架构说明与分工明细

马晋

2025 年 5 月 23 日

目录

1 分工

1.1 马晋

- 完成了基本的 MAC 模块 (参考了多种版本 FP 的比较: 排除了 FPnew，因为 FPnew 冗余过多及实现简单,pipeline 方式低效)
- MAC 模块 FP32 的 pipeline+MAC 模块中的加法部分 (同时支持 4bit 和 8bit)+FP16toFP32.sv
- transformtoAXI.sv+addrgen.sv+systolic.sv+control.sv+ENABLE.sv+tensorcore.sv
- 对整体的 tensorcore 进行了测试以及 debug, 包括找到部分 MAC 的错误
- 全局统筹项目

1.2 李亚飞

- 完成了 tensorcore 中 burst_num 和 burst_size 以及 systolic_time 和 waitwrite_time 的计算并加入
- 完成了各种 axi 的逻辑, 以及最后在 PE 中加入了写回到 AXI 的逻辑，并组装成整体的模块
- 完成了 FP16toFP32 模块 +FP32toFP16.sv 从而组合出 MAC_FP
- 组装了 MAC_FP 和 MAC_adder 从而完成顶层 MAC, 同时支持多精度 FP,INT
- 对整个 MAC 模块进行了测试以及 debug
- 协调统筹项目

1.3 王俊涵

- 完成了 mult_INT4(自写 testbench),mult_INT8(自写 testbench),CLA(自写 testbench),MAC(testbench 由马晋提供) 的验证
- 完成了 MAC 的综合 + 最终 top 的综合

1.4 补充

- 马晋和李亚飞参与了初步的架构构建, 在这个过程中马晋提出了采用分块乘法并在最后阶段做加法的想法, 两人确定了其中部分参数
- 马晋自己完成了最终的架构构建: 考虑了数据排布的影响, 否决了前面提出的矩阵乘法的顺序

- 在最终的架构构建阶段, 考虑将 systolic 与 PE 的复杂性解耦合, 用 addrgen 专门生成数据取地址, 大大提高了效率
- 马晋可能会考虑继续对于架构进行优化, 目前在 INT4 与 INT8 的情况下, 算力 » 带宽, 但是很容易将架构进行扩展, 利用多个 SRAM 使得算力得到充分运用

1.5 贡献度: 仅供参考

- 马晋: 70%
- 李亚飞: 25%
- 王俊涵: 5%