

# Lab1 设计一个FIR滤波器分离鸟类声音

## 目标

在完成本实验后，您应当学会：

- 如何使用Vitis HLS构建一个项目
- 在Vitis HLS中进行仿真、综合与IP导出

简而言之，您将掌握使用HLS进行加速核设计与部署到PYNQ的基本流程。出于篇幅限制，本实验仅介绍基本工具操作流程。

## 环境要求

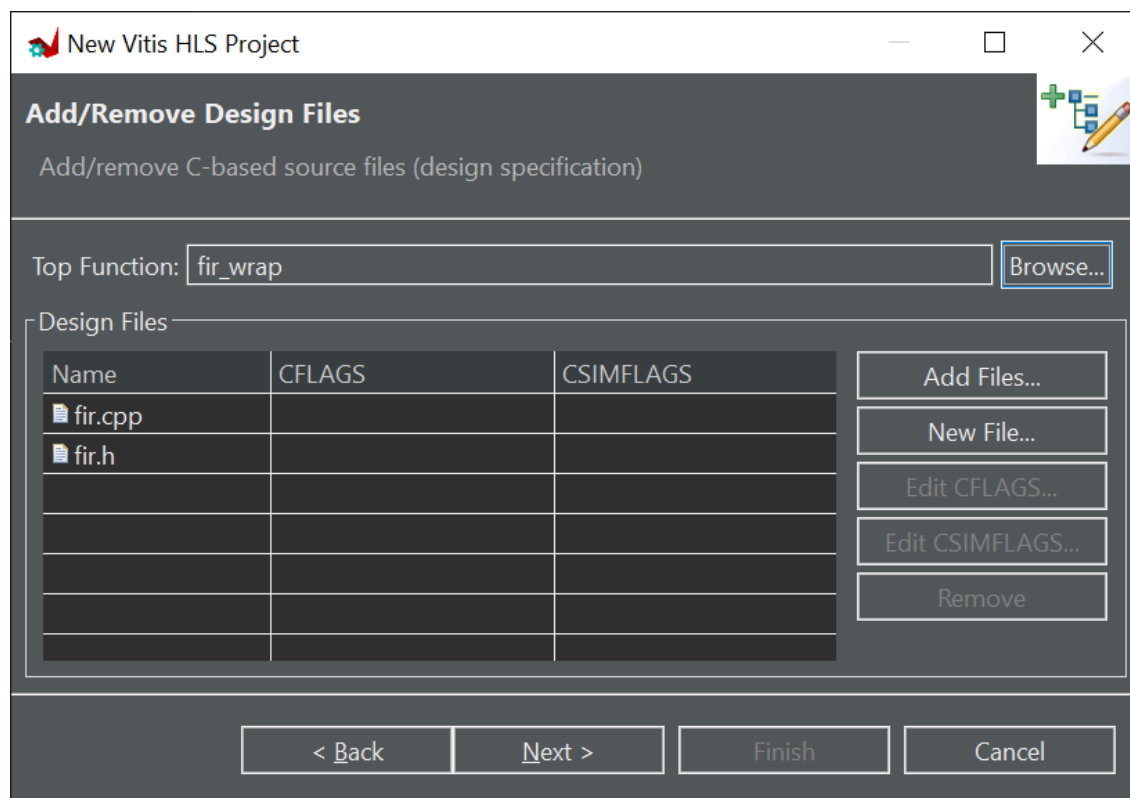
- PYNQ-Z2远程实验室服务或物理板卡
- Vitis HLS
- Vivado

## 实验步骤（Windows版本的HLS）

### 1. 在Vitis\_HLS中设计FIR

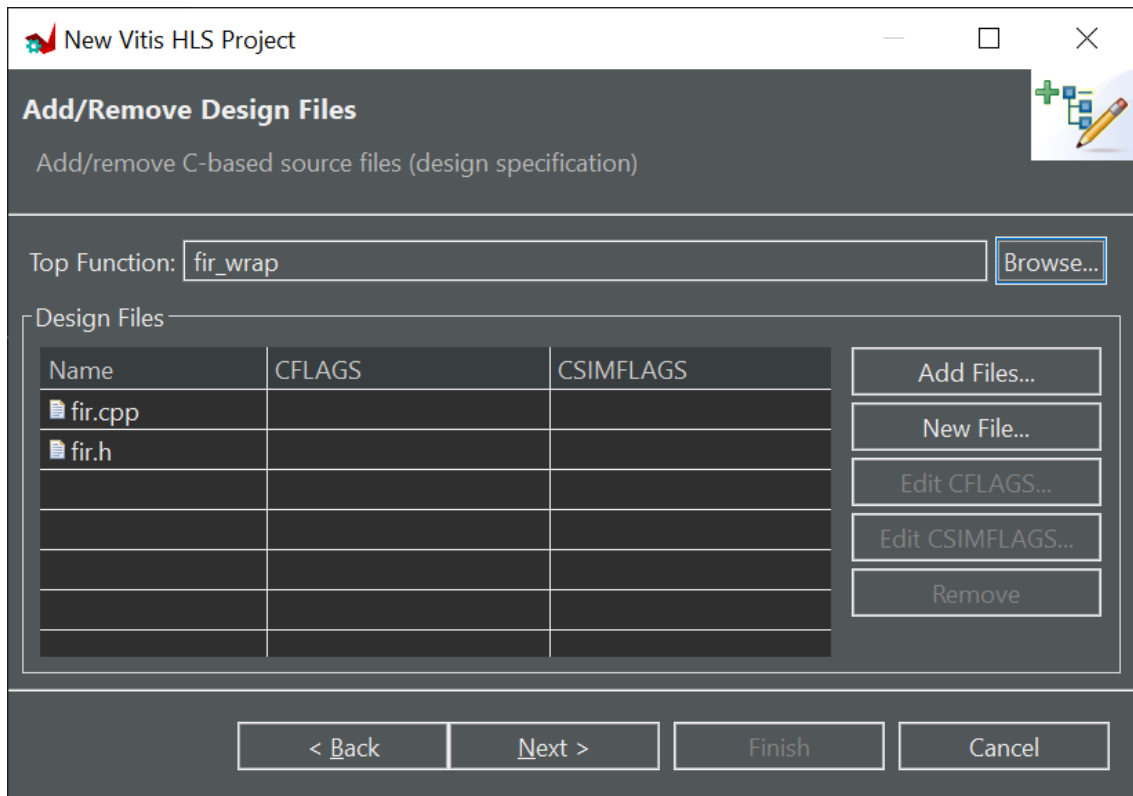
#### 1.1 创建一个新项目

1. 打开Vitis HLS软件，点击**Create Project**，创建一个新的项目
2. 在**Project name**输入项目名**fir\_hls\_prj**，点击**Browse**选择一个合适的目录位置，点击**Next**



3. 点击**Add Files...**，将src目录下的**fir.h**和**fir.cpp**添加到项目中

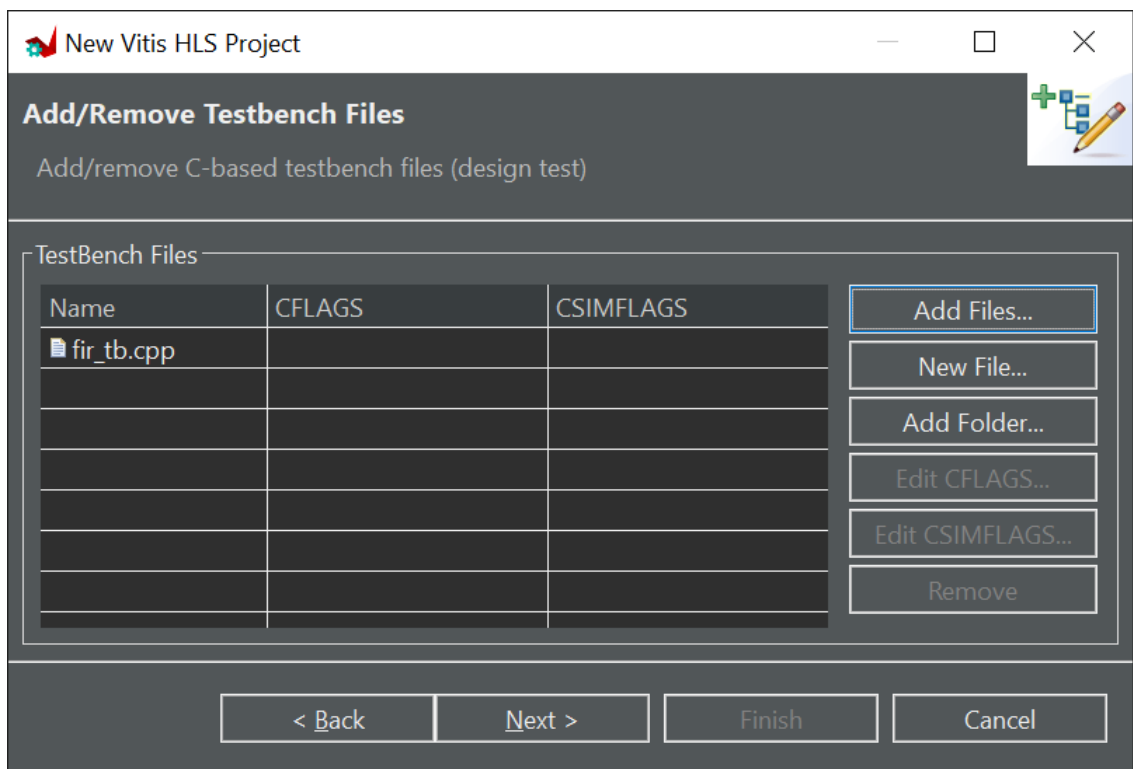
4. 点击**Top Function**栏中的**Browse**按键，选择**fir\_wrap**，这是我们进行综合时候的顶层函数，点击**Next**



The dialog box is titled "New Vitis HLS Project" and "Add/Remove Design Files". It contains a text field for "Top Function" with the value "fir\_wrap" and a "Browse..." button. Below this is a table for "Design Files" with columns "Name", "CFLAGS", and "CSIMFLAGS". The table lists "fir.cpp" and "fir.h". To the right of the table are buttons: "Add Files...", "New File...", "Edit CFLAGS...", "Edit CSIMFLAGS...", and "Remove". At the bottom are navigation buttons: "< Back", "Next >", "Finish", and "Cancel".

Name	CFLAGS	CSIMFLAGS
fir.cpp		
fir.h		

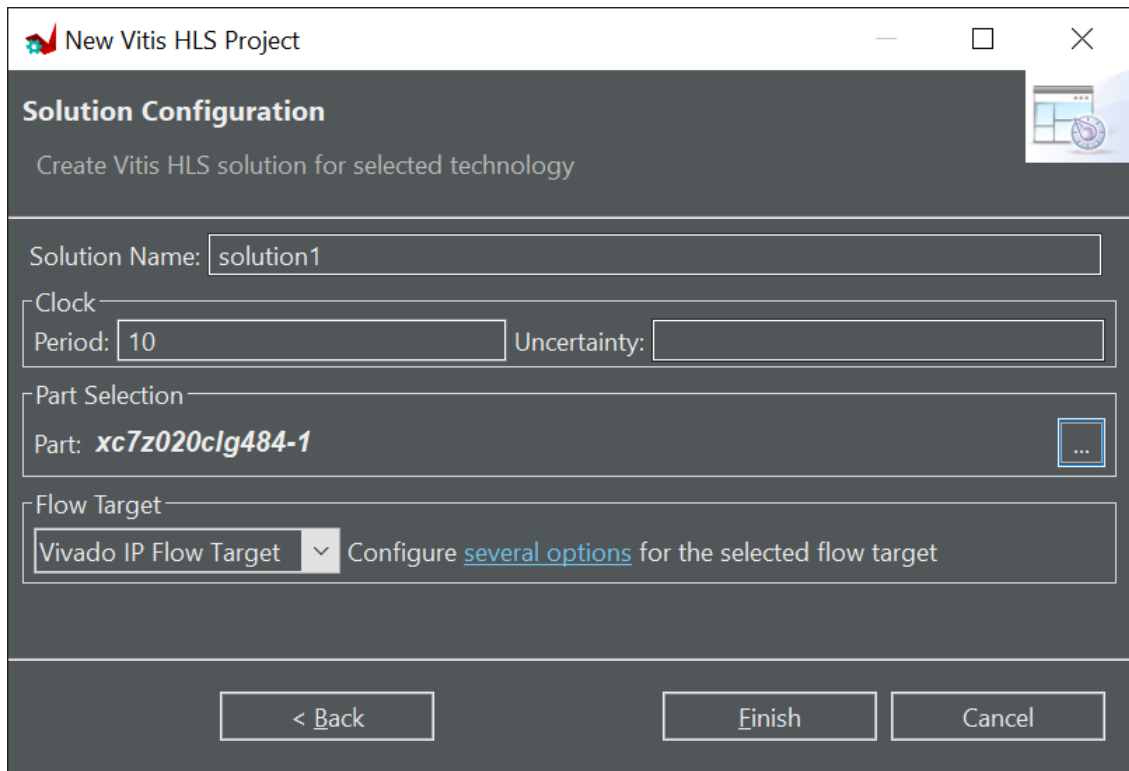
5. 点击**Add Files...**，将**src**目录下的**fir\_tb.cpp**添加到项目中，点击**Next**



The dialog box is titled "New Vitis HLS Project" and "Add/Remove Testbench Files". It contains a table for "TestBench Files" with columns "Name", "CFLAGS", and "CSIMFLAGS". The table lists "fir\_tb.cpp". To the right of the table are buttons: "Add Files..." (highlighted with a blue border), "New File...", "Add Folder...", "Edit CFLAGS...", "Edit CSIMFLAGS...", and "Remove". At the bottom are navigation buttons: "< Back", "Next >", "Finish", and "Cancel".

Name	CFLAGS	CSIMFLAGS
fir_tb.cpp		

6. 下面进入到**Solution Configuration**界面，保持其他选项不变，在**Part Selection**栏最右侧点击... 字样的按钮，在**Search**栏的搜索框中输入**xc7z020clg484-1**，即PYNQ-Z2板卡所使用的器件型号

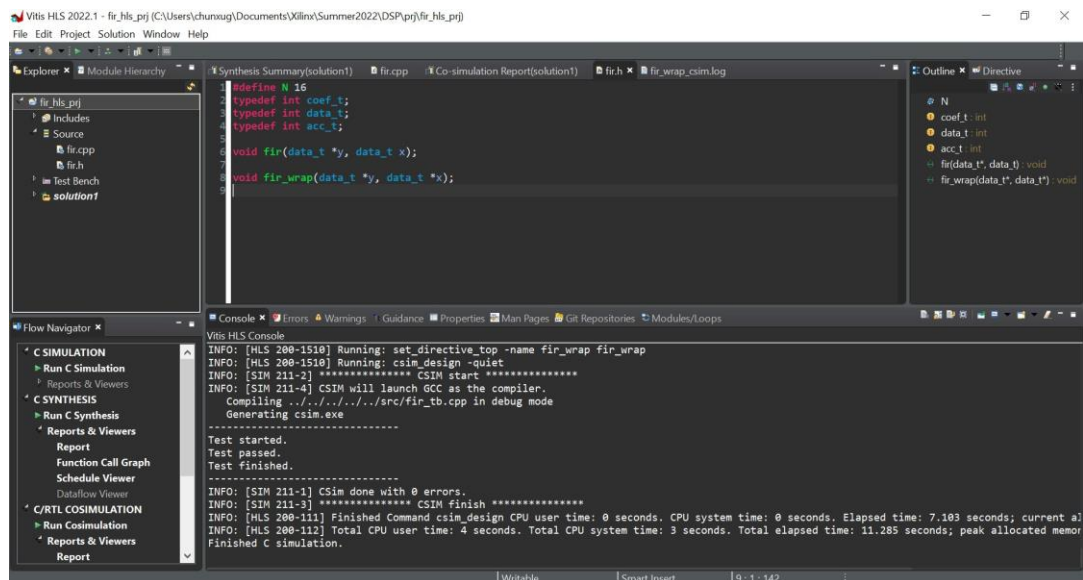


7. 点击**Finish**，完成项目的创建

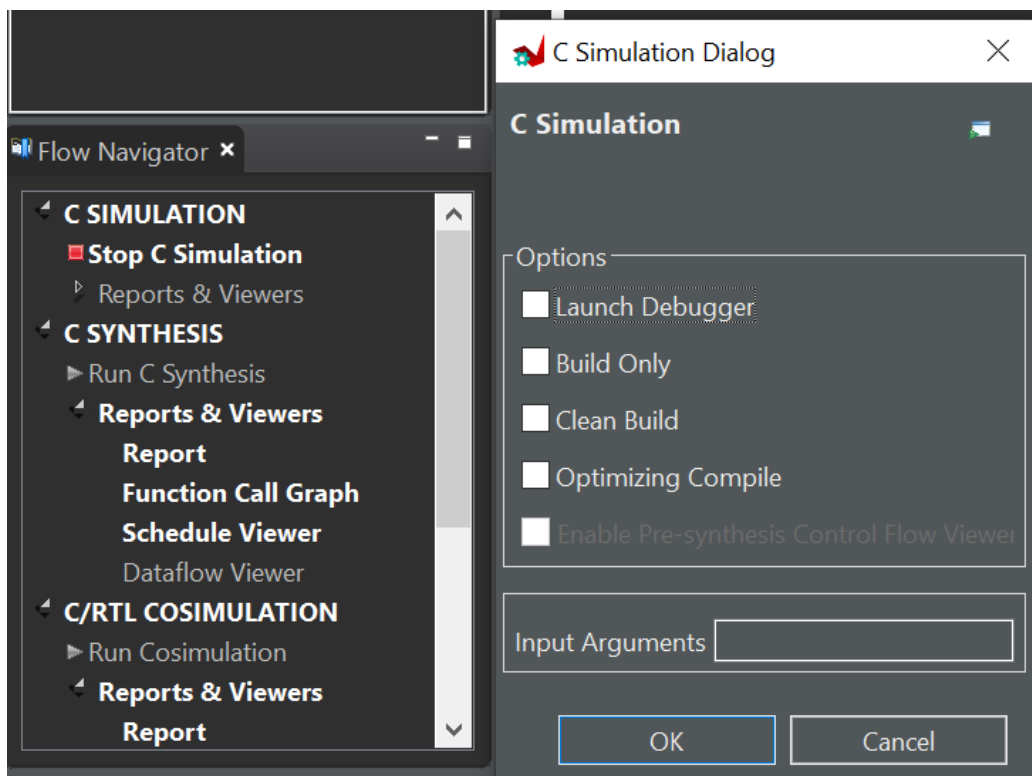
## 1.2 C-Simulation

1. 在完成项目创建后，Vitis HLS会跳转到新的界面，其由四个主要部分组成：

1. 左上方的**Explorer**，其包含了工程中的各个文件
2. 左下方的**Flow Navigator**，其展示了HLS设计中的各环节
3. 右上方的**编辑器区域**，开发者在此修改设计的代码
4. 右下方的**Console**，包含了控制台、报错信息、版本控制等



2. 下面，我们对设计进行C仿真。在左下方的**Flow Navigator**中点击**Run C Simulation**，在弹出的**C Simulation Dialog**窗口中不做改动，点击**OK**进行C仿真



3. 等待数秒，在仿真完成后Vitis HLS自动打开一个log文件，可以看到已经设计通过了C仿真

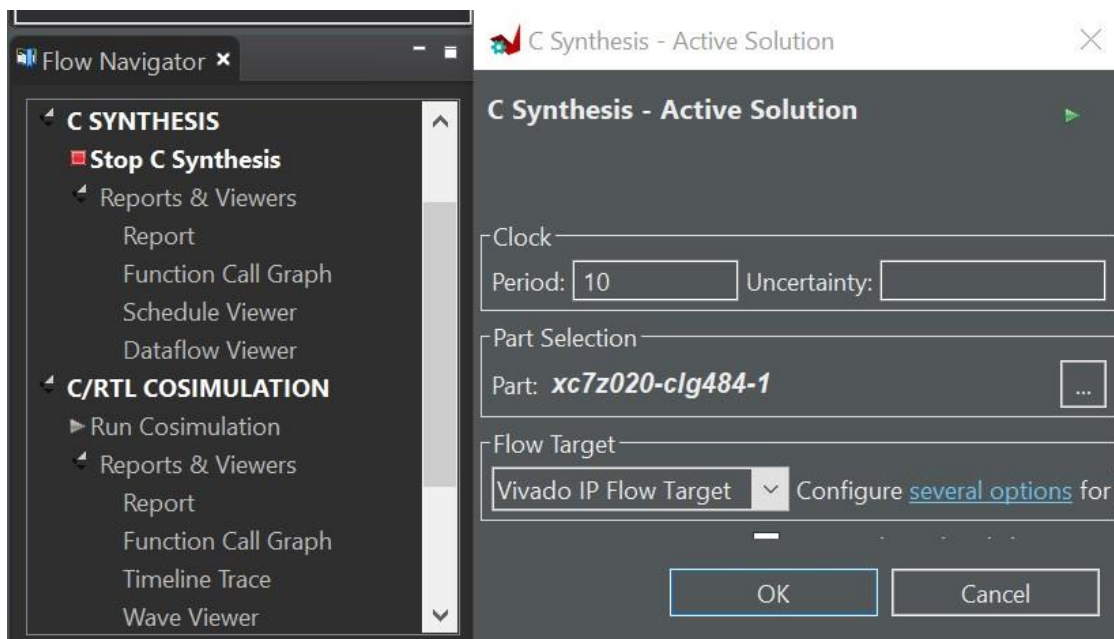
```

fir_wrap_csim.log x Synthesis Summary(solution1) fir.cpp Co-simulation Report(solution1) fir.h
1 INFO: [SIM 2] ***** CSIM start *****
2 INFO: [SIM 4] CSIM will launch GCC as the compiler.
3 make: 'csim.exe' is up to date.
4 -----
5 Test started.
6 Test passed.
7 Test finished.
8 -----
9 INFO: [SIM 1] CSim done with 0 errors.
10 INFO: [SIM 3] ***** CSIM finish *****
11

```

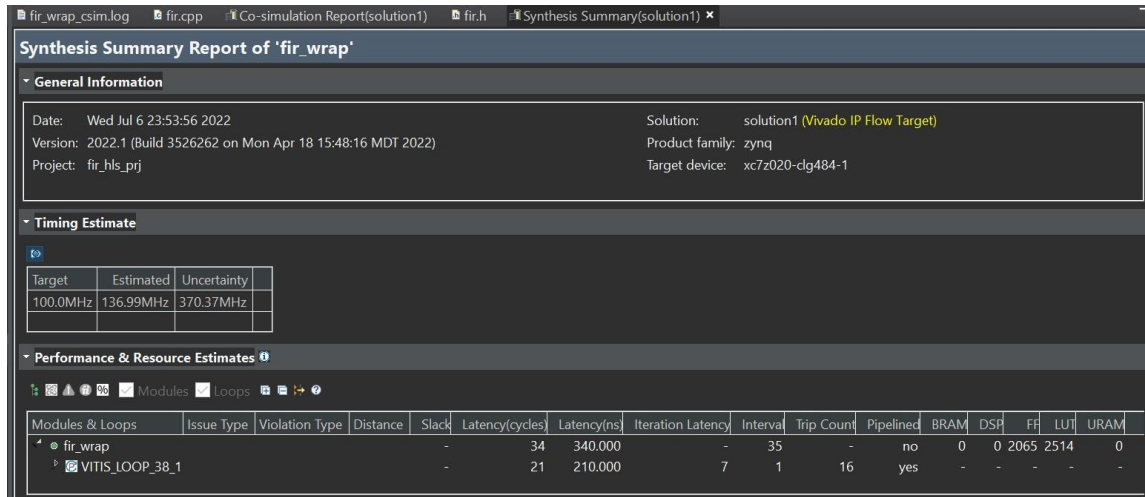
### 1.3 C-Synthesis

1. 下面，我们对设计进行C综合。在左下方的Flow Navigator中点击Run C Synthesis，在弹出的C Synthesis - Active Solution窗口中保持各选项不变，点击OK开始综合



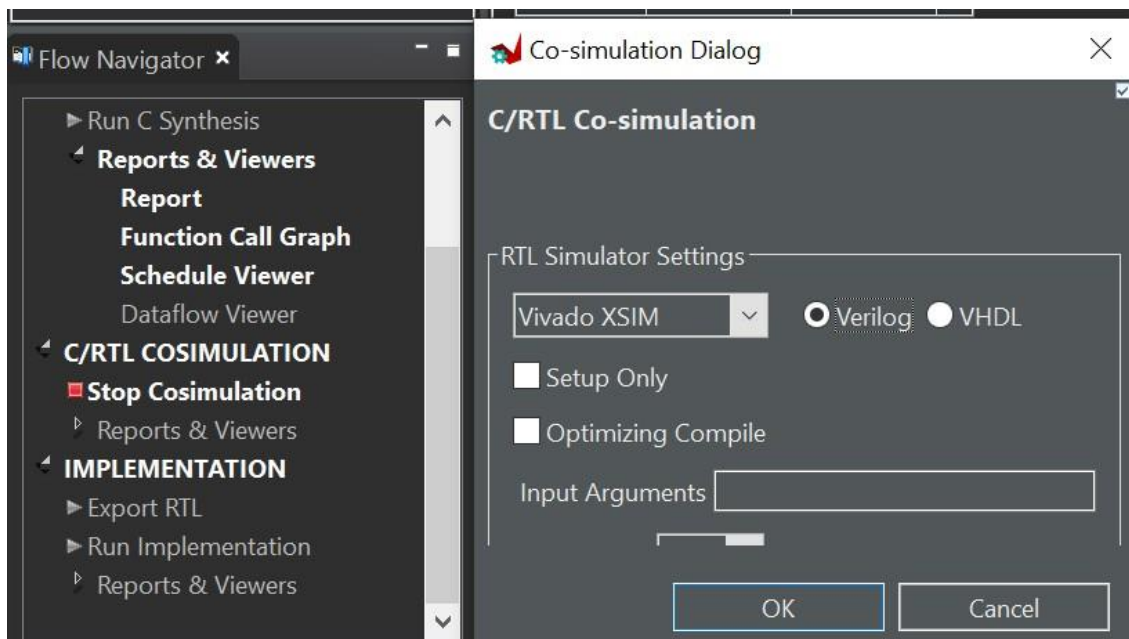
2. 等待数秒，Vitis HLS会将其综合的各步骤的信息打印在Console中

3. 综合完成后，会弹出 **Synthesis Summary(solution1)** 窗口，我们可以在此看到Vitis HLS 给出的时钟频率信息、时钟周期数和资源消耗等（在不同版本的Vitis HLS中，综合结果可能会有差异，且资源使用估算较保守，最终资源用量应以实现结果为准）

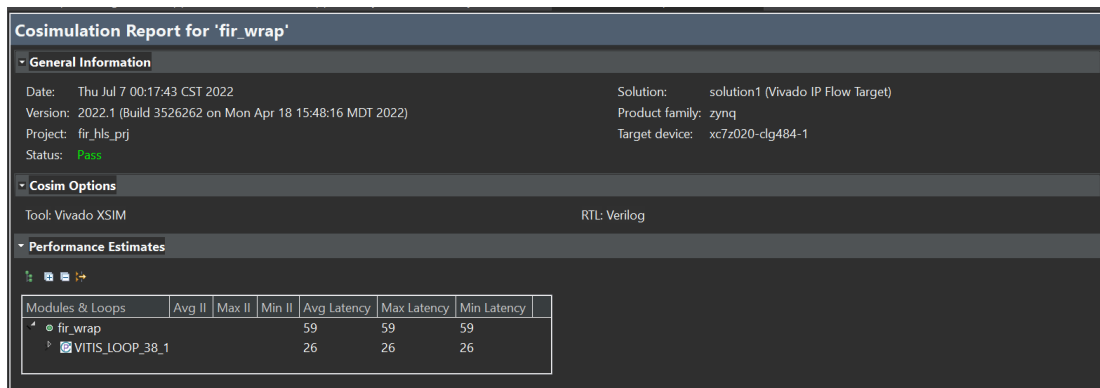


## 1.4 C/RTL Co-simulation

1. 下面，我们对设计进行C-RTL联合仿真。在左下方的**Flow Navigator**中点击**Run C/RTL COSIMULATION**，在弹出的**Co-simulation Dialog**窗口中保持各选项不变，点击OK开始综合

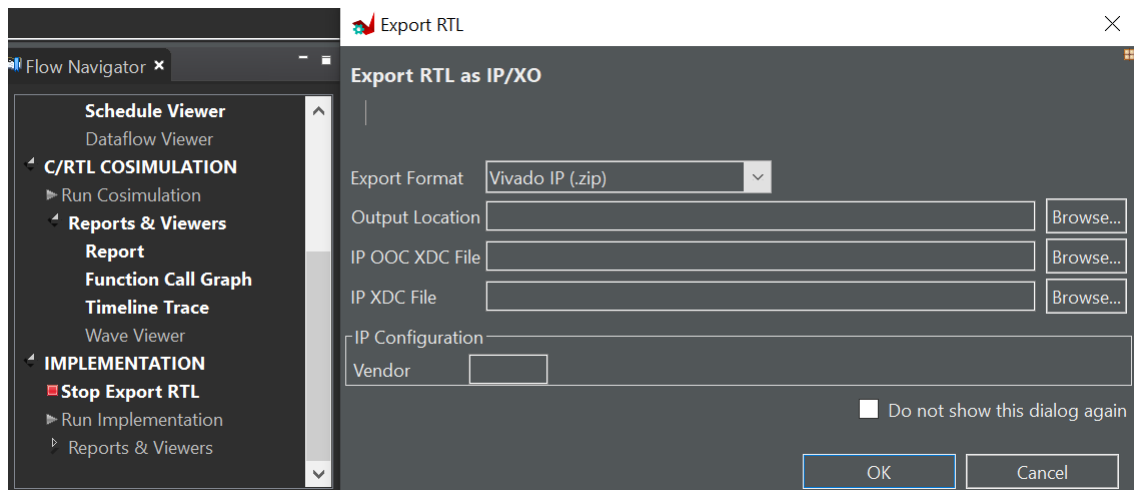


2. 等待约1分钟，C/RTL联合仿真的综合时间通常较长，仿真结束后会弹出**Co-simulation Report(solution1)** 窗口，包含了是否通过仿真、性能估测等信息



## 1.5 导出RTL

1. 下面，我们对设计进行RTL导出。在左下方的**Flow Navigator**中点击**Export RTL**，在弹出的**Export RTL**窗口中保持各选项不变，点击**OK**开始RTL的导出



2. 等待约半分钟，**Console**中打印**Finished Export RTL/Implementation**。表明RTL设计已经导出完成，你可以在**fir\_hls\_prj\solution1\impl\export.zip**找到导出的文件
3. 为了后续使用的便利，请将**fir\_hls\_prj\solution1\impl\export.zip**文件解压到其所在目录下，即得到一个**fir\_hls\_prj\solution1\impl\export** 文件夹

This PC > Documents > Xilinx > Summer2022 > DSP > prj > fir\_hls\_prj > solution1 > impl

Name	Date modified	Type	Size
export	7/7/2022 1:16 AM	File folder	
ip	7/7/2022 12:25 AM	File folder	
misc	7/7/2022 12:24 AM	File folder	
verilog	7/7/2022 12:16 AM	File folder	
vhdl	7/7/2022 12:16 AM	File folder	
export	7/7/2022 12:25 AM	Compressed (zipped)...	72 KB

4. 至此，我们已经完成了FIR加速核的设计与导出