

Altera FPGA 开发入门

Quartus II 13.0 SP1 + Modelsim SE-64 10.2c + Cyclone-V

liyh@Advanced Electronics System Lab. @201803

实验目的：这是一个 Altera FPGA 开发入门的实验，在实验完成后你将熟悉 Altera FPGA 的开发环境、设计步骤和流程。

实验形式：边实验边记录实验过程，思考，错误解决，方法等等，实验结束即提交实验记录。

实验内容及步骤：

一、打开 Quartus II 13.0 SP1 软件

鼠标左键双击桌面上的 Quartus II 13.0sp1(64-bit)图标，如下图示。如桌面上没有 Quartus II 图标，可以通过开始菜单=>所有程序=>Altera 13.0.1.232=>Quartus II 13.0.1.232 找到 Quartus II 13.0sp1(64-bit)。

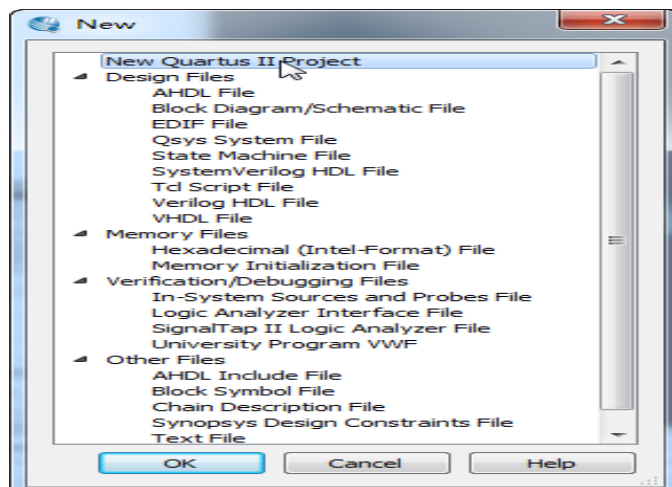


打开的 Quartus II 界面具有 Windows 操作系统下一般应用软件的基本特征，比如菜单栏、工具条、状态栏等。另外，Quartus II 还有工程管理 (Project Navigator)，任务管理 (Tasks)，信息提示 (Messages)，及工作窗口等，如下图所示。

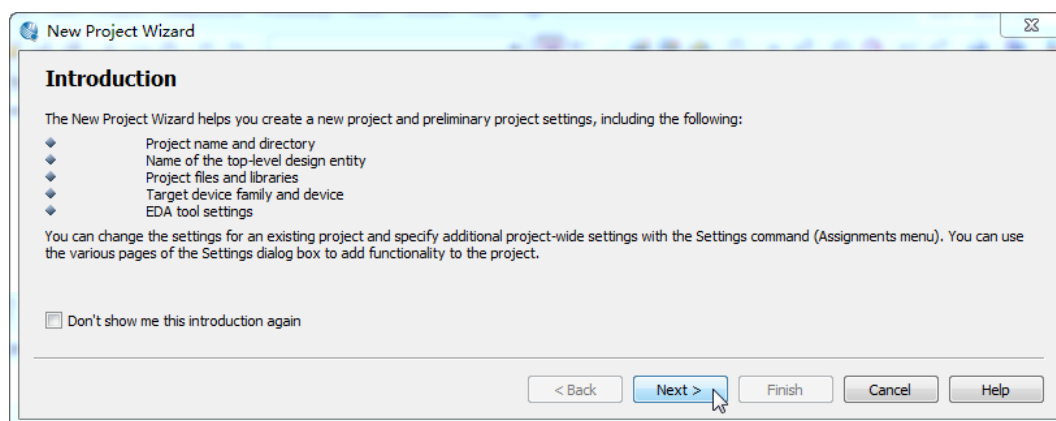


二、创建一个 FPGA 设计工程

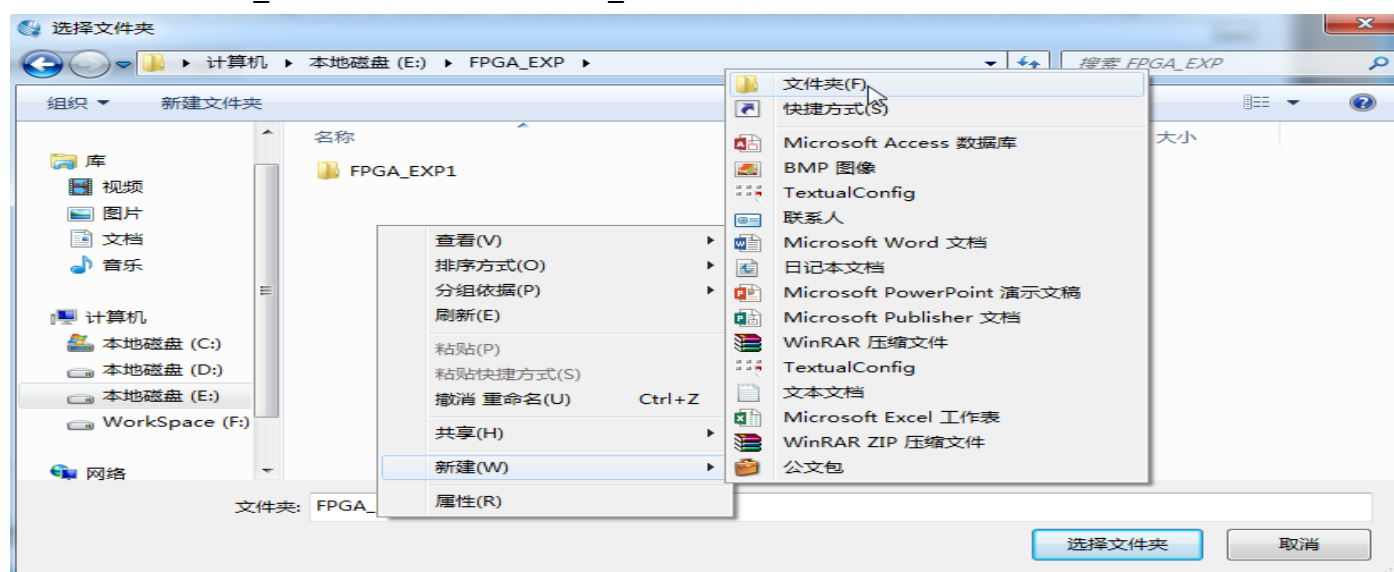
操作：鼠标左键单击（以下简称“单击”）“File”菜单，选择“New...”命令，在弹出的 New 窗口中选择“New Quartus II Project”项后，单击“OK”按钮。如下图示。（尝试其他打开“New Project Wizard”的方式，可以写入报告）



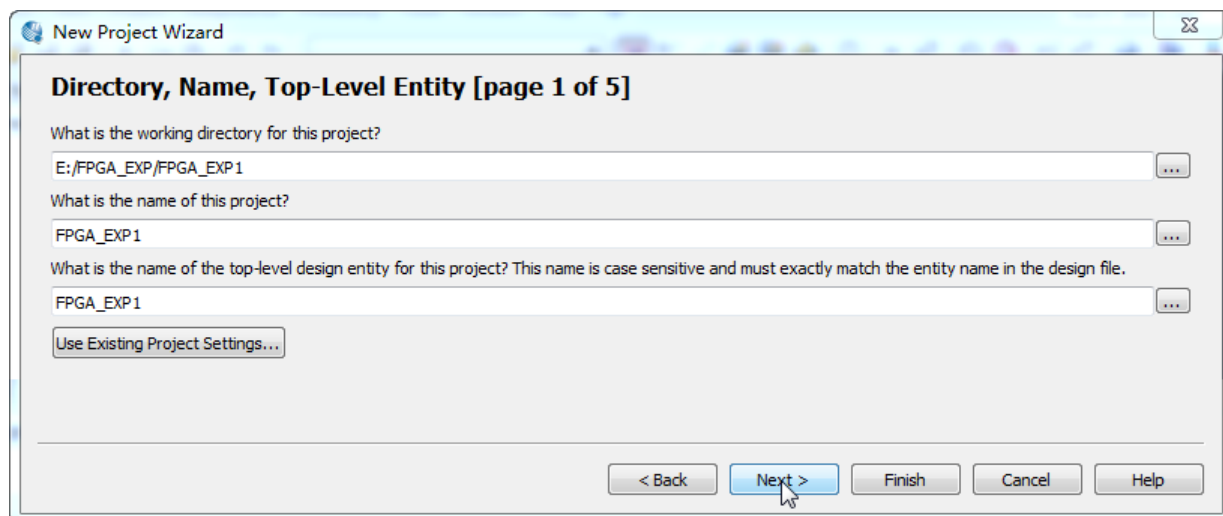
在打开的“New Project Wizard”窗口中，单击“Next>”按钮进入新工程向导第一步，如下图所示。



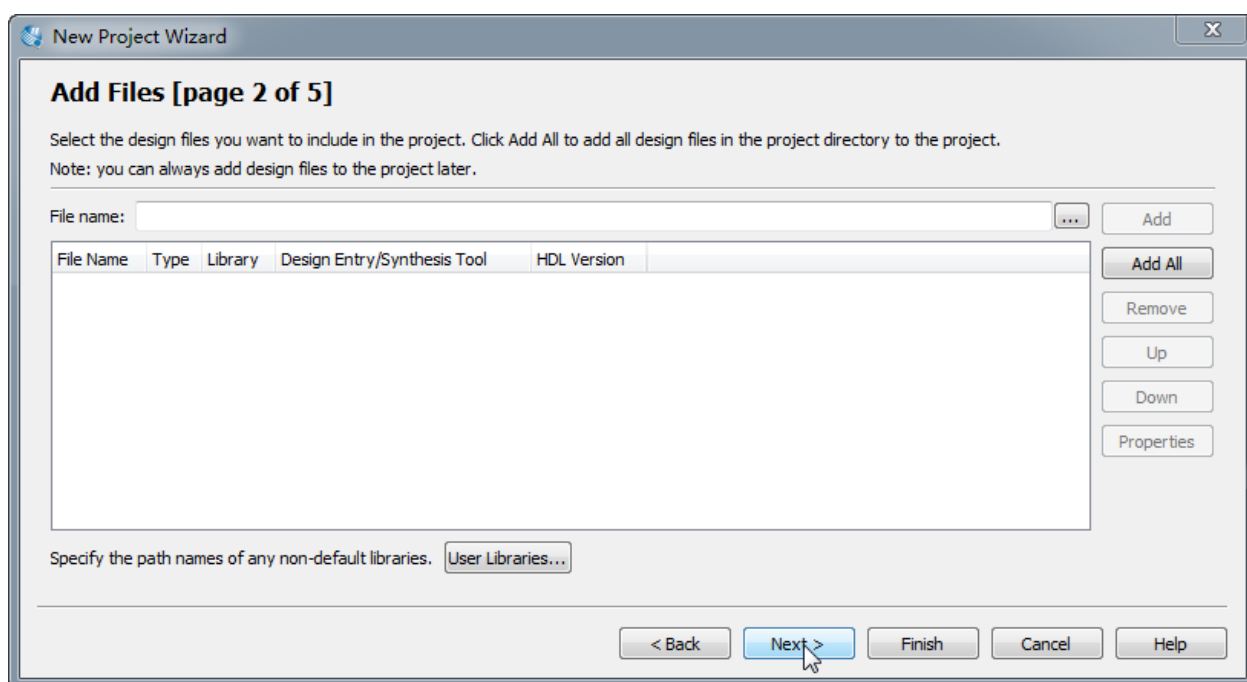
新工程创建第一步：Directory, Name, Top-Level Entity。单击“What is the working directory for this project”最右边的“...”按钮，在弹出的“选择文件夹”窗口中左边选择“计算机”下的“本地磁盘(E:)”，然后在窗口的右边单击鼠标右键，选择“新建”下的“文件夹”，并将文件夹重命名为“FPGA_EXP”。以同样的方式，再在“FPGA_EXP”文件夹下新建“FPGA_EXP1”文件夹。如下图所示。



选择好工程存放的文件夹后，指定“What is the name of this project”为 FPGA_EXP1，同时会自动确定“What is the name of the top-level design entity for this project? This name is case sensitive and must exactly match the entity name in the design file.”。单击“Next>”按钮进入新工程向导第二步，如下图所示。

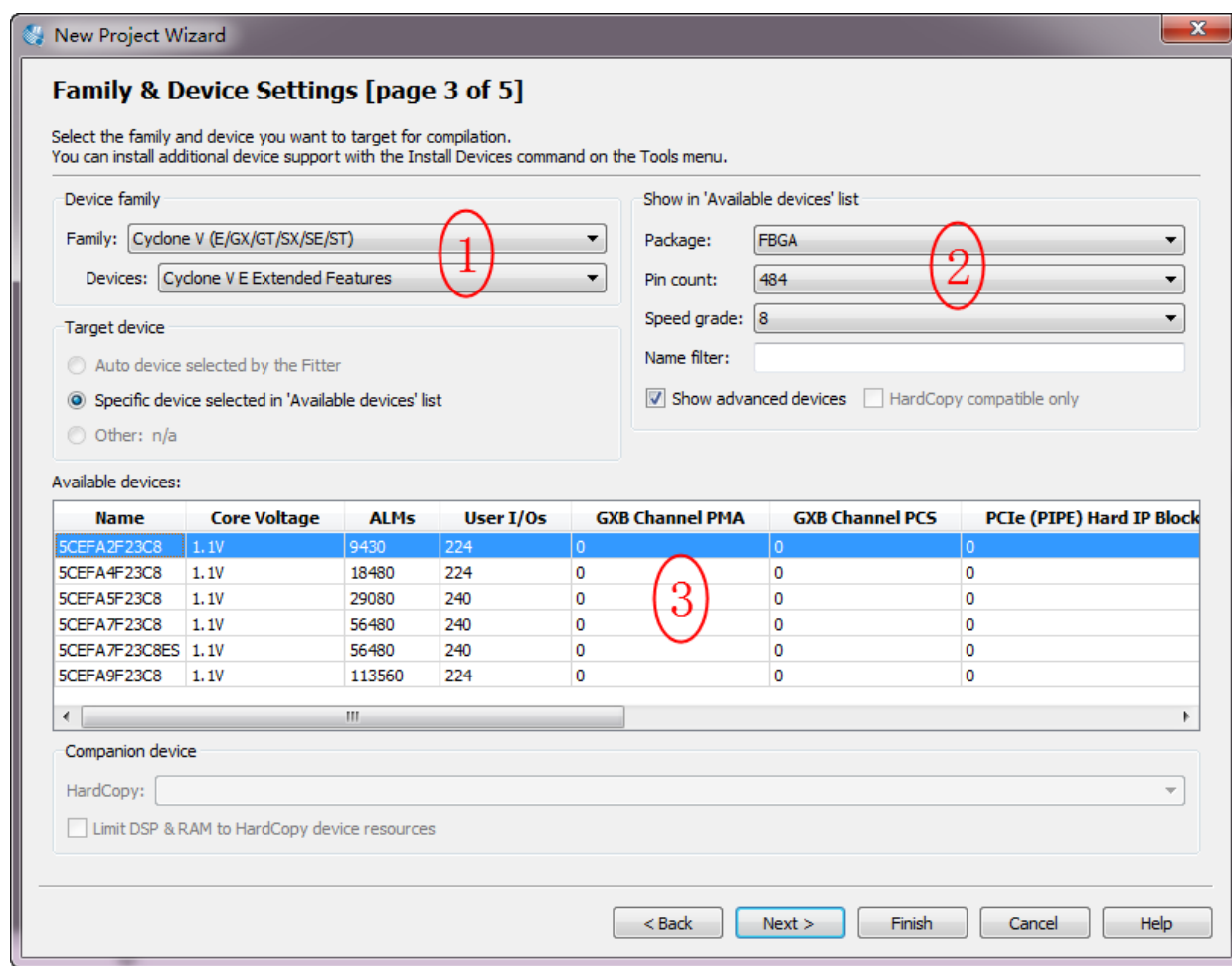


新工程创建第二步：Add Files。如果已有设计文件，可以在这一步将其添加到当前工程中来。在此工程中没有设计好的文件，因此直接单击“Next>”按钮进入新工程向导第三步。

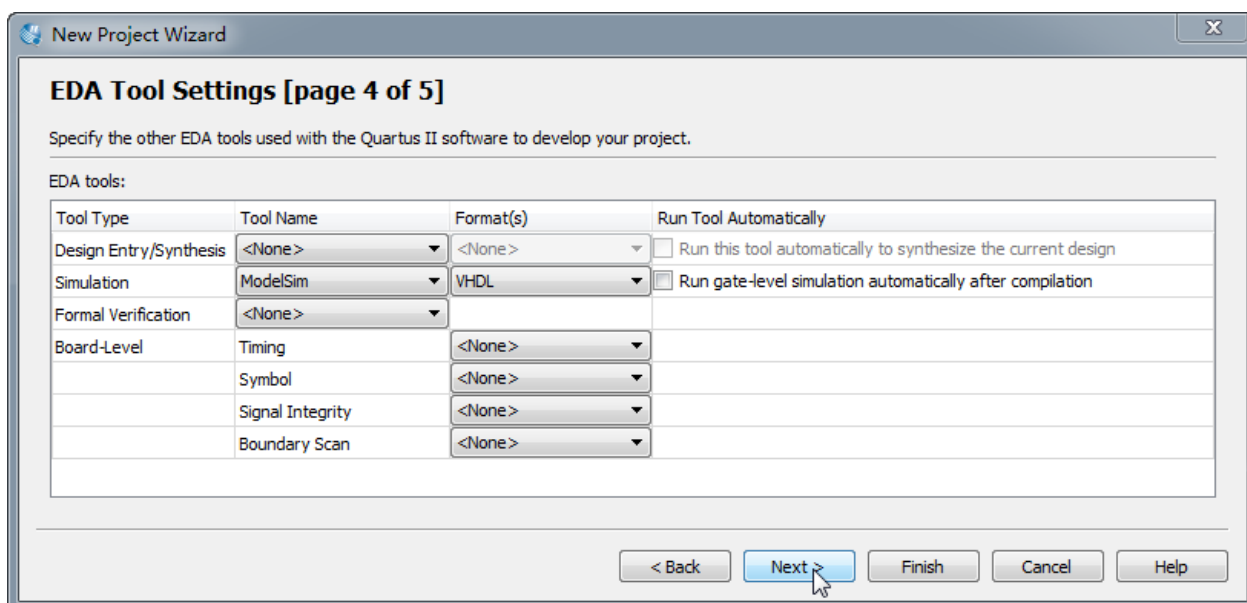


新工程创建第三步：Family & Device Settings。选择“Device family”中的“Family”为“Cyclone V”；“Devices”为“Cyclone V E Extended Features”；再选择“Show in ‘Available devices’ list 中的“Package”为“FBGA”、Pin count 为“484”和“Speed grade”为“8”；最后在“Available devices”中选中“Name”为“5CEFA2F23C8”那一行。如下图所示。

选择好目标板上的 FPGA 器件型号后，单击“Next>”进入新工程向导第四步。

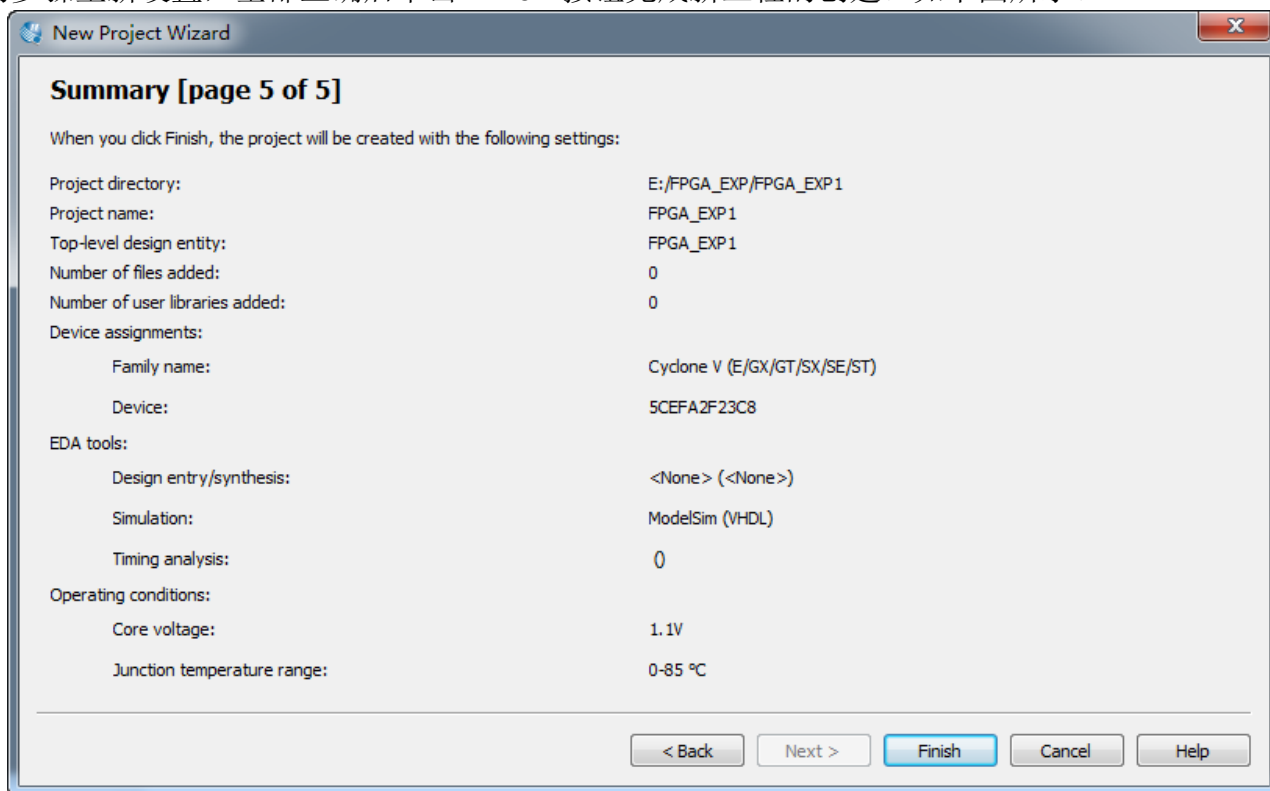


新工程创建第四步：EDA Tool Settings。选择“Tool Type”为“Simulation”那行中的“Tool Name”为“Modelsim”、“Format(s)”为“VHDL”，其他的保持默认的“None”。单击“Next>”按钮进入新工程向导最后一步。如下图所示。



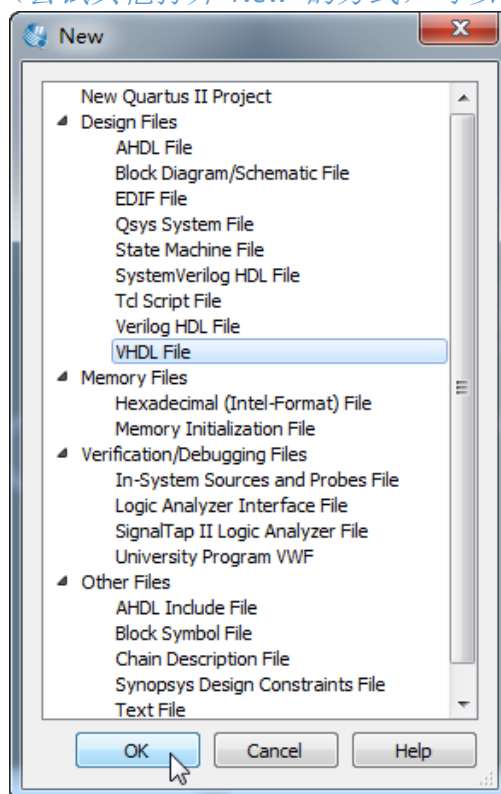
新工程创建第五步：Summary。查看新工程的设置是否正确，有不正确的单击“<Back”按钮返回前

面的步骤重新设置，全部正确后单击“Finish”按钮完成新工程的创建。如下图所示。



三、为新创建工程添加设计文件及代码

操作：鼠标左键单击“File”菜单，选择“New...”命令，在弹出的 New 窗口中选择“VHDL File”项，然后单击“OK”按钮。如下图所示。（尝试其他打开“New”的方式，可以写入报告）



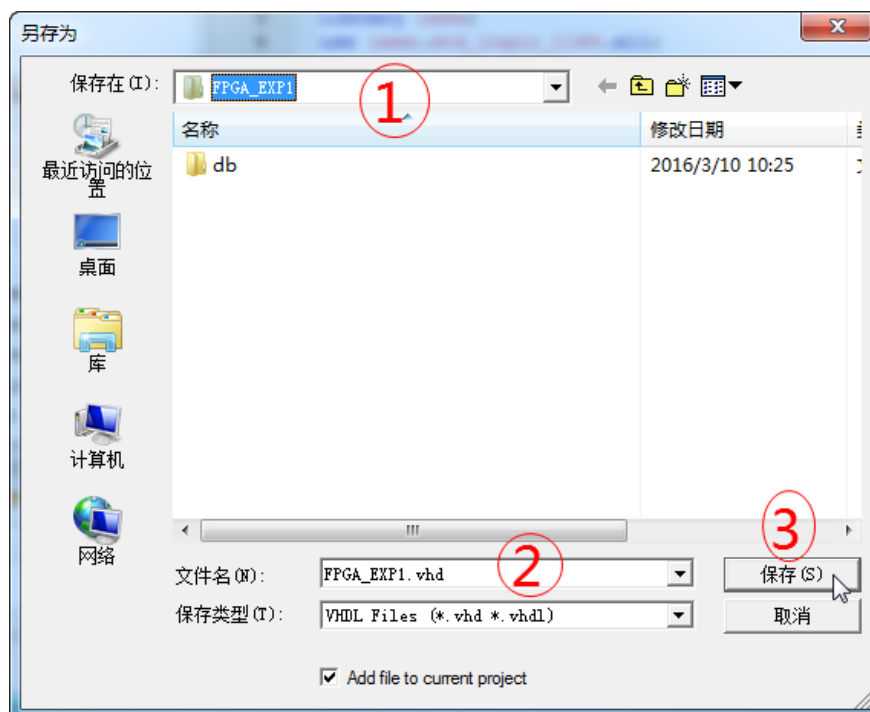
在工作区会打开 VHDL 设计文件编辑窗口，然后在此窗口中输入以下代码。

```

---FPGA_EXP1 @ 20180326
library ieee;
use ieee.std_logic_1164.all;
entity FPGA_EXP1 is
  port(
    y0_n,y1_n,y2_n,y3_n,y4_n,y5_n,y6_n,y7_n :out std_logic;
    a,b,c :in std_logic
  );
end FPGA_EXP1;
architecture arch_EXP1 of FPGA_EXP1 is
  signal abc :std_logic_vector(2 downto 0);
begin
  abc <= a & b & c;
  process(abc)
  begin
    y0_n <= '1';
    y1_n <= '1';
    y2_n <= '1';
    y3_n <= '1';
    y4_n <= '1';
    y5_n <= '1';
    y6_n <= '1';
    y7_n <= '1';
  end process;
end arch_EXP1;

```

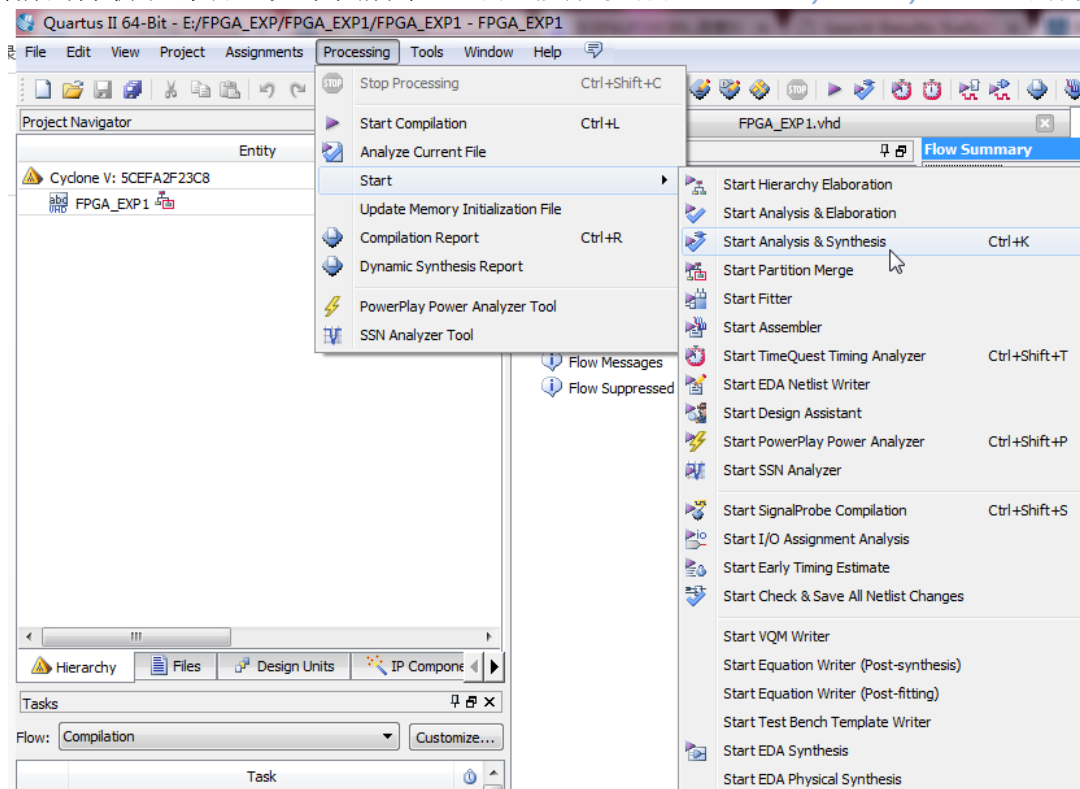
代码输入完成后，鼠标左键单击“File”菜单，选择“Save”命令，在弹出的“另存为”窗口中位置①处保存在“选择当前工程文件夹”FPGA_EXP1”、在位置②处“保存类型”选择“VHDL File”、文件名为“FPGA_EXP1.vhd”，（如这些参数已经默认且一致，就不用重新设置了），最后单击“保存”按钮完成设计文件的输入。如下图所示。（尝试其他打开“另存为”的方式）



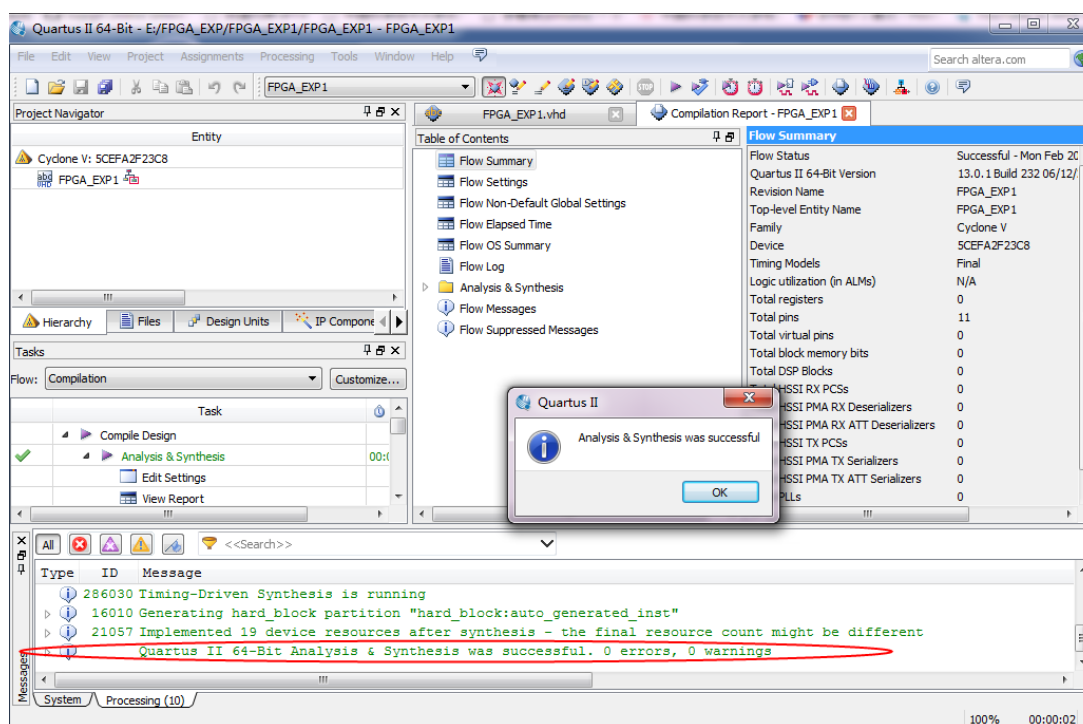
四、编译设计工程：Analysis & Synthesis、

操作：鼠标左键单击“Processing”菜单，选择“Start”子菜单中的“Start Analysis & Synthesis”命令，完

成功能仿真前的分析和综合，如下图所示。（尝试其他完成“Start Analysis & Synthesis”的方式）



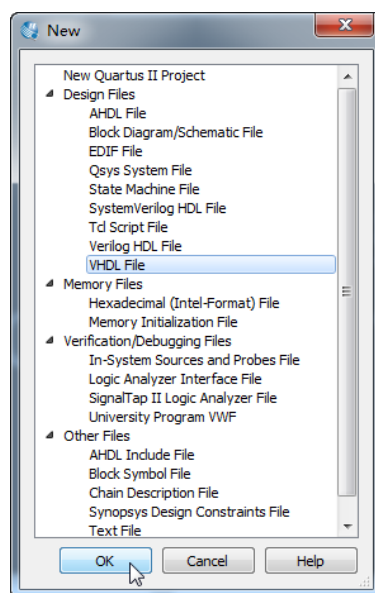
如果工程设置和代码输入正确，则会弹出“Analysis & Synthesis was successful”窗口，且在“Messages”窗口显示“Quartus II 64-Bit Analysis & Synthesis was successful. 0 errors, 0 warnings ...”，如下图所示。如果工程设置或代码输入有误，就要检查工程设置或代码输入，修改正确后再回到这一步重新执行“Analysis & Synthesis”命令，直到弹出“Analysis & Synthesis was successful”窗口。



五、为工程添加仿真文件及代码(vhdl testbench)，并成功能仿真（也称前仿真，RTL 仿真）

操作：鼠标左键单击“File”菜单，选择“New...”命令，在弹出的“New”窗口中选择“VHDL File”项后，

单击“OK”按钮。如下图所示。



在工作区会打开 VHDL 设计文件编辑窗口，然后在此窗口中输入以下代码。

```

---FPGA EXP1 @ 20170220 by lyh
library ieee;
use ieee.std_logic_1164.all;
entity FPGA_EXP1_tb is
end FPGA_EXP1_tb;
architecture arch_tb of FPGA_EXP1_tb is
    component FPGA_EXP1 is
        port(
            y0_n,y1_n,y2_n,y3_n,y4_n,y5_n,y6_n,y7_n :out std_logic;
            a,b,c :in std_logic
        );
    end component;

    signal y0_n,y1_n,y2_n,y3_n,y4_n,y5_n,y6_n,y7_n : std_logic;
    signal a,b,c : std_logic;

begin
    u_tb :FPGA_EXP1 port map(
        y0_n,y1_n,y2_n,y3_n,y4_n,y5_n,y6_n,y7_n,
        a,b,c);

    process
    begin
        a <= '0';
        b <= '0';
        c <= '0';
        wait for 20 ns;
        a <= '0';
        b <= '0';
        c <= '1';
        wait for 20 ns;
        a <= '0';
        b <= '1';
        c <= '0';

        wait for 20 ns;
        a <= '1';
        b <= '0';
        c <= '0';
        wait for 20 ns;
        a <= '1';
        b <= '0';
        c <= '1';
        wait for 20 ns;
        a <= '1';
        b <= '1';
        c <= '0';
        wait for 20 ns;
        a <= '1';
        b <= '1';
        c <= '1';
        wait for 20 ns;
        a <= '0';
        b <= '0';
        c <= 'X';
        wait for 20 ns;

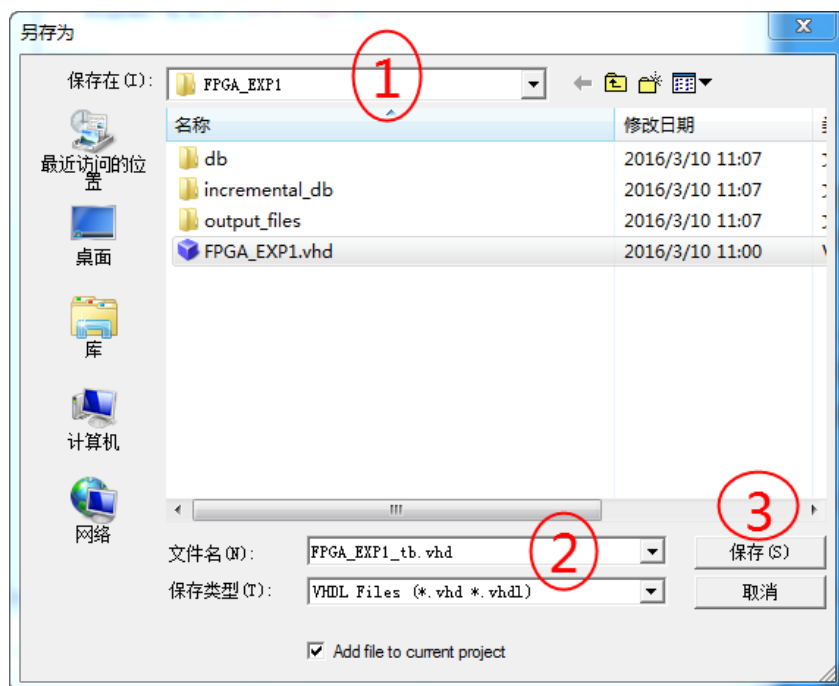
        a <= '0';
        b <= 'U';
        c <= '0';
        wait for 20 ns;

    end process;
end arch_tb;

```

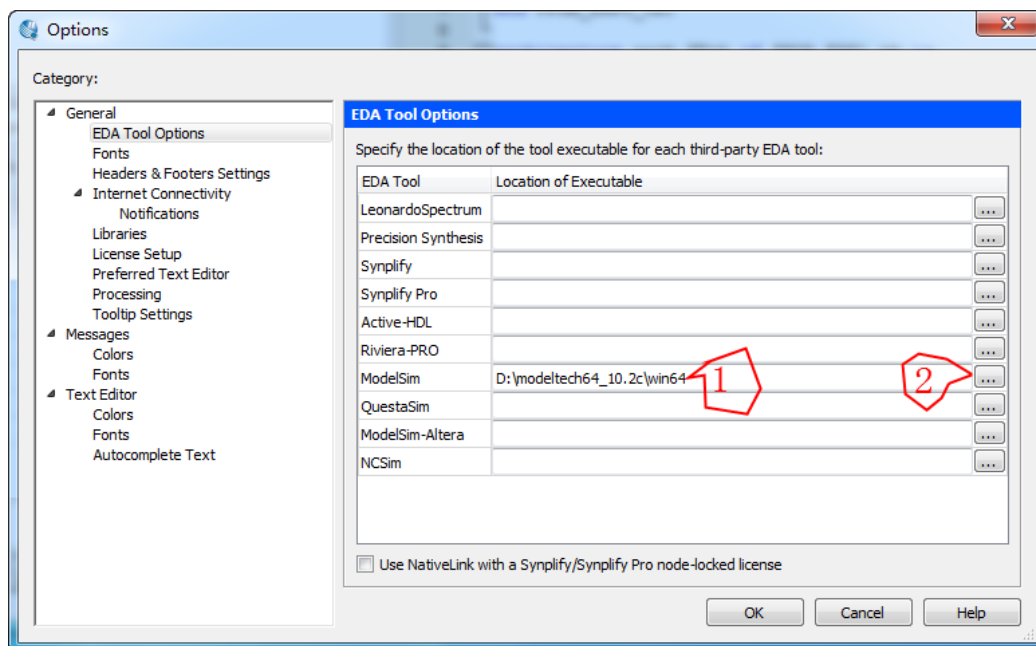
代码输入完成后，鼠标左键单击“File”菜单，选择“Save”命令，在弹出的“另存为”窗口中位置①处”

保存在“选择当前工程文件夹”FPGA_EXP1“、位置②处”保存类型”选择为“VHDL File”、文件名设置为“FPGA_EXP1_tb.vhd”，（如这些参数已经默认且一致，就不用重新设置了），最后单击“保存”按钮完成设计文件的输入。如下图所示。



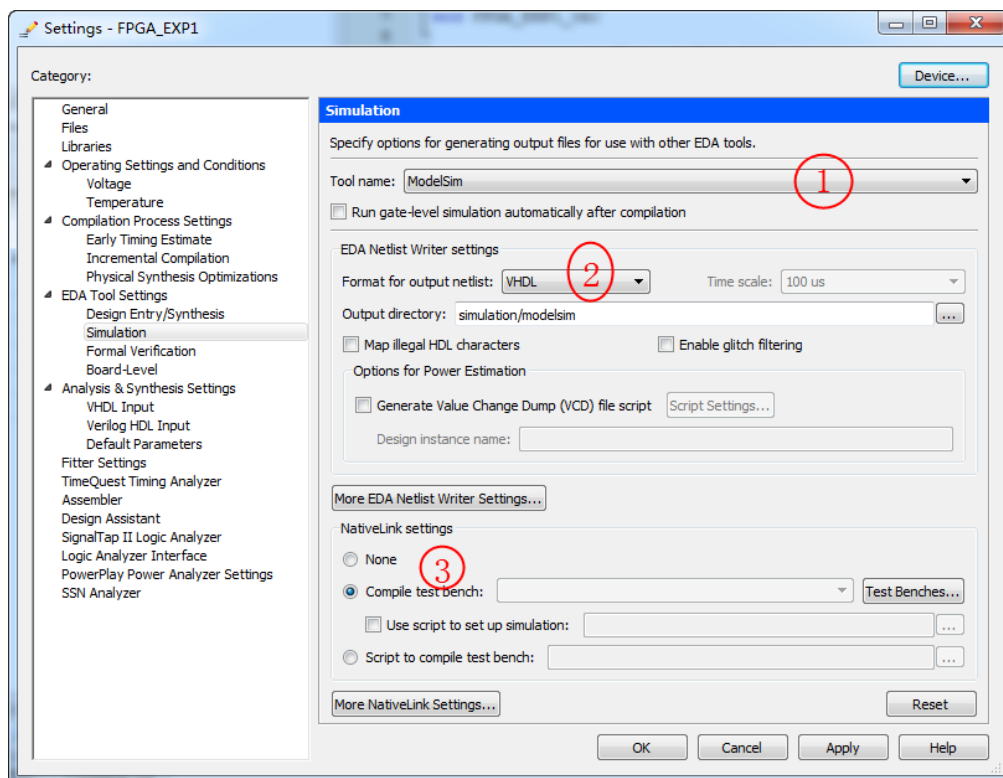
为了检查 testbench 代码的语法，请再次执行“Processing”菜单下“Start”子菜单中的“Start Analysis & Synthesis”命令。直至代码没有语法错误！

Modelsim 仿真工具的设置：单击“Tools”菜单，选择“Options...”命令，在弹出“Options”窗口的左侧单击“General”下的“EDA Tool Options”，然后按下图设置右侧 “EDA Tool Options”页面下的“ModelSim”之“Location of Executable”为“D:\modeltech64_10.2c\win64”。如果已经设置则跳过此设置。

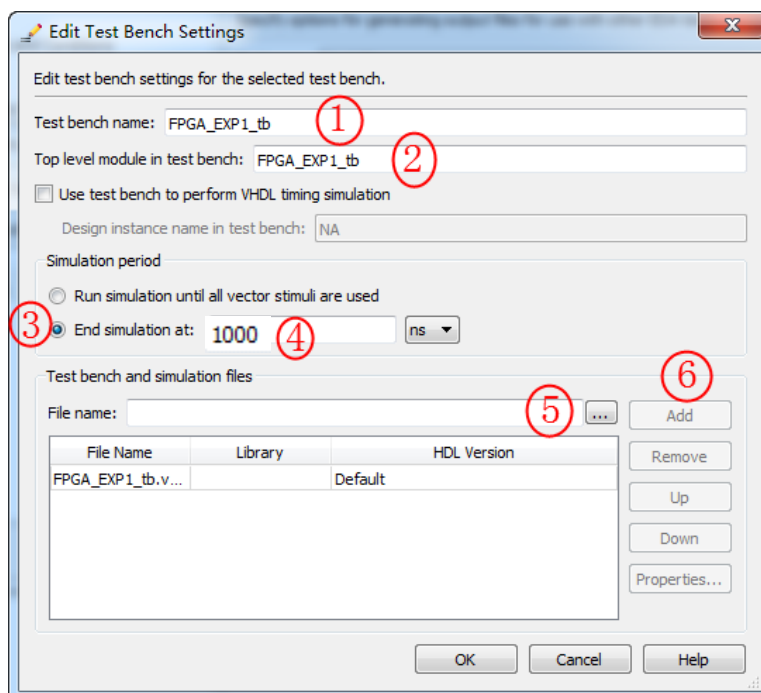


上图中可以在位置①直接输入，也可通过单击位置②的“...”按钮选择 ModelSim 工具所在文件夹。
功能仿真设置：单击“Assignments”菜单，选择“Settings...”命令，在弹出“Settings”窗口的左侧单

击“EDA Tool Settings”下的“simulation”，然后按下图设置右侧的“Simulation”参数。



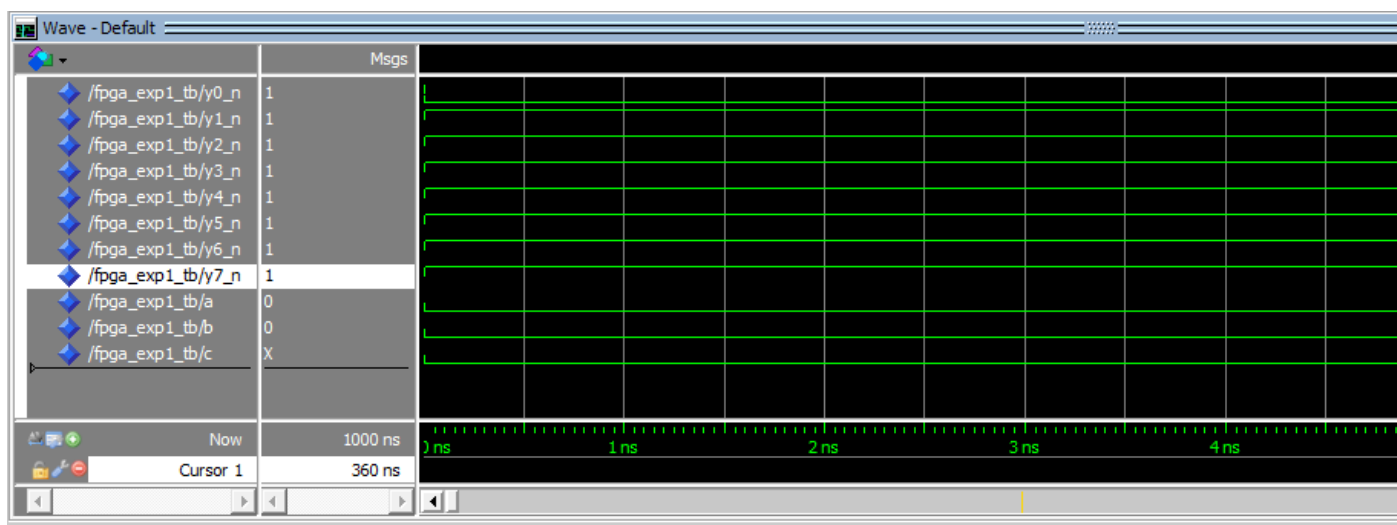
单击上图中③处最右侧的“Test Benches...”按钮，在弹出的“Test Benches”窗口中单击“New...”按钮，按下图设置“New Test Bench Settings”。



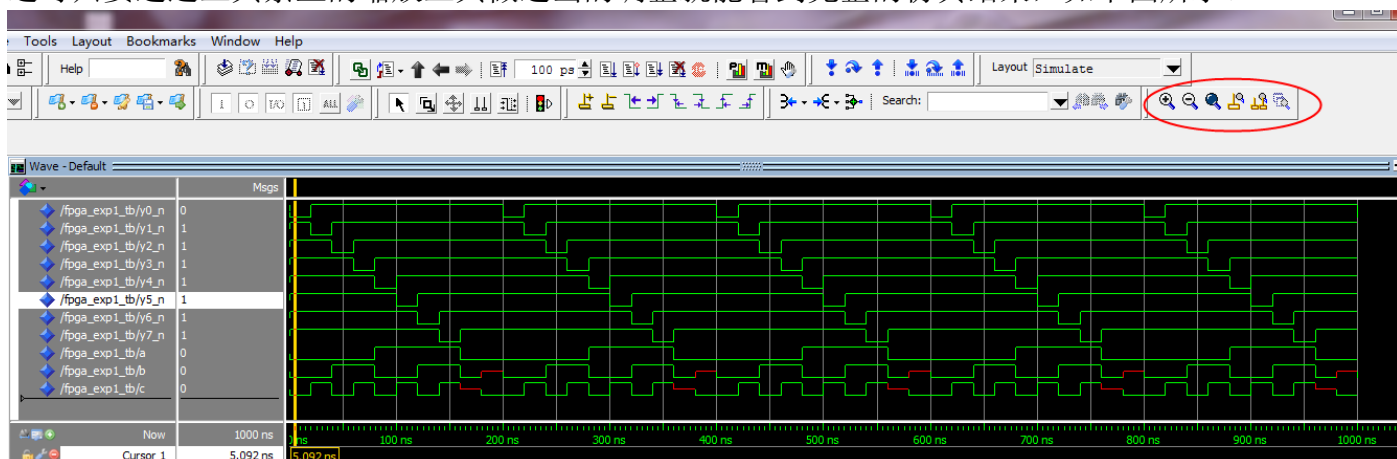
在上图位置①处输入“FPGA_EXP1_tb”，位置②会自动同步位置①的内容。在位置③，单选“End simulation at:”，并在位置④填上“100”同时选择单位为“ns”。再单击位置⑤后“...”按钮选择编辑好的“FPGA_EXP1_tb.vhd”文件，最后单击位置⑥处的“Add”按钮完成“Test bench and simulation files”的添加。

启动 ModelSim 完成设计功能仿真：单击“Tools”菜单，选择“Run Simulation Tool”子菜单中的“RTL

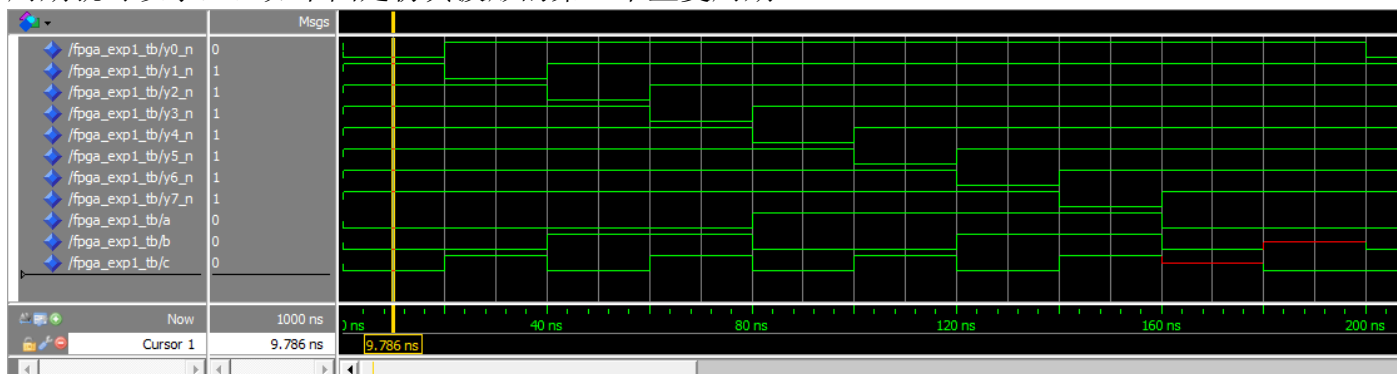
Simulation”命令来启动 Modelsim 实现功能仿真。等待 ModelSim 完全打开并跑完设定的 1000 ns，可以看到 Modelsim 的基本仿真界面及仿真结果，如下图所示。



从上图看，Modelsim 界面同一般 Windows 应用软件类似，拥有菜单栏、工具条、状态栏及各种功能窗口。在这里，我们比较关心的是 Wave 窗口中的仿真结果，而上图中没有看到完整的仿真结果。这时只要通过工具条上的缩放工具做适当的调整就能看到完整的仿真结果，如下图所示。



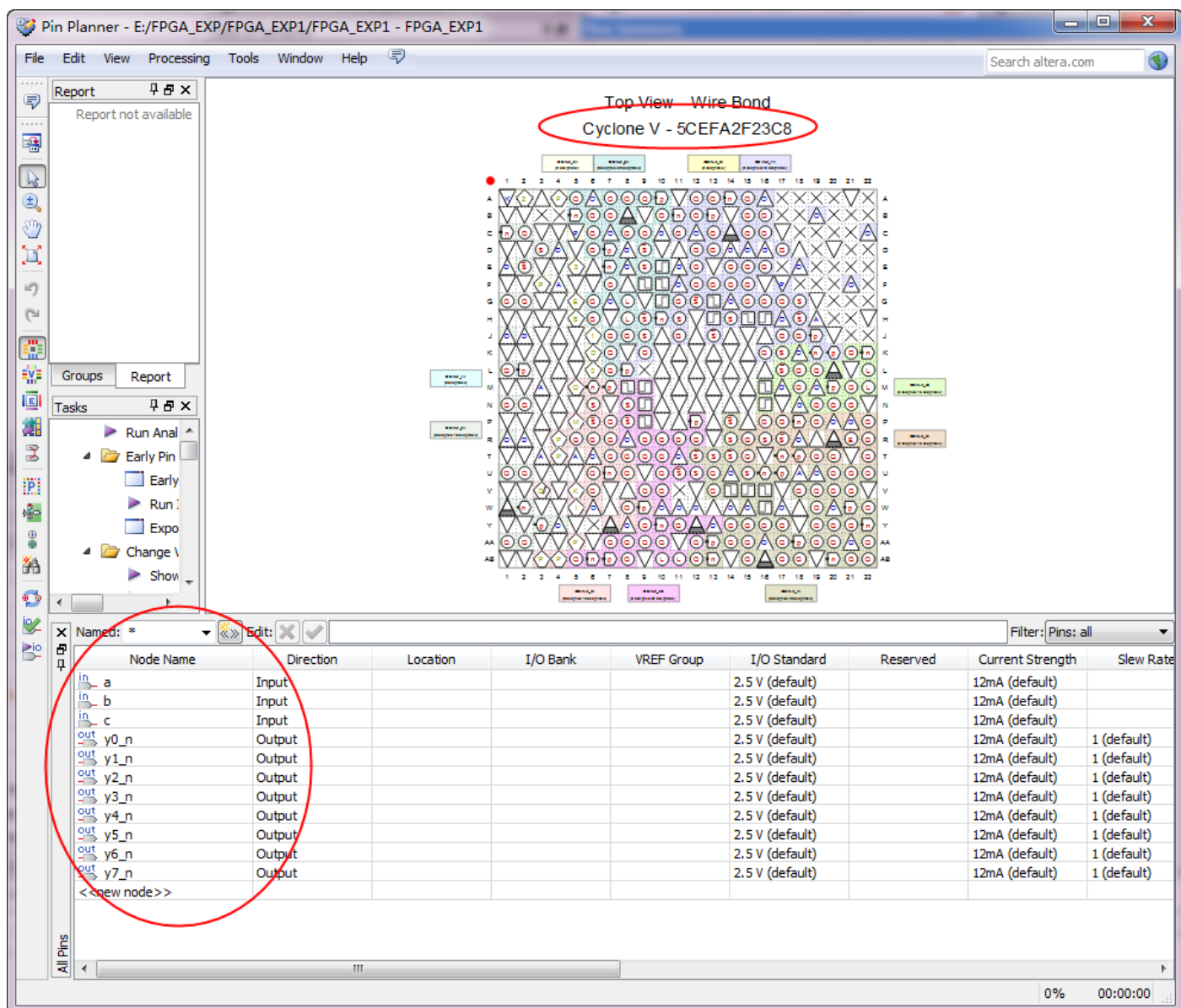
功能仿真结果分析：从上图中仿真结果可看出，仿真的结果是以 200ns 为周期不断的重复的，原因是编写 VHDL TestBench 时，没有在最后通过 wait 语句停下来。对于实验的仿真，只要分析任一个周期就可以了，比如下图是仿真波形的第一个重复周期。



在上图中 0 ns~20ns 输入 abc 为“000”，输出只有 y0_n 为低电平‘0’；20ns~40ns 输入 abc 为“001”，输出只有 y1_n 为低电平‘0’；接下来每过 20ns 依次只有一个输出为低电平，直到 160~180ns 和 180~200ns 因为输入不在译码范围而没有输出低电平。仿真结果验证了我们设计的译码器是正确的。

六、为设计工程添加约束：管脚分配

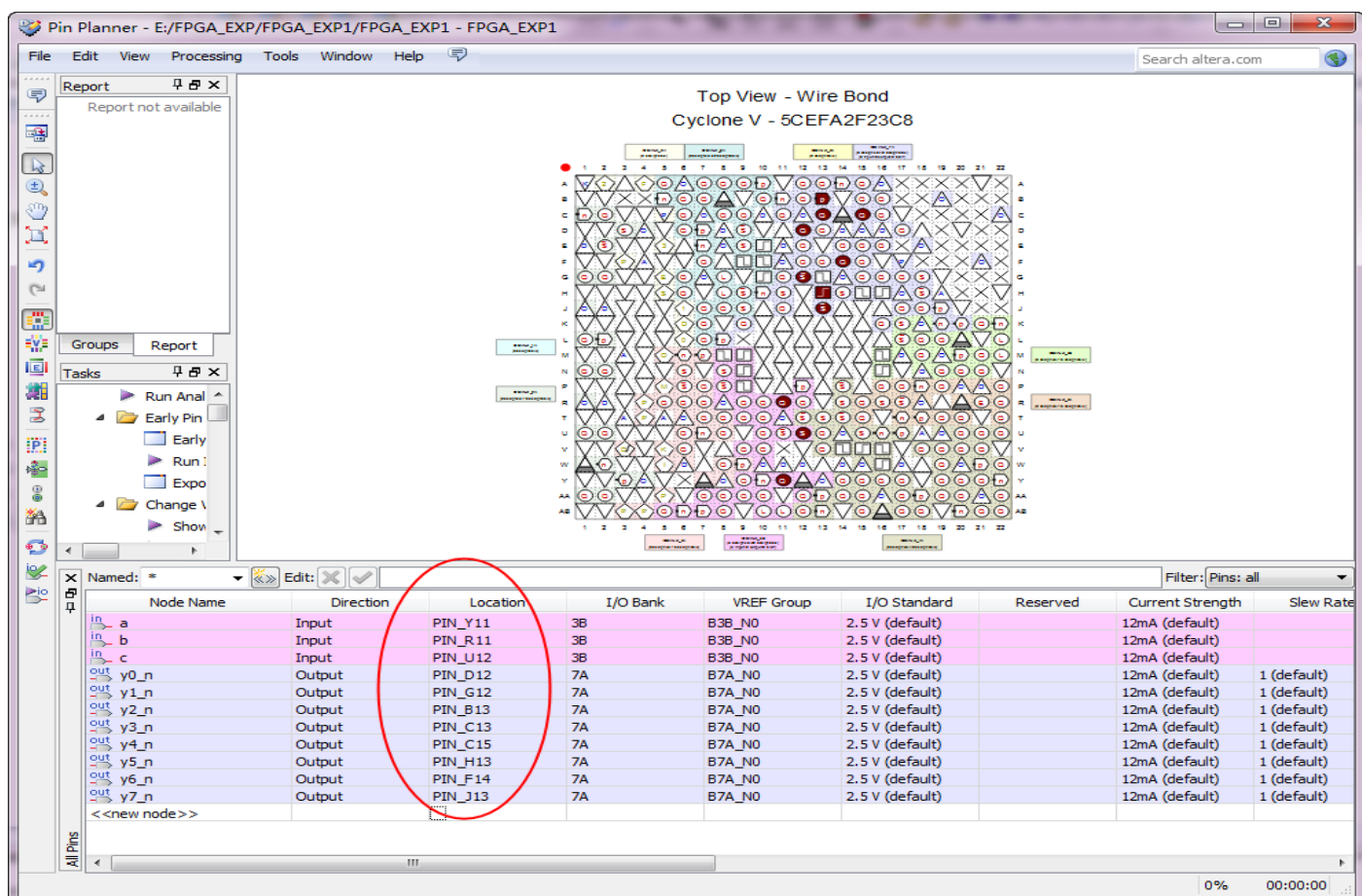
操作：单击“Assignments”菜单，选择“Pin Planner”命令打开“Pin Planner”窗口，如下图所示。注意下图中红色圈出地方与设计工程是否一致，不一致要修改一致后再往下做。



上图中，需要添加的约束是管脚（“Location”），也就是为 FPGA 内部设计中的输入和输出需要连接到对应的 FPGA 外部管脚，以便 FPGA 内部的设计与 FPGA 芯片外部的器件连接起来。为此我们通过查阅“实验指导书 altera_cyclone5.pdf”文档中的“附件一 FPGA 管脚约束”；分别为输入 a,b,c 找一个“拨码开关”对应的 FPGA 管脚，并在上图中输入信号对应的“Location”处直接输入相应的管脚编号；同样，为输出 y0/1/2/3.../6_n 找一个“LED 灯”对应的 FPGA 管脚，并在上图中输出信号对应的“Location”处直接输入相应的管脚编号。每次输入管脚编号后记得按键盘的“Enter”键，其他列参数不用处理，默认即可。管脚分配完成后的 Pin Planner 界面如下图所示。

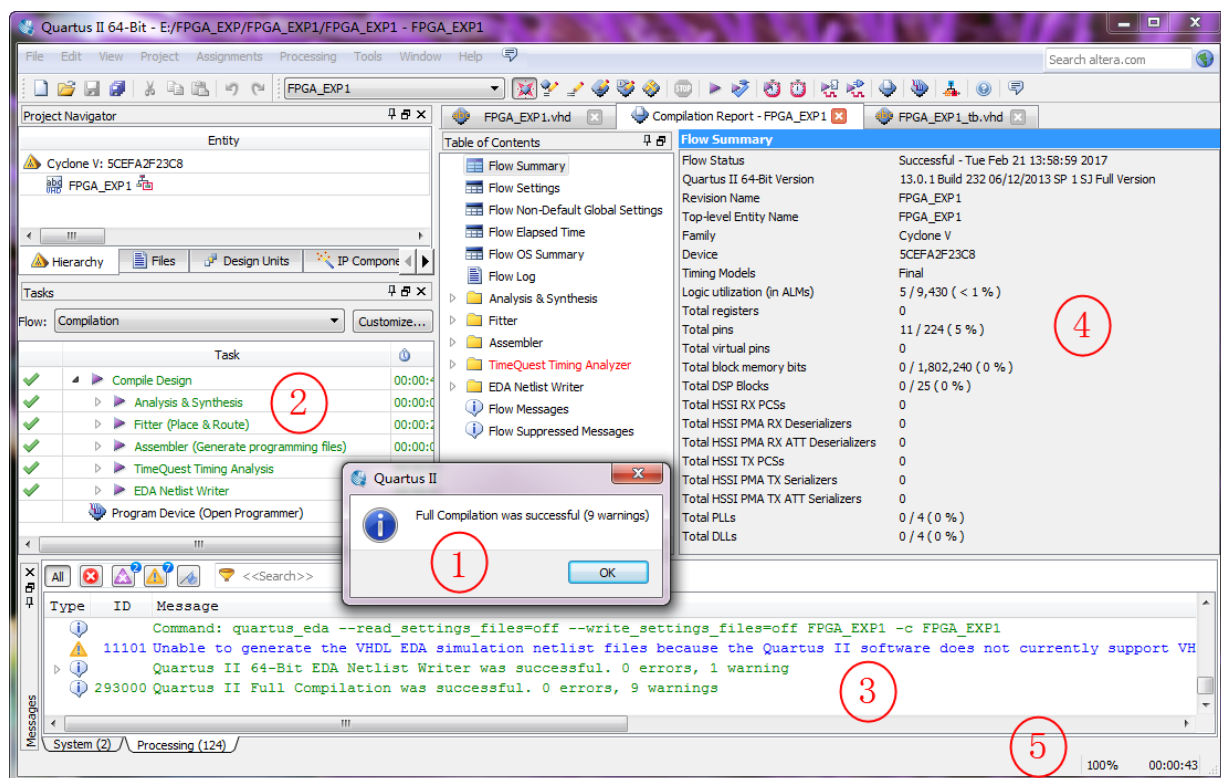
在此教程中，我们选择了 DIP0、DIP1、DIP2 作为设计中输入信号 a,b,c 的输入设备；选择了 D0~D7 作为设计中输出信号 y0/1.../7 的显示设备。实验时可以自行选择输入和输出设备为设计提供输入和显示。

在管脚约束完成以后，可以直接关闭“Pin Planner”窗口，下次修改或添加时再打开“Pin Planner”便可。



七、全编译设计工程：Analysis & Synthesis、Fitter、Assembler

操作：单击“Processing”菜单，选择“Start Compilation”命令，实现 Analysis & Synthesis、Fitter 及 Assembler 等完整的全编译过程。成功编译后会出现下图中的提示信息。



在上图中，位置①提示窗口说明全编译成功，但有 9 个警告，单击“OK”按钮后，可以在位置③查看详细的编译提示信息。位置②会在编译的过程中会指示编译进度及是否完全通过。位置⑤以百分比的方式指示编译的进度及编译过程所用的时间。位置④则是在编译正确完成后给出的各个阶段的信息，及摘要信息等。

八、硬件验证

1、硬件连接

首先在实验桌上找到 Altera FPGA 的实验箱，如下图所示。



然后把电源和 USB 数据线取下放在一旁，打开 FPGA 实验箱的盖子并放平，如下图所示。



接着用 USB 数据线（下图所示）连接 FPGA 实验箱和计算机（桌面上有 USB 延长线接口，下图所示）。连接 USB 数据线是一定要注意 USB 数据线接头的形状和方向，切不可胡乱蛮力连接。

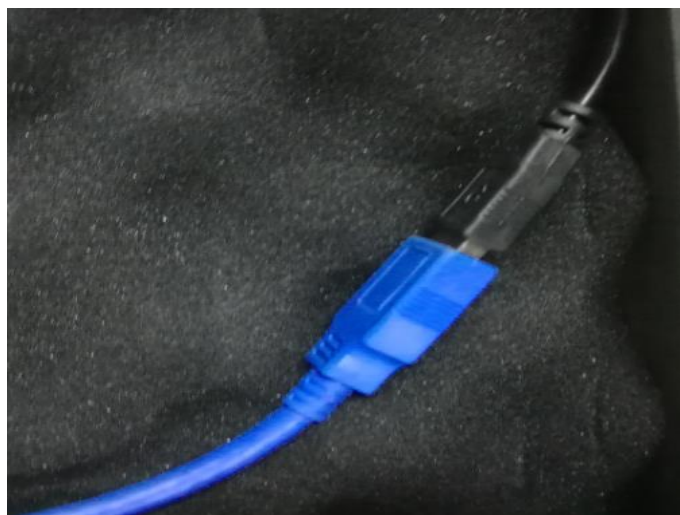
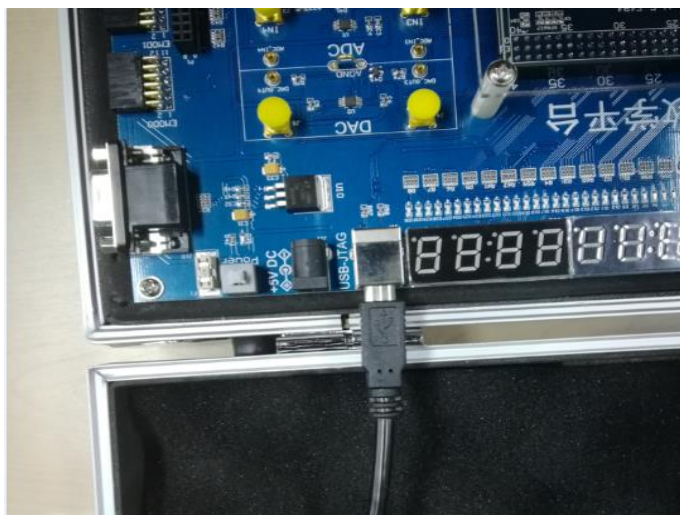


USB 数据线



USB 延长线接口

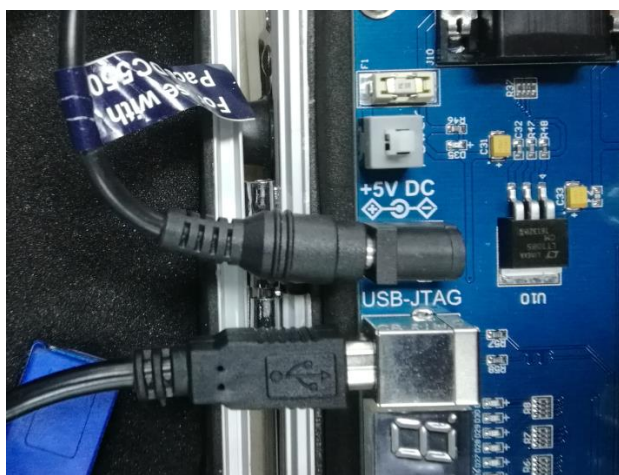
USB 数据线连接完成后的示意图如下所示。请注意 USB 数据线的连接位置。



最后连接电源(下图所示),将电源一端连接到 FPGA 实验箱的电源插座+5V DC(如下图所示),另一端连接到 220V 电源插头(如下图所示),**操作时一定要注意安全,或找老师协助**。电源连接完成后的示意图如上图所示。按下电源开关 K17(如下图所示),给 ALTERA FPGA 实验盒加电。



FPGA 实验箱外接电源

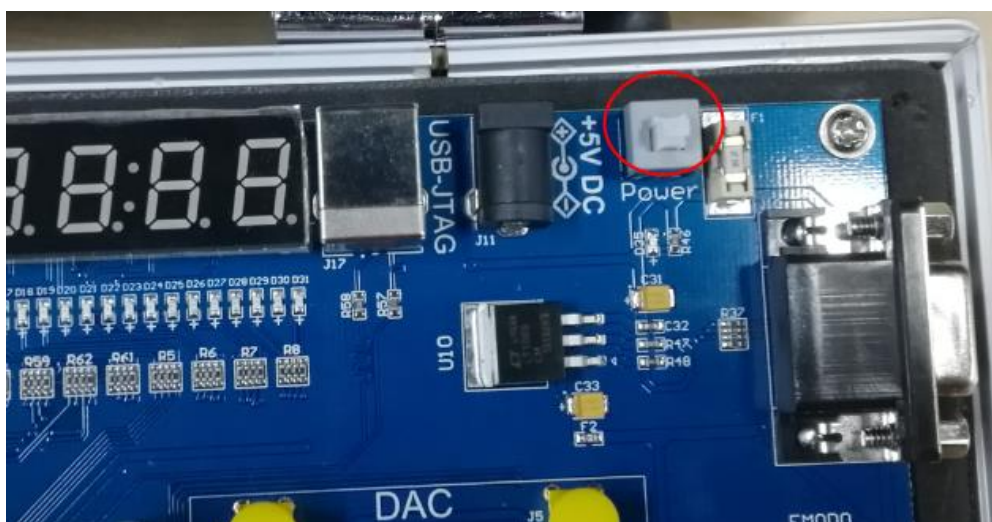


FPGA 实验箱电源连接接口及位置



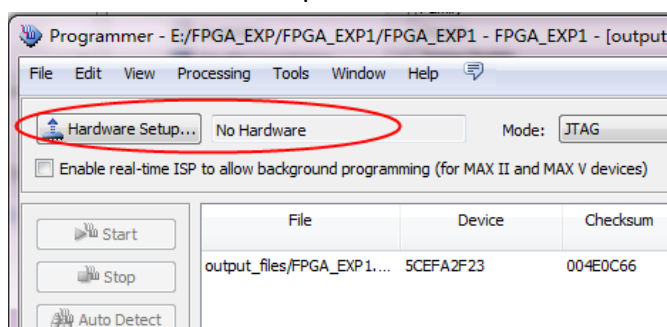
220V 电源接头及连接位置

最最后就要打开插线板开关，如上图所示；紧接着打开 FPGA 实验箱上的 POWER 开关（如下图所示）给 FPGA 实验箱加电。注意在首次给实验箱加电时要给 FPGA 实验箱的下载线安装驱动，此时请耐心等待驱动安装完成后，再往下做实验。

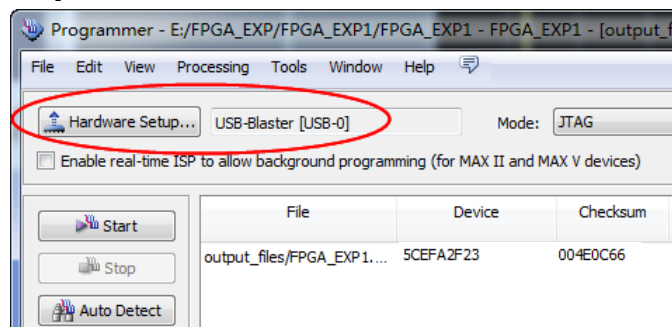


2、软件操作

单击“Tools”菜单，选择“Programmer”命令，在打开的“Programmer”窗口中如果“Hardware Setup...”按钮后显示“No Hardware”，则要单击“Hardware Setup...”按钮进行下载线设置；如果“Hardware Setup...”按钮后显示“USB-Blaster [USB-0]”，则不用设置下载线。如下图所示。

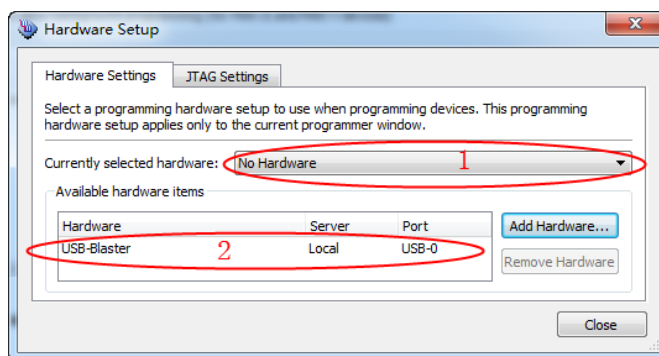


需要设置下载线

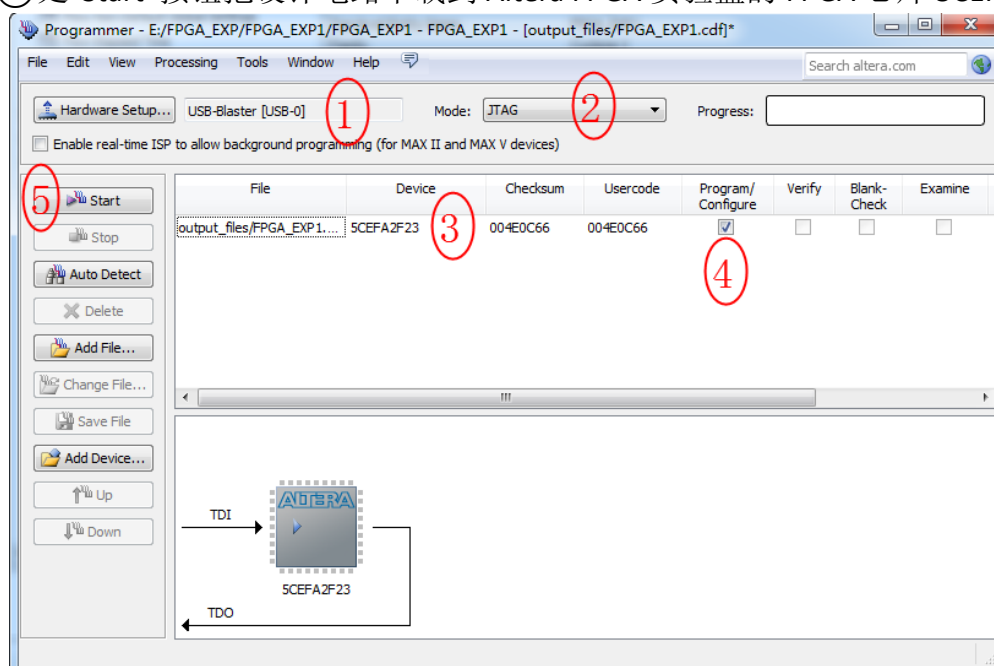


不需要设置下载线

单击“Hardware Setup...”按钮，在“Hardware Setup”窗口，通过位置①或位置②的下载线列表，来完成下载线的设定。如下图所示。

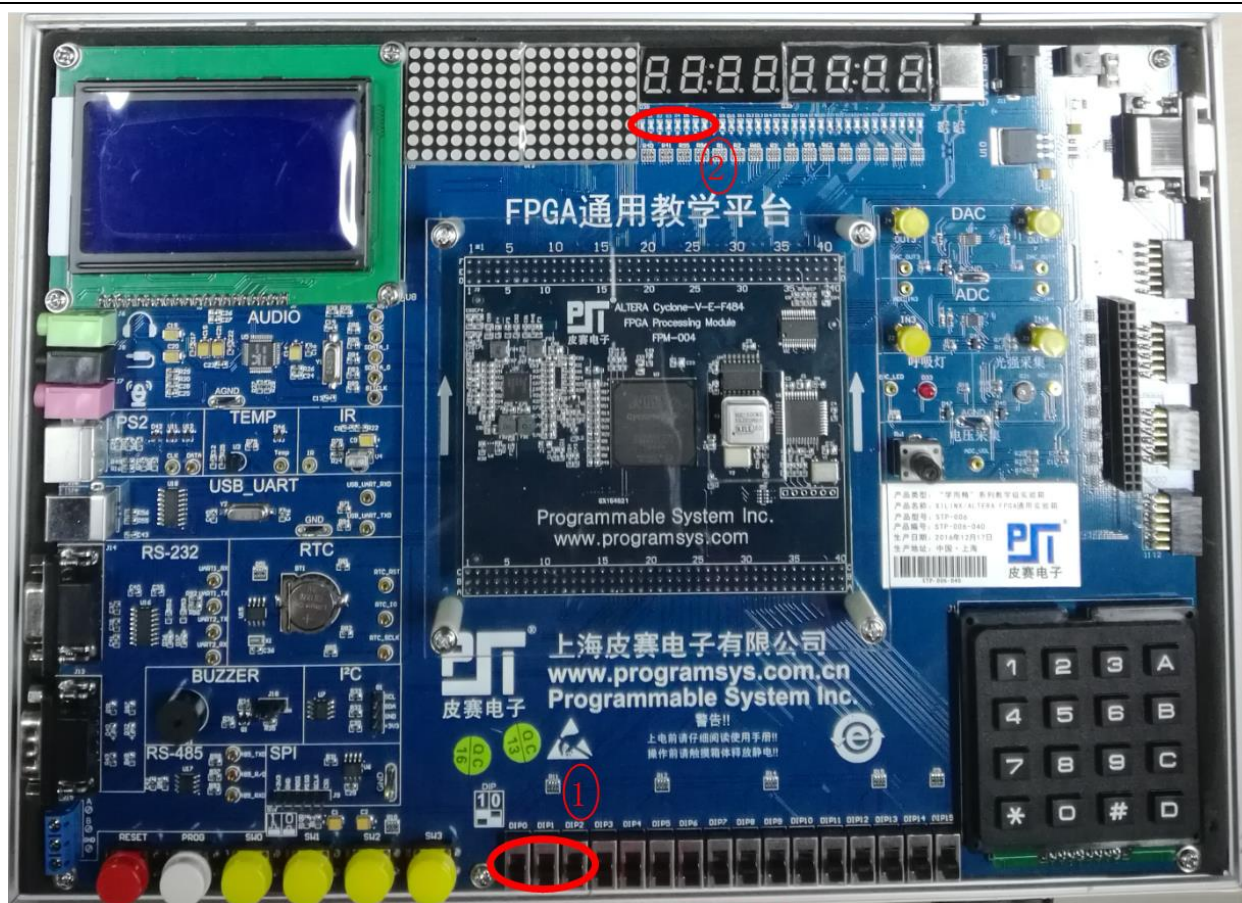


单击“Close”按钮完成下载线设置，返回“Programmer”窗口并核对下图①~④标记处信息，无误后单击⑤处“Start”按钮把设计电路下载到 Altera FPGA 实验盒的 FPGA 芯片 5CEFA2F23 中。



3、硬件操作：设计验证

根据之前为工程添加约束时的管脚分配：Altera FPGA 实验板上“拨码开关”中的 DIP0/1/2 分别通过 FPGA 的 Y11/R11/U12 管脚连接到了设计中的输入 a/b/c，而“LED 灯”中的 D0~D7 通过 FPGA 的 D12/G12/B13/C13/C15/H13/F14/J13 管脚连接到了设计中的输出 Y0/1/2/.../7_n。然后我们在实验板上分别找到 DIP0/1/2 和 D0~D7，如下图所示。



上图中位置①标出的拨码开关 DIP0/1/2，当 DIP0/1/2 拨向下（实验箱外侧）时表示相应的拨码开关给设计输入提供了低电平‘0’，反之则提供了高电平‘1’；上图中位置②标出的 LED 灯亮表示设计输出是高电平‘1’，反之则输出低电平‘0’。

接下来我们就要在操作 DIP0/1/2 的同时，观察 LED 灯 D0~D7 的状态，并记录最后硬件验证的逻辑关系如下表所示。

端口	对应器件	状态 1	状态 2	状态 3	状态 4	状态 5	状态 6	状态 7	状态 8
a	DIP0(Y11)	Down('0')	Down('0')	Down('0')	Down('0')	Up('1')	Up('1')	Up('1')	Up('1')
b	DIP1(R11)	Down('0')	Down('0')	Up('1')	Up('1')	Down('0')	Down('0')	Up('1')	Up('1')
c	DIP2(U12)	Down('0')	Up('1')	Down('0')	Up('1')	Down('0')	Up('1')	Down('0')	Up('1')
Y0_n	D0(D12)	Off('0')	On('1')	On('1')	On('1')	On('1')	On('1')	On('1')	On('1')
Y1_n	D1(G12)	On('1')	Off('0')	On('1')	On('1')	On('1')	On('1')	On('1')	On('1')
Y2_n	D2(B13)	On('1')	On('1')	Off('0')	On('1')	On('1')	On('1')	On('1')	On('1')
Y3_n	D3(C13)	On('1')	On('1')	On('1')	Off('0')	On('1')	On('1')	On('1')	On('1')
Y4_n	D4(C15)	On('1')	On('1')	On('1')	On('1')	Off('0')	On('1')	On('1')	On('1')
Y5_n	D5(H13)	On('1')	On('1')	On('1')	On('1')	On('1')	Off('0')	On('1')	On('1')
Y6_n	D6(F14)	On('1')	On('1')	On('1')	On('1')	On('1')	On('1')	Off('0')	On('1')
Y7_n	D7(J13)	On('1')	On('1')	On('1')	On('1')	On('1')	On('1')	On('1')	Off('0')

从上表中的数据可以看出，DIP0/1/2 和 D0/1/2/.../7 的逻辑关系完全符合我们设计的 3-8 译码器逻辑关系，也就是说我们的设计硬件验证完全正确。

至此，我们完成了 Altera FPGA 设计的基本流程和步骤。如果你结束了今天的实验，请备份实验数据提交实验记录，关闭所有打开的软件并关闭计算机；关闭实验盒的电源，并拆除所有相关的连线，

恢复到实验之前的整齐样子，别留下任何物品。O(∩_∩)O 谢谢！

