中国科学院大学计算机组成原理（研讨课）

实 验 报 告

学号： 2021K8009925006 姓名：冯浩瀚 专业：计算机科学与技术

实验序号：03 实验名称：定制MIPS功能型处理器设计

注1：撰写此Word格式实验报告后以PDF格式保存SERVE CloudIDE的/home/serve-ide/cod-lab/reports目录下（注意：reports全部小写）。文件命名规则：prjN.pdf，其中“prj”和后缀名“pdf”为小写，“N”为1至4的阿拉伯数字。例如：prj1.pdf。PDF文件大小应控制在5MB以内。此外，实验项目5包含多个选做内容，每个选做实验应提交各自的实验报告文件，文件命名规则：prj5-projectname.pdf，其中“-”为英文标点符号的短横线。文件命名举例：prj5-dma.pdf。具体要求详见实验项目5讲义。

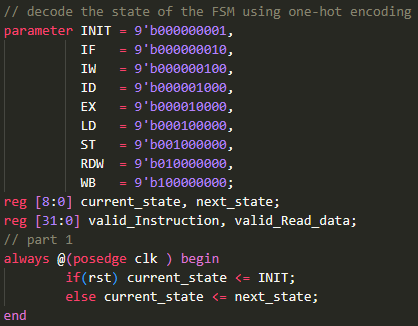
注2：使用git add及git commit命令将实验报告PDF文件添加到本地仓库master分支，并通过git push推送到SERVE GitLab远程仓库master分支（具体命令详见实验报告）。

注3：实验报告模板下列条目仅供参考，可包含但不限定如下内容。实验报告中无需重复描述讲义中的实验流程。

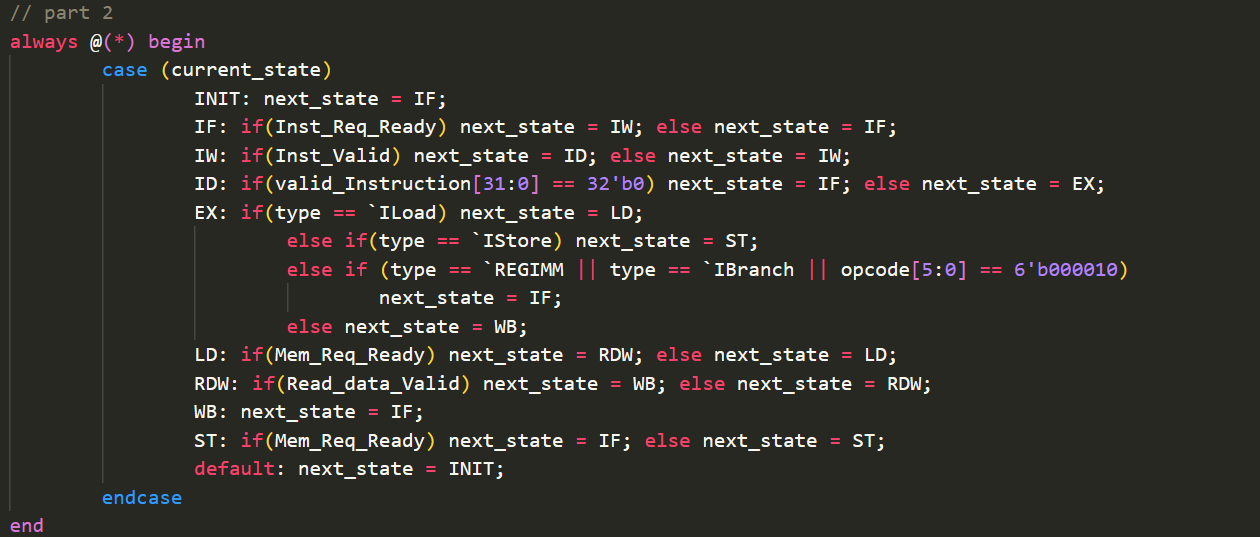
1. 逻辑电路结构与仿真波形的截图及说明（比如关键RTL代码段{包含注释}及其对应的逻辑电路结构图{自行画图，推荐用PPT画逻辑结构框图，复制到word中}、相应信号的仿真波形和信号变化的说明等）
2. 处理器真实内存访问通路

该部分相比上个实验中单周期mips处理器的代码最大的变动是加入一个三段式状态机。

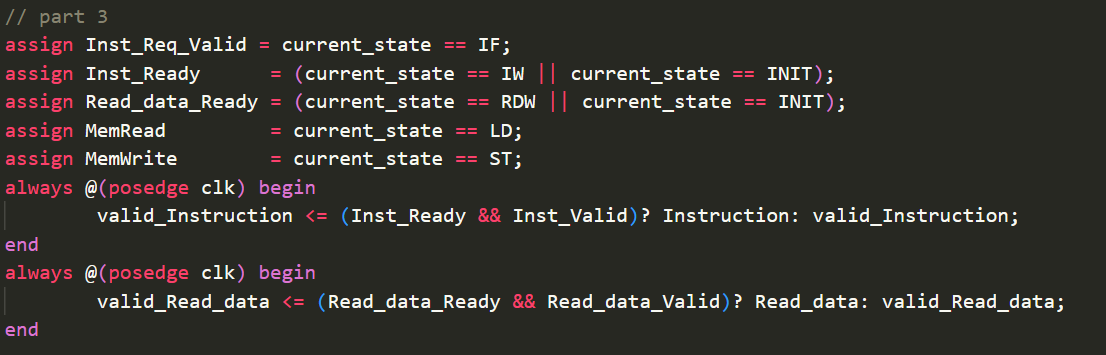
第一段用独热码将9个状态编码，并且用时序逻辑描述状态寄存器的同步状态改变。



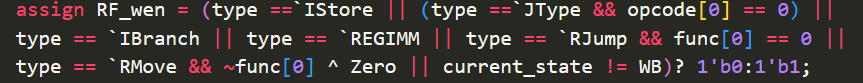
第二段用always块的组合逻辑描述状态机的状态根据当前状态以及其他输入信号的改变。此处注意case语句要有default避免出现锁存器



第三段用always时序逻辑以及assign组合逻辑，根据状态机当前状态，描述握手信号以及添加的valid\_Instruction和valid\_Read\_data寄存器的同步变化。此二者保证了指令和读出的数据随着状态发生变化而变化或保持不变。

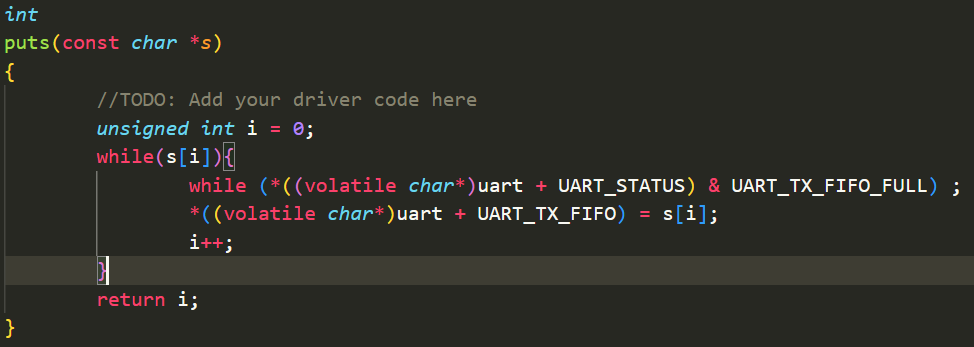


此外，还有一些细节也发生了变动。例如RF\_wen等在单周期cpu中根据指令类型决定信号高低的使能信号在实验中应该改为由当前状态决定信号高低。

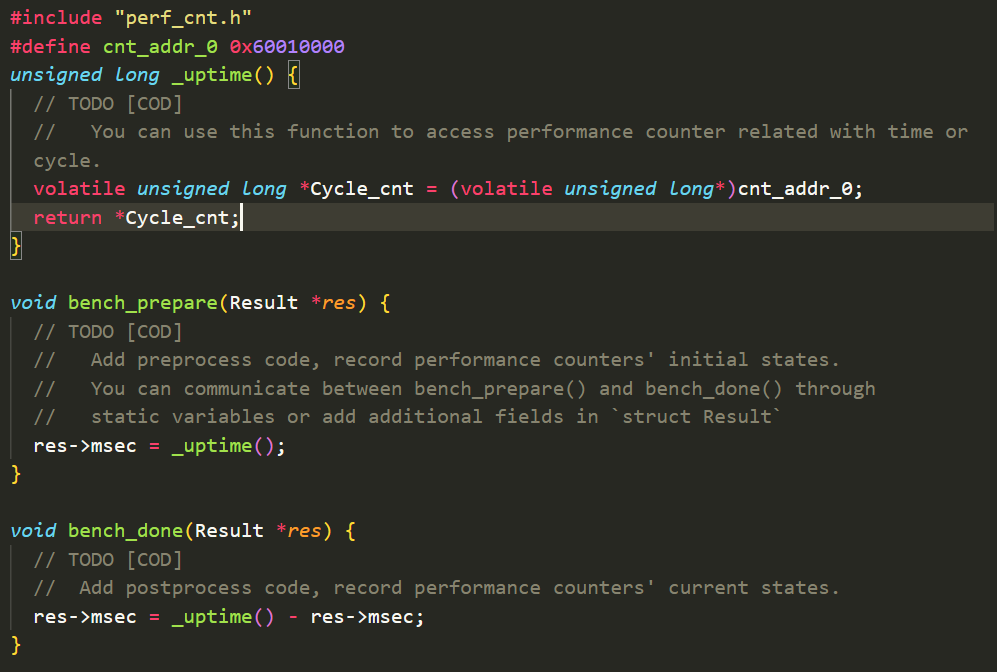


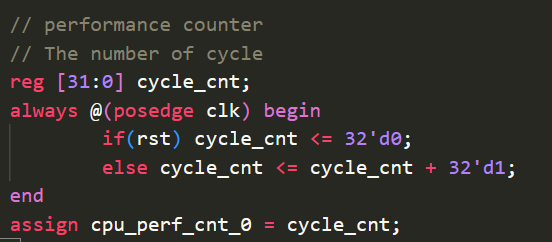
1. 外设控制器访问方法与字符串打印功能实现

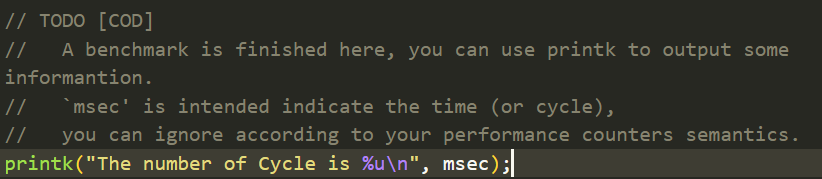
这里将\*uart强制类型转换成char类型指针，保证了指针加1同时地址偏移量也是1。并且用volatile关键字使得编译器对访问该变量的代码不再进行优化，从而可以提供对地址的稳定访问。



1. 处理器核性能计数器及性能评估方法

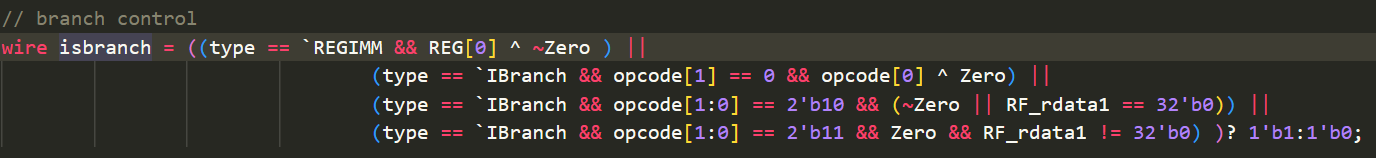
我选择加入的性能计数器是周期计数器，因此不需要在perf\_cnt.h中新增结构体，只需修改perf\_cnt.c中代码如图所示，修改\_uptime()函数，在bench\_prepare和bench\_done中调用，计算msec值，最终在bench.c中使用printk()函数输出。





1. 实验过程中遇到的问题、对问题的思考过程及解决方法（比如RTL代码中出现的逻辑bug，逻辑仿真和FPGA调试过程中的难点等）
2. 上次实验单周期mips处理器中存在的bug在测试样例中没有显现，在本次试验的测试样例中显现：

例如在描述REGIMM类型以及I类型分支指令中是否分支的isbranch时出现错误。



1. 新增真实内存访问通路后出现的bug：

例如需要新增PC\_pre寄存器来存储未到达IF状态时的PC值，在IF状态再更新为PC，在最初编写代码时未考虑到，使得J类型指令的PC值总出现错误，仔细思考后改正。

1. 对讲义中思考题（如有）的理解和回答
2. 

答：volatile 关键字是一种类型修饰符，用它声明的类型变量表示可以被某些编译器未知的因素更改。遇到这个关键字声明的变量，编译器对访问该变量的代码就不再进行优化，从而可以提供对特殊地址的稳定访问。当要求使用 volatile 声明的变量的值的时候，系统总是重新从它所在的内存读取数据，即使它前面的指令刚刚从该处读取过数据。而且读取的数据立刻被保存。有些变量是用 volatile 关键字声明的。**当两个线程都要用到某一个变量且该变量的值会被改变时，应该用 volatile 声明，该关键字的作用是防止优化编译器把变量从内存装入 CPU 寄存器中。如果变量被装入寄存器，那么两个线程有可能一个使用内存中的变量，一个使用寄存器中的变量，这会造成程序的错误执行。volatile 的意思是让编译器每次操作该变量时一定要从内存中真正取出，而不是使用已经存在寄存器中的值。**

（[C/C++ 中 volatile 关键字详解 | 菜鸟教程 (runoob.com)](https://www.runoob.com/w3cnote/c-volatile-keyword.html)）

去掉volatile后，编译器将从cpu寄存器中访问uart+UART\_STATUS地址，这样每个指令循环后取出的该地址的值可能不会正确改变。

1. 在课后，你花费了大约 5 小时完成此次实验。
2. 对于此次实验的心得、感受和建议（比如实验是否过于简单或复杂，是否缺少了某些你认为重要的信息或参考资料，对实验项目的建议，对提供帮助的同学的感谢，以及其他想与任课老师交流的内容等）

本次实验基于上个实验的代码，总体较为容易。但是debug过程较为繁琐：custom\_sym的shuixianhua测试样例需要花费2h的时间（到后期才意识到如果不需要该测试样例的运行结果可以直接将其cancel，不影响后续pipeline），fpga\_emu过程需要反复修改emu\_settings.yml文件并推送到远程仓库…这些都给实验带来考验。但是这也训练了我通过波形定位bug的能力；另外，本次实验“软硬结合”，编写VHDL的同时也帮助我回忆巩固C语言的知识，还新学习了volatile关键字的用，这些都使我受益匪浅。

最终，感谢老师以及助教团队的辛勤付出。