## 基于QuartusII的实例

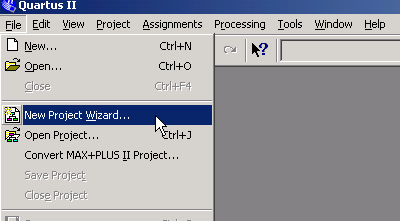
### **实验一 实验板上的KEY1按钮控制FPGA核心板上的LED灯——图形设计输入**

**目的**：通过该实例学习，可以了解FPGA的基本开发流程，熟识quartusII软件基本功能的使用。

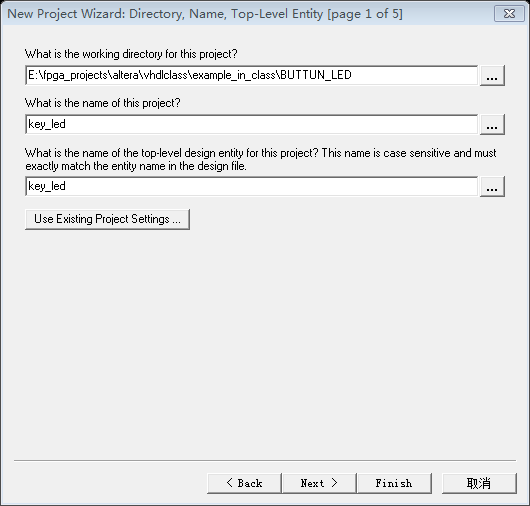
**原理**：利用一个常开按钮（实验板上的KEY1，在板上标“S2”）作为输入（常开时输入1，闭合时输入0），经过一个反相器后输出到核心板的第一个LED。KEY1常开时，LED灭，按下（闭合）实验板上的KEY1，该LED亮。

* 1. 建立工程

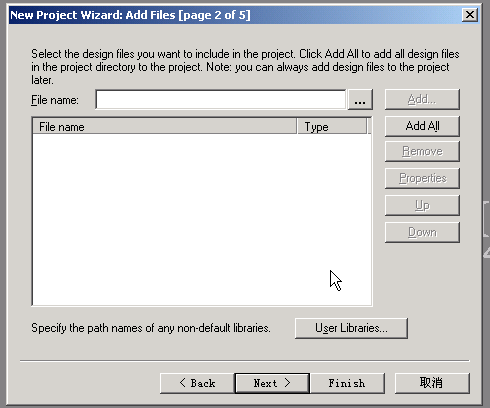
运行QuatrusII软件（以下简称Q2），建立工程，File🡪New Project Wizad如



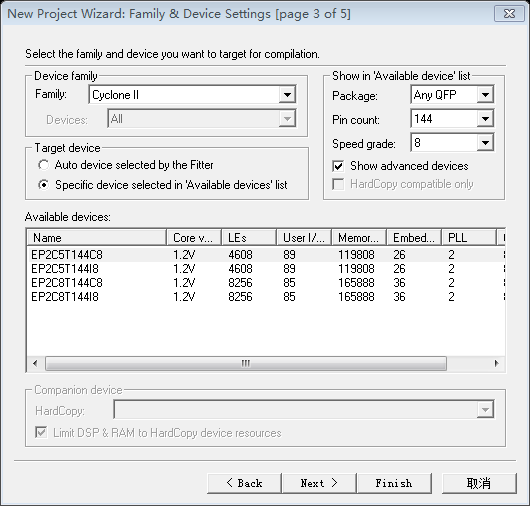
点击New Project Wizard 后弹出指定工程名的对话框，在Diectory, Name, Top-Level Entity中如下图填写：



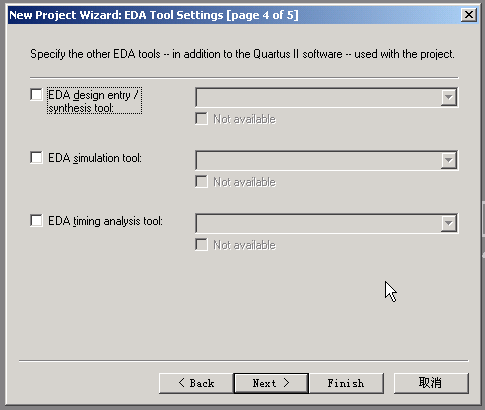
按Next按钮，出现添加工程文件的对话框：



在这里我们先不用管它，直接按Next进行下一步，选择FPGA器件的型号：



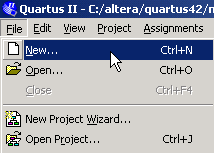
在Family下拉筐中，我们选择CycloneII系列FPGA，然后在“Available devices：”中根据核心板的FPGA型号选择FPGA型号。执行下一步出现对话框：

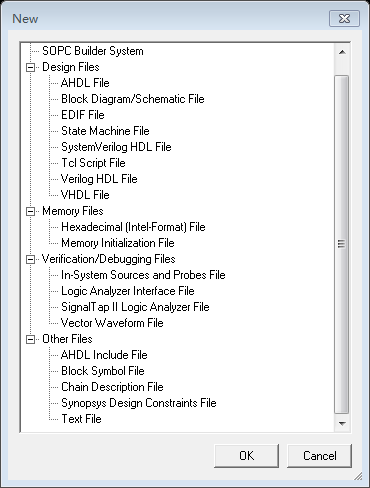


这里是选择其它EDA工具的对话框，我们用Q2的集成环境进行开发，因此这里不作任何改动。按Next进入工程的信息总概对话框，按Finish按钮即建立一个空项目。

* 1. 建立顶层图

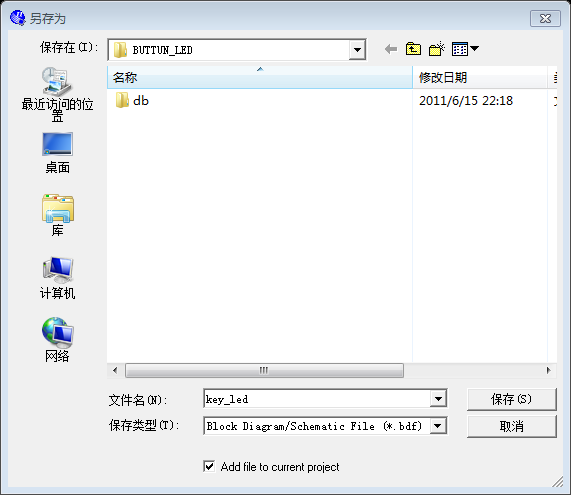
执行File🡪New，弹出新建文件对话框：





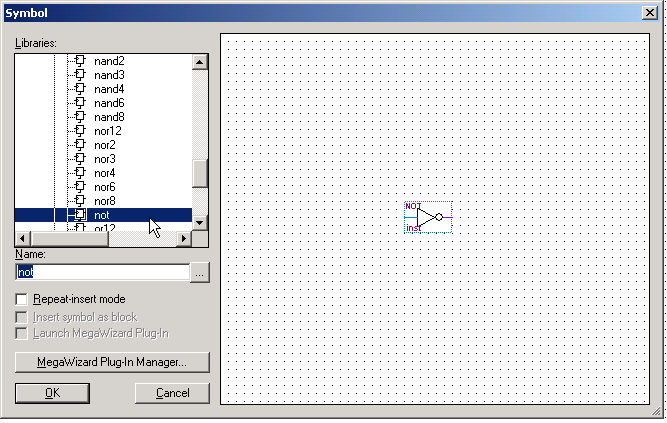
选择“Block Diagram Schematic File”按OK即建立一个空的顶层图，缺省名为“Block1.bdf”，我们把它另存为（File🡪Save as），接受默认的文件名，并将“Add file to current project”选项选上，以使该文件添加到工程中去。

如图所示：



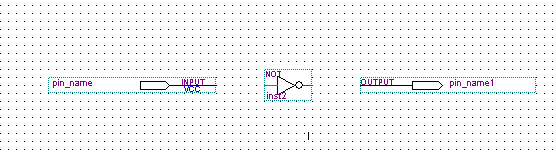
* 1. 添加逻辑元件（Symbol）

双击顶层图图纸的空白处，弹出添加元件的对话筐：

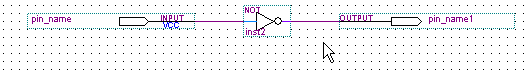


在Libraries里寻找所需要的逻辑元件，如果知道逻辑元件的名称的话，也可以直接在Name一栏敲入名字，右边的预览图即可显示元件的外观，按OK后鼠标旁边即拖着一个元件符号，在图纸上点击左键，元件即安放在图纸上。

在图纸上分别添加非门（not）、输入（input）、输出（output）三个symbol，如图所示：



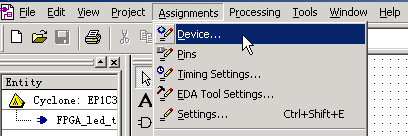
连线，将鼠标移到symbol连线端口的那里，鼠标变成图示模样：连线1，按下左键拖动鼠标到另一个symbol的连线端。本例中，这三个symbol的连线如下图所示：



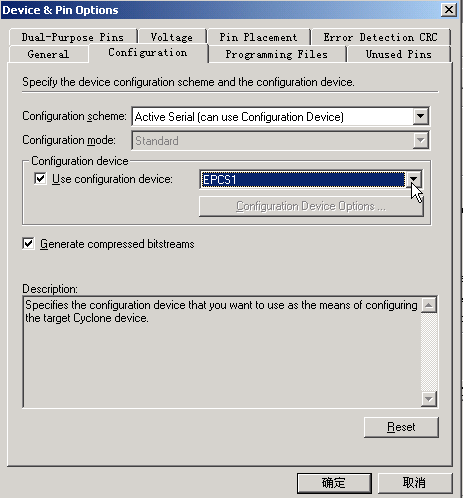
分别双击input和output symbol的名字“pin\_name”、“pin\_name1”，将它们的名字改为Key1，LED：

连线3

* 1. 设置。在建立工程的时候我们选定了芯片型号，其实也可以在这一步设定芯片型号，在菜单Assignments🡪Device：



根据核心板的FPGA选择芯片型号，如根据我们的实验板选择EP2C5T144C8。点击设置对话框的“Device & Pin Options”按钮弹出Device & Pin Options对话框，并选择该对话框的Configuration选项卡，如下图所示：



在Configuration选项卡中，选择如上图所示内容(注意：这里一定要根据开发板的具体EPCS芯片情况进行选择。(QII软件默认是AUTO,其AUTO选择与实际不一定会相符)。

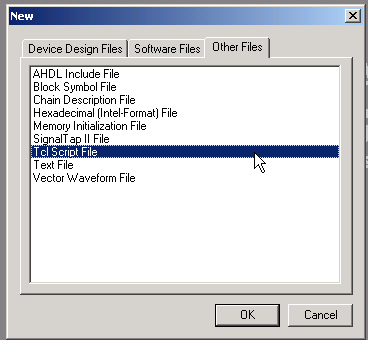
**请务必在Unused Pins标签中选择reserve all unused pins　:　as input tri-stated！！否则有损坏实验板的危险!!**

其余留缺省设置即可。按确定退出该对话筐，返回到顶层图界面。

* 1. 分配管脚

为芯片分配管脚可以用QuartusII软件里的“Assignments🡪Pins”菜单，也可以用tcl脚本文件。用Tcl文件进行配置可重用性好，易于管理，因此本文介绍用tcl的方法。对于另一种方法，可以参考QuartusII软件的帮助文档。

在工程目录下建立一个name为Setup.tcl的file。File🡪New，选择other files页面：



有关tcl文件的更详尽内容可参考QuartusII的帮助文档，在此，我们已经提供了本实验板的一个sncu\_pins.tcl文件，已经分配了实验板上外设的管脚。在实际项目中，该文件也可以根据具体管脚分配要求来改写。

#scnu\_pins.tcl

set\_global\_assignment -name RESERVE\_ALL\_UNUSED\_PINS "AS INPUT TRI-STATED"

set\_global\_assignment -name ENABLE\_INIT\_DONE\_OUTPUT OFF

set\_location\_assignment PIN\_17 -to clk

#led

set\_location\_assignment PIN\_71 -to led

#seg7

set\_location\_assignment PIN\_65 -to seg7com\[0\]

set\_location\_assignment PIN\_67 -to seg7com\[1\]

set\_location\_assignment PIN\_69 -to seg7com\[2\]

set\_location\_assignment PIN\_70 -to seg7com\[3\]

set\_location\_assignment PIN\_53 -to seg7data\[0\]

set\_location\_assignment PIN\_55 -to seg7data\[1\]

set\_location\_assignment PIN\_57 -to seg7data\[2\]

set\_location\_assignment PIN\_58 -to seg7data\[3\]

set\_location\_assignment PIN\_59 -to seg7data\[4\]

set\_location\_assignment PIN\_60 -to seg7data\[5\]

set\_location\_assignment PIN\_63 -to seg7data\[6\]

set\_location\_assignment PIN\_64 -to seg7dp

#SDRAM

set\_location\_assignment PIN\_112 -to sd\_data\[0\]

set\_location\_assignment PIN\_104 -to sd\_data\[1\]

set\_location\_assignment PIN\_103 -to sd\_data\[2\]

set\_location\_assignment PIN\_101 -to sd\_data\[3\]

set\_location\_assignment PIN\_100 -to sd\_data\[4\]

set\_location\_assignment PIN\_99 -to sd\_data\[5\]

set\_location\_assignment PIN\_97 -to sd\_data\[6\]

set\_location\_assignment PIN\_96 -to sd\_data\[7\]

set\_location\_assignment PIN\_129 -to sd\_data\[8\]

set\_location\_assignment PIN\_132 -to sd\_data\[9\]

set\_location\_assignment PIN\_133 -to sd\_data\[10\]

set\_location\_assignment PIN\_134 -to sd\_data\[11\]

set\_location\_assignment PIN\_135 -to sd\_data\[12\]

set\_location\_assignment PIN\_136 -to sd\_data\[13\]

set\_location\_assignment PIN\_139 -to sd\_data\[14\]

set\_location\_assignment PIN\_137 -to sd\_data\[15\]

set\_location\_assignment PIN\_76 -to sd\_addr\[0\]

set\_location\_assignment PIN\_75 -to sd\_addr\[1\]

set\_location\_assignment PIN\_74 -to sd\_addr\[2\]

set\_location\_assignment PIN\_73 -to sd\_addr\[3\]

set\_location\_assignment PIN\_113 -to sd\_addr\[4\]

set\_location\_assignment PIN\_114 -to sd\_addr\[5\]

set\_location\_assignment PIN\_115 -to sd\_addr\[6\]

set\_location\_assignment PIN\_118 -to sd\_addr\[7\]

set\_location\_assignment PIN\_119 -to sd\_addr\[8\]

set\_location\_assignment PIN\_120 -to sd\_addr\[9\]

set\_location\_assignment PIN\_79 -to sd\_addr\[10\]

set\_location\_assignment PIN\_121 -to sd\_addr\[11\]

set\_location\_assignment PIN\_81 -to sd\_ba\[0\]

set\_location\_assignment PIN\_80 -to sd\_ba\[1\]

set\_location\_assignment PIN\_94 -to sd\_ldqm

set\_location\_assignment PIN\_126 -to sd\_udqm

set\_location\_assignment PIN\_86 -to sd\_cs

set\_location\_assignment PIN\_87 -to sd\_ras

set\_location\_assignment PIN\_92 -to sd\_cas

set\_location\_assignment PIN\_93 -to sd\_we

set\_location\_assignment PIN\_122 -to sd\_cke

set\_location\_assignment PIN\_125 -to sd\_clk

#key

set\_location\_assignment PIN\_88 -to key1

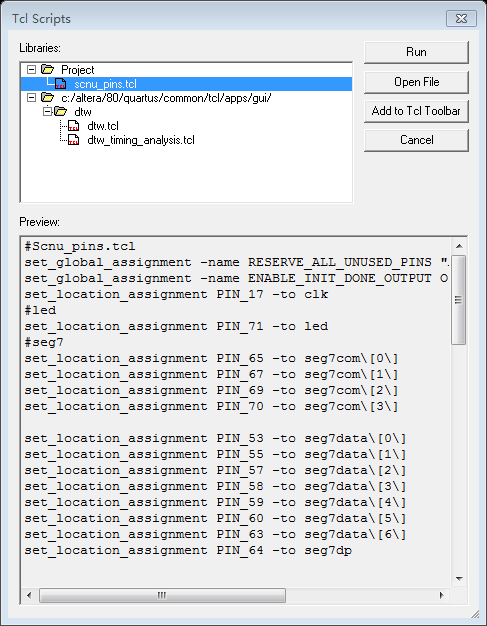
set\_location\_assignment PIN\_89 -to key2

set\_location\_assignment PIN\_90 -to key3

set\_location\_assignment PIN\_91 -to key4

set\_location\_assignment PIN\_72 -to key5

保存到工程目录下，并注意在保存对话框选上“Add file to current project”选项。然后打开Tools -> Tcl Scripts,选中刚才编辑的Script文件：scnu\_pins.tcl,并点击Run。



注意：建立工程时如果路径名有中文字符或者路径名有空格字符，则tcl Script文件将无法运行。请留意QuartusII软件的输出信息框查看是否已正常运行该tcl文件。

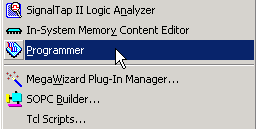


* 1. 编译。按主工具栏上的编译按钮即开始编译：

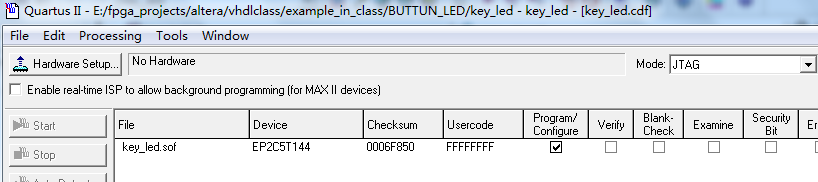


QuartusII下面的Message 窗口会显示一些编译信息，若编译成功弹出提示。

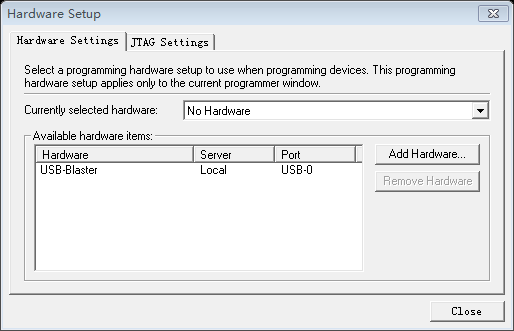
* 1. 下载。
     1. 下载设置：如果第一次使用下载线下载配置文件到FPGA，则需要在Q2软件设置下载线的型号等信息。先将下载线的一头接到PC的USB，执行菜单Tools🡪Programmer（如下图）



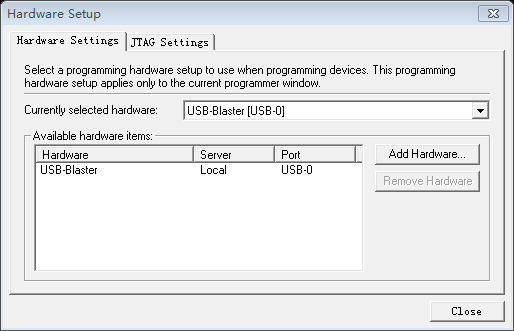
打开下载界面，若看到No Hardware，表示还没有设置下载线，如下图：



点击Hardware Setup，弹出“Hardware Setup”对话框：



点击“No Hardware”下拉筐，选择USB-BLASTER：



点击“Close”返回下载界面，可看到下载线已经被添加：



* + 1. 将USB-BLASTER下载线一头与PC连接，另一头插入到JTAG口（实验板上下面那个）或EPCS1（实验板上上面那个）的下载口。

**注意：插上下载线时请务必是在断电的情况下操作，否则有损坏FPGA下载口的危险！！**

* + 1. FPGA核心板接上5V电源。
    2. 如果只是调试一下设计是否成功，可通过JTAG口把芯片的配置信息下载到FPGA芯片内，掉电后配置信息丢失。此时，下载界面的“Mode：”下拉列表应选择“JTAG”，并选择工程中.sof后缀的文件进行下载。（注意记得在“Program/Configure”那个方框那里打上“√”，其它“Verify”、“Blank Check”等可根据需要选择。）
    3. 如果要下载到非易失Flash配置芯片（掉电后配置信息不丢失）里去，则将下载线插到EPCS1芯片的下载口（实验板上上面的插口）。并在下载界面的“Mode：”下拉列表应选择“Active Serial Programming”，并选择工程中.pof后缀的文件进行下载。一般情况下使用JTAG下载即可，等整个设计都完成了不需要再修改后才把最后的POF下载到EPCS芯片中。

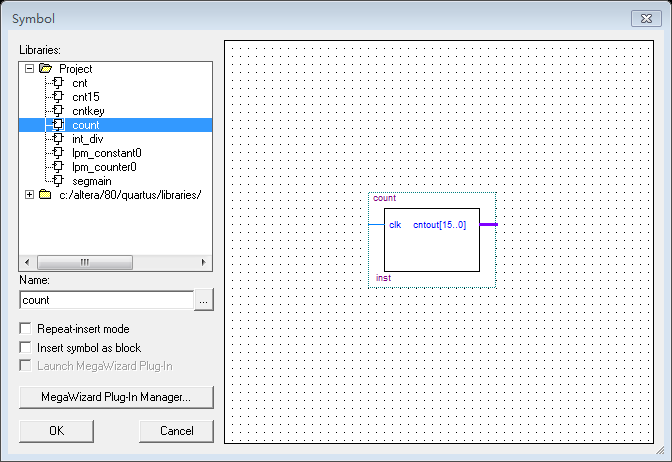
另外注意：插拔下载线要先断电，在插下载线的时候注意需要用手托着PCB板底部，以免会因为插入的时候过于用力导致PCB板变形从而使FPGA脱焊的危险。我们知道EPCS芯片的擦写次数是有限度的，一般我们不需要经常性下载POF到EPCS芯片中，在验证，调试，烧写flash的时候一直使用JTAG即可，只有当调试都OK了到的确需要保存配制信息的时候才使用EPCS。于此下载线就可以一直插在JTAG口，这样就不需要频繁地插拔电源，以及插拔下载线，以最大可能地降低损坏板子的风险，延长开发板的使用寿命。

补充说明：在第3步“添加逻辑元件（Symbol）”，我们在这个例子里面只是添加QuartusII的库里自带的一些逻辑符号，如果我们需要添加我们用VHDL编写的模块符号，必须先生成该VHDL对应的模块符号。

在工程里打开需生成模块符号的VHDL文件，如count.vhd

从“文件”->“Creat/Update”->“Creat Symbol File For Current File”即可，注意参考QuartusII的输出信息框那里是否提示成功生成，若错误，可能是因为该vhdl文件有语法错误。

生成模块符号后，可以在插入符号对话框里的“project”列表里找到生成的符号。



另外，对于多模块设计，必须指定其中一个文件为顶层文件，如指定原理图文件作为顶层文件，那么就应该在顶层文件对输入输出信号分配管脚。设为顶层文件的方法是：打开该文件，选择菜单：“Project”->“Set As Top-Level Entity”

**实验二 实验板上的KEY1按钮控制FPGA核心板上的LED灯——VHDL设计输入**

1. **按实验一方法建立QuartusII工程；**
2. **建立VHDL文件，注意QuartusII要求文件名必须与实体名一致；**
3. **往工程添加设计文件，并设置顶层文件；**
4. **分配管脚，若用tcl文件分配管脚，注意tcl文件中的管脚名应与顶层文件中的端口名一致。**

**实验三 设计8-3优先级编码器，用一个数码管显示编码，用板上的5个按键模拟8路输入中的部分输入。**