实验1. VCS+Verdi的使用及Makefile的编写

1. **实验目的**
2. 学习使用VCS进行systemverilog的编译与波形输出
3. 学习使用verdi进行波形观察
4. 学习Makefile的编写
5. **软件介绍**

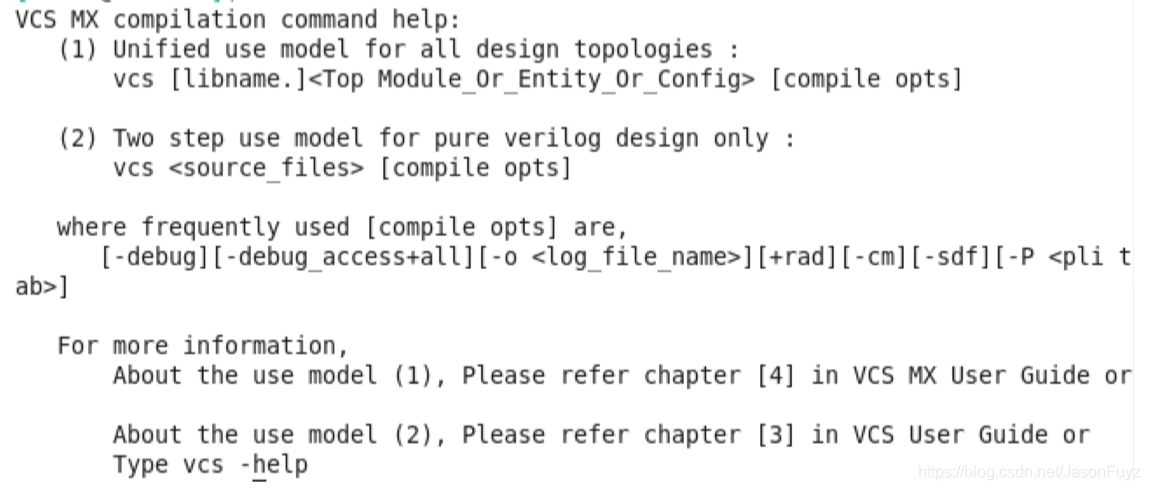
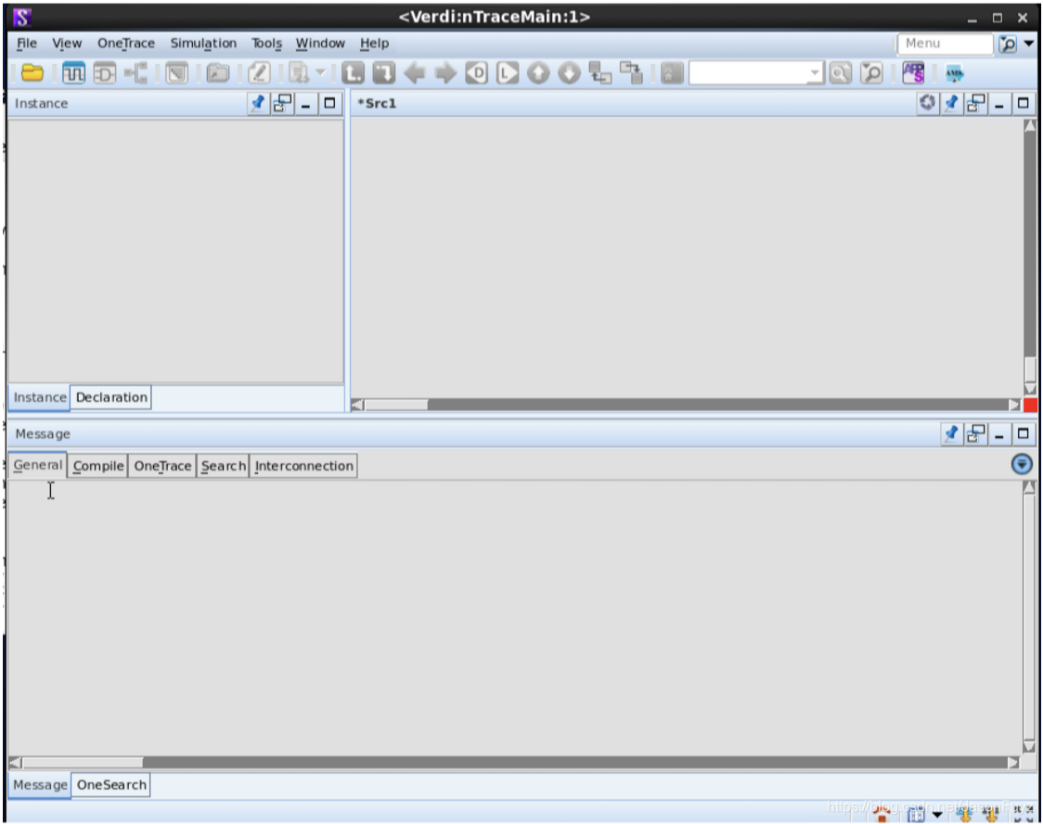
业界领先的设计人员在从事最先进的设计工作时为其验证环境选择 Synopsys VCS© 功能验证解决方案。 实际上，绝大多数的 32nm 及以下制程的设计均采用 VCS 进行验证。 全球顶尖的半导体公司大多采用 VCS 作为其主要验证解决方案，VCS 可提供高性能仿真引擎、约束条件解算器引擎、Native Testbench (NTB) 支持、广泛的 SystemVerilog 支持、验证规划、覆盖率分析和收敛以及完整的调试环境。

Verdi是Synopsys公司针对复杂SoC的调试问题所开发的一个调试平台，具有强大的分析引擎，是广泛应用的EDA工具之一，它具有良好的可视化操作界面：源代码窗口以树型排布的方式展示了设计的层次结构，在具体的设计代码部分可实现信号驱动的查找、仿真结果反标等功能。波形窗口可以导入两种格式的仿真文件，VCD和FSDB，在波形窗口可以对感兴趣的信号进行可视化分析，可以极大的提高调试效率。原理图窗口可以以电路的形式展示设计代码，既可以显示某一层的电路结构，也可以显示信号的局部电路图，有助于使用者理解各个模块之间的层次关系。

Verdi主要用于生成fsdb模型，同VCS使用的vcd文件相比，verdi使用的fsdb相当于vcd文件经过霍夫编码压缩之后的精简版，可用于查看fsdb波形并追踪RTL代码。虽说verdi、modelsim都是用来调试波形， modelsim与verdi相比，最大的缺点是波形不会全dump，wave窗口拉不全的话需要重新跑，而verdi边运行边查看。

1. **VCS与verdi使用方法**

1．）可以Terminal输入vcs，verdi来验证下环境有没有配置成功（若安装成功一般会如下图所示，但出现下图也不能确保环境一定是正确的）；

* **Terminal输入vcs会显示如下：  
  **
* **Terminal输入verdi会直接调用出verdi的GUI  
  **

2）创建工作环境及代码准备

1. 在home目录中创建工作目录把代码拷贝到work下

> mkdir ~/work

> cp /mnt/hgfs/共享目录/.\*.sv ~/work

> vim ~/work/\*.sv (查看文件内容)

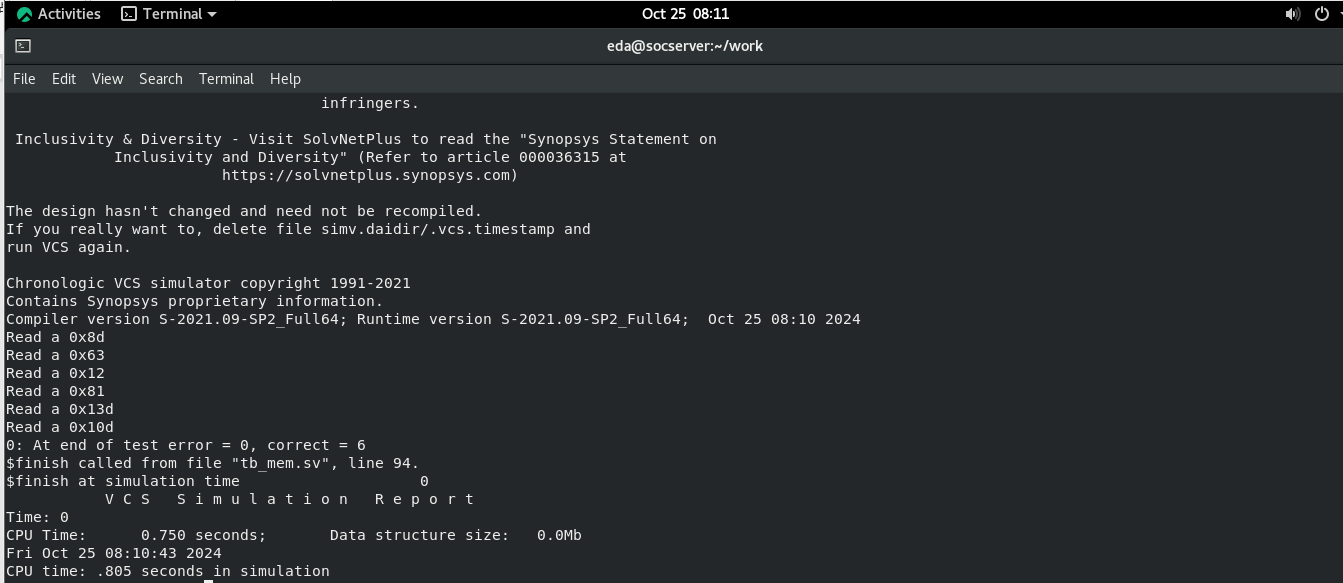
1. vcs编译

> vcs -R -full64 +v2k -debug\_access +define+FSDB -sverilog mem.v tb\_mem.v -l sim.log

-R表示自动运行仿真，+v2k表示使用Verilog-2001标准，-fsdb表示支持对fsdb相应操作，+define+FSDB相当于在verilog头文件里加上`define FSDB，-sverilog表示支持system verilog，输入.v文件的顺序可以不同（顺序是随意的）-l vim.log表示将终端显示的信息在vim.log中储存；

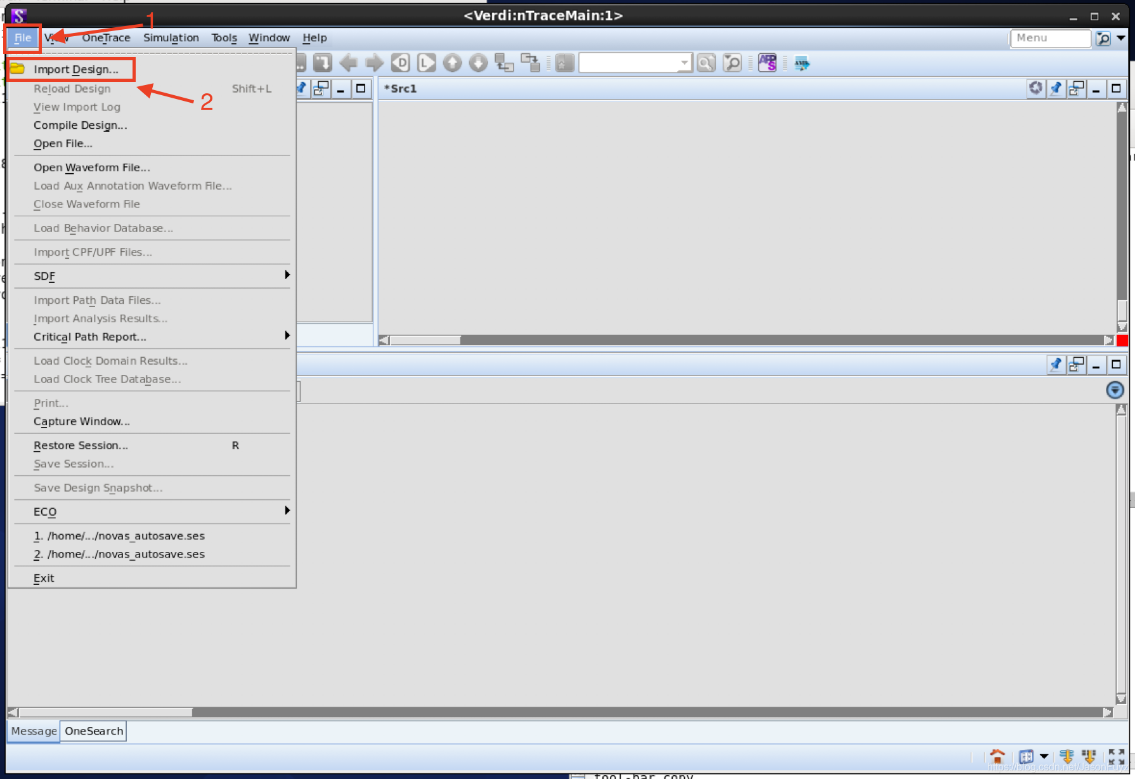
> vim run.log

编译后Terminal显示下图，并在文件夹中出现fsdb文件时，代表vcs已经对mem编译仿真成功了：

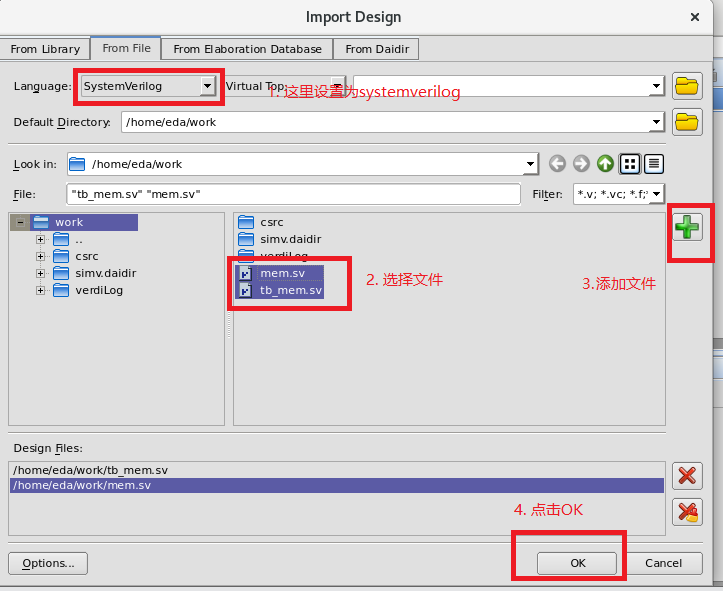


3) 现在可以打开verdi来看波形了，在Terminal输入：

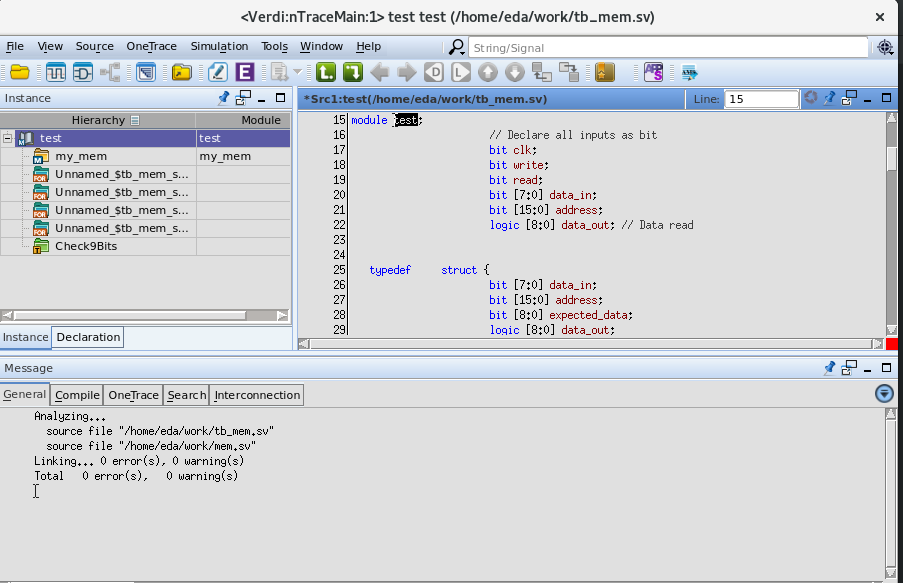
> verdi



>添加源文件



> verdi导入文件成功



> 生成导入fsdb文件，以显示波形：

1. Top层末尾加入如下语句

//dump fsdb

initial begin

$fsdbDumpfile("mem.fsdb");

$fsdbDumpvars(0);

end

1. 生成源文件列表

find ./ -name "\*.sv" > file.list

1. **Verdi使用**

**B站搜素“Verdi工具入门指导”**

**https://www.bilibili.com/video/BV1sz4y1W7jL/?spm\_id\_from=333.337.search-card.all.click&vd\_source=09eed9a5c3afed41888398e84ec8fb97**

1. **Makefile**

vcs\_sim:

./simv -l sim.log +notimingcheck +nospecify -k ucli.key

run\_verdi:

verdi -sv -f ./file.list -ssf top.fsdb &

clean:

rm -rf simv.daidir [csrc](https://zhida.zhihu.com/search?content_id=232830759&content_type=Article&match_order=1&q=csrc&zhida_source=entity)

DVEfiles verdiLog \*.log \*.conf \*.vpd \*.key \*.fsdb simv \*.dump

collect\_error:

perl collect\_vcs\_error\_warning.pl

vcs\_compile:

vcs +v2k -f ./file.list \

-full64 \

-sverilog \

-timescale=1ns/1ps \

-debug\_access+cbk \

-l compile.log \

-P ${VERDI\_HOME}/share/PLI/VCS/LINUX64/novas.tab \ #加载动态库

${VERDI\_HOME}/share/PLI/VCS/LINUX64/pli.a \ #加载表格文件

-o simv

参考<https://zhuanlan.zhihu.com/p/651156338>

1. **实验要求**

1）参考以上过程，设计ATM四端口仿真与验证环境（课本4.10）。

2）使用接口简化端口。Dump输出波形。

3）编写Makefile实现仿真验证的自动化。

4）整理出vcs，simv的参数列表。并解释。