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：05 实验名称：深度学习算法与硬件加速器

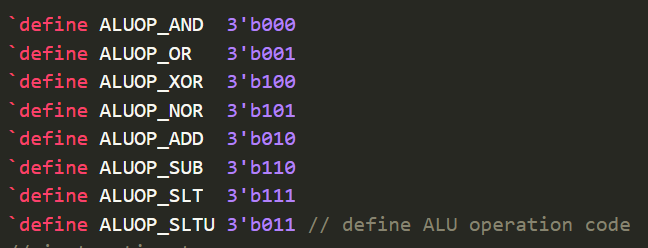
注1：撰写此Word格式实验报告后以PDF格式保存SERVE CloudIDE的/home/serve-ide/cod-lab/reports目录下（注意：reports全部小写）。文件命名规则：prjN.pdf，其中“prj”和后缀名“pdf”为小写，“N”为1至4的阿拉伯数字。例如：prj1.pdf。PDF文件大小应控制在5MB以内。此外，实验项目5包含多个选做内容，每个选做实验应提交各自的实验报告文件，文件命名规则：prj5-projectname.pdf，其中“-”为英文标点符号的短横线。文件命名举例：prj5-dma.pdf。具体要求详见实验项目5讲义。

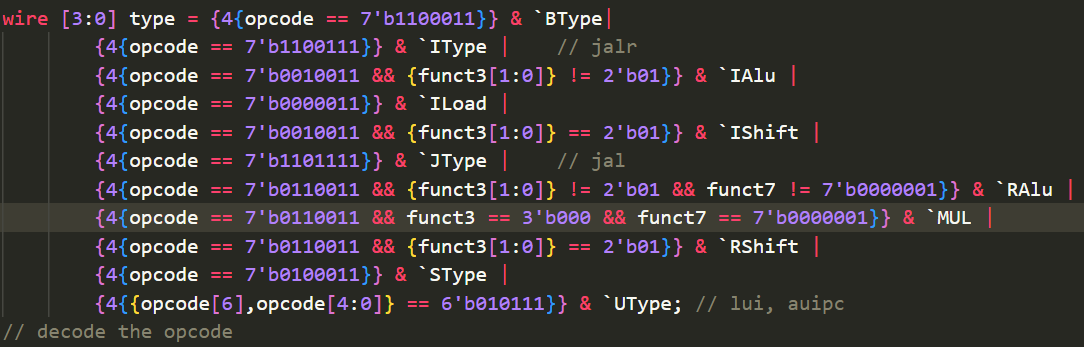
注2：使用git add及git commit命令将实验报告PDF文件添加到本地仓库master分支，并通过git push推送到SERVE GitLab远程仓库master分支（具体命令详见实验报告）。

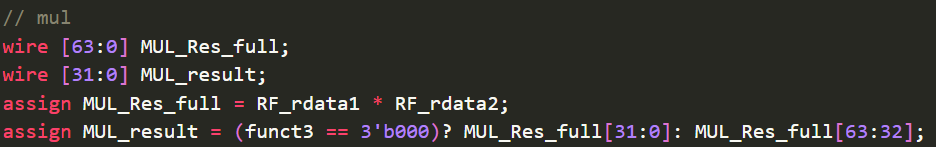
注3：实验报告模板下列条目仅供参考，可包含但不限定如下内容。实验报告中无需重复描述讲义中的实验流程。

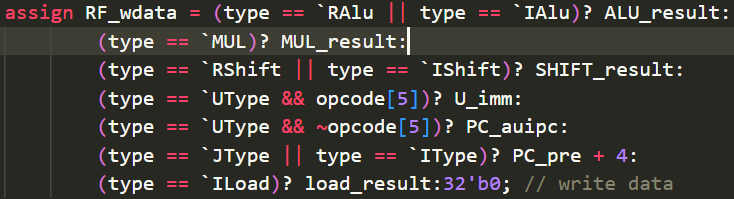
1. 逻辑电路结构与仿真波形的截图及说明（比如关键RTL代码段{包含注释}及其对应的逻辑电路结构图{自行画图，推荐用PPT画逻辑结构框图，复制到word中}、相应信号的仿真波形和信号变化的说明等）
2. 在基于RISC-V的custom\_cpu中添加MUL指令通路：

实验中并没有采用直接修改ALU模块的做法，原因是原来alu.v的译码器是3-8译码器，而原有alu已经拥有8种操作（如图），故再加入MUL操作需要改动的地方过多，因此实验中采用了另外加入一个MUL的指令类型（type），与Ralu类型指令并列的做法。



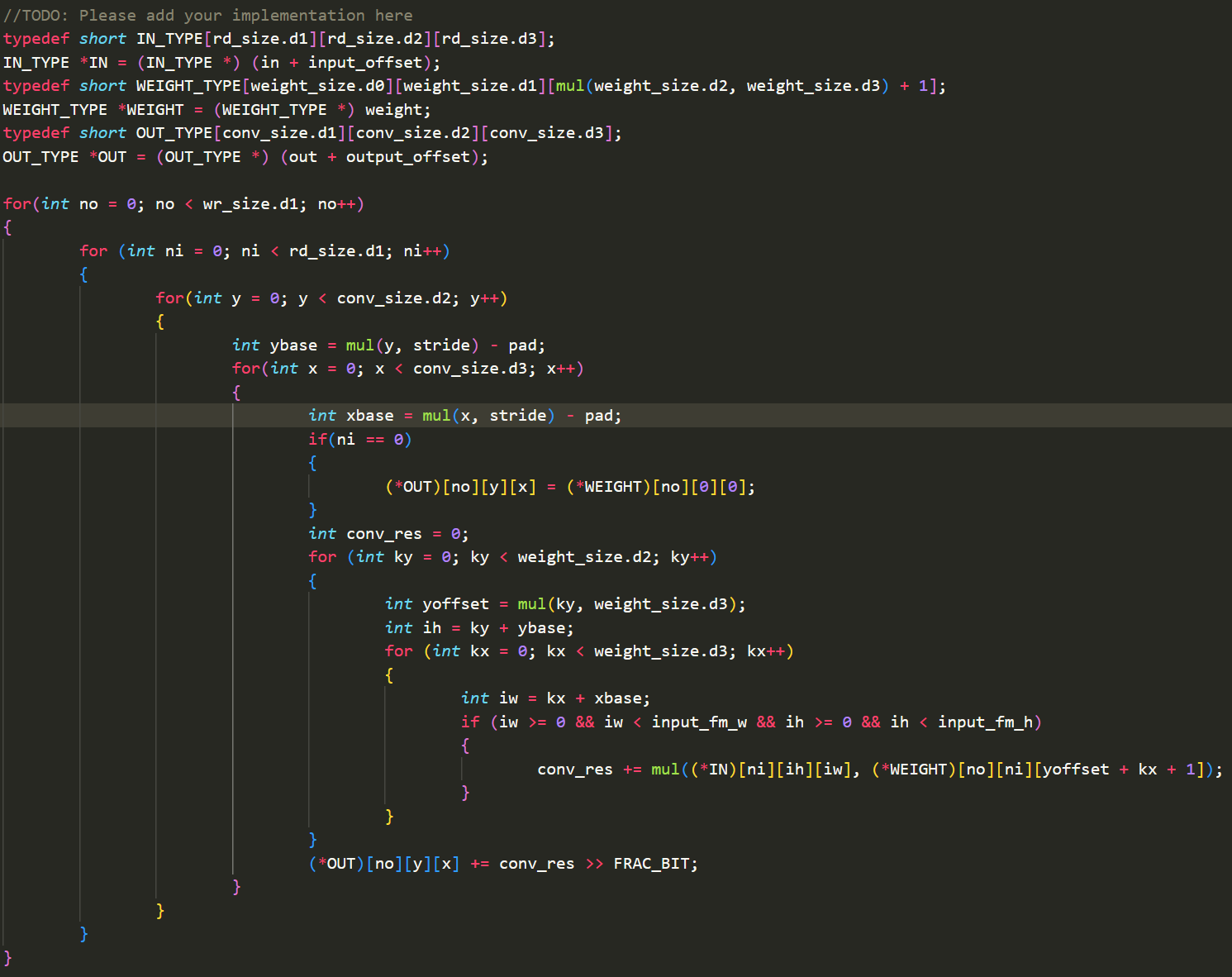






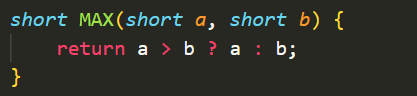
1. 修改测试软件代码
2. 卷积算法

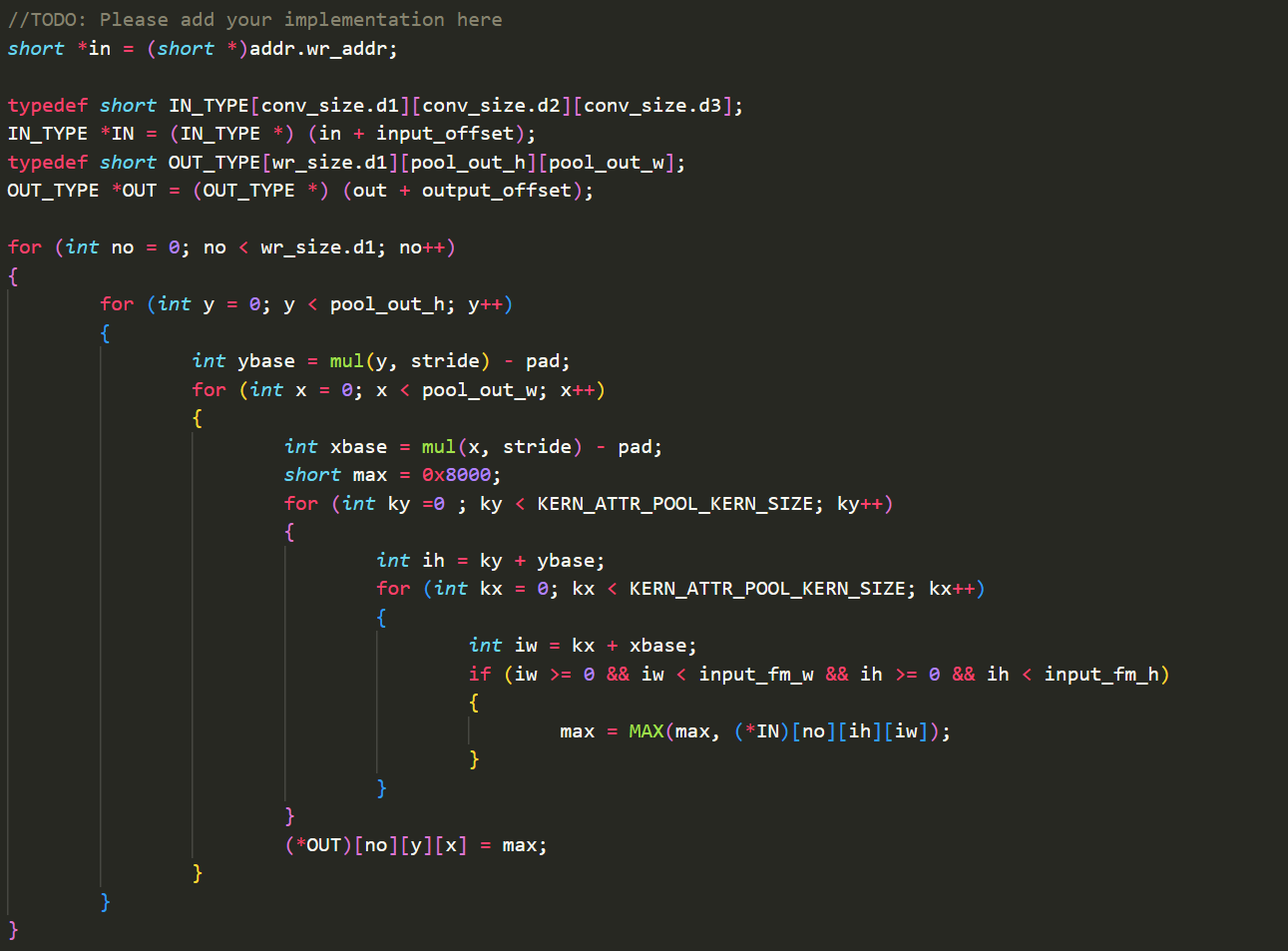
用多层嵌套循环的方式，将二维的卷积分成若干次的一维卷积，并设置累加数组（conv\_res），将结果累加：



1. 池化算法

池化算法与卷积算法思路大致相同，只是将关键的累加步骤替换为最大值的更新，因此还需要提前定义一个用于取最大值的MAX()函数：





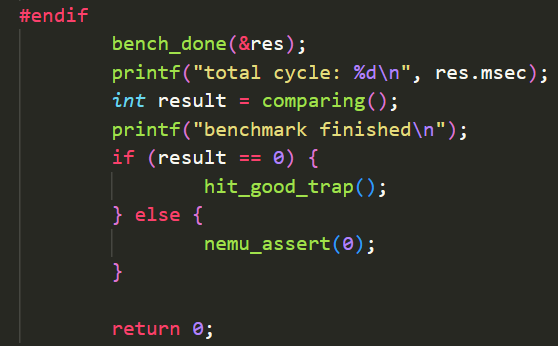
1. 实验过程中遇到的问题、对问题的思考过程及解决方法（比如RTL代码中出现的逻辑bug，逻辑仿真和FPGA调试过程中的难点等）
2. TIME OUT!!!

初期的代码存在很多重复计算的问题，将只需要计算一次的操作嵌套进了多次循环中，因此每次上板测试都出现超时。

另外，还有一种解决方法是将三维数组的运算用指针与地址取代，化三维为一维，大大缩短时钟周期数。

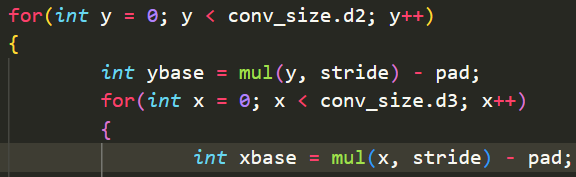
1. 关于性能计数器

给出的代码模板没有引用perf\_cnt.h头文件，需手动添加；打印性能计数操作需要在hit\_good\_trap()之前完成，否则将出现错误。



1. 对讲义中思考题（如有）的理解和回答
2. 请思考如果使用边界填充，算法应如何修改。

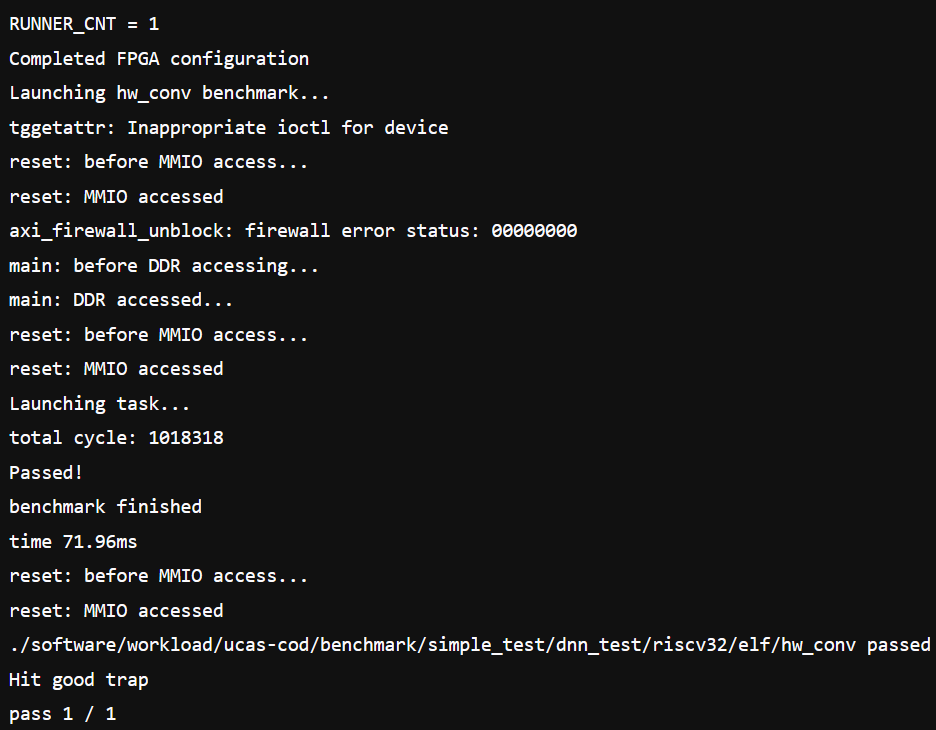
答：在对每个通道卷积或池化时，均考虑了pad和stride的影响，只需要根据需要在代码最前边的宏定义中修改二者的值。



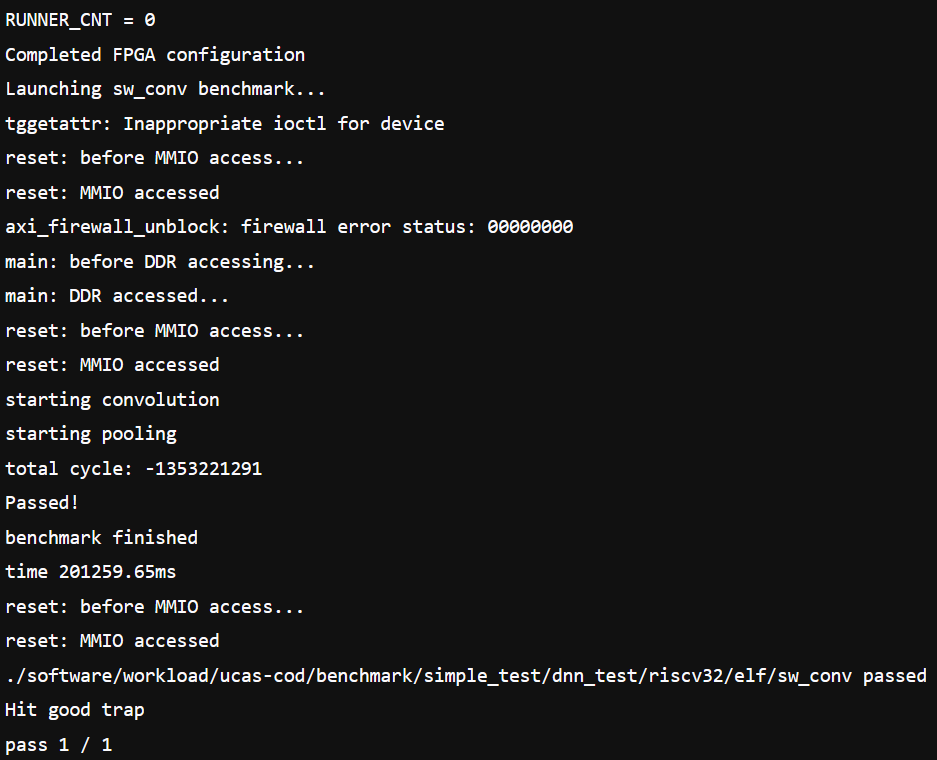
1. 在软件算法实现中，需考虑如何避免出现溢出和精度损失。乘法、加法运算的中间结果可使用32-bit定点数来表示，请同学们思考如何扩展？

答：在对conv\_res累加的时候，不考虑移位，在所有结果累加后跳出循环，再进行移位。

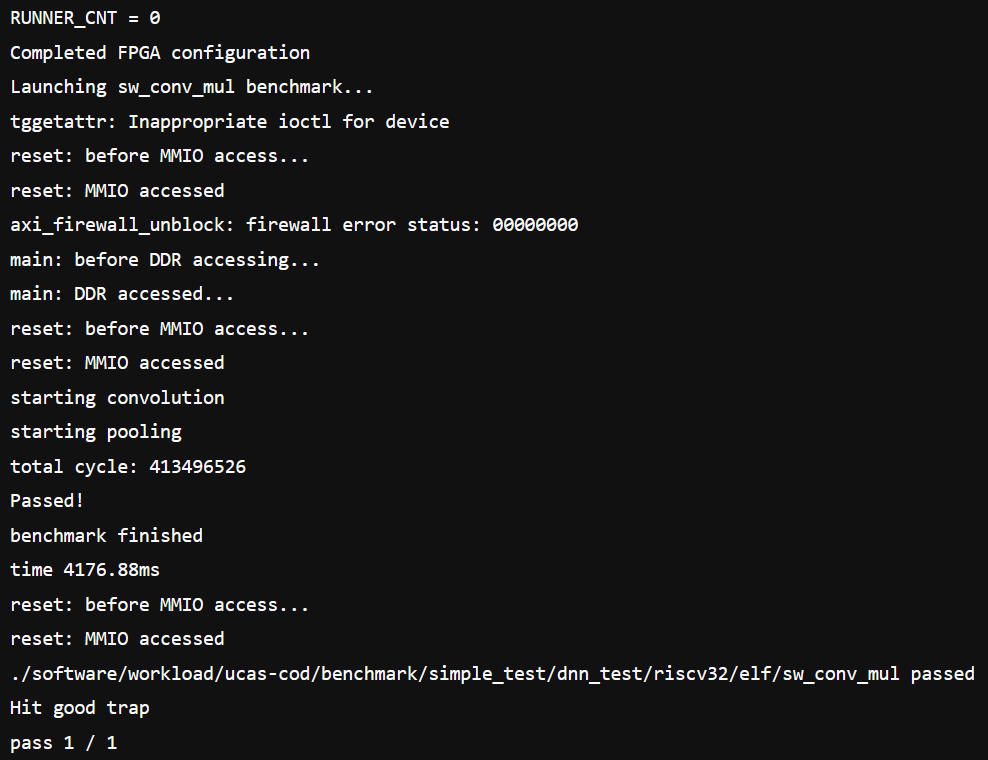
1. 分析不同实现方法的性能差异

答：hw\_conv：

sw\_conv：



sw\_conv\_mul：



可以看到，不使用mul指令实现乘法的软件方法的时钟周期数发生了溢出，说明其时钟周期数远大于另外两个方法。而使用mul指令乘法的软件方法所需的时钟周期数是硬件方法的406倍，由此可见利用硬件加速器可以较好提高深度学习算法的效率。

1. 在课后，你花费了大约10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）

本次实验资料讲义讲解较为详尽，因此总体难度不大。只是由于平台提供的调试工具有限，一旦代码出现bug需要花费较长的时间进行调试。通过本次实验我了解了深度学习的基础算法，并且对硬件加速器有了更深的理解。