**哈尔滨理工大学**

计算机科学与技术学院

**实 验 报 告**

（2021-2022第二学期）

**课程名称：** 集成电路验证技术

**班 级：** 集成19-2

**学 号：** 1914020208

**姓 名：** 黄羽铧

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实验名称 | **APB接口的存储器验证** | | | 专 业 | 集成电路设计与集成系统 |
| 姓 名 | 黄羽铧 | 学 号 | 1914020208 | 班 级 | 集成19-2 |

**一、实验目的：**

1、熟悉APB接口

2、继续熟悉VCS基本功能、界面和使用

3、熟悉基于transaction的验证平台的模块和结构

**二、实验内容：**

1、编写一个APB接口存储器设计的验证平台

**三、实验设备及软件环境：**

**设备：**计算机电脑

**软件环境：**Linux, VCS, DVE, VMware, Vim

**四、实验过程及结果：**

**实验步骤：**

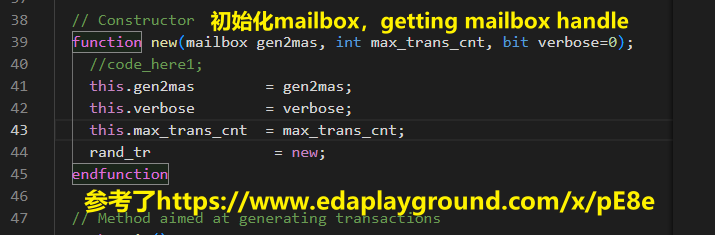
1. 分析apb\_trans.sv中APB事务定义，并完成以下代码
   * 加display功能，打印出事务中的数据
   * 增加copy功能，返回class的一个拷贝
2. 分析事务产生apb\_gen.sv代码，并完成以下代码
   * 在apb\_env/apb\_gen.sv中编写类的构造函数。在构造函数中完成初始化mailbox、transactor对象、和transactions的最大数量设置
   * 在任务main中，使用循环把max\_trans\_cnt个随机的transaction传递给mailbox
3. 分析验证组件apb\_master.sv代码，完成以下代码
   * 分析read、write、idle等BFM 函数功能
   * 编辑构造函数，初始化mailbox、创建apb的transaction tr，
   * 在任务main中，使用无限循环去监控mailbox中是否有transaction，并将拿到的transaction解析，根据事务类型不同利用read、write、idle等BFM task做激励驱动和数据监测
4. 分析测试Program程序test.v ,并完成以下代码
   * 完成mailbox, 事务产生器，APB Master 验证组件的初始化。
   * 对APB Master调用 reset方法，然后fork 出Main 方法.
5. 分析Makefile内容，利用Make或自构建的命令行 运行验证平台，并观察结果。如有必要使用图形界面调试代码

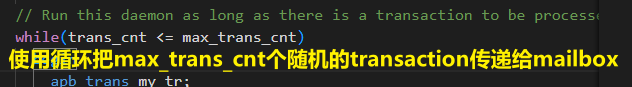
**实验结果截图：**

1. 对Transaction进行分析，并完成相关代码

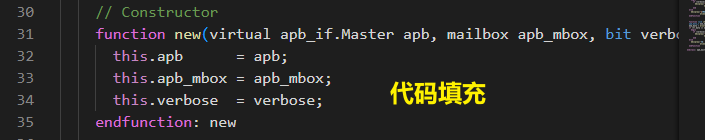


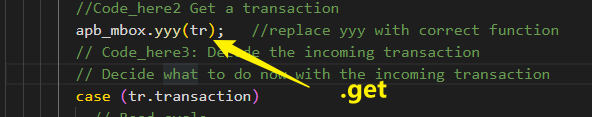
1. 中途觉得gvim没有设置好补全，就直接转到共享文件夹里回到VScode里进行编辑

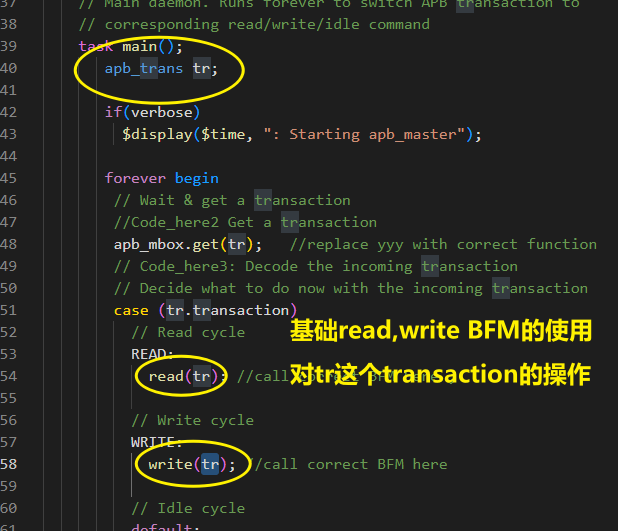


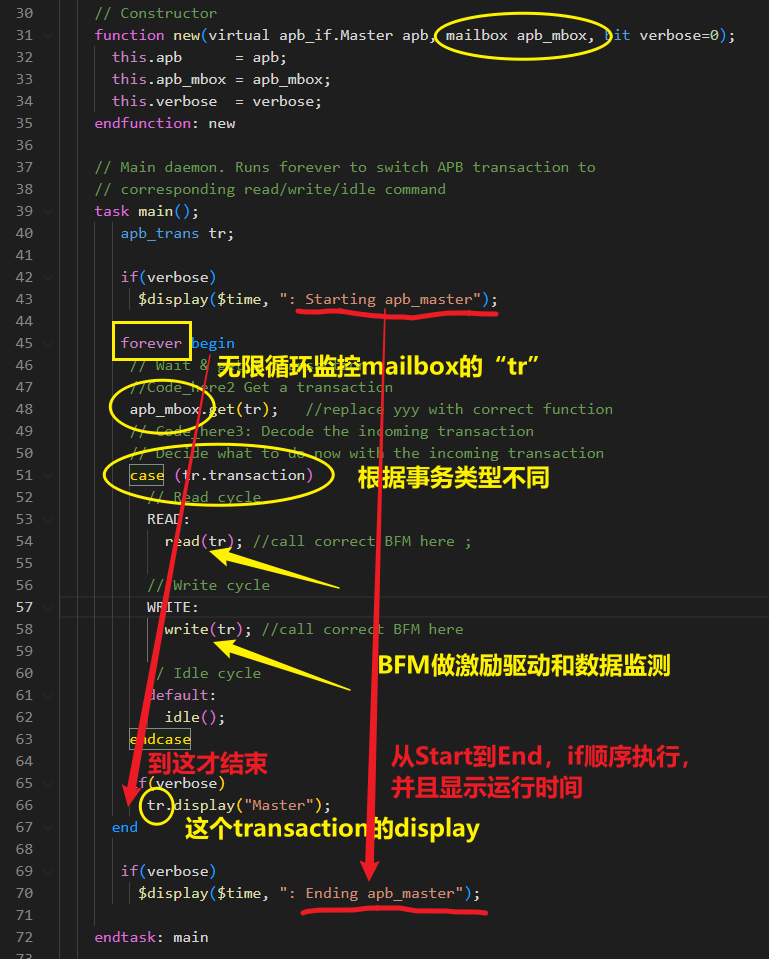


1. 分析apb\_master.sv,详情看图内注释

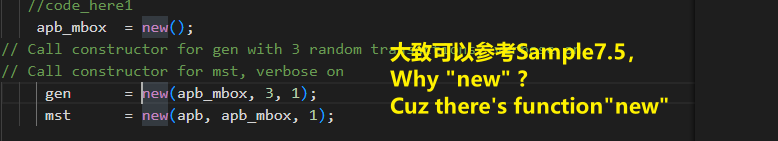






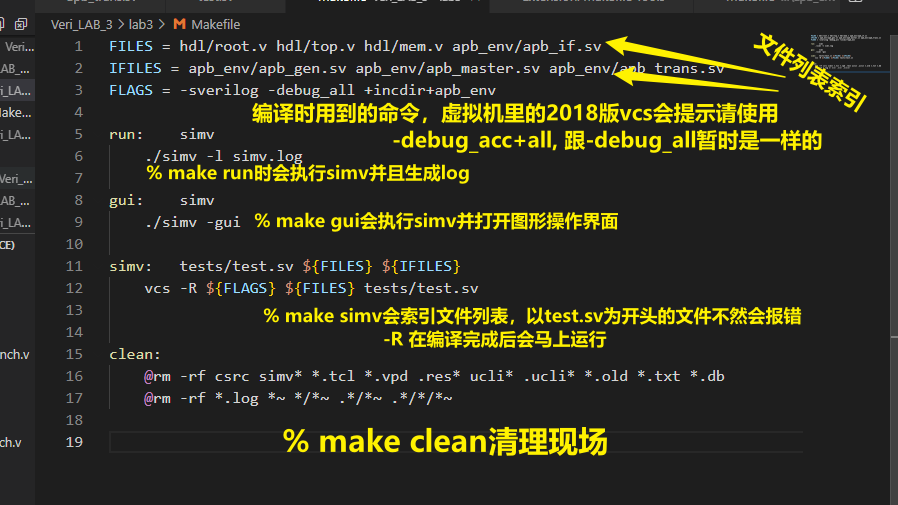


1. 分析测试program程序test.v

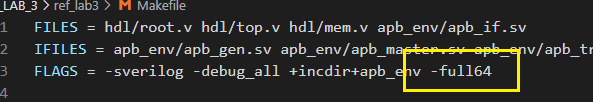




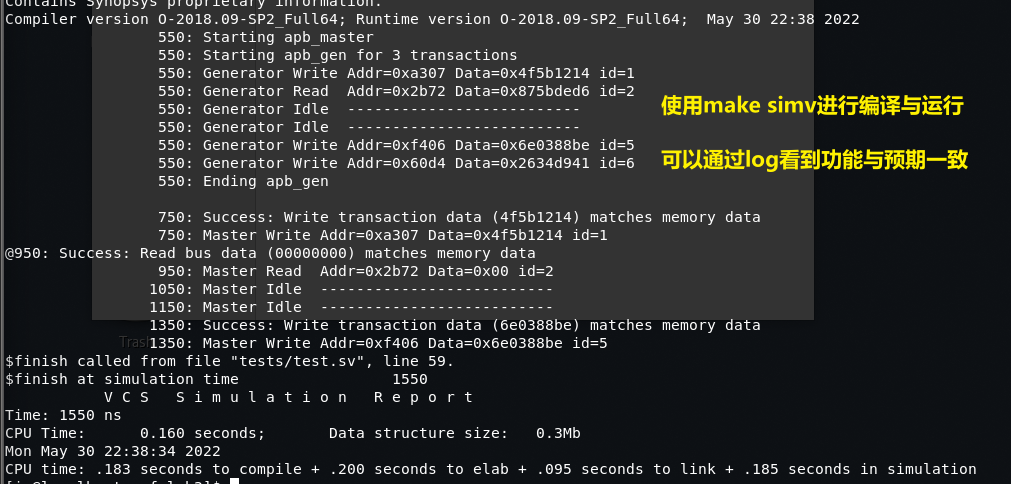
1. 分析Makefile



在这里同样出现了无法编译的情况，在Makefile中加入-full64



输出的log查看运行情况：



**实验结果分析：**

圆满完成了实验任务，成功完成了代码的编译以及仿真，成功通过makefile运行了vcs编译以及仿真，输出结果与预期一致。

**五、总结：**

本次实验搞明白了之前做作业的时候很多不明白的地方，对于transaction事务有了更深入的理解，建立了对mailbox的认识，增加了对于fork join应用的认知。

此外，对于实现对一个使用APB接口的ram的相关测试平台的编写有了新的认知，在后面剩余的作业以及设计起到了重要的指导作用。

实验成绩： 指导教师： 年 月 日