

P4 Tips

和上次差不多，还是只提一些细节上的问题以及课上测试相关的东西，不过多介绍模拟的CPU的代码实现。

非官方，仅仅是个人经验，其中一些内容相信与否请自行斟酌。

水平很有限+知识遗忘，可能干货不多，请见谅。

课下

- 可以翻译你的P3电路，但是也不要过度依赖它，不然做P5的时候会感觉没电路很难受。
- 规范导线和接口的命名，建立一套舒服且清楚的命名方案，这个对P4以及之后的实验都有好处。
- 阻塞赋值和非阻塞赋值，要注意一下，display的时候可能会因为这个地方出现bug，因为非阻塞赋值在过程块执行完之后才能完成赋值，这个时候来个display得不到修改之后的结果。
- 一个去年课下弱测没测出来的错误：jr 指令的跳转，可以跳各个寄存器里存的合法的地址值，而不只是31号寄存器的。
- 关于reset：有一些输出比正常慢一个周期的情况，有可能是reset的问题，请仔细斟酌一下。另外，课下一定别忘了测reset，不要光测那些指令。
- 关于debug：
 - 主流debug方法就是Pre or P1的教程里面给出的设置断点，并加入中间变量，然后手动控制每次往后跑一小段时间（dbq我忘了那个按钮叫啥了）。开始可能感觉不友好，但是越用越熟练，只要有hack数据，bug用不了多久就能干掉。
 - 还有个无意中发现的debug方法：强行综合。假设你刚把P4代码写完，想测一下对不对，在用数据测之前可以先双击综合（Synthesize-XST），或许在几分钟之后或许会有成千上万个报错，根据那些报错（主要看error，warning可以先不管），可以发现比较明显的连线错误或者笔误。当然了，一些报错也有可能是你描述的那个东西本身就是不可综合的，对于那些部分暂时忽略就好。
- 课下准备的时候，可以参考P3怎么准备的，试着加一些指令试试。更有兴趣的同学，请看cscore上的英文版指令集，在里面找一些功能比较复杂的指令加一加试试，比如P5可能出现的bgezalr的弱化版bgezal等。
- 课下还是不建议加过多的指令，一是加多了之后课上不好加，二是为了一个P4消耗太多精力不值得，不如多学学P5。
- 如果还有时间，准备一下自动化测试吧，P5-P7都会用得到。魔改版Mars的获取可以参考一下[这](#)[个](#)，不太会自动化测试的可以看看牟哥的github学习一下。

课上

- 未来几个P的考题一般还是跳转+计算+访存，也有可能出现跳转+简单计算+复杂计算的情况。P4要加的指令要比后面两个P的简单不少，英文指令集中的比较难的指令大概就是P4天花板难度了。
- 还是推荐先加比较简单的计算指令，然后提交之后验证课下的CPU的正确性。课下bug导致的Wrong Answer的分布仍**可能**具有一定的规律性。
- 仔细读课上指令的RTL，不要想当然。
- 有同学可能想课上用自己写的测试工具，但请注意在测试工具提交窗口提交的测试工具在去年似乎是无法下载的。所以你可能要藏到设计文档的提交窗口里面（bushi

