****

**实验4**

**MIPSfpag系统：中断实验**



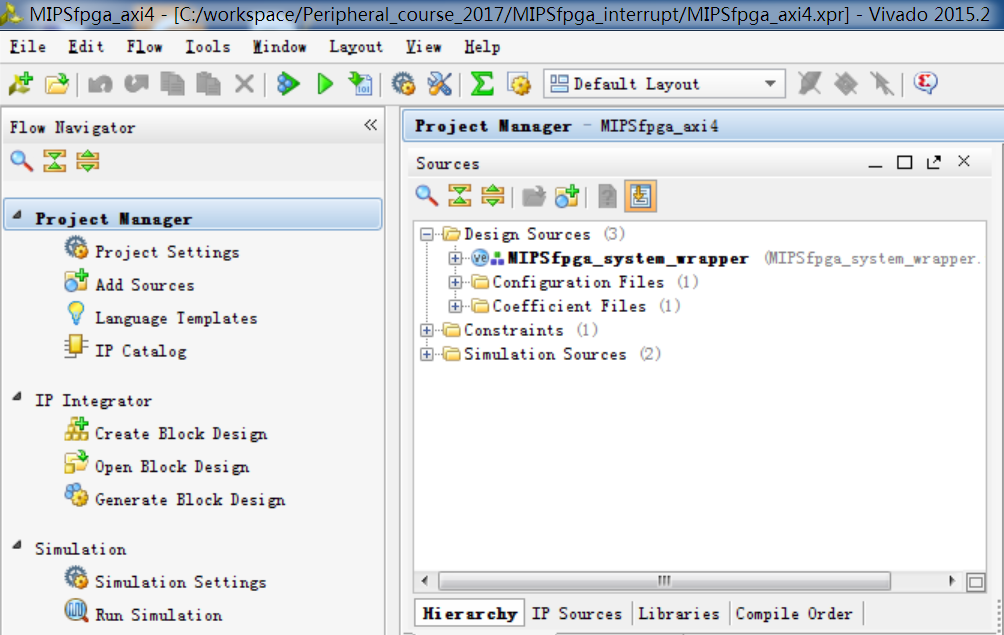
**实验4：MIPSfpga系统之中断实验**

# 一、概述

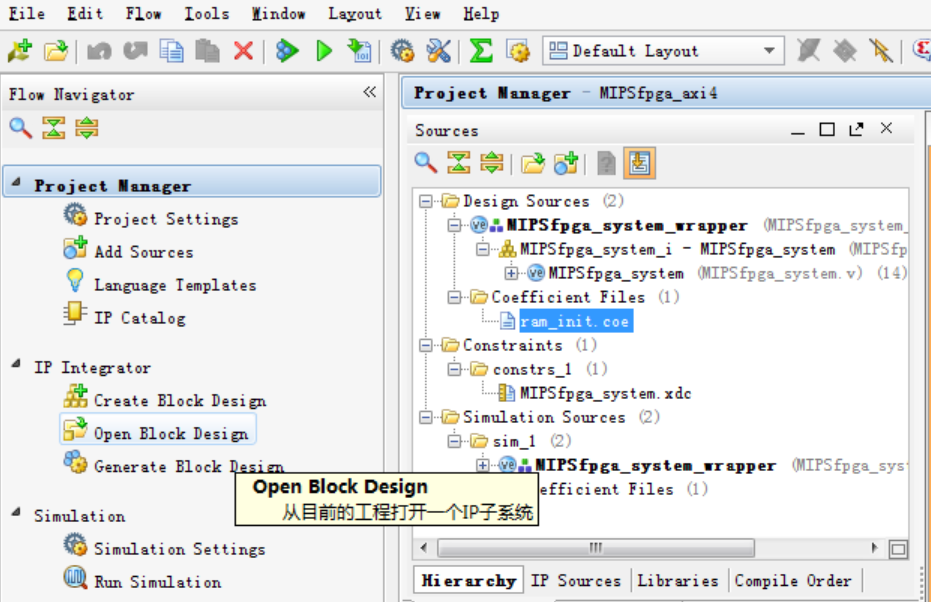
本实验将学习如何使用MIPSfpga处理器的中断系统。首先，在实验3的基础上，将UART、自定制IP模块等基于AXI4的外设模块的中断信号线连接到MIPSfpga处理器；然后，编写中断处理程序进行演示。

# 二、MIPSfpga处理器系统搭建

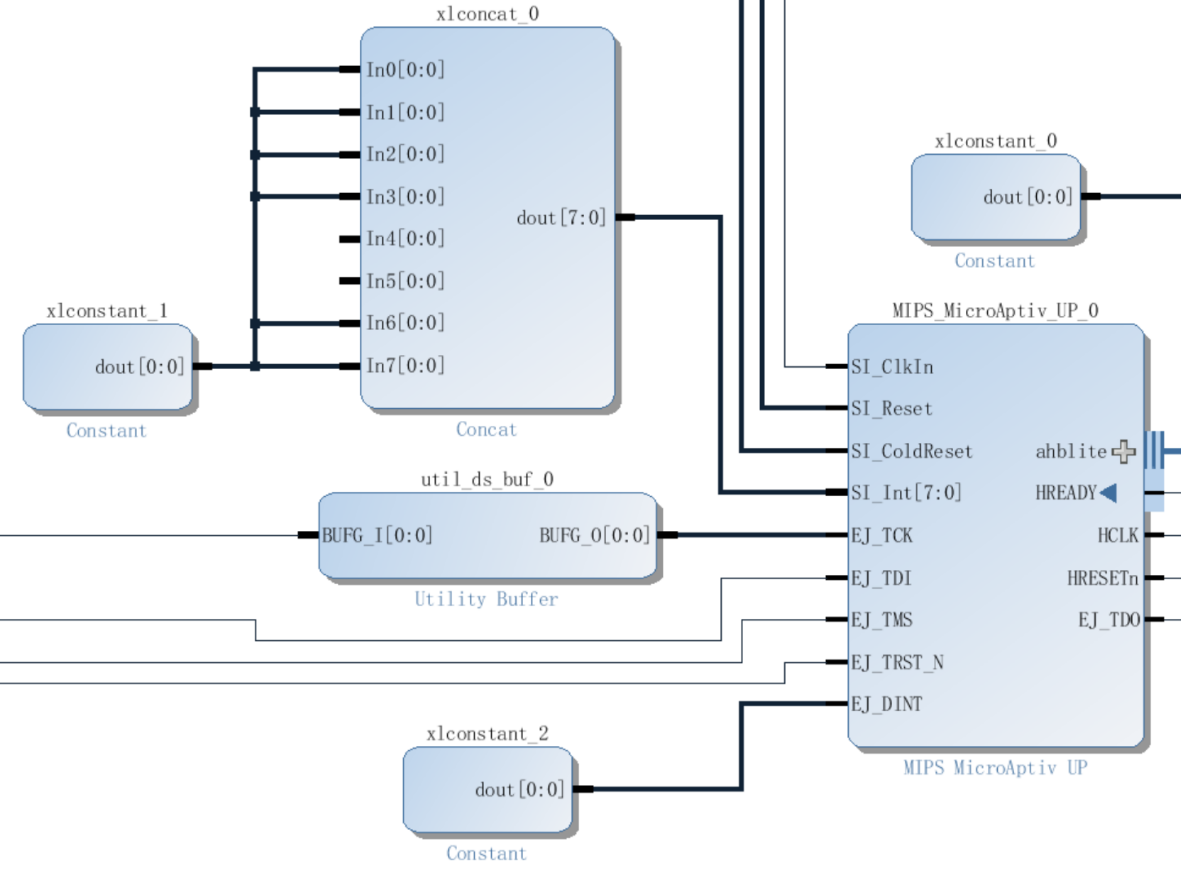
1. 复制实验3的工程，并将其目录从MIPSfpga\_CustomIP更名为MIPSfpga\_interrupt，启动Vivado，然后打开该工程。（因为只修改了该工程的目录名称，因此该工程的名称仍然是MIPSfpga\_axi4）



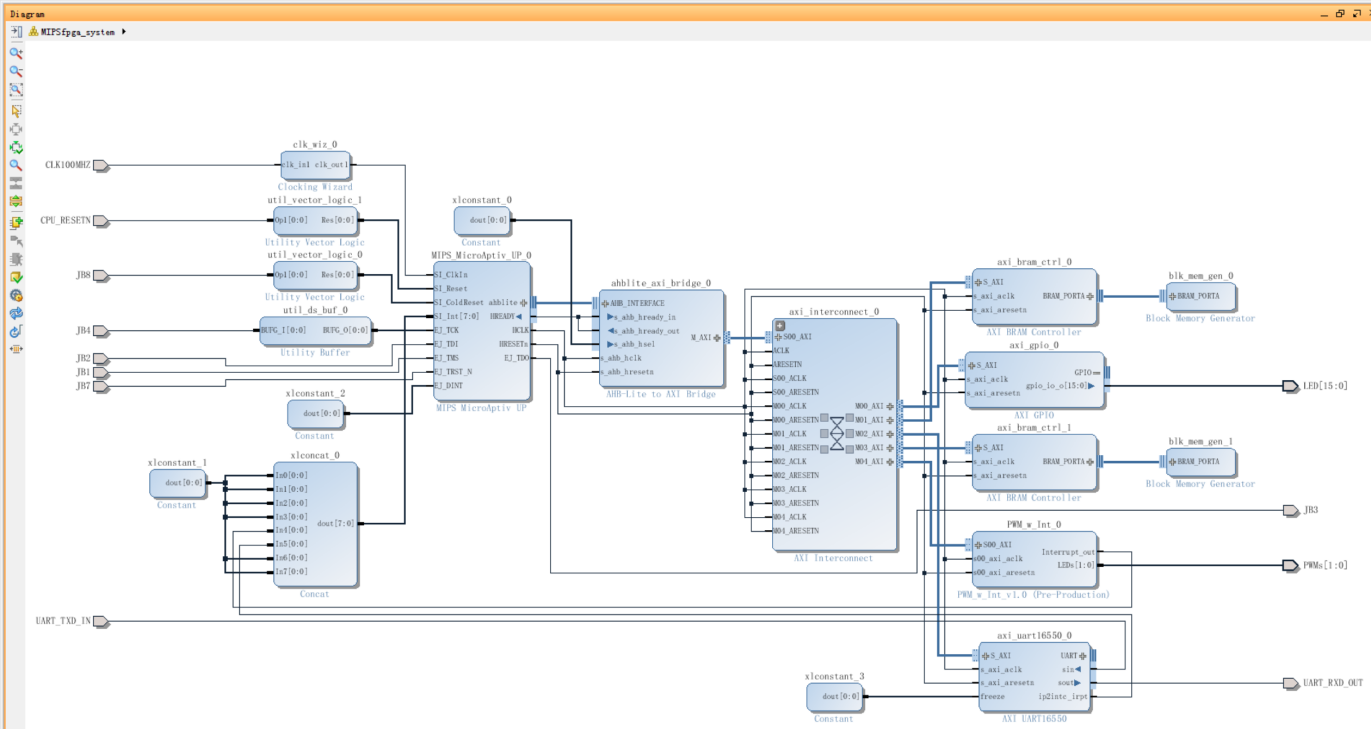
1. 点击Open Block Design打于先前的设计。



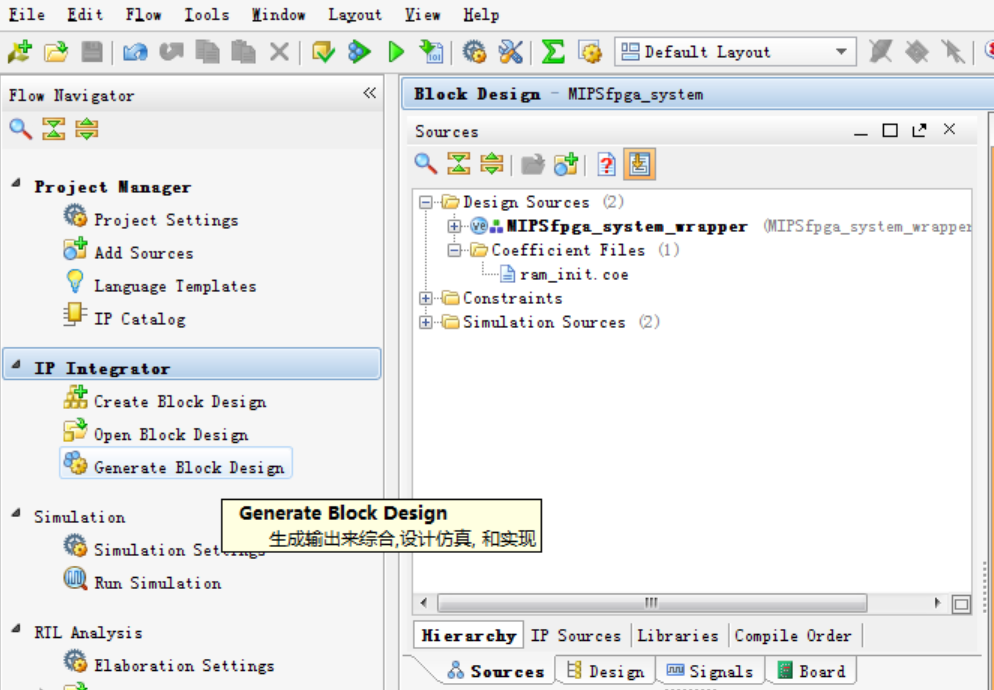
1. 将连接MIPSfpga处理器SI\_INT[7:0]引脚的Constant模块断开，并将其Const Width修为1，再添加一个8位的 Concat模块；将Concat模块的dout[7:0]引脚连接到MIPSfpga处理器的SI\_INT[7:0]；除Concat模块的In4和In5引脚外，其它的Inx引用都连接到前面修改的Constant模块。完成后如下图所示。

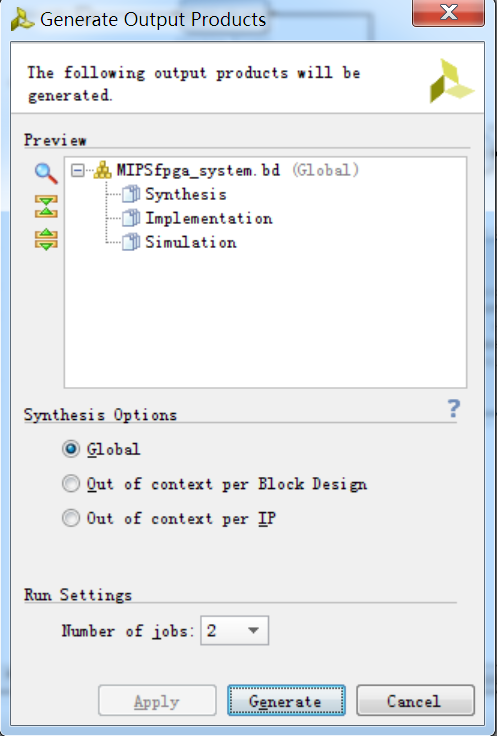


1. 将PWM\_w\_Int模块的Interrupt\_out引脚连接到上面Concat模块的In4引脚，UART16550模块的ip2intc\_irpt引脚连接到上面Concat模块的In5引脚。完成后，整个MIPSfpga处理器系统如下图所示。



1. 点击Validate Design，对设计的正确性进行校验。校验过程中如果出现警告，点击OK忽略。
2. 点击Generate Block Design，弹出对话框后选择Generate更新MIPSfpga\_system\_wrapper文件。





1. 最后，点击Generate Bitstream按键，生成bitstream文件。

# 三、应用程序编译、调试和执行

1. 进入MIPSfpga\_interrupt\_C目录，在该目录下打开boot.S，init\_cp0.S，mini\_crt0.S，exception\_entry.S和main.c等文件，仔细阅读这些文件，学习MIPSfpga处理器的中断服务是如何实现的。
2. 用鼠标右键选择打开cmd命令窗口。在该命令窗口中输入make进行编译生成elf可执行文件。使用make clean命令可以将编译的结果清除。
3. 连接Nexys4 DDR开发板bit文件下载线缆和MIPSfpga调试器。
4. 打到Codescape\_Scripts目录，在该目录下用鼠标右键选择打开cmd命令窗口。在命令窗口中输入如下命令运行loadMIPSfpga.bat批处理文件：

loadMIPSfpga.bat C:\workspace\MIPSfpga\_Peripheral\_2017\MIPSfpga\_interrupt\_C

1. 同时打开一个串口终端，将波特率设置为115200，观察演示程序的运行情况。

# 四、动手实践

1. 编写自定制外设模块PWM的中断服务程序，要求当PWM模块有中断请求时，将PWM的输入值设置为0。
2. 尝试关闭MIPSfpga处理器系统的定时器中断，消除MIPSfpga处理器系统演示程序在运行一段时间后会不断进入定时中断的Bug。