**Logisim转Verilog使用手册**

1. **使用环境**

操作系统：Windows64位操作系统

Java JDK版本：1.7及其以上

1. **操作步骤**

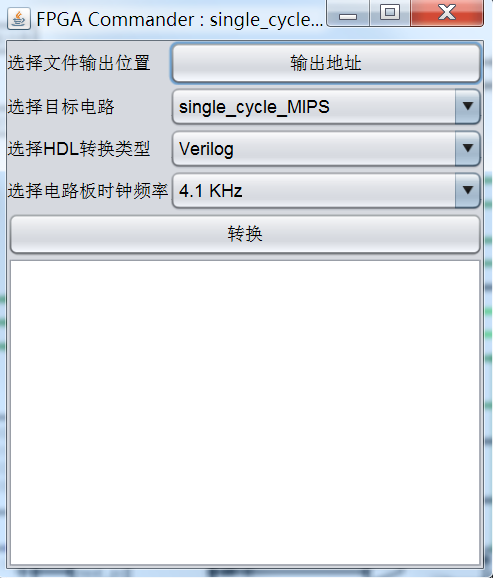
这一版本的Logisim是在原有Logisim Evolution版本的基础上，增加了原理图一键直接转Verilog功能，即将在Logisim中画的电路转换成Verilog语言文件，且直接支持使用Vivado生成可以在Nexys4 DDR开发板上运行的比特流（bitstream）。因此，设计电路的过程跟原来Logisim的使用方法一样，这里就不再赘述了；电路设计完成且验证正确后，按照下面的步骤实现Verilog语言的转换：

1. 通过Logisim右上角的File菜单，选择Save Verilog选项（快捷键Ctrl+V）打开Logisim转换Verilog功能对话框，如图1所示；



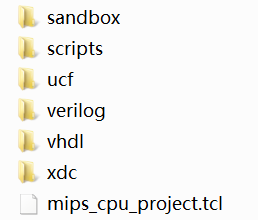
**图1 Logisim转Verilog功能菜单**

1. 弹出Logisim转Verilog对话框，如图2所示；在该对话框中首先需要完成4个选项的设置，它们分别为：（1）选择转换后的Verilog语言文件保存的位置，点击Workspace按钮，选择生成的Verilog文件保存的位置；（2）选择需要转换的主电路，通过拉框选择需要转换的电路；（3）选择将转换成的硬件描述语言的类型，目前只支持Verilog，因此这个选项使用默认值即可；（4）选择转换后电路运行的主时钟频率，通过下拉框选择期望的时钟频率。



**图2 Logisim转Verilog对话框**

1. 选择完成后，点击图2所示的Download按钮，即可一键生成相应的Verilog文件，转换后的Verilog文件都保存在步骤2中指定的目录中；
2. 转换完成后，指定目录中保存的内容如图3所示；其中真正有用的是2个目录和一个tcl脚本文件，一个是verilog目录，它里面按照Logisim原理图的结构保存着相应的Verilog文件；另一个是xdc目录，它里面其实只有一个文件，这个文件名固定，为LogisimToplevelShell.xdc，是Vivado生成Nexys4 DDR开发板上运行的比特流（bitstream）所需的约束文件，该文件的内容可能需要根据实际情况进行修改；tcl脚本文件则是用来自动生成Vivado的工程及比特流（bitstream）的；



**图3 转换后的Verilog文件目录**

1. 打开Vivado，在Vivado脚本窗口中浏览到保存Verilog转换文件的目录，source该目录中的tcl脚本文件，即可自动生成相应电路的Vivado工程及比特流（bitstream），Vivado脚本文件的使用请参看相应的文档。
2. **注意事项**
3. 由于转换成Verilog语言的电路最终需要在Vivado中生成比特流（bitstream），因此在使用Logisim设计电路图时，元器件、引脚、标签（Label）等等的命名需要符合Vivado和Verilog语言的命名要求，即不能使用中文、不能有空格等等（具体参看Vivado和Verilog语言的命名规则），否则在Vivado中可能会出现错误；
4. 部分目录还不支持（即不能进行转换）的元件：

Wiring：Pull Resistor、Transistor、Transmission Gate

Gates：Controlled Buffer、Controlled Inverter

Arithmetic：Multiplier、Divider、Bit Adder、Bit Finder

Input/Output：Joystick、Keyboard、LED Matrix、TTY

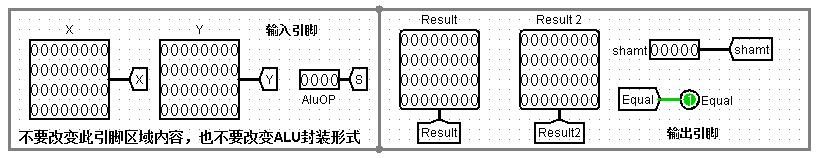
1. 部分元件转换前需要添加Label，它们分别是：

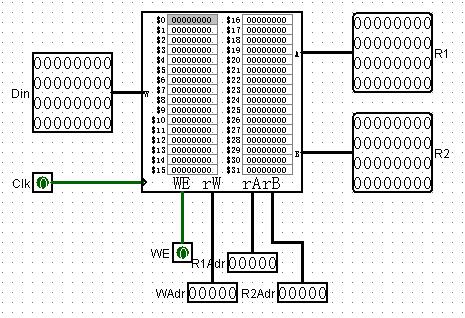
Input/Output：Button、Led、7-Segment Display、Hex Digit Display

Memory：Ram、Rom

Wiring：Pin

1. 电路中不可以出现三态；例如：Pin的属性栏中的“Three-state？”要选为No，Multiplexer的属性栏中的“Disabled Output”要选为“Zero”等；
2. 电路需要连接正确，例如：不允许在模拟过程中出现红色或蓝色的线等等。
3. **新增元器件说明**
4. 在输入/输出（Input/Output）中新增了FPGADigit元件，该元器件主要包括1个32位的输入、1个1位的输入（时钟）、2个8位的输出（默认输出为全0），在绘制的电路图中没有实际功能，主要用于在Nexys4 DDR开发板上通过数码管进行显示；该元器件在进行Verilog转换时，除了生成Blackbox.v文件外，可在相同目录（arithmetic文件夹）中自动生成Counter.v、decoder3\_8.v、display\_sel.v、divider.v和sevenseg\_dec.v文件，自动生成的Verilog文件代码均固定；
5. MIPS ALU和MIPS Regifile器件，它们都是封装好的二级电路，转换时将在circuit文件夹中生成.v文件，而不转换其内部电路；MIPS ALU二级电路如图4所示，它包含2个32位的输入X和Y、1个5位的输入shamt、1个4位的输入AluOP、2个32位的输出Result和Result2，1个一位的输出Equal；MIPS Regifile二级电路如图5所示，这包含1个32位的输入Din、2个一位的输入Clk（时钟）和WE、3个5位的输入WAdr、R1Adr和R2Adr、2个32位的输出R1和R2；以上所有命名均不可更改；

**图4 MIPS ALU二级电路**



**图5 MIPS Regfile二级电路**

https://blog.csdn.net/yohe12/article/details/104790430