****

**实验1**

**基于MIPSfpga系统的GPIO接口实验**



**实验1：基于MIPSfpga系统的GPIO接口实验**

# 一、概述

使用Vivado开发环境，中通过IP集成的方法，搭建了一个MIPSfpga处理器的计算机系统；该系统主要包括：MIPSfpga处理器、内存、GPIO、UART等基于AXI4接口的外设模块；编写C语言程序，利用Imagination Technologies公司提供的编译器等工具，生成可在MIPSfpga处理器运行的二进制代码。

# 二、学习如何搭建MIPSfpga处理器系统

1. 新建一个目录，该目录不要包含中文和空格，将**MIPSfpga\_axi4**工程复制到该目录。例如，新建C:/workspace/MIPSfpga\_Peripheral\_2017目录，将**MIPSfpga\_axi4**工程复制到这个目录中。
2. 启动Vivado 2015.2，点击***Open Block Design***菜单进入图形化的IP集成环境，打开**MIPSfpga\_axi4**工程。在Diagram窗口中观察**MIPSfpga\_axi4**的各部分的属性，例如可在Address Editor窗口中观察内存、GPIO、UART等模块地址分配情况。

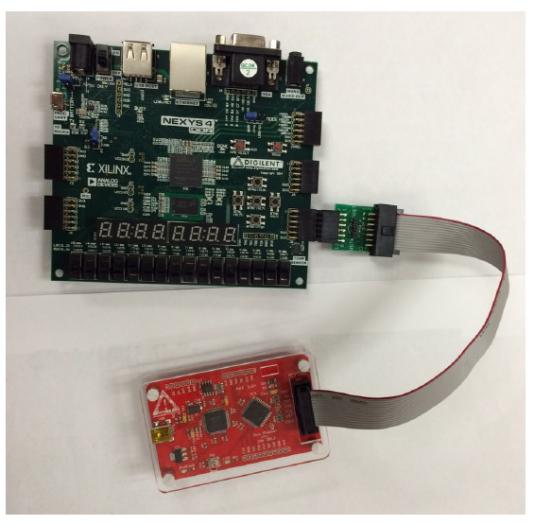
**注：**感兴趣的同学可以学着自己去搭建这样的一个系统！

# 三、应用程序编译、调试和执行（big picture）

1. **编辑源文件：**进入**MIPSfpga\_axi4\_C**目录， 编辑main.c文件即可对进行系统进行相应的操作（正确的情况下）。

**MAKE**：编辑完成后，在**MIPSfpga\_axi4\_C**下用鼠标右键选择打开**cmd**命令窗口。在此命令窗口中输入**make，**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。

1. 连接Nexys4 DDR开发板和bit文件下载线缆，同时将MIPSfpga的调试器（下图中的红色小电路板）按照下图所示连接到Nexys4 DDR开发板。连接完成后首先将Bitstream（**MIPSfpga\_axi4**系统描述）下载到Nexys4 DDR开发板，然后按CPU\_RESET按钮，准备接收1中生成的elf形式的程序。



1. 退出MIPSfpga\_axi4\_C目录，进入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：

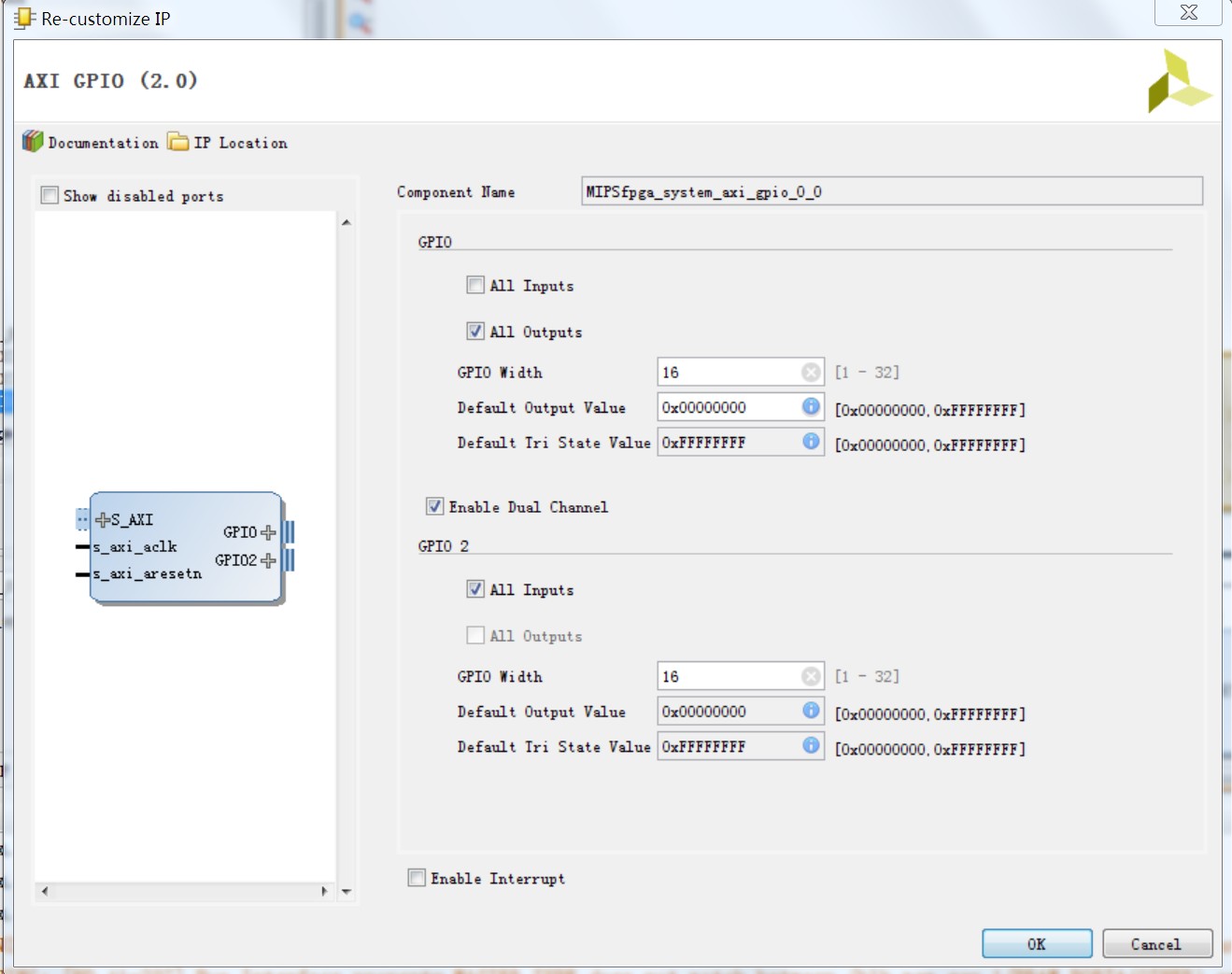
loadMIPSfpga.bat C:\workspace\MIPSfpga\_Peripheral\_2017\MIPSfpga\_axi4\_C

（下载到**MIPSfpga\_axi4系统**）

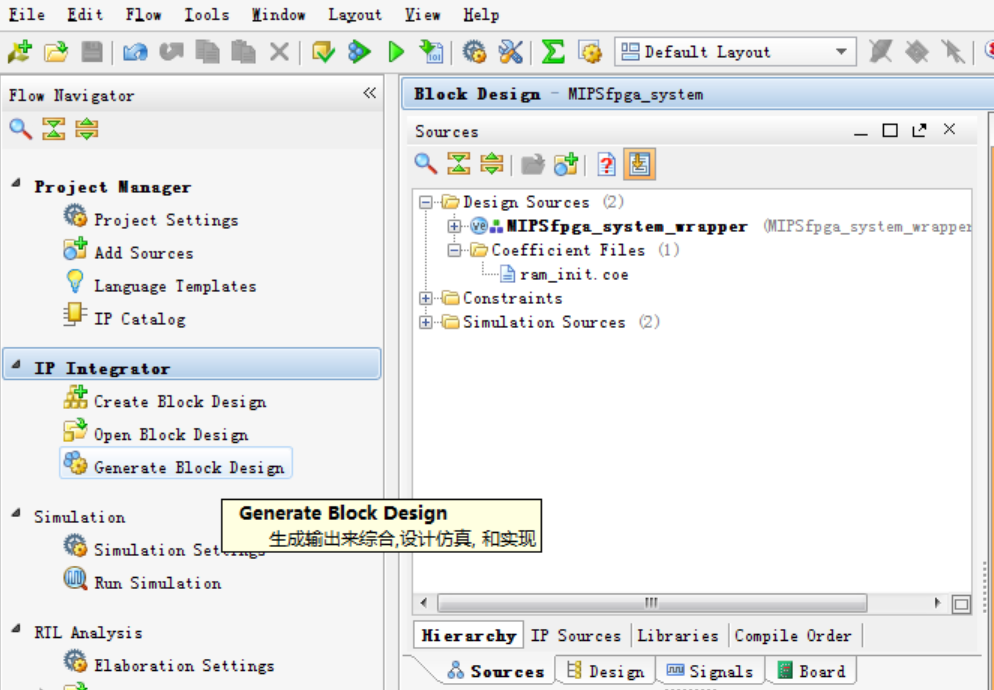
1. 观察程序的运行情况。

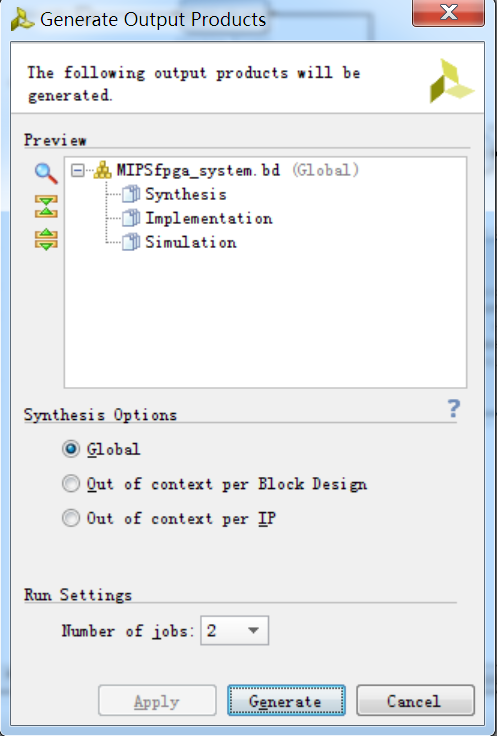
# 四、动手实践(details, one step at a time!)

1. 打开**MIPSfpga\_axi4**工程，点击Open Block Design菜单进入图形化的IP集成环境，双击axi\_gpio\_0模块添加16个输入引脚。



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. 添加约束文件，即将16个GPIO的输入引脚绑定到开发板的滑动开关。最后，点击Generate Bitstream按键，生成bitstream文件。Mipsfpga综合实现后观察时序,检查能否满足CPU运行时钟的频率要求。
2. 参照前面“应用程序编译、调试和执行”章节编写演示程序，要求通过开关的不同输入控制LED灯的显示。