

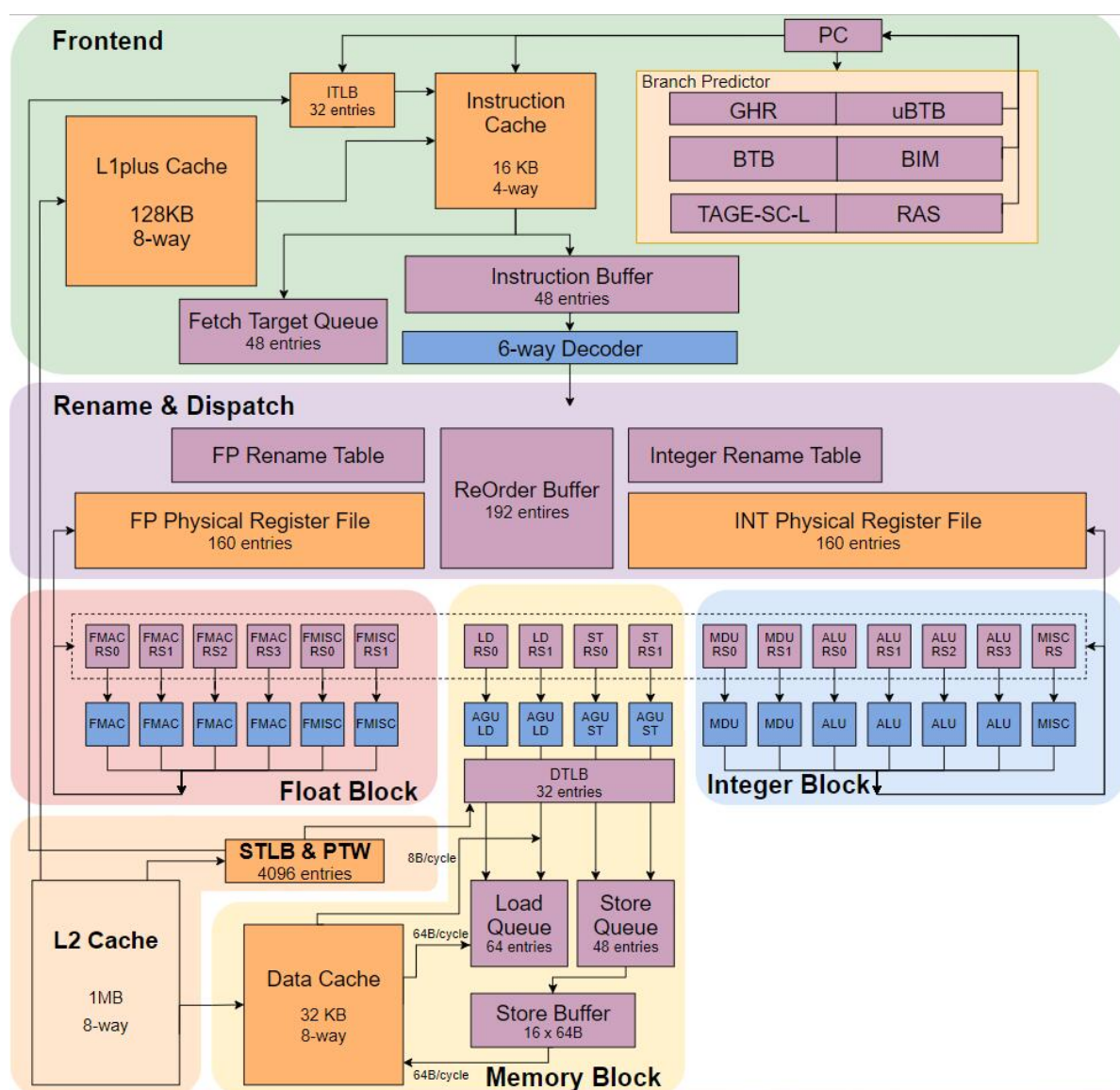
XiangShan Cache 第一次作业

蔡洛珊 2018K8009929051

一、XiangShan简介

香山是一款由中国科学院计算所和鹏城实验室联合发起的高性能开源RISC-V处理器项目。目前，香山处理器的第一版（雁栖湖）已经投片，第二版南湖还在开发，因此我在本次课程的大作业中主要阅读第一版的代码。香山采用 Chisel 硬件设计语言开发，Chisel将面向对象和函数式语言之类的软件工程优势引入数字系统设计中，以scala语言为基础，表示同步的数字电路，其代码本身是scala程序，在编译运行后生成verilog描述的电路。由于Chisel带来的一些敏捷设计的特征，香山的开发过程本身也是在验证高性能处理器是否能更加敏捷地开发，而我也将在阅读源码的过程中学习和分析面向对象思想为硬件敏捷开发带来的具体优势。

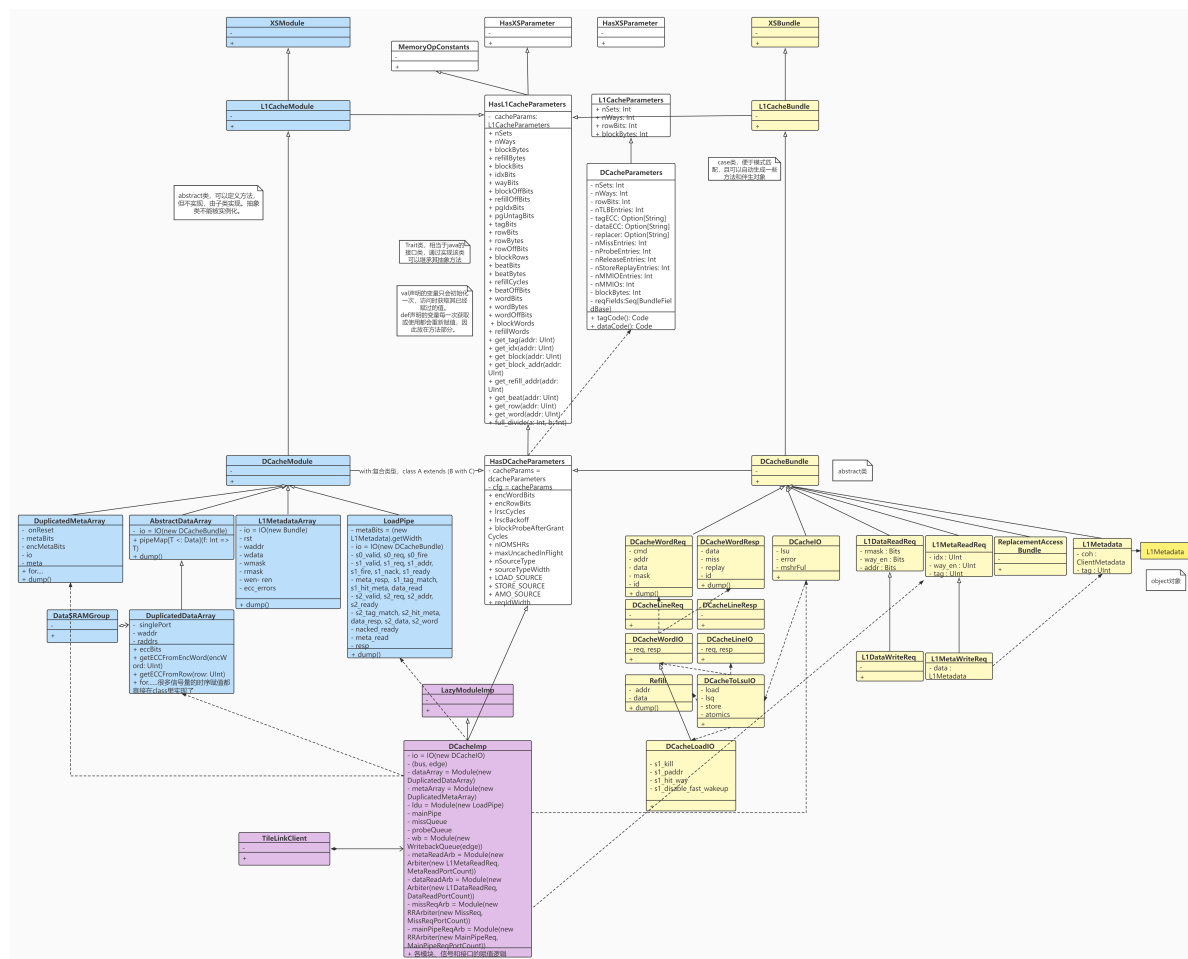
主要功能：“雁栖湖”架构是一个11级流水、6发射、4个访存部件的乱序处理器核，微架构图如下：



可以分为前端、后端、访存三个主要功能模块。前端包括取指、分支预测、指令缓冲等单元，顺序取指。后端包括译码、重命名、重排序缓冲、保留站、整型/浮点寄存器堆、整型/浮点运算单元。访存子系统按照load和store分割开，包括两条load流水线和两条store流水线，以及独立的Load Queue和Store Queue，Store Buffer等，缓存包括L1Cache(ICache、DCache)、L2Cache、TLB和预取器等模块，在访存部件内。

二、XiangShan Cache 设计类关系

我主要选择L1Cache模块进行阅读，并梳理其类关系。L1缓存包括指令缓存 ICache 和数据缓存 DCache，二者结构相似只是 DCache 的访问模式比 ICache 更加丰富，指令是只读不可写的且位宽固定，而数据可读可写且有字节访问、字访问和行访问等形式。以DCache为例，类图如下：



三、XiangShan Cache 功能分析

缓存位于CPU和主存之间，是规模较小但速度很高的存储器。Cache可以存储CPU刚用过或循环使用的一部分数据，如果CPU需要再次使用该部分数据时可从Cache中直接调用，减少了CPU从内存获取数据的等待时间，从而提高系统的效率。

ICache用于存放CPU最近访问的指令，DCache用于存放CPU在最近Load/Store指令中访问的数据。在上述类图中，可以看到DCache的源码中涉及近50个类，可以大致分为用于作为模块被继承的abstract类Module，用于作为接口传入参数的trait类Parameters，用于作为IO请求的继承的abstract类Bundle和用于作为实例被实现的实现类Imp。它们首先从顶层中继承 `xs...` 或从现有的开源rocket-chip、berkeley-hardfloat、SiFive block-inclusivecache等代码中继承 `LazyModule` 的属性和方法，然后定义L1Cache的通用属性，再由此扩展到DCache特有的 `dataArray`，`MetadataArray` 等模块，`HasDCacheParameters` 等参数，`DCacheWordReq`，`WordResp`，`LineReq`，`LineResp` 等IO请求，最后由 `DCacheImp` 类调用这些类，连接相应接口，为模块信号赋值，生成真正的DCache。这样DCache就可以根据CPU发来的访存req查找dataArray中是否存在对应的数据，若命中则完成读写返回resp，若缺失则向下一级cache或DRAM发送读写请求，查找对应数据并完成替换。