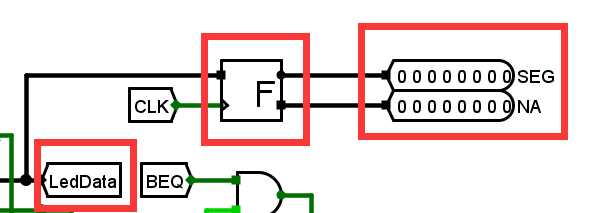
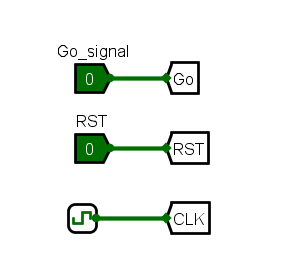
**Logisim转Verilog个人过程文档**

**一．Logisim使用部分**

1. 修改电路名称，保证命名符合规范（如果不符合规范会报错），一般来说只要不使用中文和特殊字符即可（空格可以出现）。
2. 删除多余的输入和输出引脚（包括辅助判断的标签和输入输出、数字显示管等部件，**以及mips probe器件也就是显示IR的灰色显示器件**）；增加新的输入输出引脚，输入引脚为RST（总复位）、Go（继续运行）以及时钟信号，需要注意的是时钟信号需要使用时钟源，RST和Go使用输入引脚或按键（笔者使用引脚）；输出引脚为SEG和NA（命名随意），如图，需增加器件FPGADigit作为输出中转。





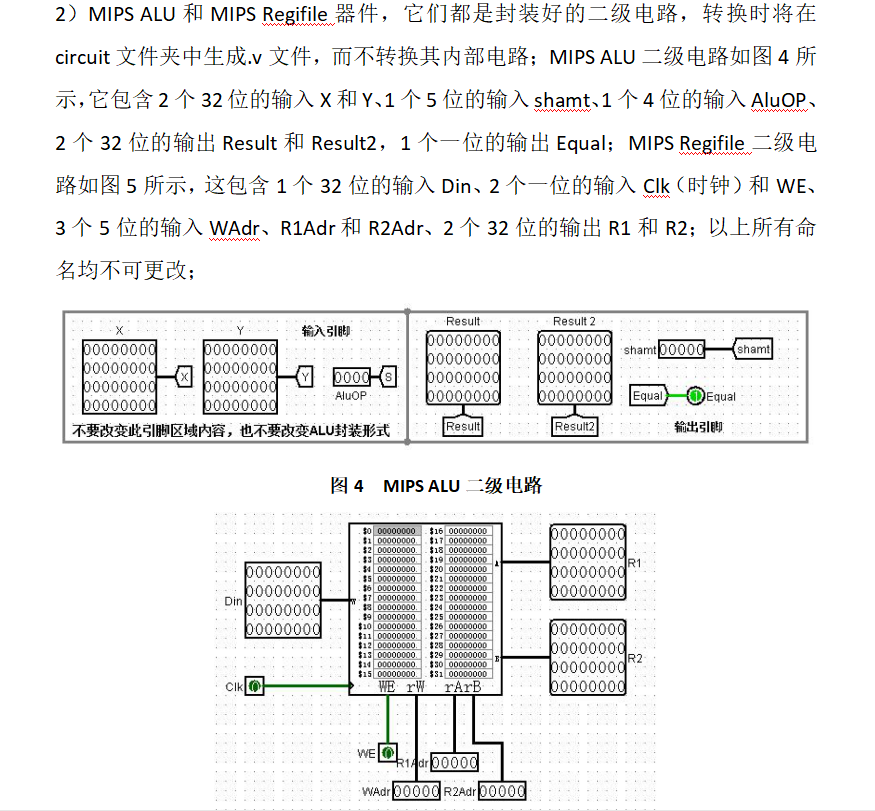
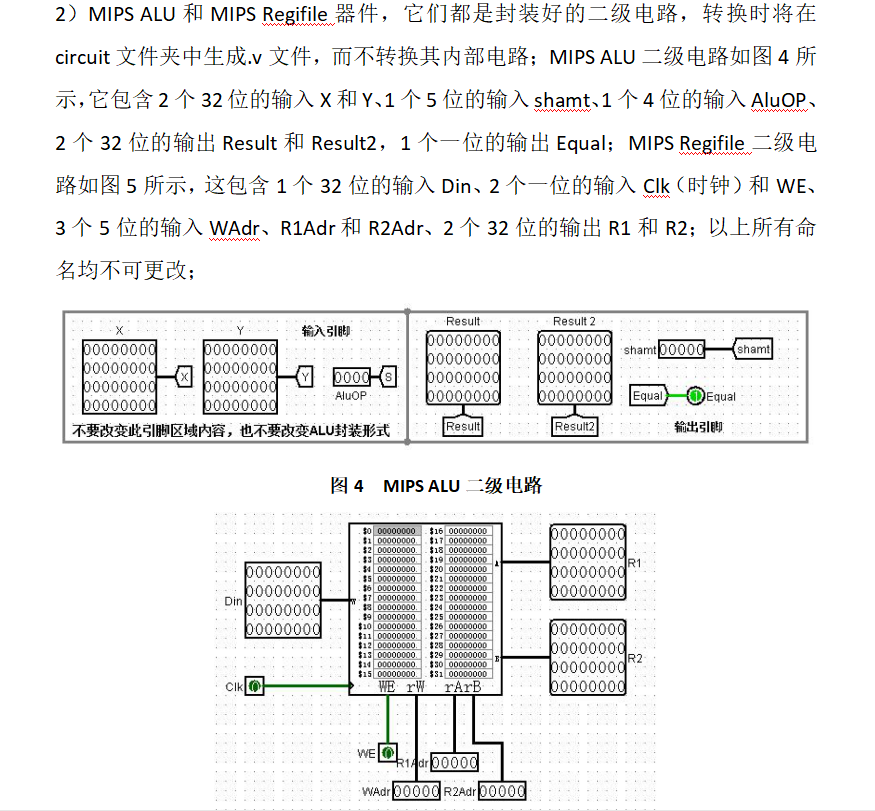
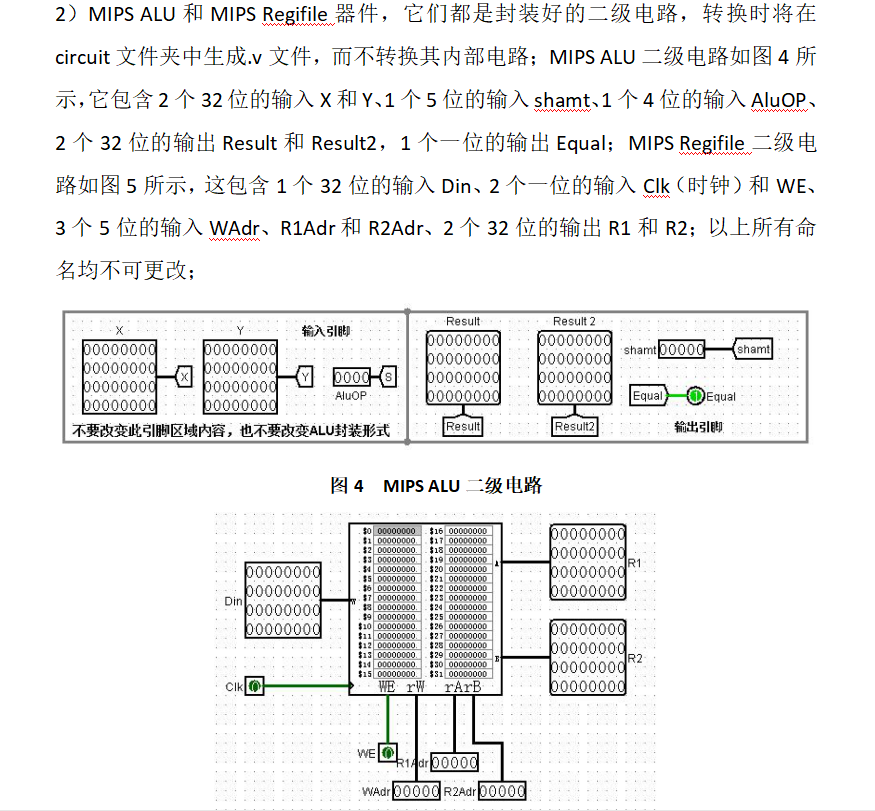
1. 修改新增加的引脚的名称，需要符合Verilog的命名规则，此外引脚的三态需要改成“否”，否则无法完成转化。



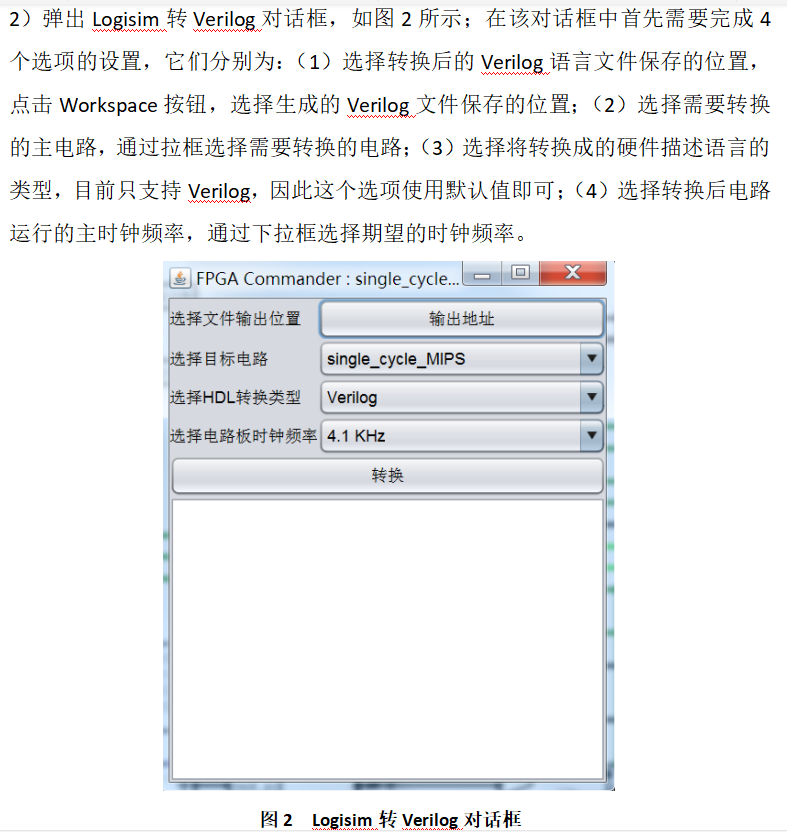
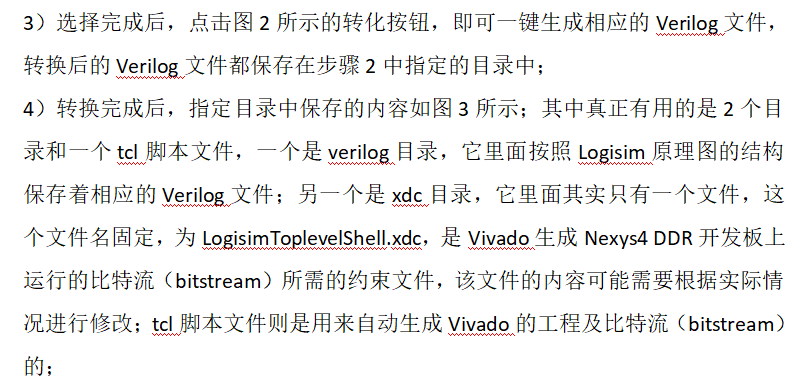
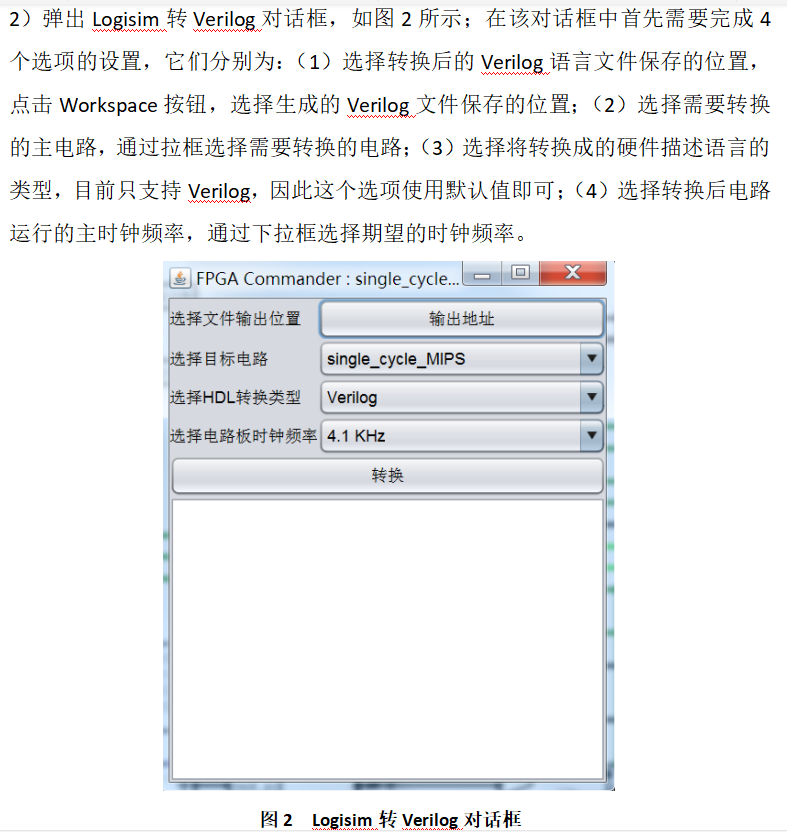
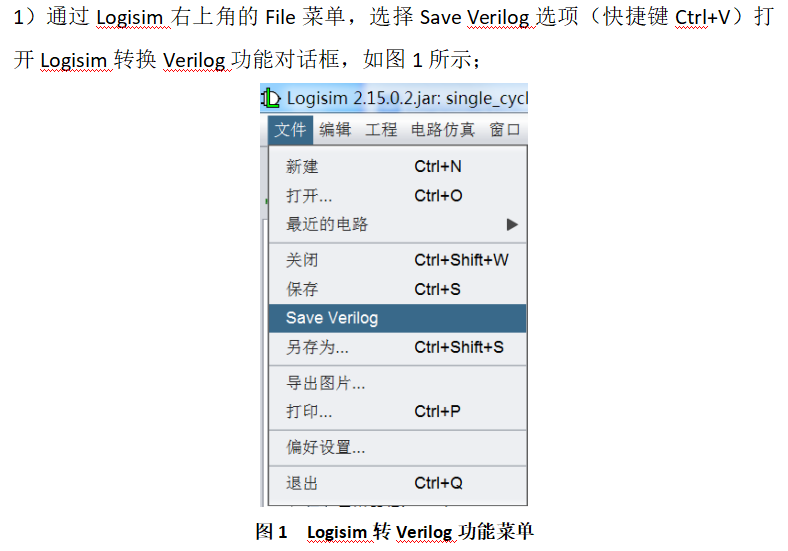
1. 修改所有的隧道标签名称，其命名需要符合Verilog的命名规则，建议直接改成xxx\_xxx的形式或者纯英文；并且，为ROM、RAM添加标签。
2. 保证所有的输入引脚不要悬空（笔者通过尝试发现，很多寄存器的使能端悬空仍可转换，但是仍然建议自行赋值，以免仿真时出现问题），**包括自己设计的控制器的输入引脚**（笔者卡在这个地方卡了很久）。
3. 修改所有的多路选择器的三态选项，选择禁用时输出0，包括其他具有三态功能的器件，包括所有的复用器（Plexers）器件都需要设置，**此外不得使用三态门。**



1. 按照要求修改MIPS\_Regifile电路中含有#符号的输出引脚名称。



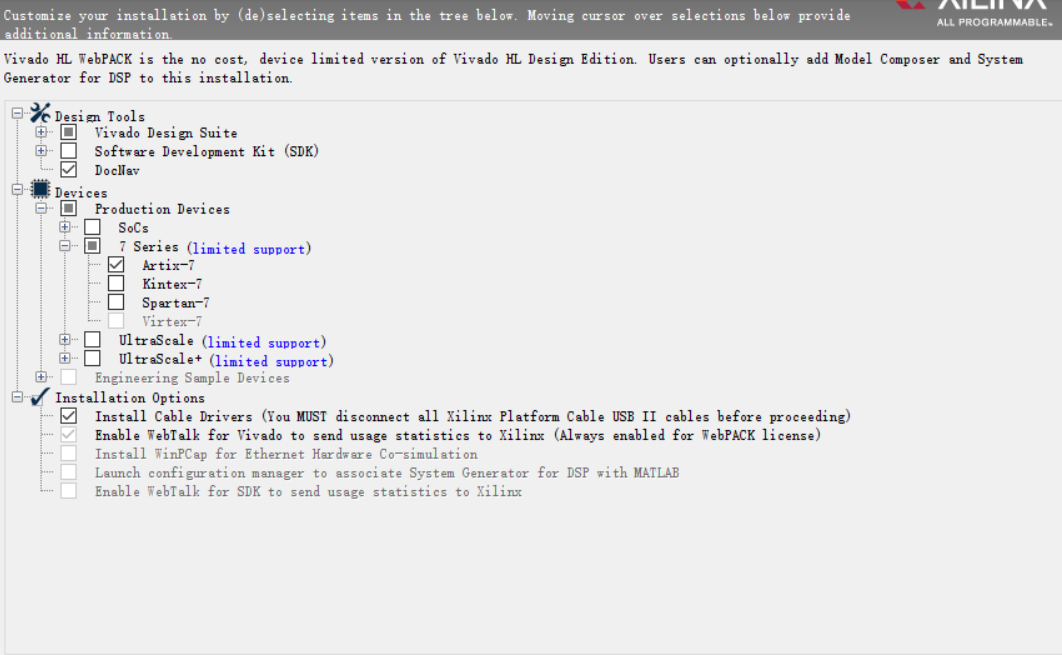
1. 按照如下步骤转化：



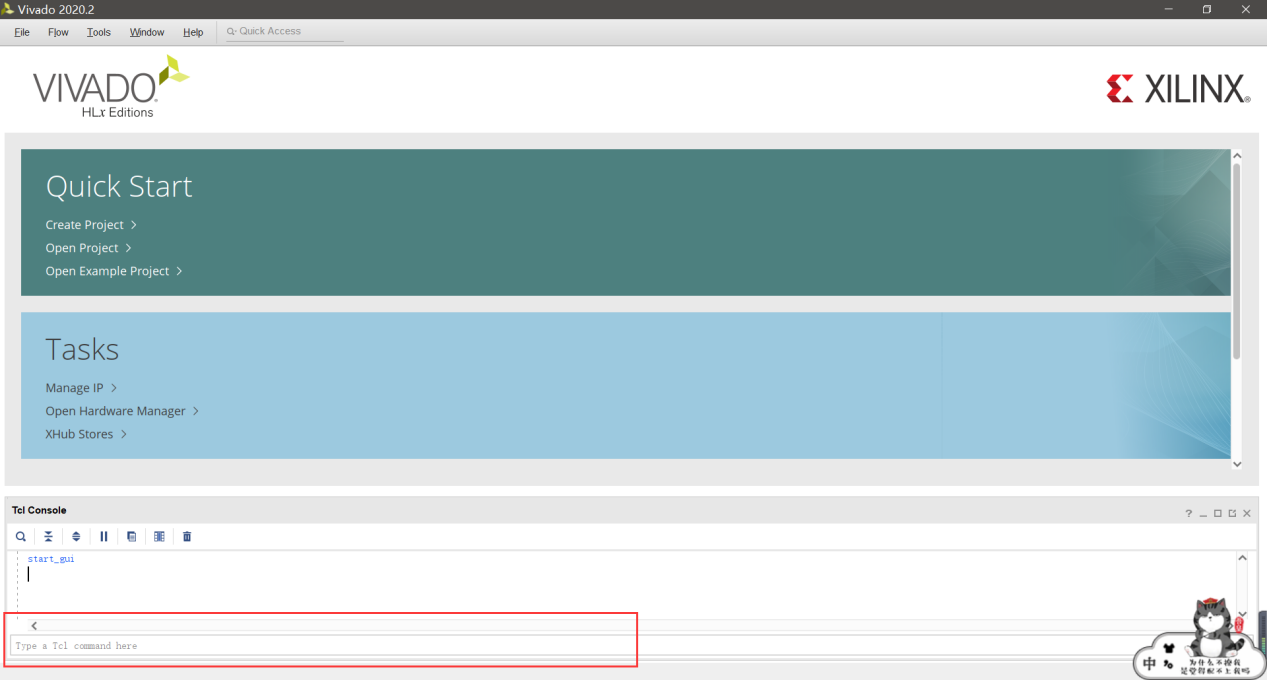
**一．Vivado使用部分**

1. 1. 安装vivado，（可找624的李响拿2020版的安装包，最新版的安装包大小较大共有36G大小，有需要可以找我复制，附2015版的百度云安装包4.8G：<https://pan.baidu.com/s/14wv9x7gs1mC6aTg377ZZAQ> 提取码：9h0c），安装组件如下，安装路径和使用项目路径不得存在中文和空格，具体配置如下：

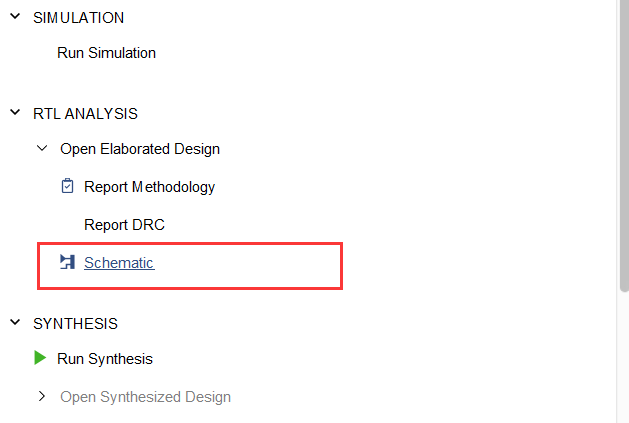
建议：在选择安装版本时，建议选择 Vivado HL WebPACK，组件选择如下：



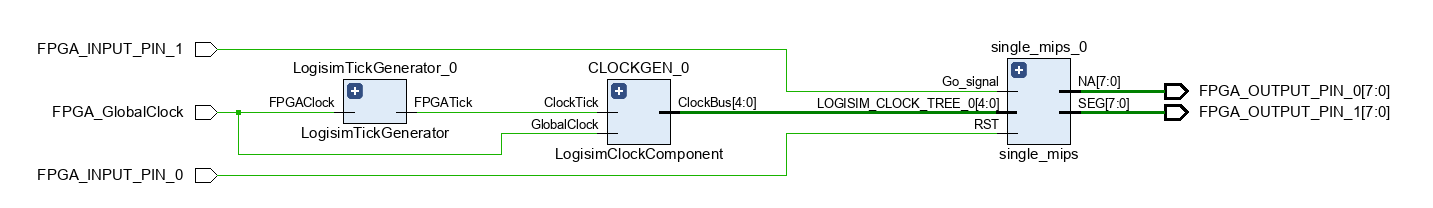
1. 打开vivado，在输入命令行的地方输入cd + .Tcl所在的路径名，如C:/xxx，然后输入source+.tcl的文件名，即可自动生成项目。

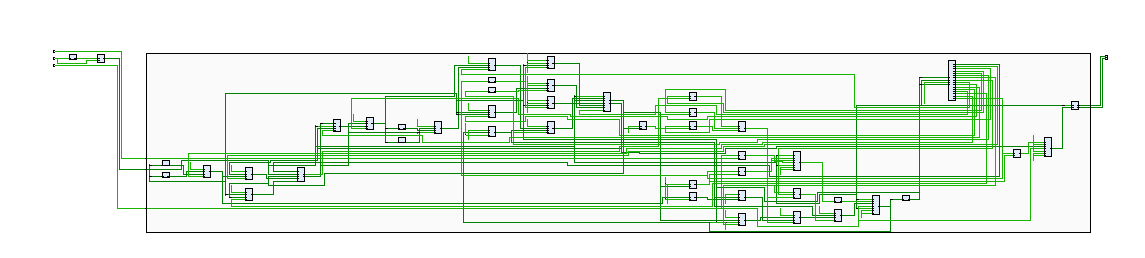


1. 点击schematic即可生成原理图，如果遇到问题则直接点击错误原因即可转到对应位置修改，常见错误原因是端口不匹配的问题，遇到多余端口，可以直接删除，不影响结果。

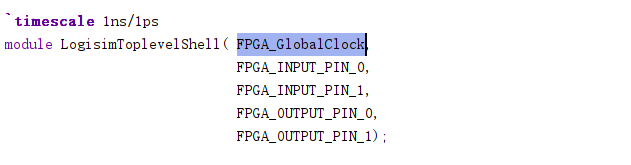


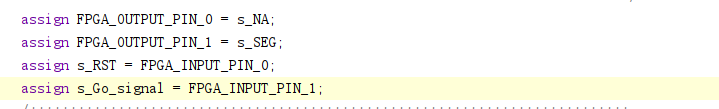
1. 最终原理的效果图如下：

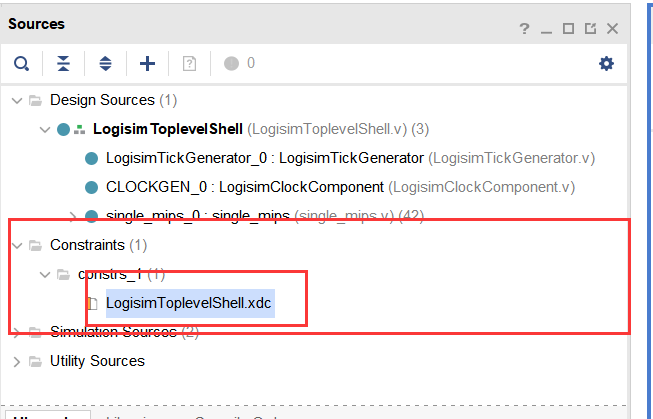




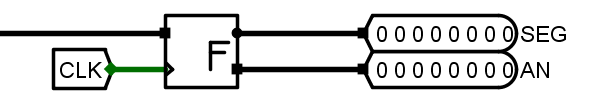
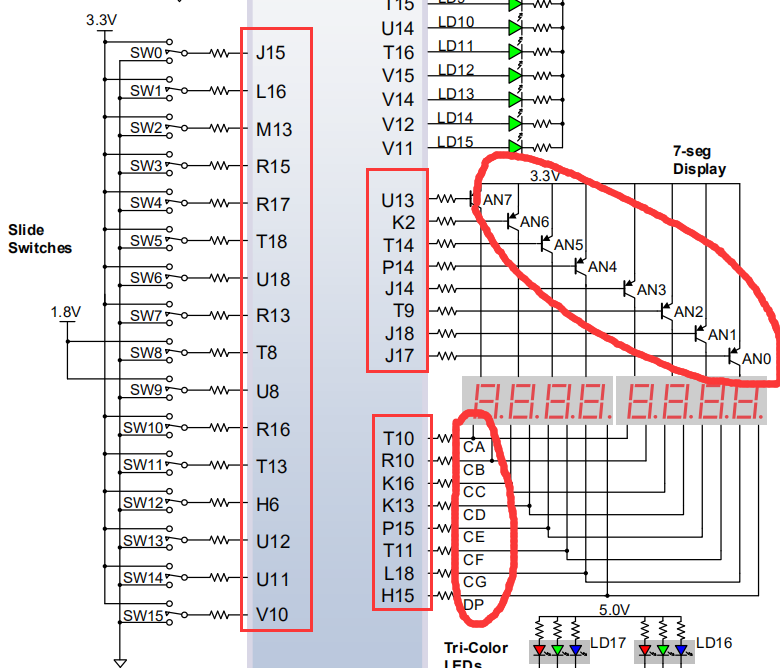
1. 此外，单周期流水线的CPU的顶层封装只需要5个输入输出引脚，分别是时钟、RST、Go以及两个FPGADigit的两个输出，如果存在其他引脚可以删除，使用结构化语句对顶层封装以及mips接口进行连接，顶层接口以及赋值的具体语句如下图：





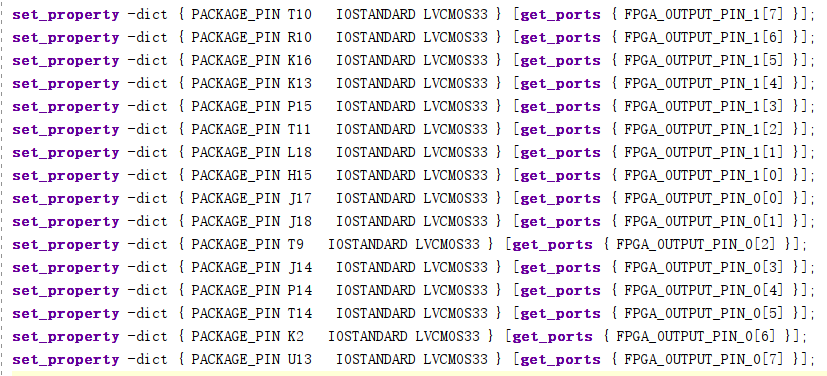
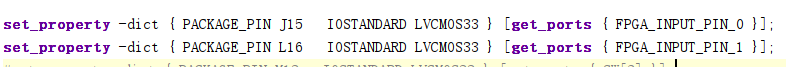


1. 修改如上图所示，源文件夹下的约束文件（.xdc），参考如下Artix-7的接口结构进行接口的绑定，左侧为开关，右侧为数字显像管，开关的编号和具体开关的对应关系，可以参考开发板上的标签，数显管的标签从上到下是按照从高位到低位排列的，因此，绑定时也需要从高位到低位，AN部分对应FPGADigit的下方输出，CX部分对应FPGADigit的上方输出。

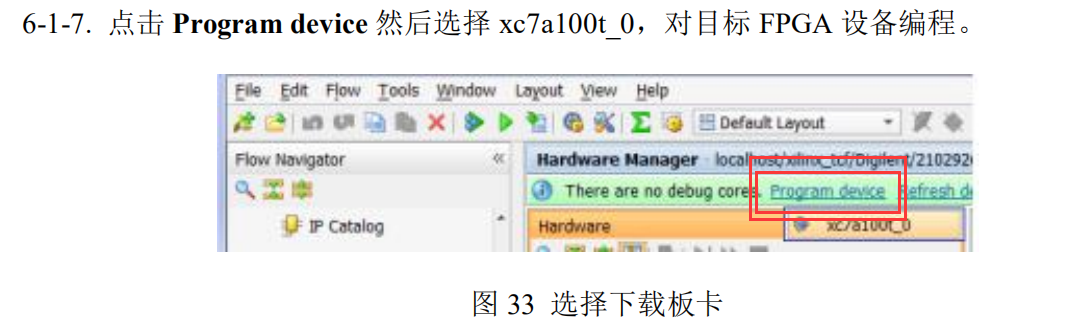
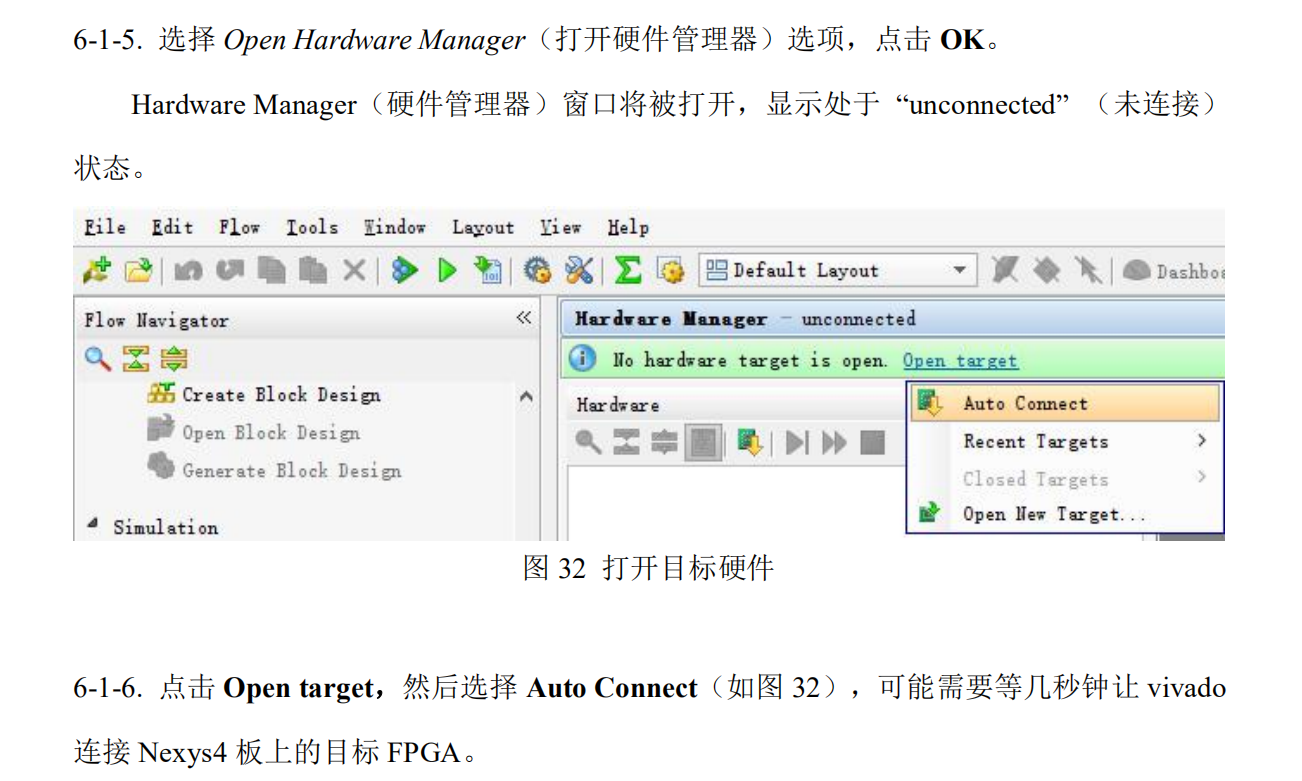
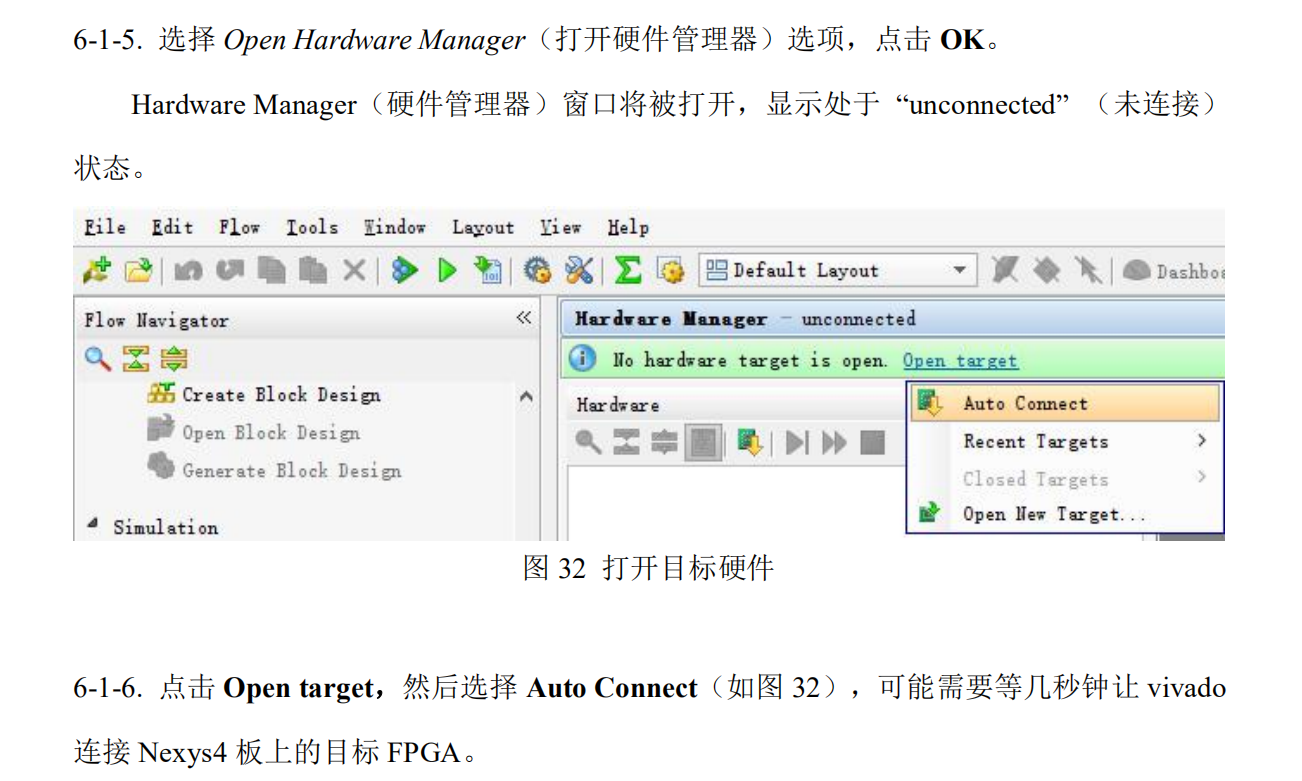
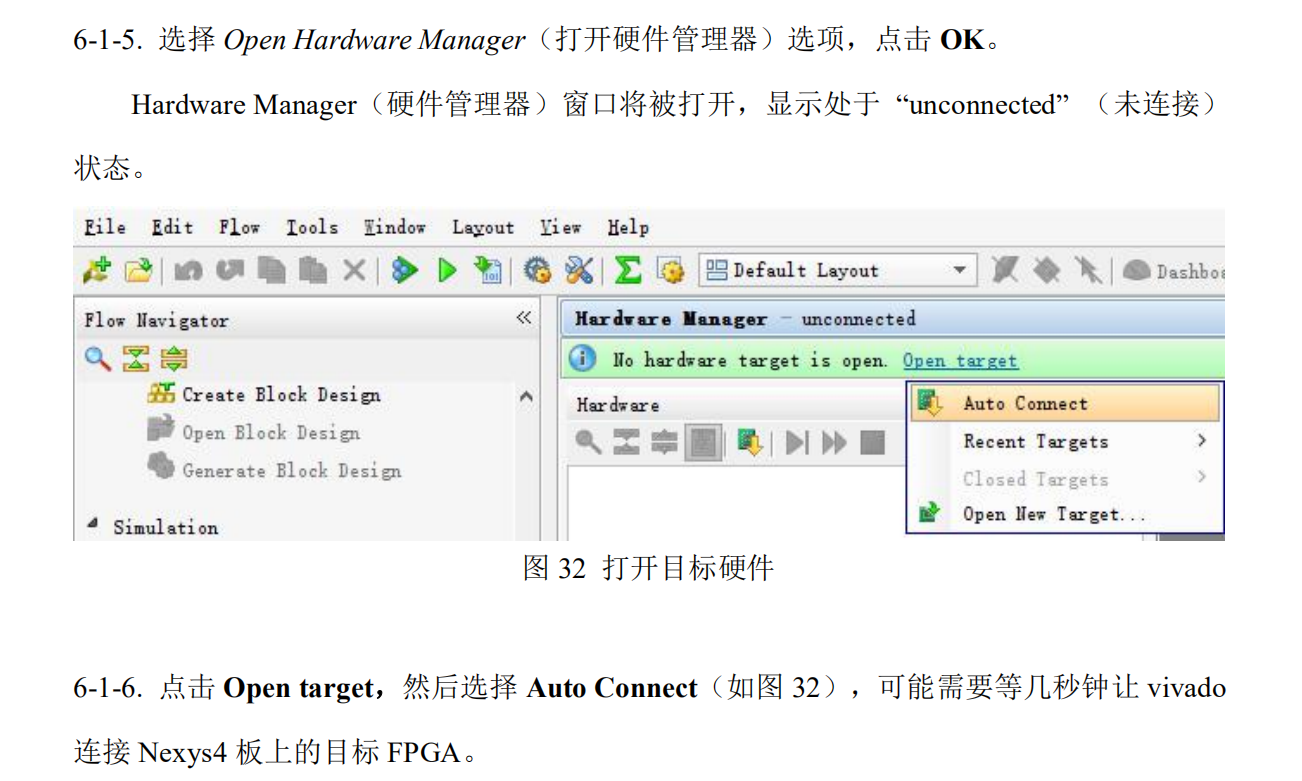
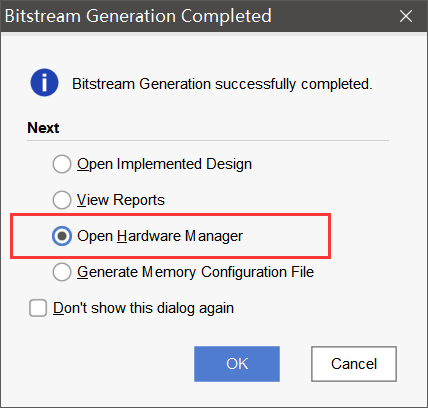
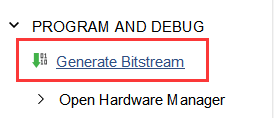
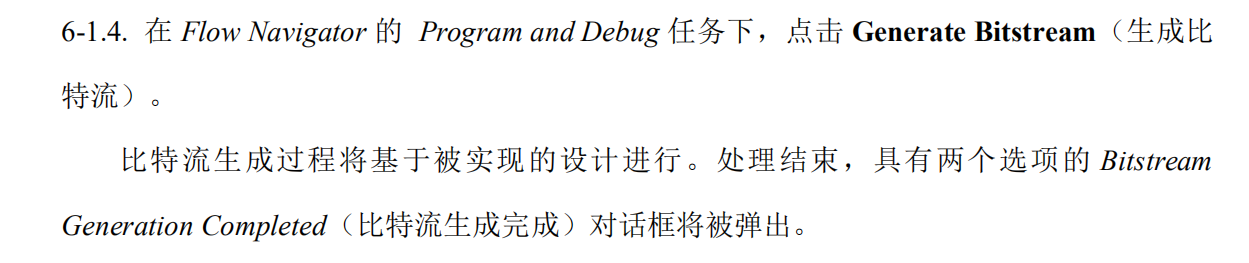
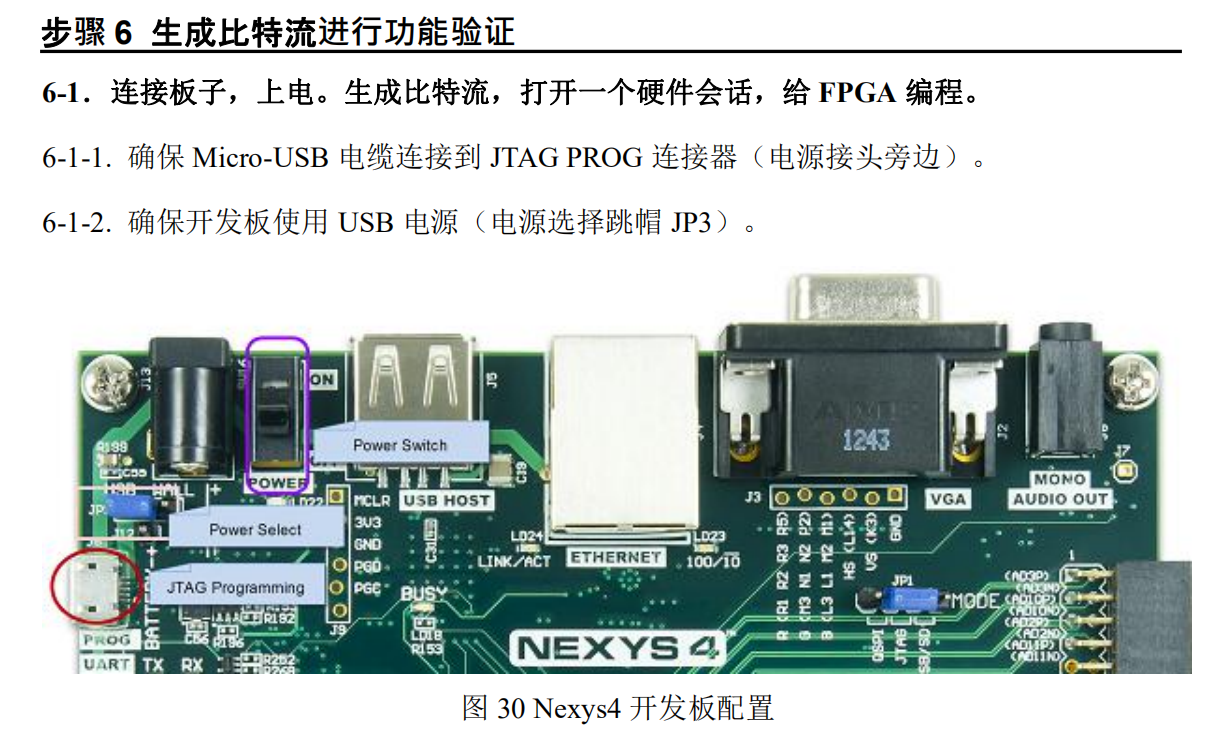


1. 绑定完成后，效果如下所示，需要注意的是，如果信号为1位信号，不要加“[0]”，否则无法生成比特流（至少在2020版存在这个问题，其他版本未知）。

**PS.2020.3.9版本的转化ALU存在问题，equal指令可能需要修改**



1. 绑定完成后，点击生成比特流进行生成（弹出的提示一路确认即可），然后领取开发板进行上板实验。



1. 完成上述步骤后即可开始测试。
2. 注意事项： 寄存器的清零（RST）可以使用异步清零，但是数据载入（包括PC以及LedData的载入需要同步载入，也就是时钟端必须连到系统时钟，然后通过使能信号对数据载入进行控制，不可将载入信号直接连到时钟端，会因毛刺会造成数据错误）。