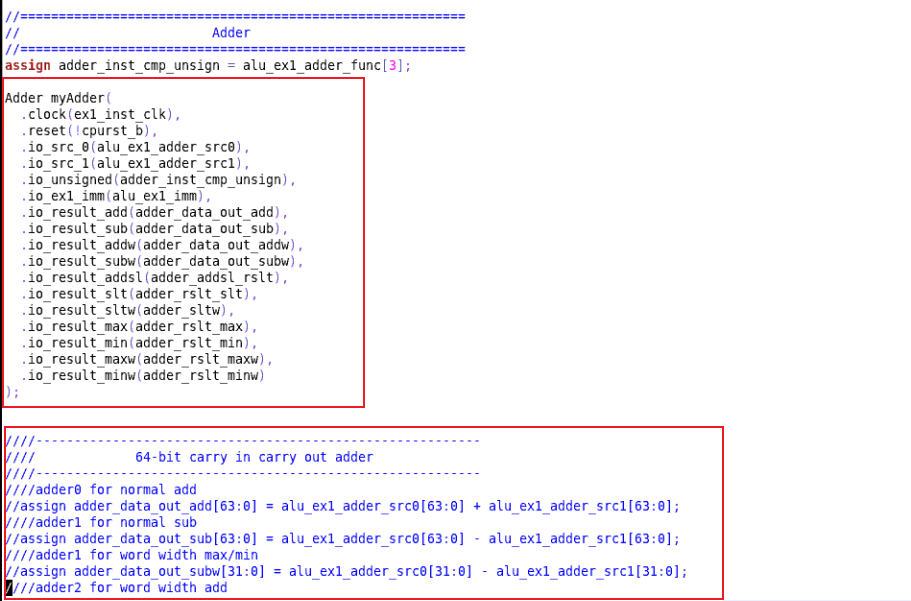
# 玄铁模块替换与仿真

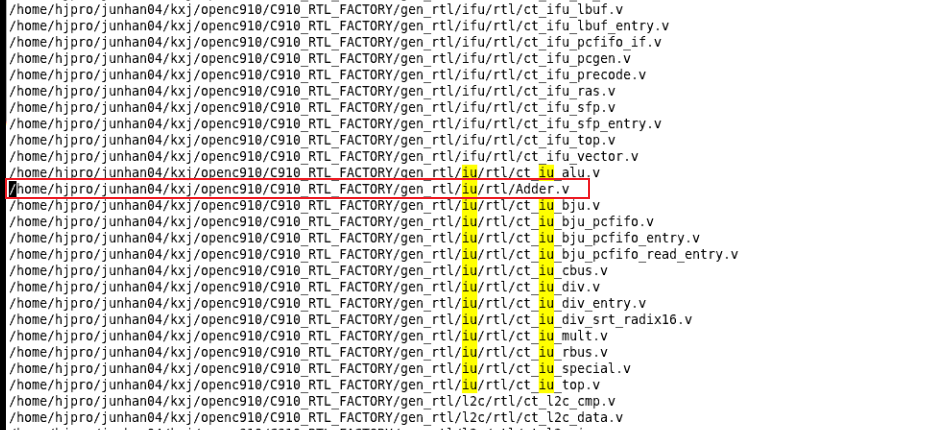
openc910的源文件位于/home/ECDDesign/ECDDesign\_share/openc910中，openc910-main即为我们要使用的c910源文件。大家可以把这个文件夹复制到自己的目录下，然后进行下面的操作运行。（如果不需要更换源文件，则可以跳过前四步，直接进行仿真的操作）

1. 首先我们将我们的替换模块的rtl放置到对应的模块文件夹中，例如我们要替换ALU中的Adder模块，我们就把Adder.v放到openc910/C910\_RTL\_FACTORY/gen\_rtl/iu/rtl的目录下。

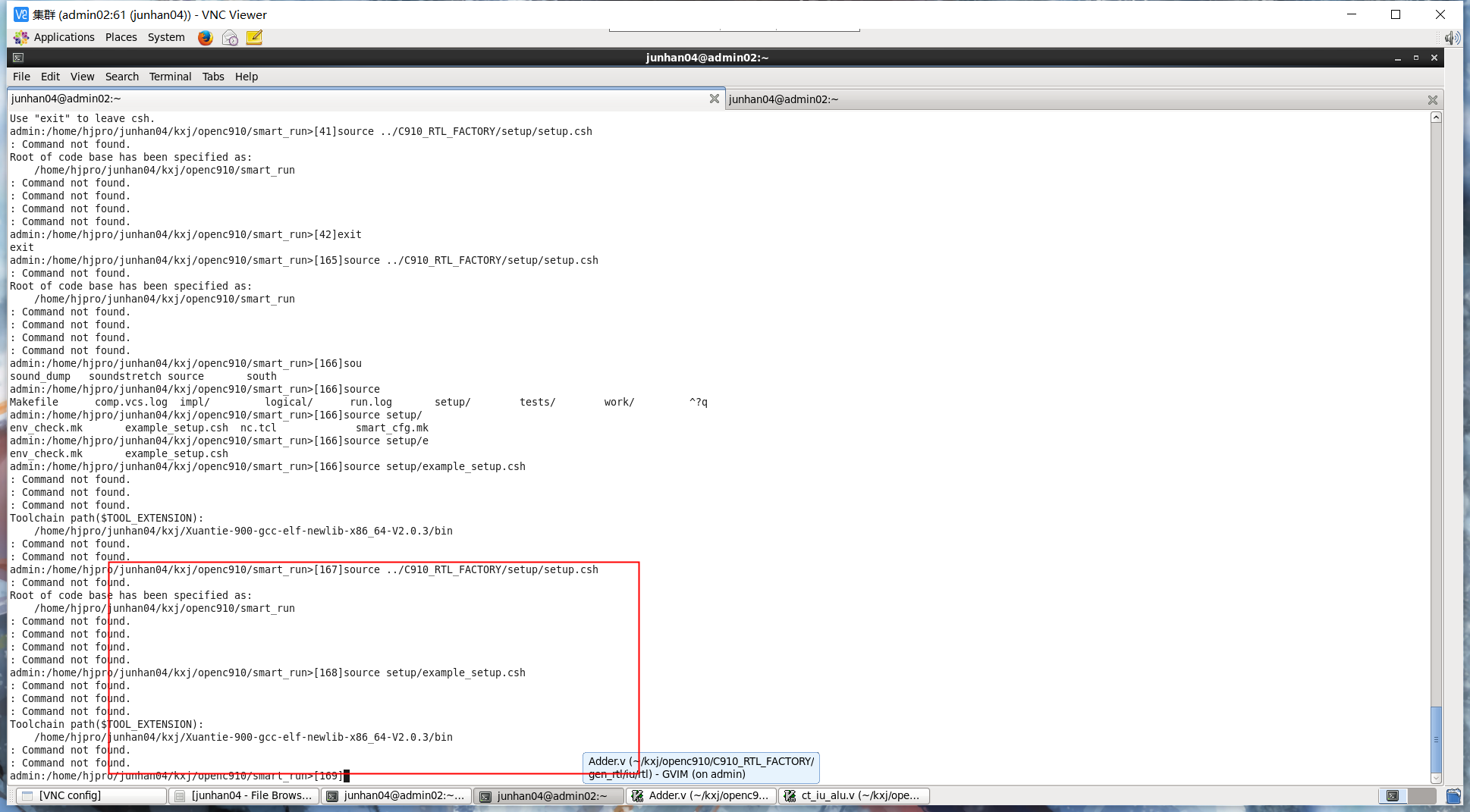
2. 然后我们需要将替换模块的上层模块的连接关系进行更改，例如ALU中的Adder逻辑将由我们自己的Adder.v来实现，所以我们将Adder逻辑注释，然后例化我们的Adder模块来进行连接。



3. 然后我们需要让仿真器在进行编译仿真模型的时候能够索引到自定义的源代码，所以需要更改openc910/C910\_RTL\_FACTORY/gen\_rtl/filelists下的filelist文件，主要在iu中加入我们的Adder。

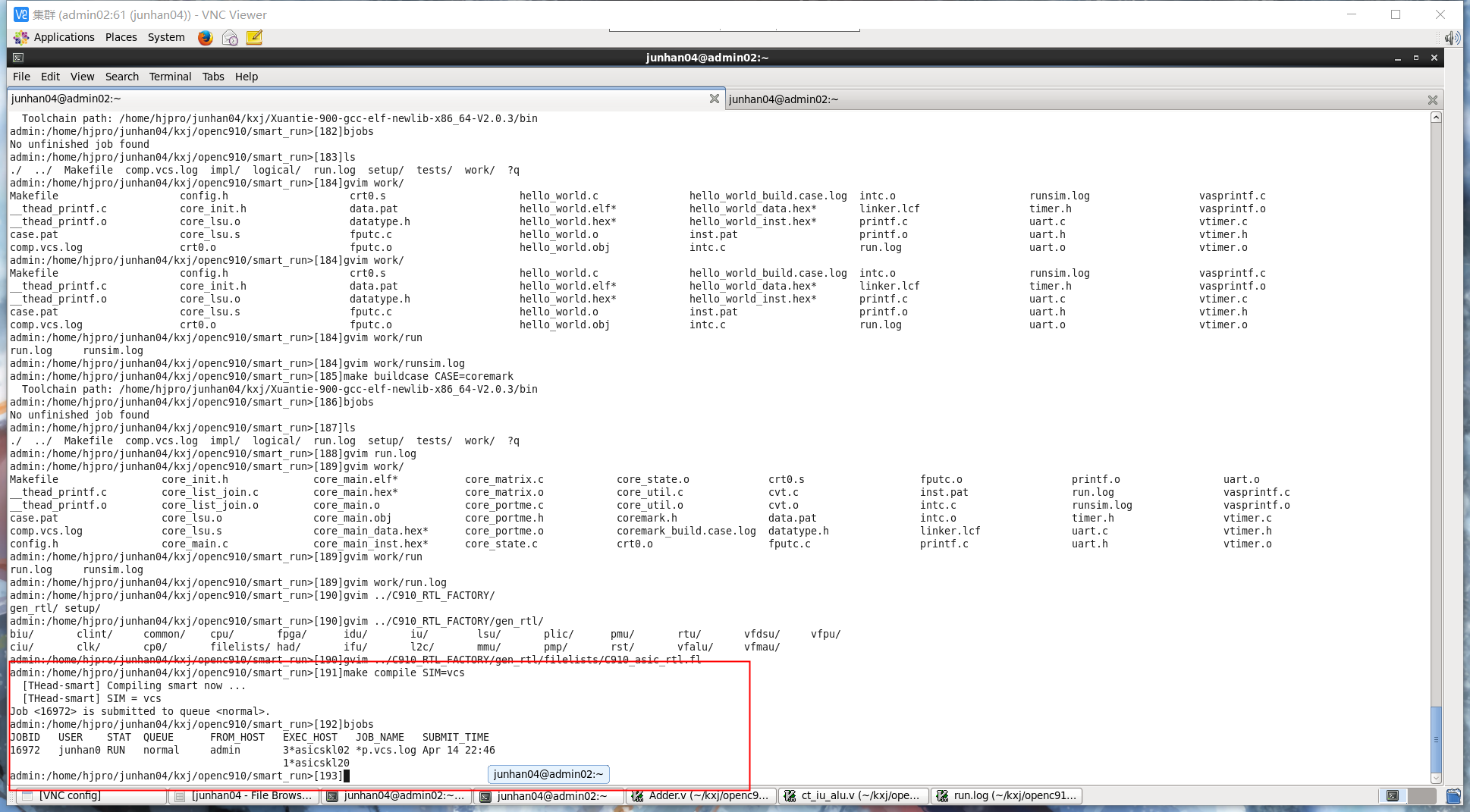


4. 完成了这些修改后，我们就成功把我们自定义的模块替换原有模块了，下面就是在集群上进行仿真的操作。

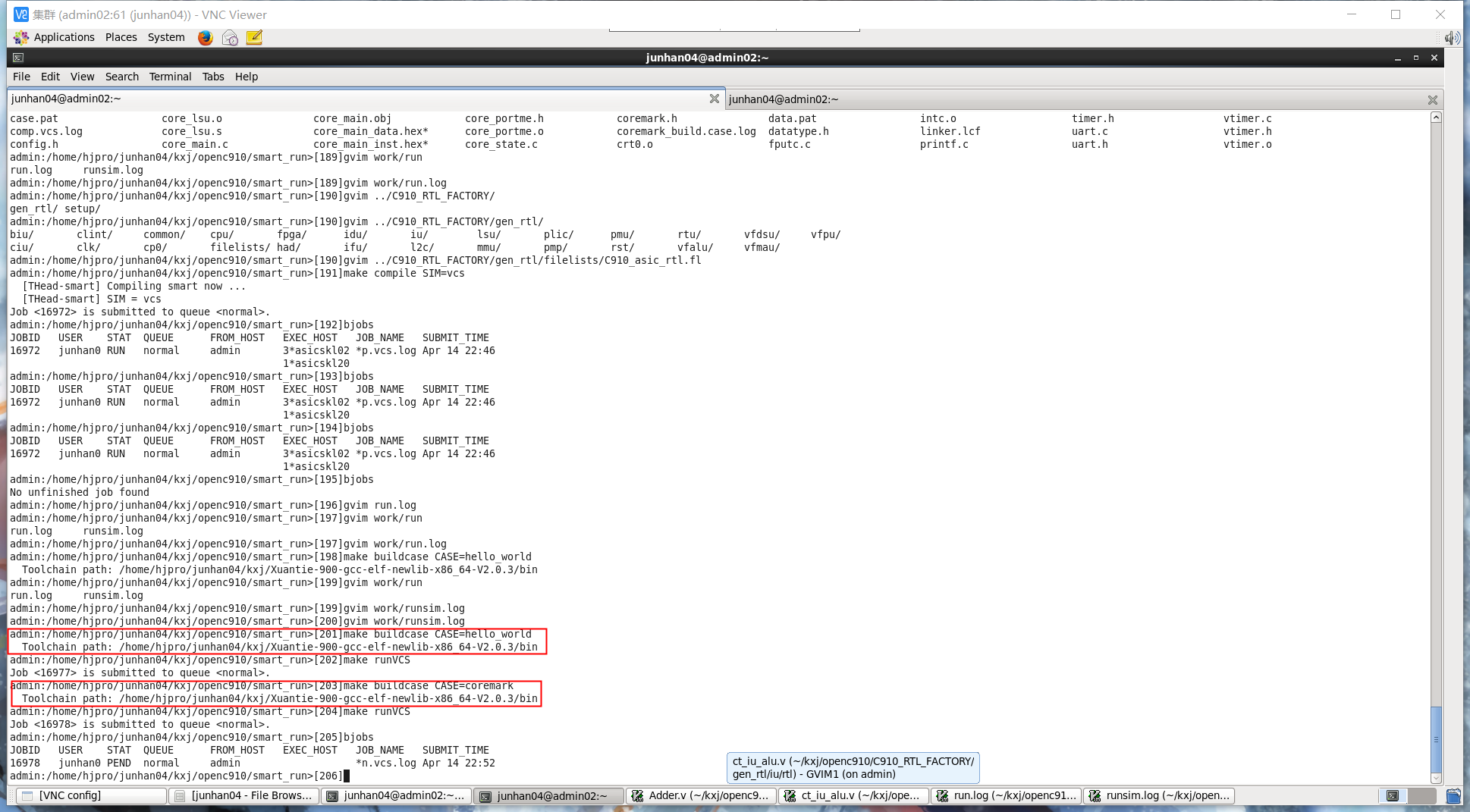
5. 我们先进行环境变量脚本的设置，首先在openc910-main/C910\_RTL\_FACTORY目录下运行source setup/setup.csh，我们需要进入到smart\_run文件夹下并运行source setup/example\_setup.csh。这两个脚本主要给出了源代码的路径以及RISC-V工具链的路径。

**由于在集群上运行有一定负载的工作需要使用bsub进行提交，所以我们将openc910的makefile进行了一定的更改以适用集群环境。在每个命令后，其任务会提交到集群进行执行，查看是否执行完成可以用bjobs来查看。**

6. 然后我们可以利用smart\_run文件夹下的Makefile文件进行仿真命令的运行。首先我们运行make compile SIM=vcs进行vcs的仿真模型的编译，主要是根据C910源代码进行编译。这一步会在work目录生成simv可执行文件。其运行的命令行输出可以在work目录下的run.log查看。（run.log不会自动清空，最新的可以在最后查看）



7. 之后我们使用make buildcase CASE=hello\_world可以生成hello\_world的hex文件以供仿真模型读取仿真。（此任务负载较小，是直接运行的，输出可以实时查看）



8. 最后我们运行make runVCS就可以启动simv的仿真模型去进行仿真了。命令行输出结果可以在work目录下的runsim.log查看。

