

实验 5 IP 调用实验

Advanced Electronics System Lab, 202309

一、实验目的

- 1、学习 Altera IP 核及 SignalTap II 工具的使用；
- 2、了解 FPGA 基本结构中的存储器；
- 3、熟悉 Verilog HDL Test Bench（测试平台/测试激励）的设计；
- 4、熟练掌握 Altera FPGA 的开发环境、设计步骤和流程。

二、实验内容

Altera 宏功能模块是复杂的高级构建模块，可在 Quartus II 设计文件中与逻辑门和触发器等基本单元一起使用，方便用户设计。Altera 提供的参数化宏功能模块和 LPM 功能均针对 Altera 器件结构做了优化，必须使用宏功能调用（MegaWizard）才可使用一些 Altera 专用的宏功能模块，如存储器、DSP 模块、PLL 等。本实验以一个 ROM 存储器为例介绍宏功能调用的基本步骤。

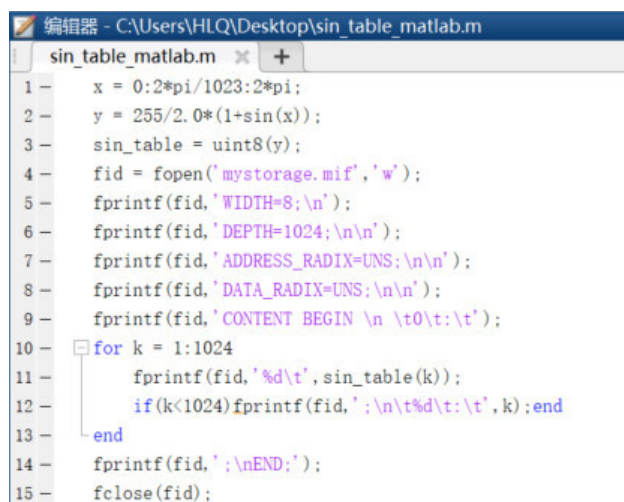
本实验的内容是首先调用一个 Altera 的宏功能模块 ROM，容量是 1024×8bit，并在 ROM 中存放正弦波一个完整周期的数据。再编写顶层实体，将 ROM 模块当作元件调用，实现一个正弦波产生器。最终的结果采用 Quartus II 中自带的嵌入式逻辑分析仪 SignalTap II 来观察。SignalTap II 的采样部件可以随设计文件一起下载于目标芯片中，采样部件可以捕捉目标芯片内部的信号，但不影响硬件电路的正常工作。

三、实验步骤

1、产生正弦波数据文件

利用 Matlab 产生一个周期的非负正弦波，然后对其采样，取 1024 个数据点，每个数据点用 8 位二进制表示，并将此数据文件存为 mystorage.mif。具体步骤为：

- 1) 打开 matlab；
- 2) 在 matlab 中，切换当前文件夹到 sin_table_matlab.m 所在的文件夹。
sin_table_matlab.m 文件如图 1 所示；
- 3) 打开 sin_table_matlab.m，并点击菜单上的按钮“运行”；
- 4) 最后会在当前的文件夹下产生存储器初始化文件 mystorage.mif；
- 5) 也可以直接利用实验室提供的存储器初始化文件 mystorage.mif。



```

1  x = 0:2*pi/1023:2*pi;
2  y = 255/2.0*(1+sin(x));
3  sin_table = uint8(y);
4  fid = fopen('mystorage.mif','w');
5  fprintf(fid,'WIDTH=8;\n');
6  fprintf(fid,'DEPTH=1024;\n\n');
7  fprintf(fid,'ADDRESS_RADIX=UNS;\n\n');
8  fprintf(fid,'DATA_RADIX=UNS;\n\n');
9  fprintf(fid,'CONTENT BEGIN \n \t0\t:\t');
10 for k = 1:1024
11     fprintf(fid,'%d\t',sin_table(k));
12     if(k<1024) fprintf(fid,';\n\t%d\t:\t',k);end
13 end
14 fprintf(fid,';\nEND:');
15 fclose(fid);

```

图 1 sin_table_matlab.m 文件代码

2、创建一个新的 Quartus II 工程

按照“实验 1_Altera FPGA 开发入门”文档中的说明，新建一个 Quartus II 工程，工程名为 FPGA_EXP5。将存储器初始化文件 mystorage.mif 复制到这个新建的 Quartus II 工程文件夹下。

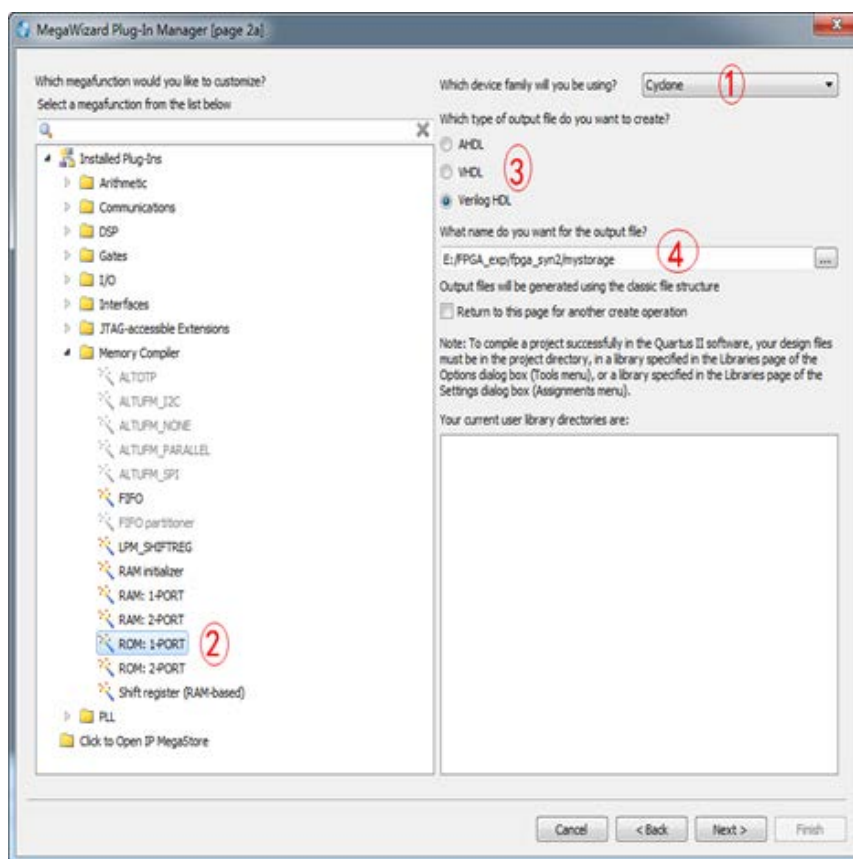
3、利用 MegaWizard 生成 ROM 或 RAM 存储器

在 QuartusII 中利用 MegaWizard 生成一个 1024*8bits 的存储器（RAM 或 ROM），并利用正弦波数据文件 mystorage.mif 来初始化此存储器。下面给出本实验中调用宏功能生成存储器的基本步骤。

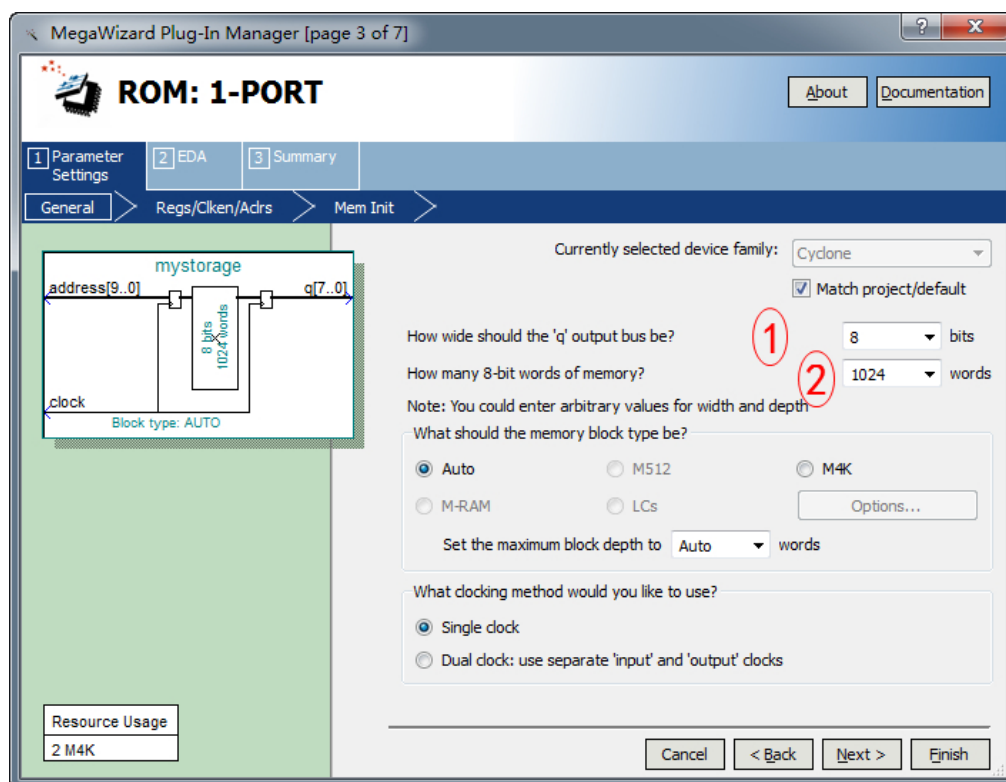
1) 启动 MegaWizard: 选择“Tools”菜单下的“MegaWizard Plug-In Manager”命令。

2) 在弹出的【page 1】窗口中选择“Which action do you want to perform”为“Create a new custom megafunction variation”。并单击“Next>”按钮进入下一步。

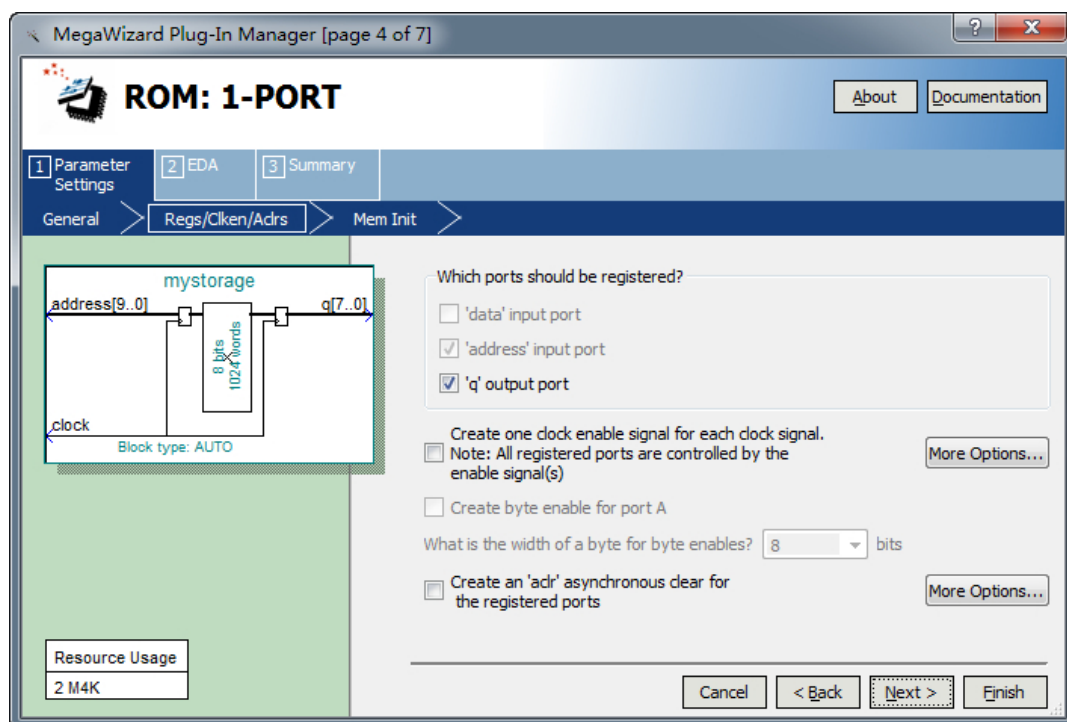
3) 在【page 2a】页面，要选择“Which device family will you be using?”为“Cyclone V”；“Which megafunction would you like to customize?”为“Memory Compiler”中的“ROM: 1-PORT”；“Which type of output file do you want to create?”为“Verilog HDL”；同时设置“What name do you want for the output file?”为“mystorage”，并保存到 Quartus II 的工程文件夹下（按工程文件夹实际路径填写），如下图所示。



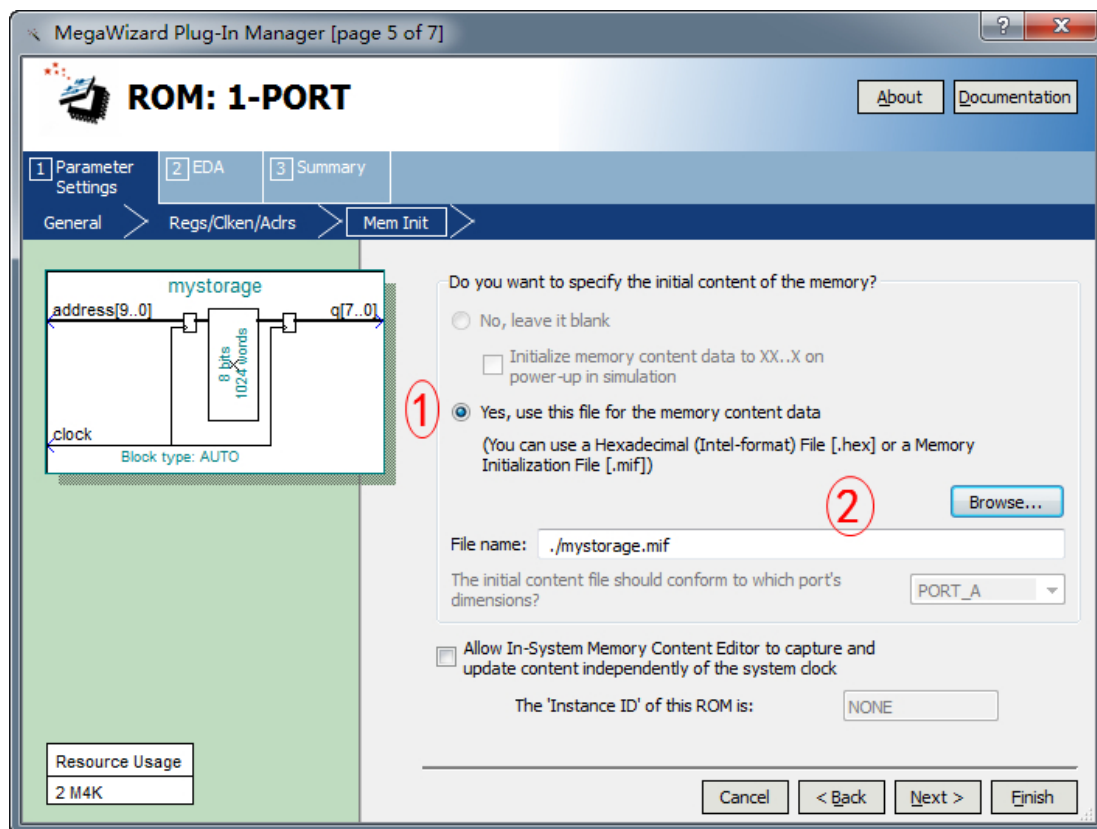
4) 在【page 3 of 7】页面，选择“‘How wide should the ‘q’ output bus be’”为“8”； “How many 8-bit words of memory”为“1024”； 其它默认设置。如下图所示。



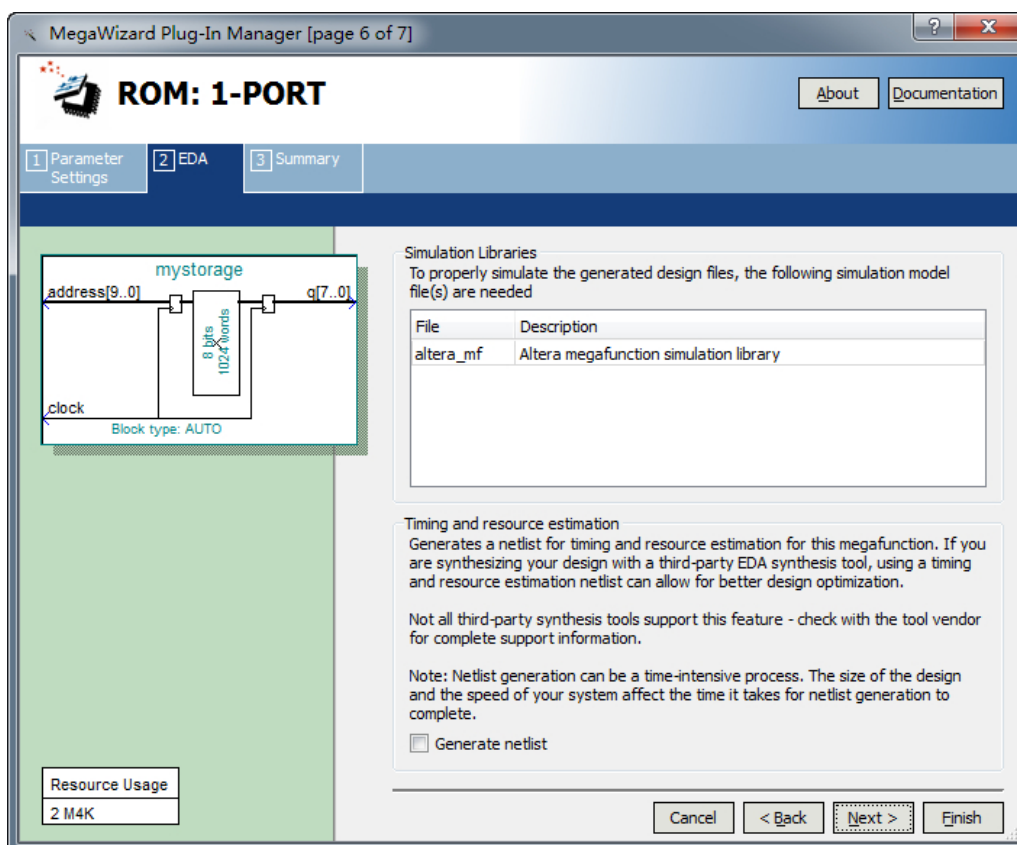
5) 在【page 4 of 7】页面，默认设置，如下图所示。



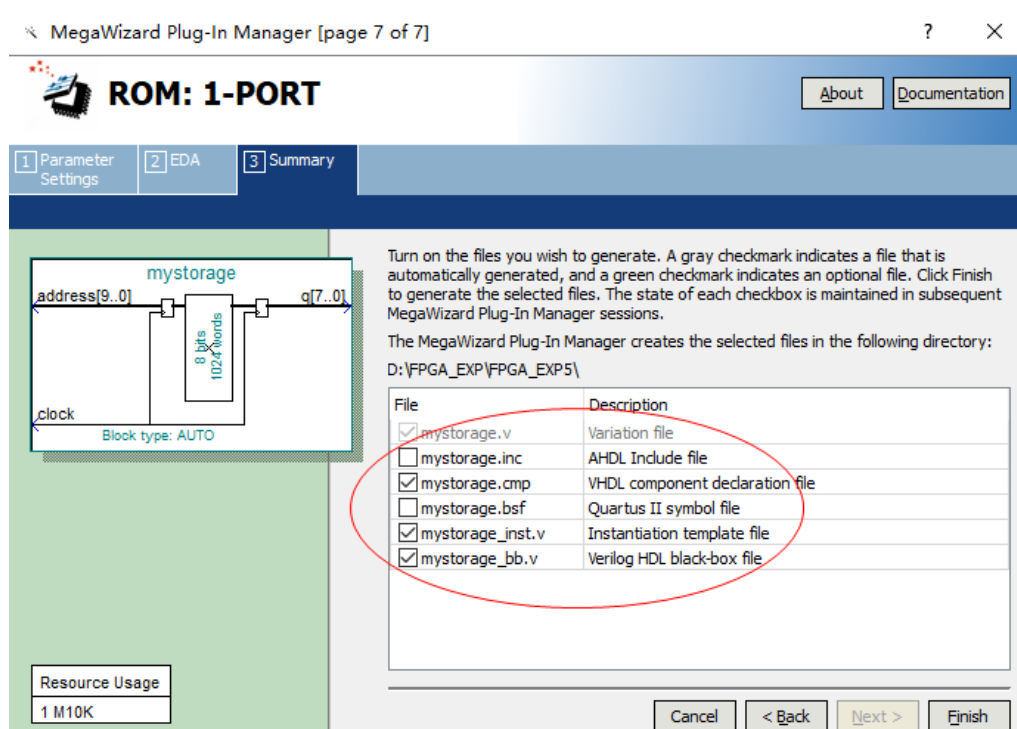
6) 在【page 5 of 7】页面，选择“Do you want to specify the initial content of the memory”为“Yes, use this file for the memory content data”，并通过“browse”按钮选择当前工程下的存储器初始化文件“mystorage.mif”，如下图所示。



7) 在【page 6 of 7】页面，默认设置，如下图所示。



8) 在【page 7 of 7】页面，可以选择生成相关的文件，如下图所示。点击“Finish”之后，会出现弹窗询问是否要将 ROM 加入到工程中，选择“Yes”，则可以在工程中看到 mystorage.qip，打开可看到文件 mystorage.v。

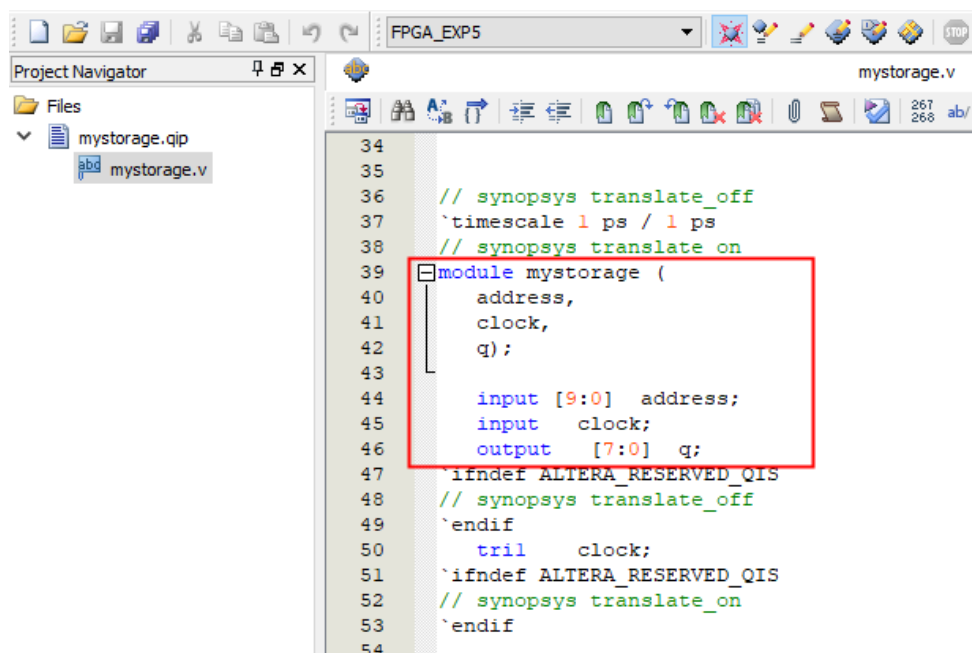


4、编写顶层模块的 Verilog 文件

编写顶层模块的 Verilog 程序，模块名为 FPGA_EXP5，与工程名一致。按一定的方式（如顺序）将存储器中的数据读出并输出到 D/A 端口。具体操作步骤如下：

1) 为工程添加顶层模块 Verilog 程序。模块中的输入端口有复位和时钟，输出端口为 8 位的正弦数据。

2) 将存储器 mystorage 作为元件，在顶层模块中进行调用：在“Project Navigator”窗口选择“Files”页面，展开“mystorage.qip”后，双击“mystorage.v”，可以打开“mystorage.v”文件查看 mystorage 模块的端口定义。如下图所示。



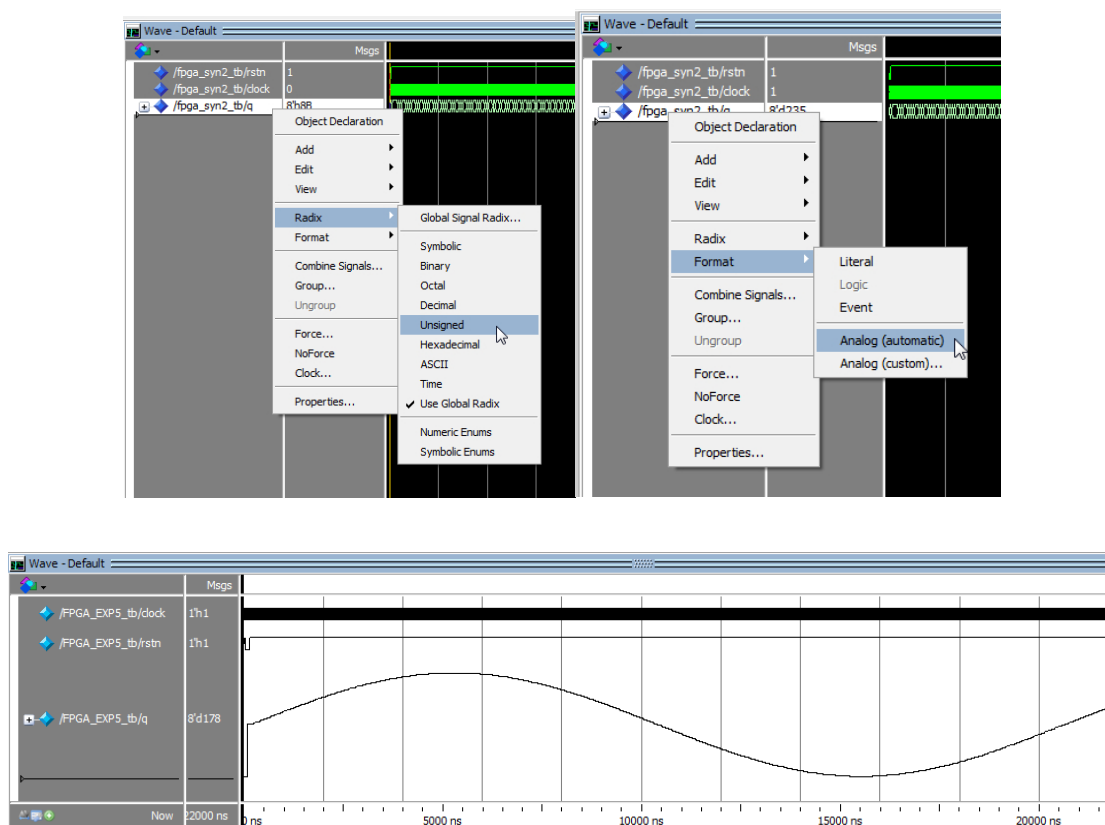
3) 在顶层模块中添加一个 always 语句设计一个计数器，计数值作为存储器 mystorage 的地址，以便 mystorage 存储器中存储的正弦波数据依次输出。

5、功能仿真

编写 FPGA_EXP5 的 Test Bench 文件，并设置相关参数后完成功能仿真。仿真时钟的周期用 20 ns，仿真的时间长度要大于 20 us，才能看到正弦波的一个完整周期。

对仿真结果可以采用模拟图的方式查看，具体设置方式为：

- ① 右键单击输出端口 q，选择 Radix 下的 Unsigned。
- ② 右键单击输出端口 q，选择 Format 下的 Analog(automatic)。



6、查看 RTL 级电路图

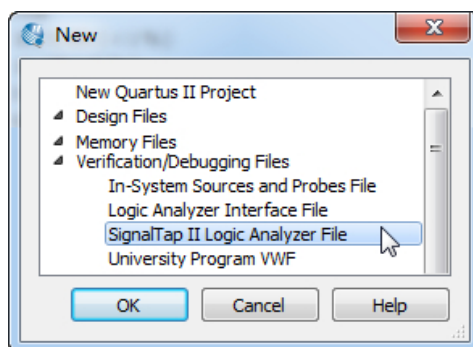
在仿真结束后,通过“Tools”菜单“Netlist Viewers”子菜单下的“RTL Viewer”命令查看设计电路的 RTL 级的电路结构。

7、管脚分配

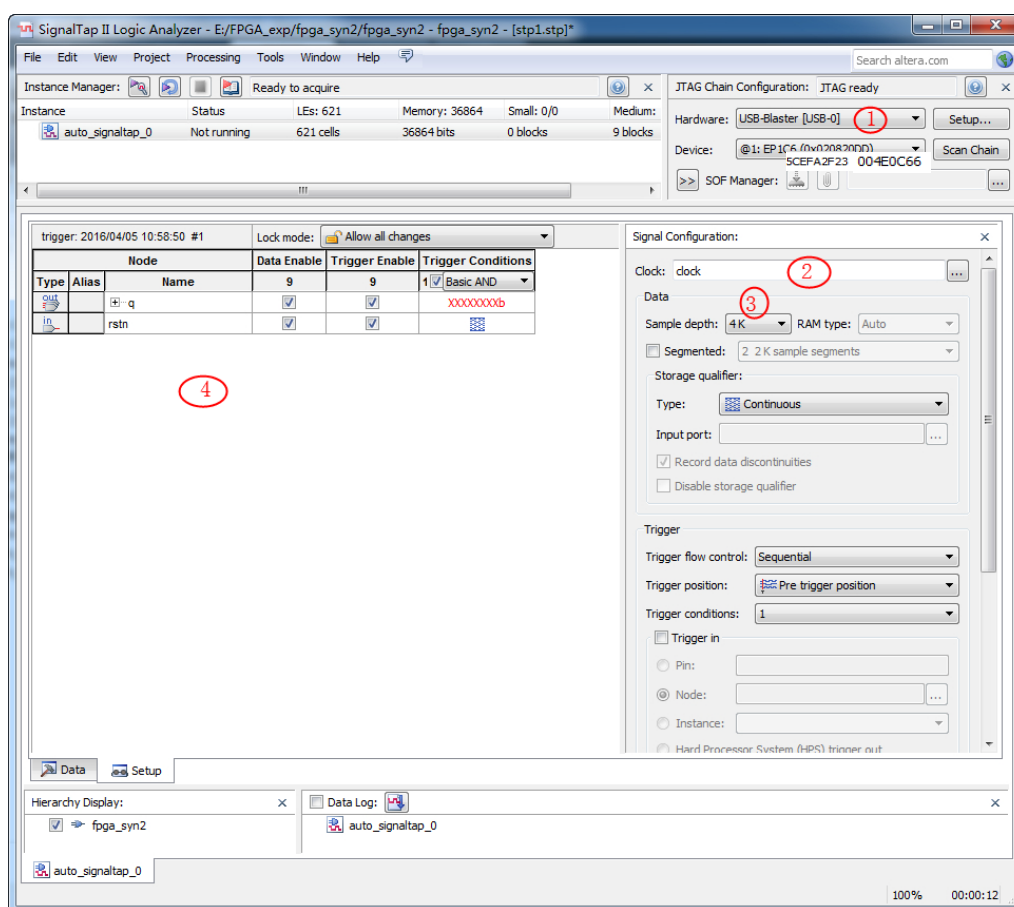
复位端口可以分配到拨码开关或者按键开关上,时钟端口分配到 50MHz 的外部时钟信号管脚,8 位输出数据分配到 EMOD0 的 EMOD0[0~7]上。

8、为工程添加 Signal Tap II 模块

- 1) 连接下载线,给 FPGA 目标板加电;
- 2) 添加 SignalTap II 模块:单击“File”菜单,选择“New...”命令,在打开的“New”窗口中选择“SignalTap II Logic Analyzer File”,进入 SignalTap II Logic Analyzer 窗口。



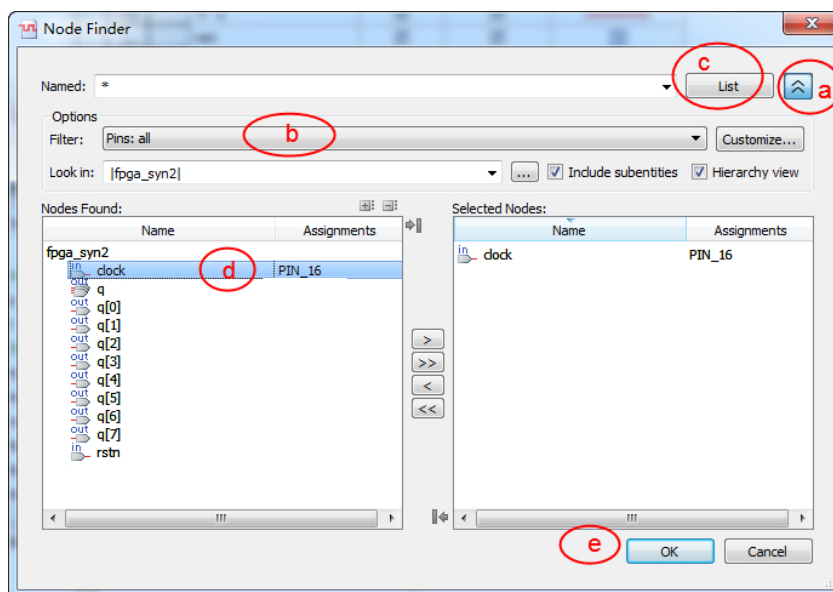
3) 设置 Signal Tap II



在上图的 SignalTap II Logic Analyzer 窗口中，需要做如下设置：

① 如果“Hardware”和”Device”后的参数正确就不用设置了，否则要通过”Setup...”按钮完成设置。

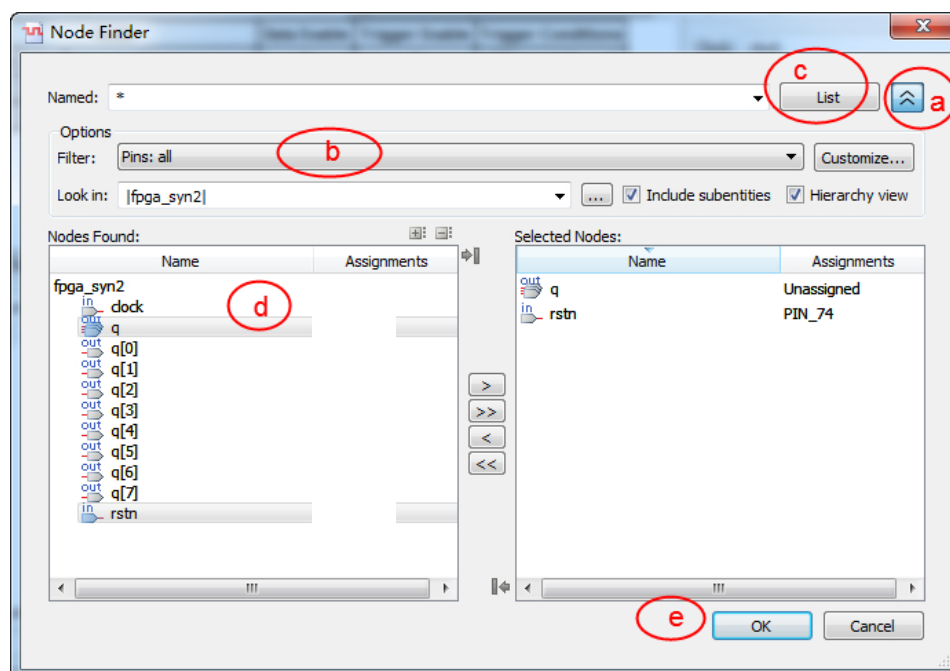
② 通过“Clock:”最右侧的”...”按钮设置 SignalTap II 的工作时钟为设计中的时钟。



在上图中先单击 a 处的按钮展开“Options”项，再在 Options 项中的 Filter 里选择“Pins:all”后，单击 c 处 List 按钮，然后在 d 处双击鼠标左键。最后单击“OK”按钮完成时钟的设置。

③ 设置 Sample Depth 为 4K，其它参数默认。

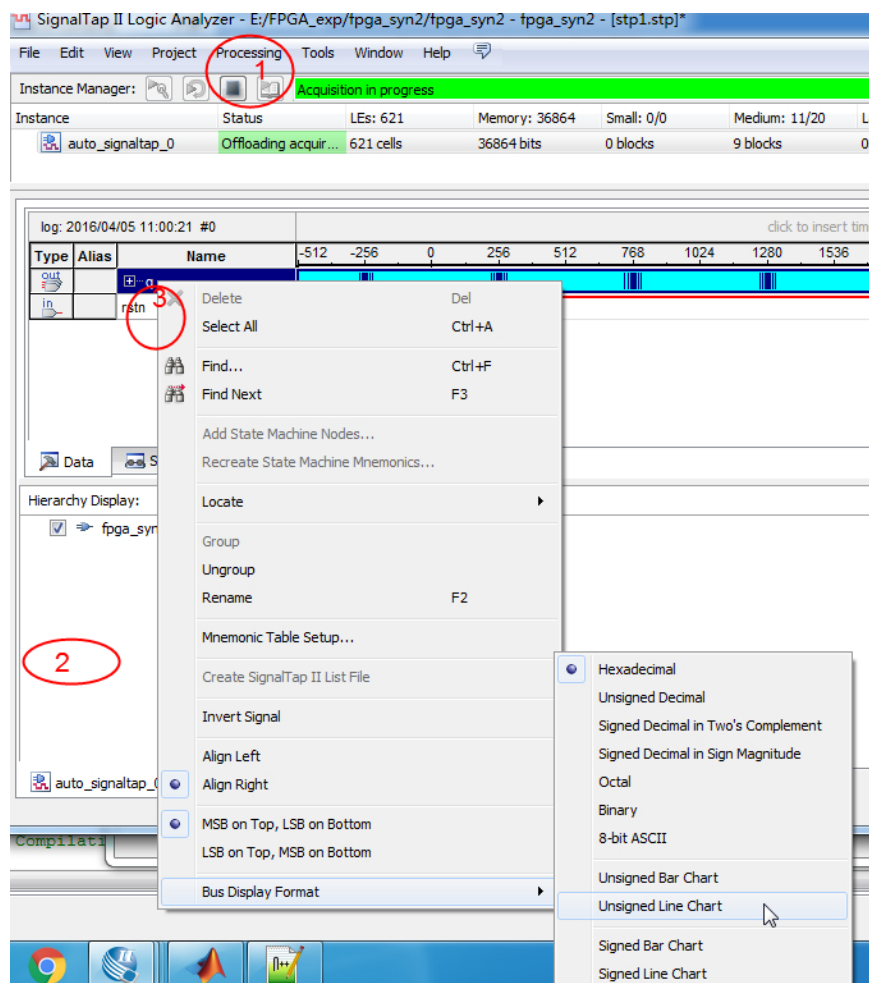
④ 在位置 ④ 处双击鼠标左键，在弹出的 Node Finder 窗口中，选择 q 和 rstn，方法同第 ② 步时钟的设置。如下图所示。



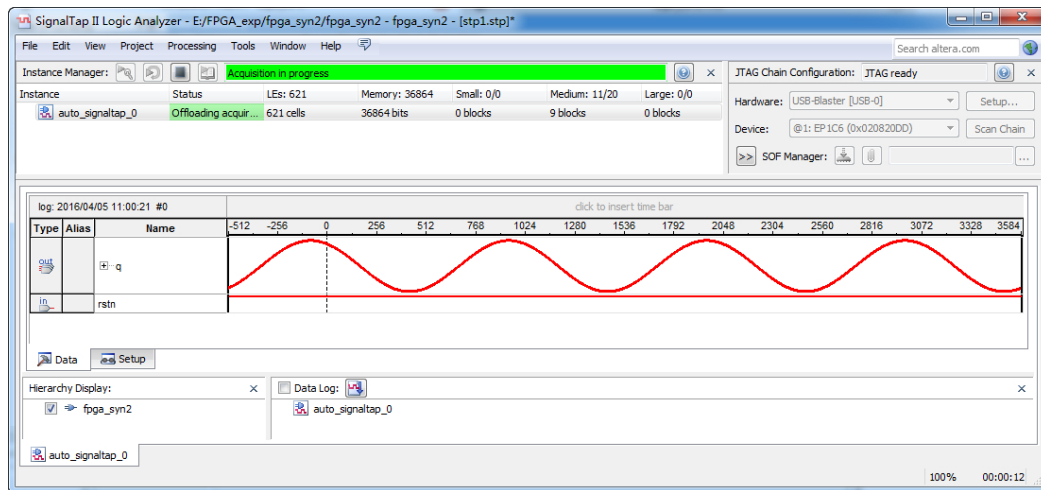
⑤ 最后通过“File”菜单中的“Save”或“Save As...”保存 SignalTap II 文件到当前工程文件夹下，并在弹出的“Do you want to enable SignalTap II File “文件名.stp” for the current project”窗口中，单击“Yes”按钮。

9、通过 signalTap II 查看硬件运行结果

- 1) 重新编译工程（Start compilation），生成配置文件；
- 2) 下载配置文件到目标芯片（Programmer）；
- 3) 单击 SignalTap II 界面中的“Processing”菜单下的“Autorun Analysis”命令，启动 SignalTap II。如下图所示；
- 4) 将“setup”页面切换的“data”页面，如下图中的位置②。
- 5) 在下图中的位置③处的输出 q 上单击鼠标右键，选择弹出菜单中“Bus Display Format”子菜单下的“Unsigned Line Chart”命令。



- 6) 最后可以通过 SignalTap II 看到设计的硬件输出结果为连续的正弦信号输出。如下图所示。



四、结束

至此，我们完成了 Altera FPGA 综合设计实验的流程和步骤。如果你结束了今天的实验，请备份实验数据和实验记录，关闭所有打开的软件并关闭计算机；关闭实验箱的电源，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！

