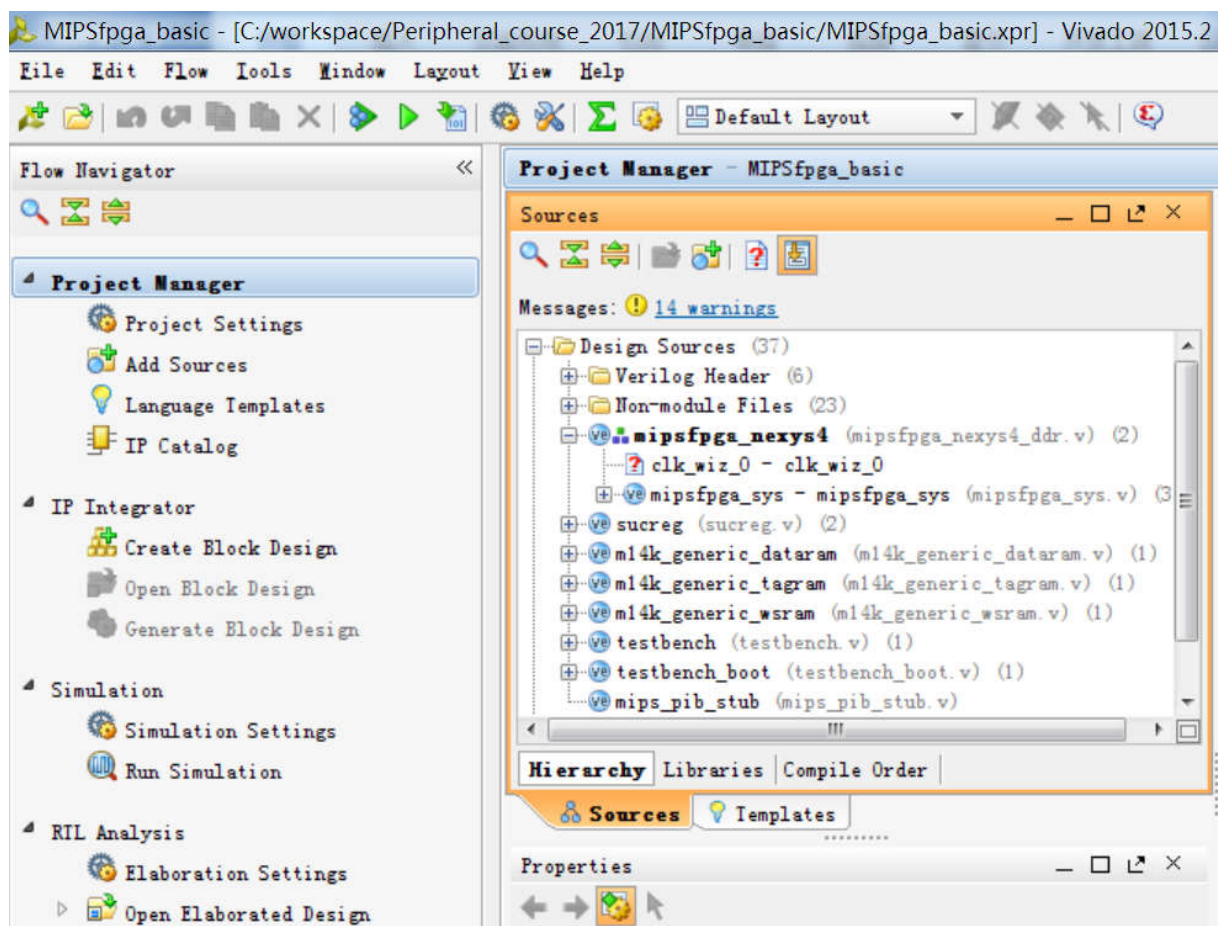
同样在 MIPSfpga\_rtl\_up 目录中找到约束文件（后缀为.xdc），将其加入工程。



找到 Nexys4 DDR 开发板对应的 FPGA 型号，将其添加进工程。

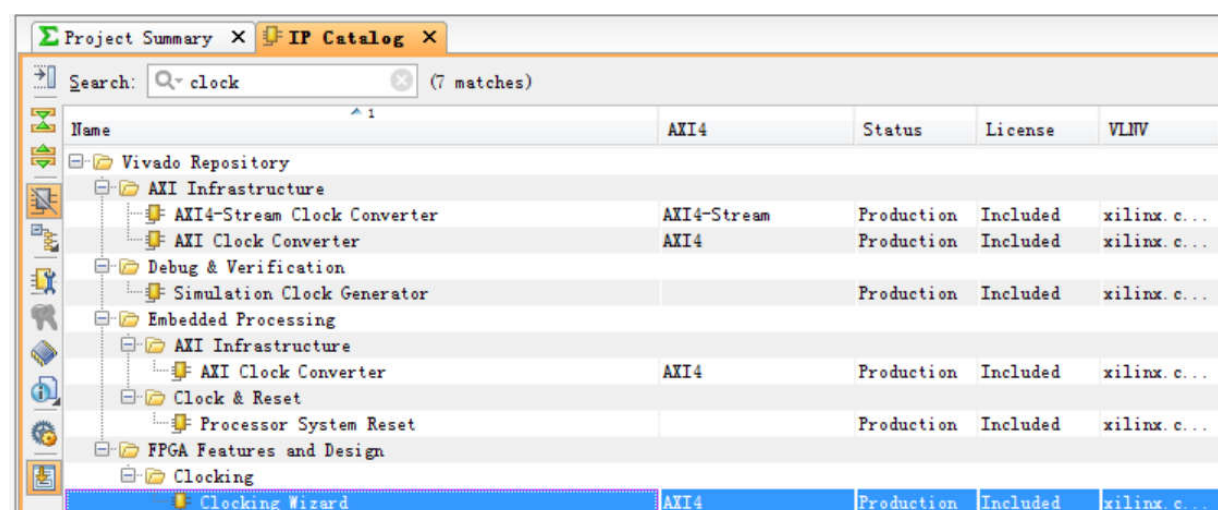


完成新建工程，结果如下所示。

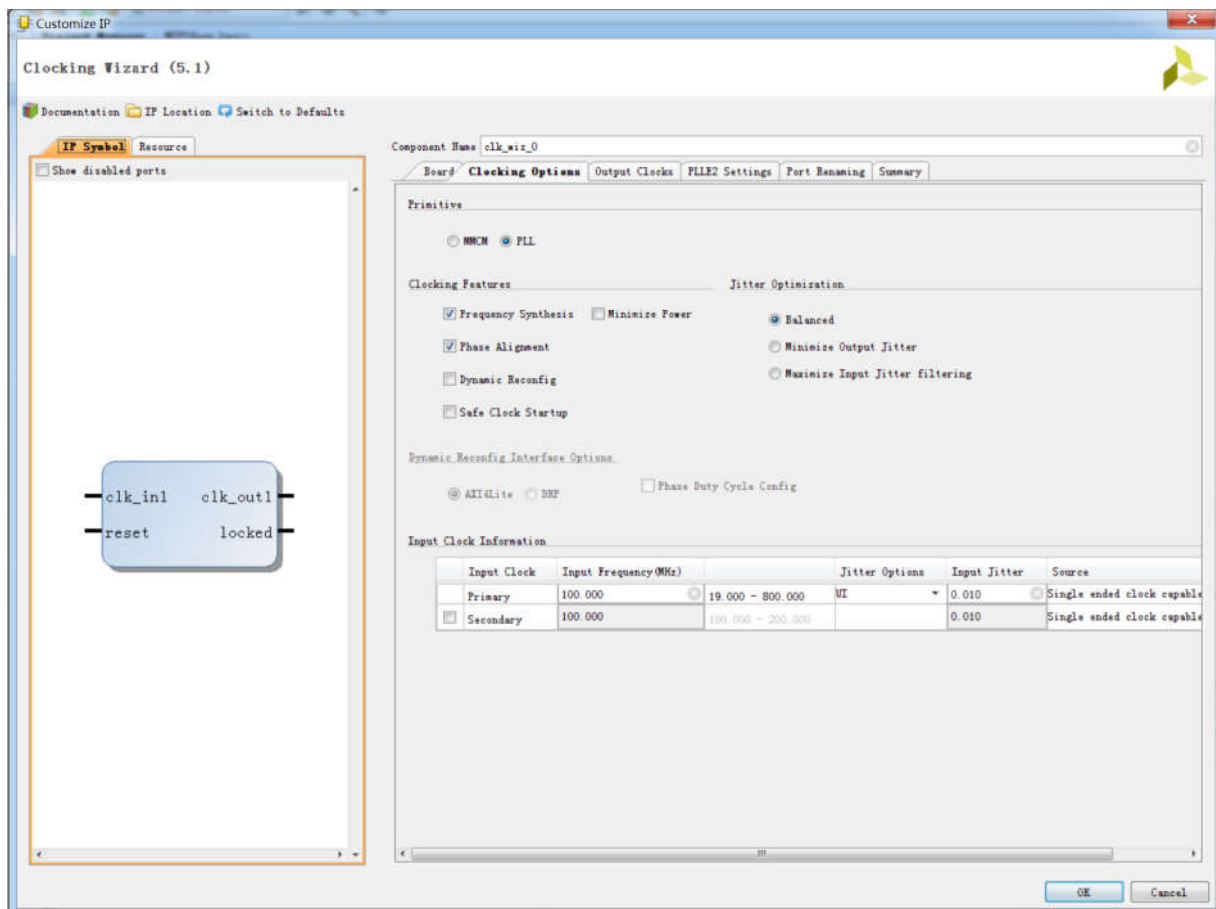


## 2.3 添加时钟模块

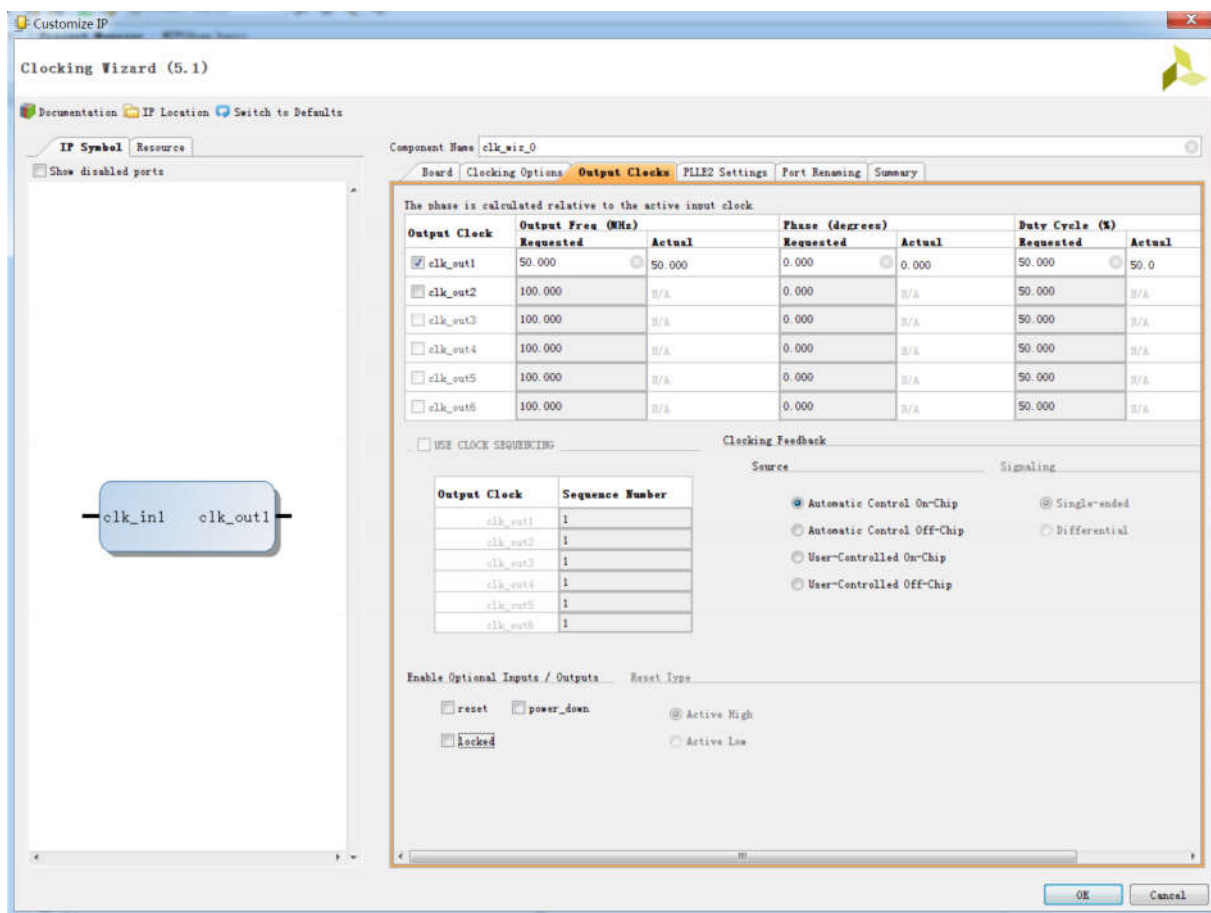
打开 IP Catalog, 找到 Clocking Wizard 模块, 双击添加。



在时钟选项中，选择 PLL 方式。

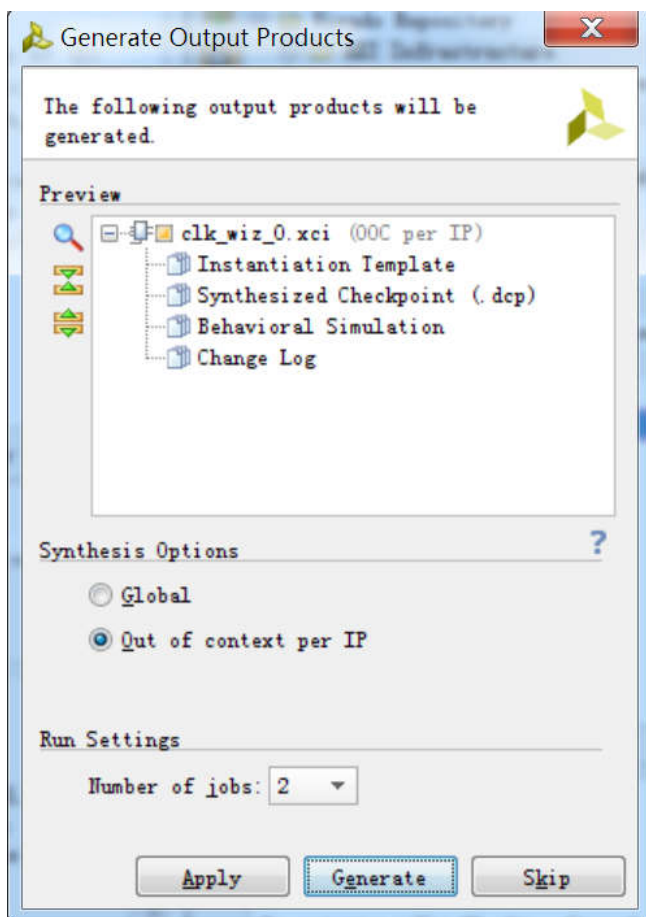


将输出时钟的频率设置为 50MHz（MIPSfpga 处理器在 Nexys4 DDR 开发板上最高可以跑到 62MHz，为了保险起见，同时为了方便以后外设模块的实现，这里只选择 50MHz）。同时取消 reset 和 locked 信号的选项。

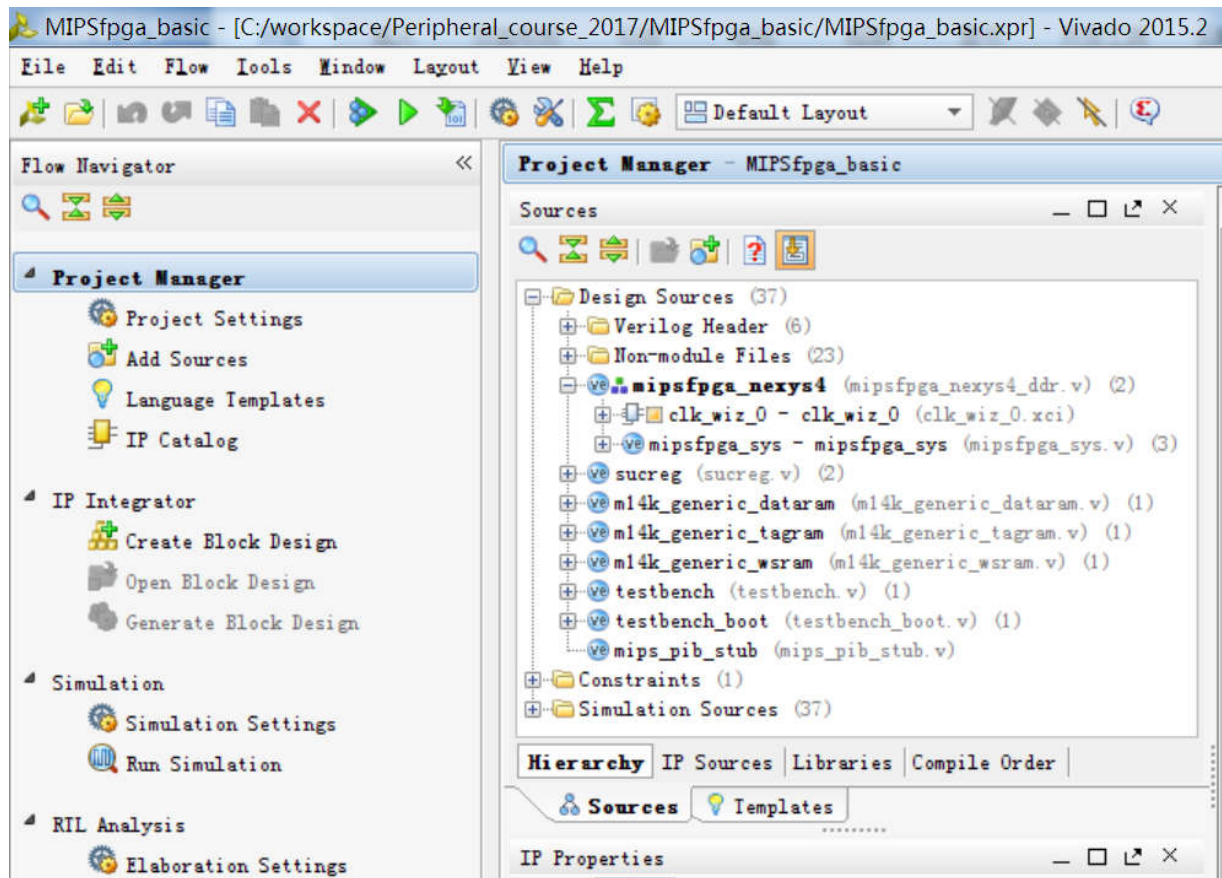


时钟模块成功添加后，会弹出一个“Generate Output Products”对话框，选择“Generate”。



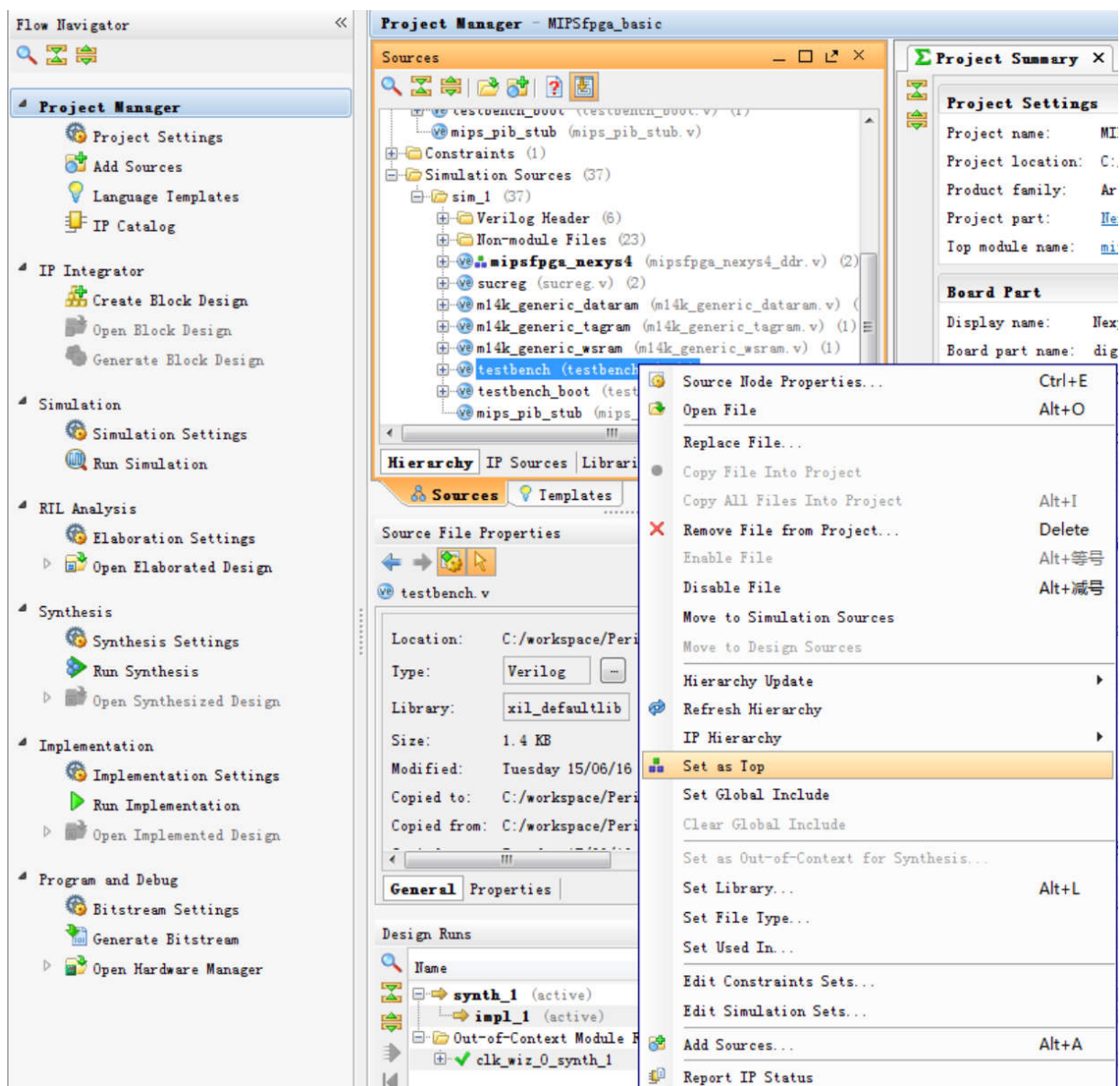


生成完成后，clk\_wiz\_0 模块前的“？”消失。

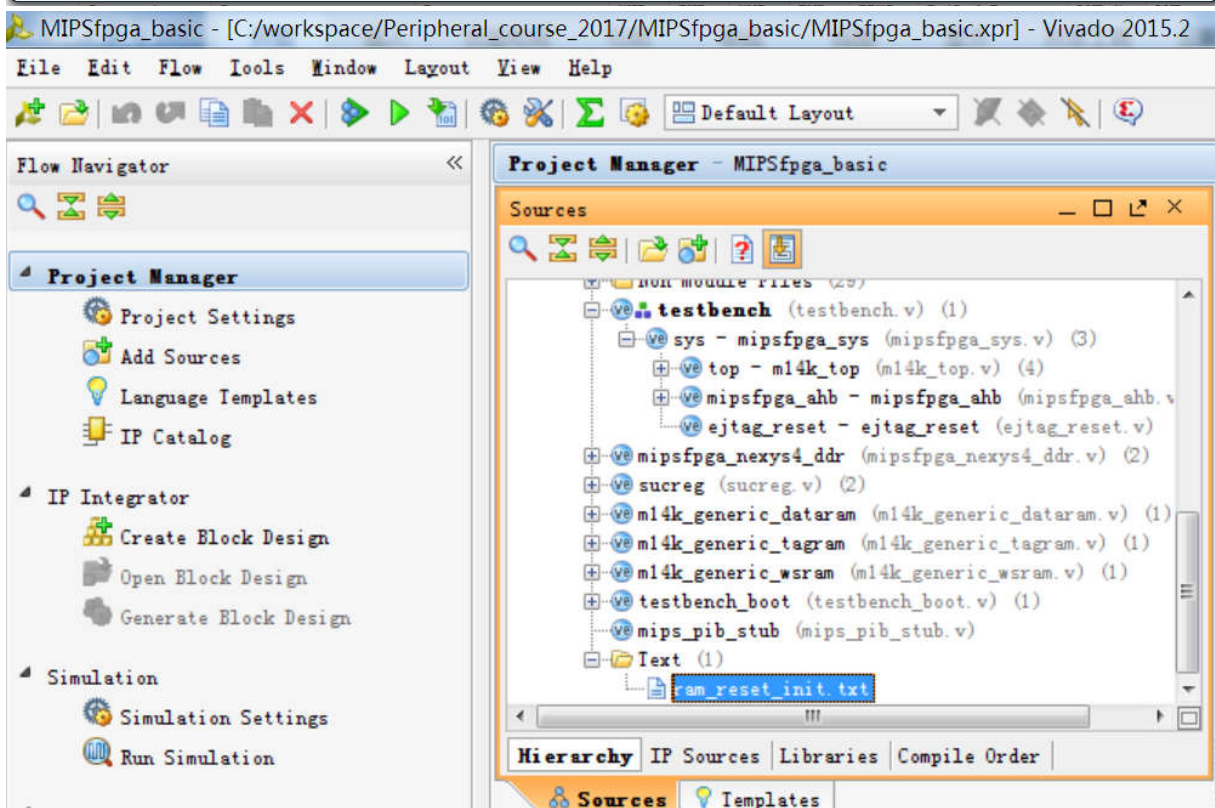
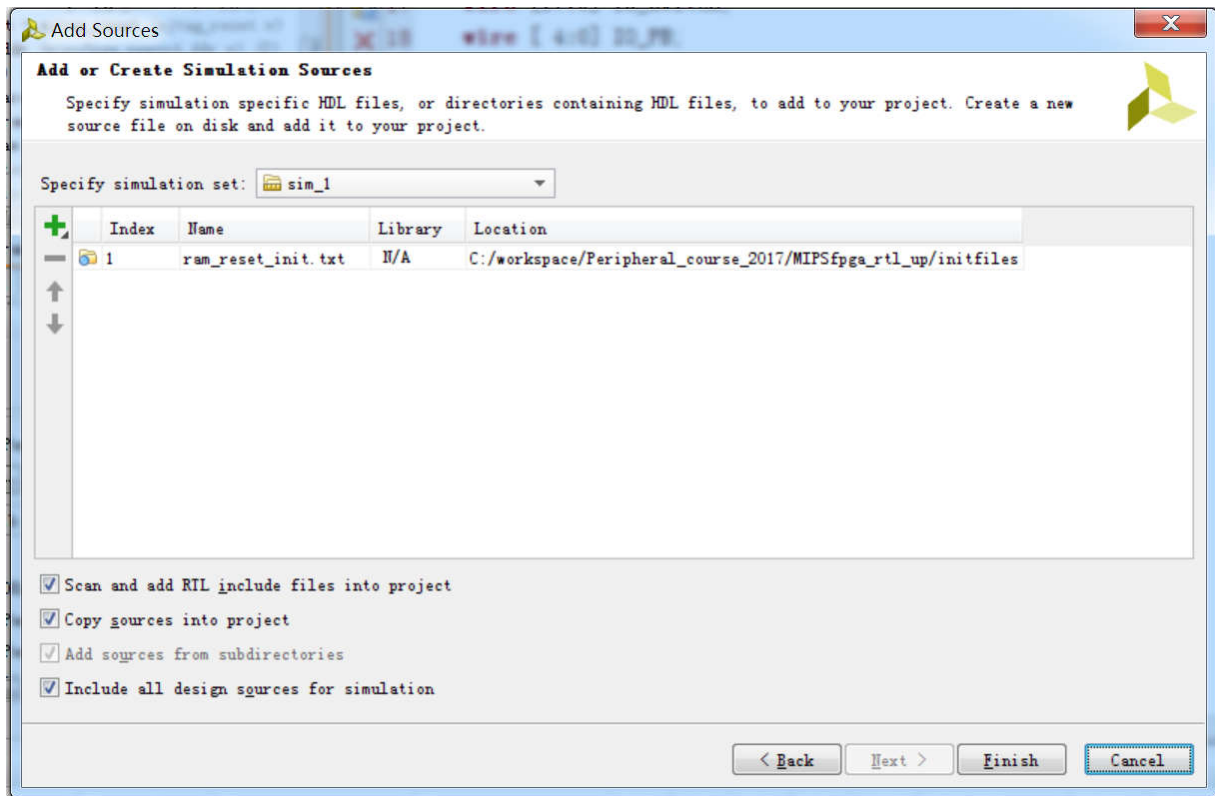


## 2.4 对 MIPSfpga 进行仿真

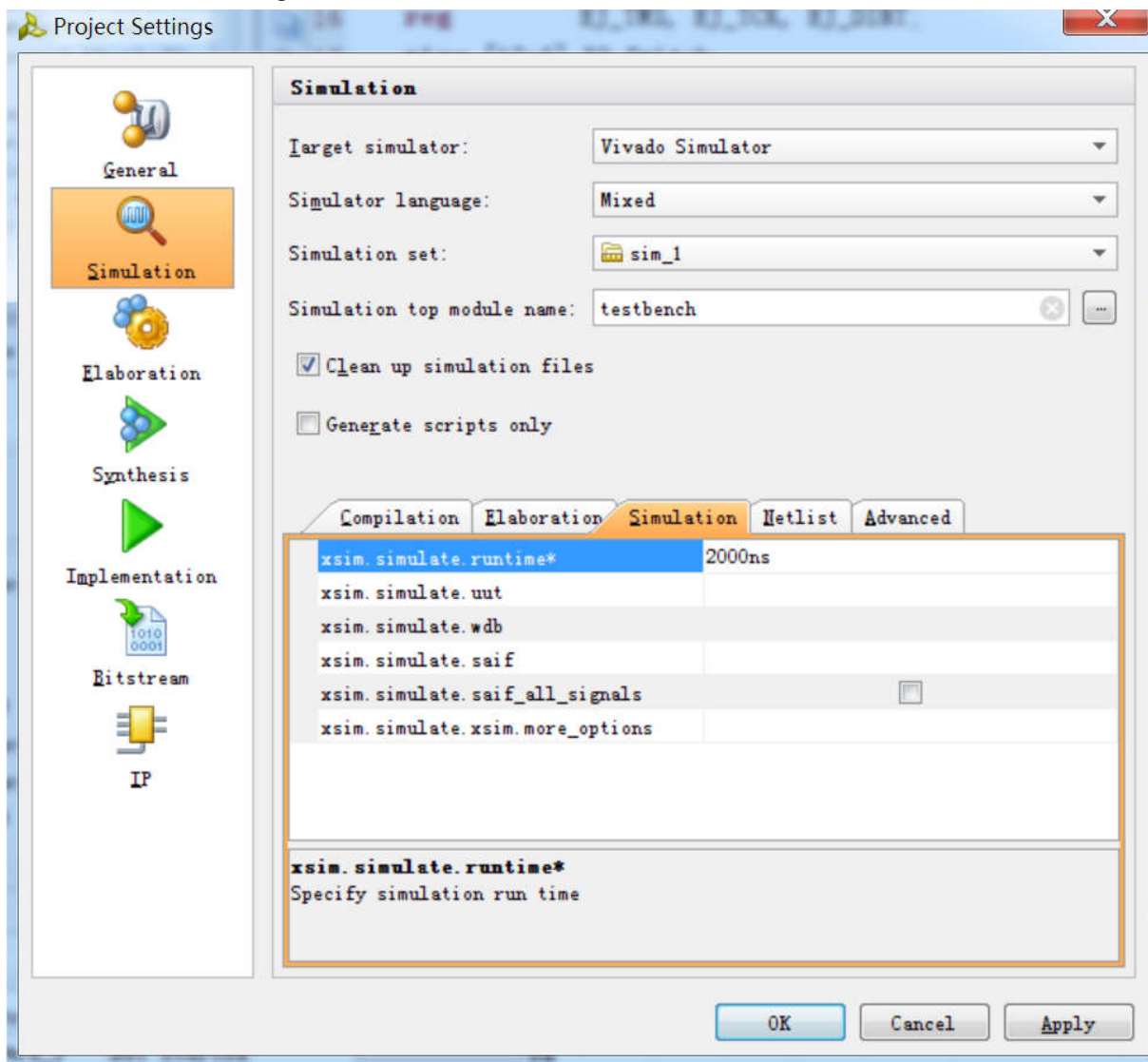
找到 testbench.v 文件，将其设置为顶层文件。



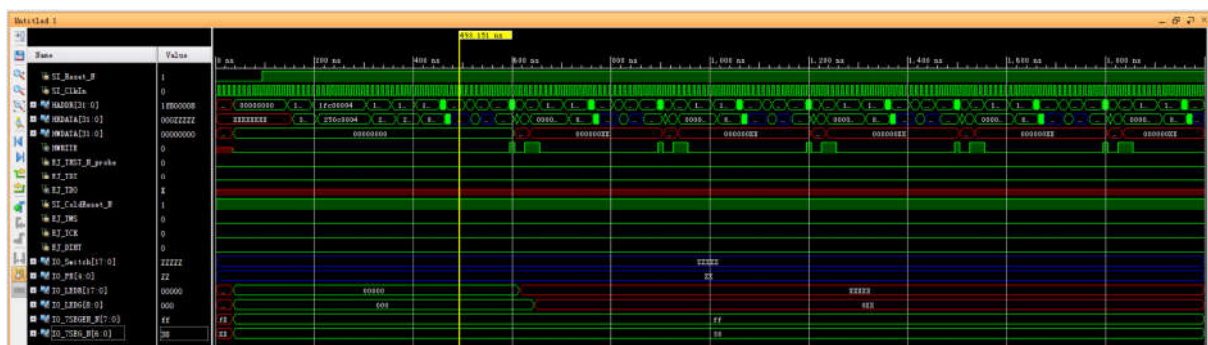
选择 *Add Sources*，通过 *Add or create simulation sources* 选项将 MIPSfpga\_rtl\_up/initfiles 目录下的 ram\_reset\_init.txt 文件添加进工程。



选择 *Simulation Settings* 将仿真时长设置为 2000ns。

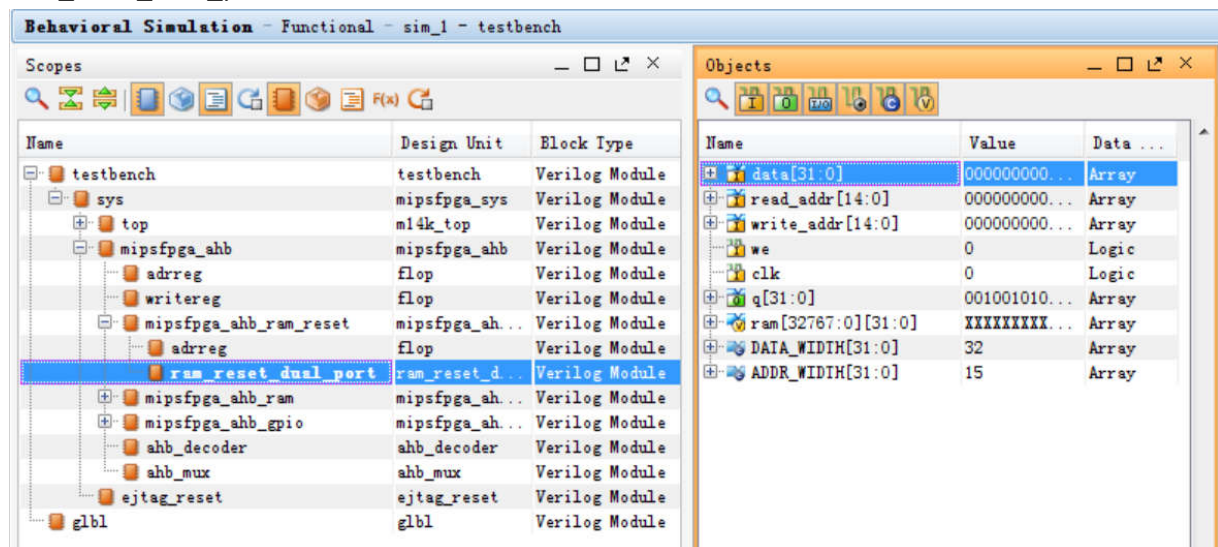


点击 *Run Simulation > Run behavioral simulation* 进行行为仿真，并观察仿真波形。

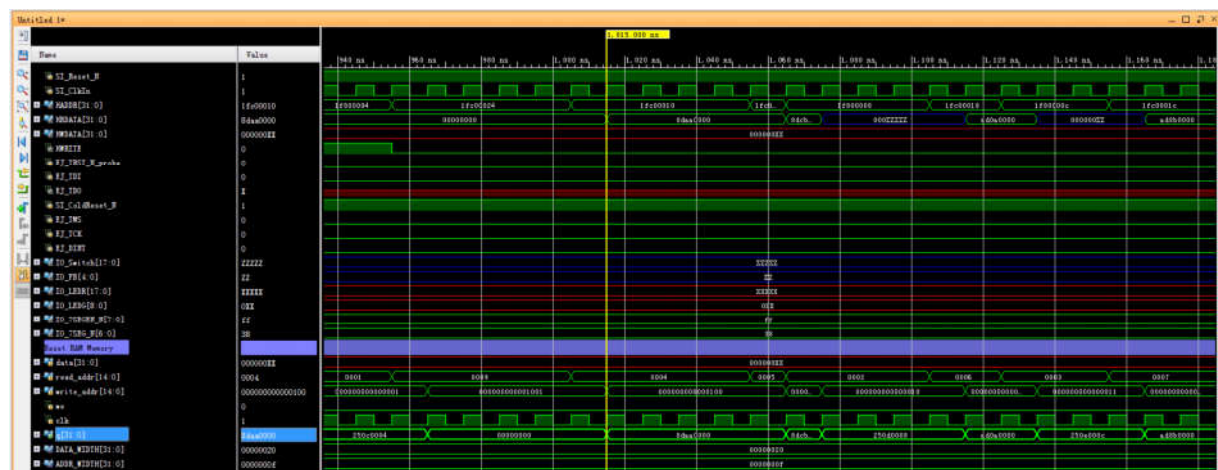




展开 *testbench*，找到 *testbench* → *sys* → *mipsfpga\_ahb* → *mipsfpga\_ahb\_ram\_rese*，将 *ram\_reset\_dual\_port* 加入仿真。

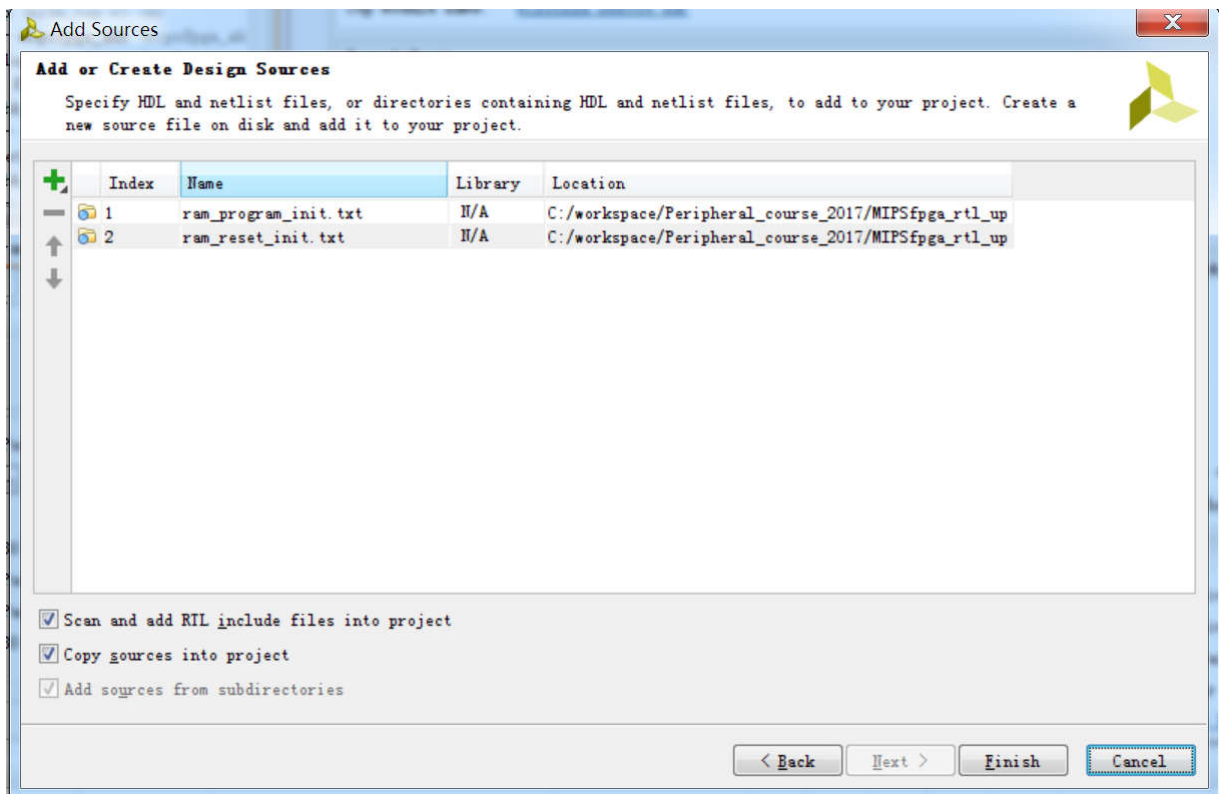
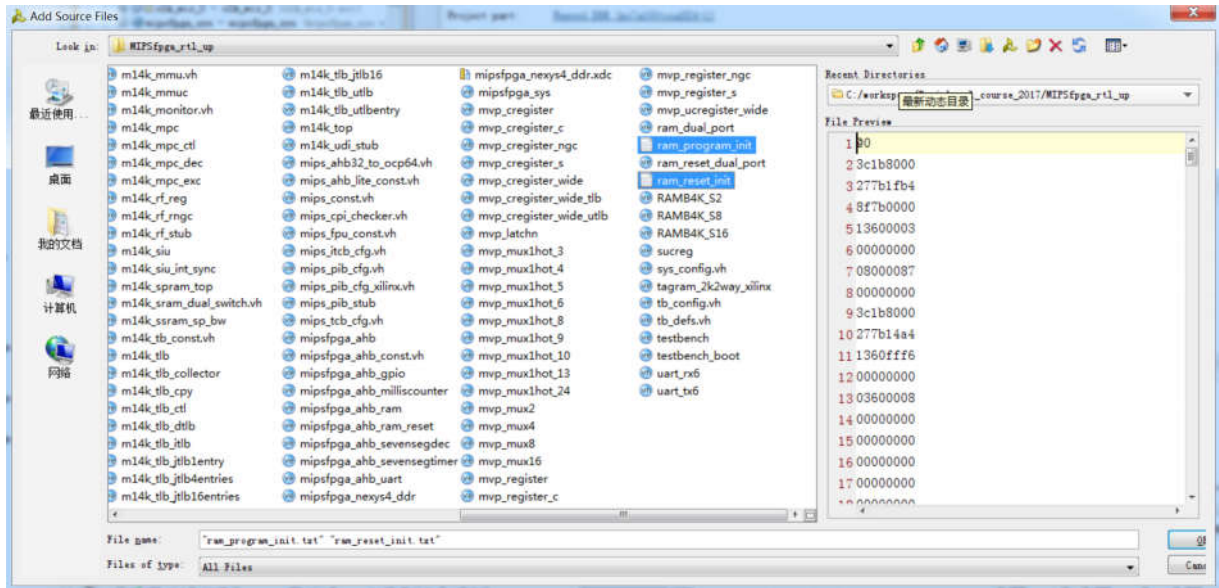


在波形窗口中点击鼠标右键，选择 *New Divider*，输入 **Reset RAM Memory** 并点击 **OK**。然后将 *ram\_reset\_dual\_port* 模块中的全部信号加入波形窗口。再将仿真时间修改为 **2us**，并点击 **Restart** 按钮，然后选择 **Run**。

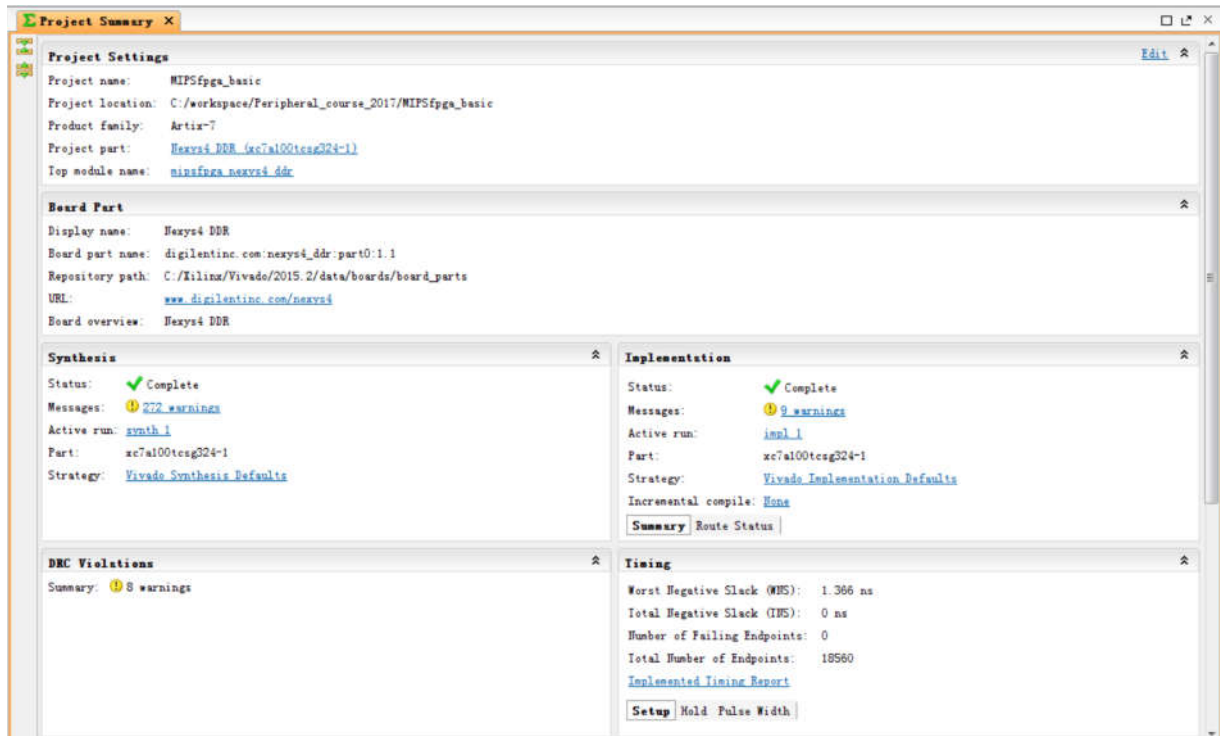


## 2.5 对 MIPSfpga 进行综合和实现

将 MIPSfpga\_rtl\_up/initfiles 目录下的 *ram\_reset\_init.txt* 和 *ram\_program\_init.txt* 文件复制到 MIPSfpga\_rtl\_up 目录中。选择 *Add Sources*，通过 *Add or create design sources* 选项将 MIPSfpga\_rtl\_up 目录下的 *ram\_reset\_init.txt* 和 *ram\_program\_init.txt* 文件添加进工程。



点击 *Generate Bitstream* 按键，生成 bitstream 文件。Mipsfpga 综合实现后观察时序能否满足 CPU 运行时钟的频率要求。



## 2.6 将 MIPSfpga 下载到 Nexys4 DDR FPGA 开发板运行

按照下图将 FPGA 下载线连接到 Nexys4 DDR 开发板的 USB Programmer Port；打开 Power Switch 电源开关；下载 bitstream 文件。然后，按 Processor Reset 按钮，观察 MIPSfpga 运行 ram\_reset\_init.txt 文件中的程序。



