

# 为SimpleCPU增加Sv39模式TLB支持

---

申请人：陈泓佚

项目编号：210060193

项目导师：陈国凯

## 项目任务

为使用Chisel语言实现的SimpleCPU增加Sv39模式TLB支持。RISC-V的TLB为硬件管理，包含Page Table Walker模块，需要在TLB Miss的时候，由硬件对内存中的页表数据结构进行逐级地访问，完成地址的翻译。

## 项目详细方案

---

### 1. Chisel语言的学习

Chisel是一门敏捷硬件开发语言，其规避了Verilog HDL的部分不合理的设计，对于硬件设计更加优雅而友好。本项目采用Chisel语言，Chisel基于Scala，其入门门槛较高，需要花费一部分时间进行学习，初步打算是先过完Chisel-Bootcamp和Chisel-Book中的内容，然后通过阅读SimpleCPU的代码，逐步熟悉Chisel语言的使用。

### 2. 熟悉RISC-V指令集特权态虚拟内存管理部分

题目中要求的虚拟内存管理机制为Sv39，在RISC-V指令集手册卷2中有详细的描述。需要阅读RISC-V指令集手册中相关的部分，归纳出相应的算法后，进行硬件层面的设计。

### 3. 实现硬件PTW

基于上一点，RISC-V处理器的TLB中需要有硬件PTW机制，即使用硬件按照Sv39规定的页表格式进行虚拟内存的访问。首先需要了解SimpleCPU中内存总线接口部分的代码，然后Chisel语言编写硬件Page Table Walker，在没有TLB的情况下，测试硬件Page Table Walker实现的正确性。

### 4. 实现TLB

实现了PTW之后，就可以开始实现TLB。TLB事实上是作为PTW的Cache而存在的。在Rocket处理器中，TLB的结构比较复杂，分为好几级，能够缓存不同层级的页表。在SimpleCPU上实现TLB，有助于提升Page Table Walk的效率，避免对于内存的过多访问。

## 项目开发时间计划

---

7月1日-7月10日：熟悉SimpleCPU代码结构

7月10日-7月25日：学习Chisel语言，熟悉RISC-V体系结构

7月25日-8月15日：硬件PTW的实现与测试

8月15日-8月30日：硬件TLB的实现与测试

9月1日-9月30日：后期的优化和报告的编写

