### 实验二 IP核调用

一、实验目的：

（1）掌握Intel提供的IP核的使用方法和生成方法；

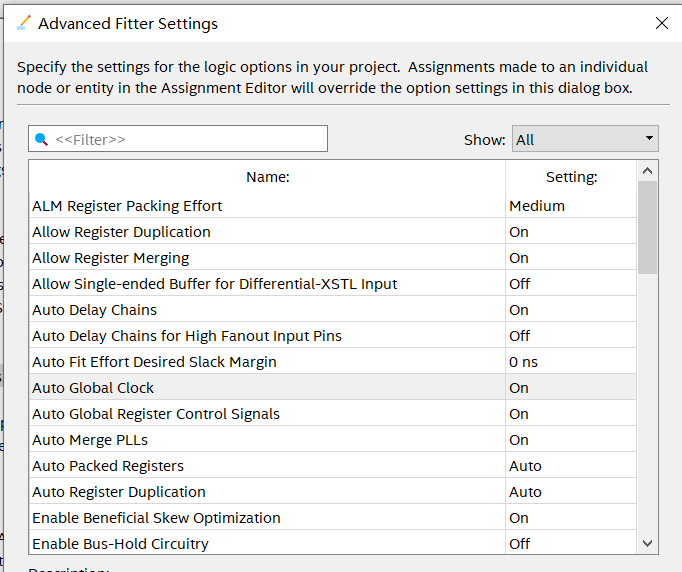
（2）掌握FIFO存储器的应用与设计方法。

二、实验原理

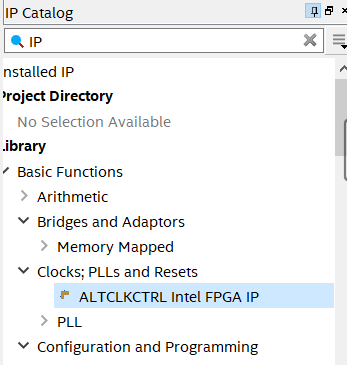
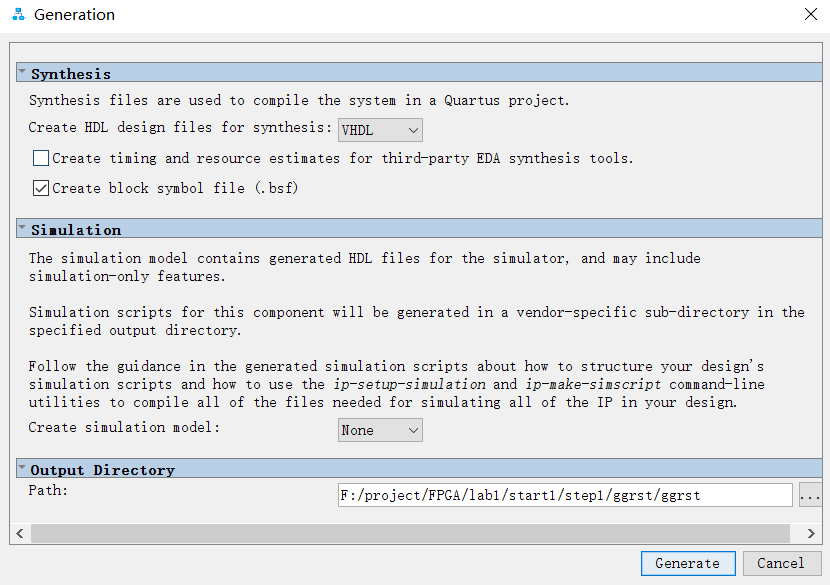
随着FPGA容量的发展和业界需求。FPGA芯片厂商应客户要求，并与部分客户合作，将一些从基础到复杂的电路功能模块做成参数化的用户可定制模块。这些模块的知识产权属于原厂，或者合作的第三方，所以叫知识产权核（IP Core）。用户可以参数化的使用这些电路功能模块，但不知道其内部的电路结构和源码。使用免费的IP核，相当于站在前人的肩膀上，帮助用户更快速的进行设计和构建系统。学习FPGA，就一定要学习如何使用IP核。

1. 实验步骤

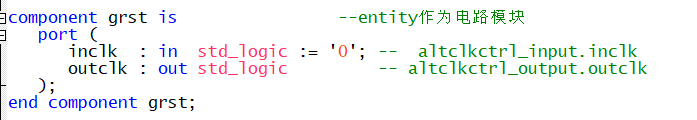
在实验一的sync代码运行和仿真过程中，设置界面将全局时钟自动开启，但是实际中需要根据实际情况设计全局变量，因此引入IP核。

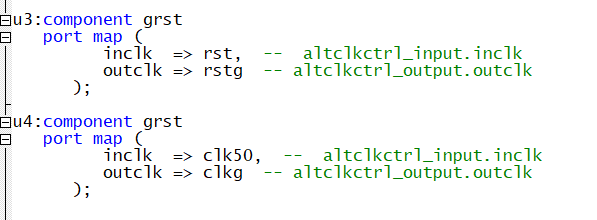


在左侧栏找到IP核点击进入，设置IP核的文件夹以及相关语言

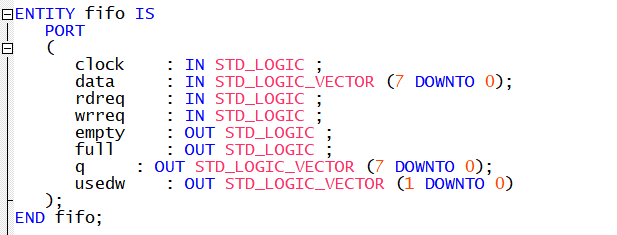
生成IP核后将后缀为.qip的文件加入工程文件夹，要使用时，整个IP核文件要作为一个子电路模块，需要在原sync文件中添加相关定义和声明。首先定义子电路模块的变量定义，在u3，u4部分添加该电路模块的实际操作。

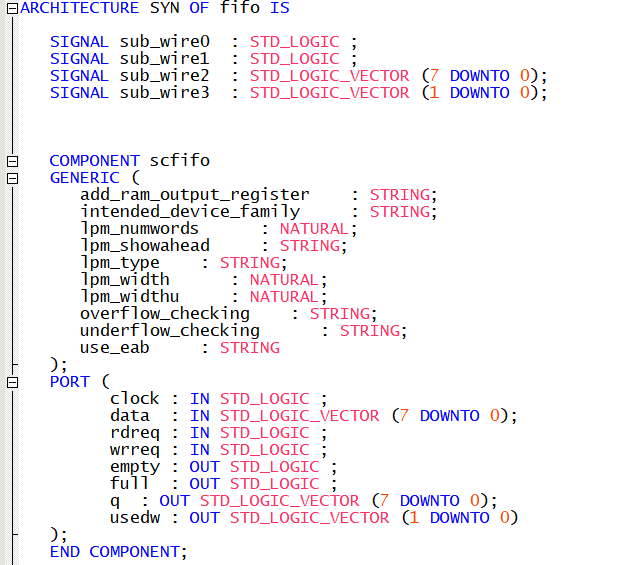




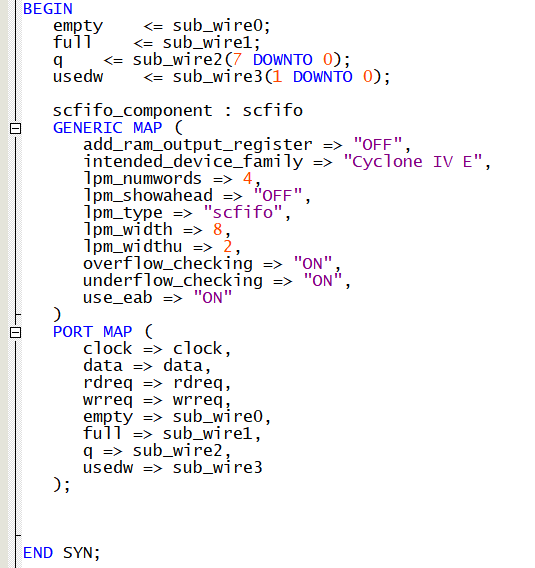
添加之后再进行编译仿真，通过编译的分析结果可以得知clk已变为全局时钟。

引入FIFO。在右侧栏找到FIFO，确定保存位置、语言之后选择位宽和深度，可以增加异步清零，确认FIFO空和慢的标志，配置完成之后点击generate生成FIFO。





以上两部分分别是对fifo的变量定义声明



以上部分为对电路的描述。

四、思考题

1.使用FIFO没有读FIFO地址、写FIFO地址的问题，其实是内部实现了地址。请画出你理解的FIFO内部的地址管理电路框图。

原理如下：

每往FIFO写入一个数据，内部的写地址加1，读地址不变，但是当写地址再次等于读地址时，此时FIFO已满，不能再写了。

每往FIFO读出一个数据，内部的读地址加1，写地址不变，但是当读地址等于写地址时，此时FIFO已空，不能再读了；