

# NSCSCC 2022 初赛设计报告

东北大学秦皇岛分校 1 队  
易一泓、吴桐、左金徽、谭歌

## 1、设计简介

本次比赛 CPU 采用顺序单发射七级流水,采用哈佛结构,实现了初赛要求的 57 条指令、分支预测、中断例外和与外界交互的 AXI3 接口。并设计了支持 burst 传输的 L1-cache。

## 2、设计方案

### 2.1 总体设计思路

七级流水核心部分位于文件夹 imports→news 下,顶层模块 mycpu\_top.v 位于文件夹 new 下, my cpu 通过 sram-like 接口与外界进行交互,从转换桥将接口信号转化为 AXI3 的接口,集中 AXI 的访存请求仲裁处理以后以 AXI 的接口信号与 my cpu 外界交互。

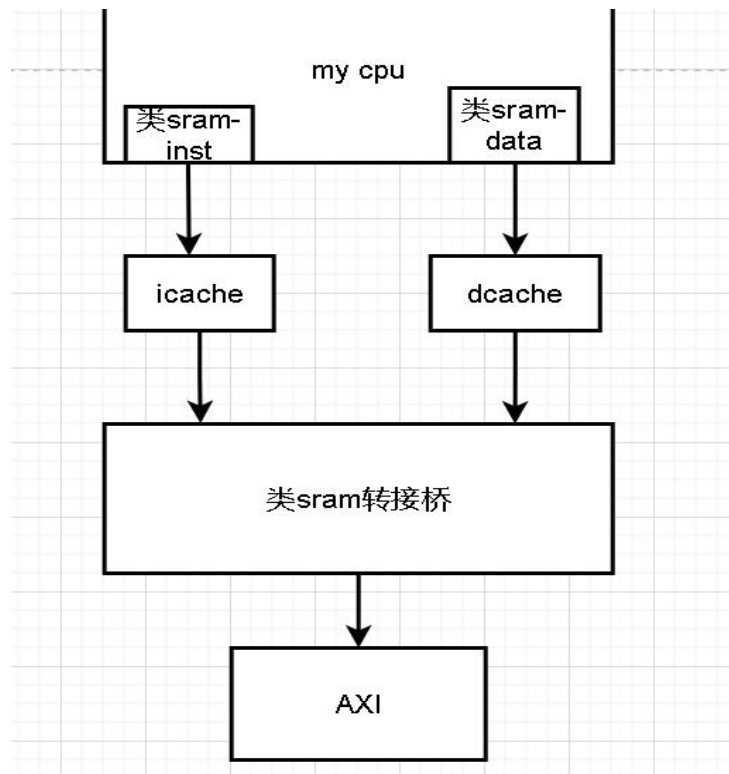


图 1、顶层结构图

### 2.2 CPU 流水线设计

不同于一般的五级流水线设计,本次设计的 CPU 流水线在一定范围内将占用时间比

---

较长的译码和访存流水段进行了拆分，设计成了七级流水线，以期获得比传统五级流水线更高一些的主频。设计的流水线也成功完成了流水线取指，译码，执行，访存，写回功能，也实现了处理异常相关指令的协处理器 CP0，hilo 寄存器等等。

### 2.2.1 取指阶段 fetch

此部分根据传入的取值地址发出取值请求，从存储器中取出下一条待执行的指令。

地址产生条件：复位信号保证 next\_pc 为 0xbfc0000。

### 2.2.2 译码阶段 decode

此部分接收指令并对指令进行解码。

暂停：在译码第一阶段，这一部分主要将上一级流水线传来的信息接受传递至 decode\_pt1 模块。取值模块未暂停时，正常向下传递信息；取指模块暂停而译码模块未暂停时，传递信号为 0；取指模块和译码模块都暂停时，传递信号不变。

相关性检测：在译码第二阶段，这一部分以数据前推来判断是否有数据。

### 2.2.3 执行阶段 execute

此部分主要负责逻辑运算与算数运算，并传递到下一阶段进行数据选择；数据前堆。

### 2.2.4 访存阶段 memory

此部分主要负责接收访存响应，访问存储内存数据，生成最终结果。

访存第一阶段：传递读写请求，在下一周期进行传递。

访存第二阶段：保存上一周期的 flush、resetn、stall 信息，用于数据选择。

data\_tran 模块：存储前一周期的数据，根据 flush、resetn、stall 信息进行数据选择。

### 2.2.5 回写阶段 write back

此部分主要负责通用寄存器的写入，协处理器 CP0 的读写。

### 2.2.6 类 sram 接口

State	保存是否处于传输过程中，当处于传输过程中 req 为 0；
addr_ok	进入传输状态
data_ok	跳过传输状态

## 2.3、分支预测设计

在取指阶段根据取指地址 PC 查询并得到预测结果，然后将预测结果传递到译码阶段。

在译码阶段，会根据预测的结果来决定是否要进行跳转。

在执行阶段，判断分支指令的实际跳转方向。

在访存阶段，根据分支指令的实际跳转方向，判断分支预测是否预测成功。若预测失败，则清空取指跟译码两个阶段的流水线，置为正确的跳转地址

## 2.4、AXI 设计

使用 xilinx 的 ip 核 AXI crossbar 来处理 CPU 的对外总线，AXI crossbar 会对 CPU 中的 axi 总线信号进行仲裁，并在仲裁之后将总线信号发送到 CPU 外部的总线。AXI crossbar 放置于文件夹 ip 下。

Number of Slave Interfaces	2	▼
Number of Master Interfaces	1	▼
Crossbar Optimization Strategy	Current Settings	▼
Protocol	AXI3	▼
Address Width	32	⊗ [12 - 64]
Data Width	32	▼
ID Width	4	▼

图 2、AXI crossbar 设定参数

## 2.5、cache 设计

### 2.5.1 I-cache

采用 4 路组相联的结构，组相联使得一个数据块可以放在多个 cache line 中，通过合理的替换算法，可以大大降低 cache 的冲突缺失率。每路容量为 4KB。

1 个 cache 行中包含 8 个 data bank，LRU 替换算法。

### 2.5.2 D-cache

采用 4 路组相联的结构，组相联使得一个数据块可以放在多个 cache line 中，通过合理的替换算法，可以大大降低 cache 的冲突缺失率。每路容量为 4KB。

1 个 cache 行中包含 8 个 data bank，LRU 替换算法，写回写分配。写回策略使得只有 cache 处在脏状态且被替换回内存时才需要访问内存，提升了一定效率。

### 3、设计结果

实现了顺序单发射七级流水

主频：80Mhz

#### 3.1 设计交付物说明

| -my cpu/ 目录，本次 CPU 设计整体。

| | --imports/ 目录，存放源码。

| | | --sources/ 目录， 存放源码。

| | | | --imports/ 目录， cache 与 cpu 核心部分的源码。

| | | | --new/ 目录， 类 sram 接口。

| | | | --xilinx\_ip/ 目录， Xilinx IP， 包含 clk\_pll、inst\_ram、data\_ram。

| | --ip/ 目录，本次设计所用所有 IP 核，包含 AXI crossbar 和 Bank RAM。

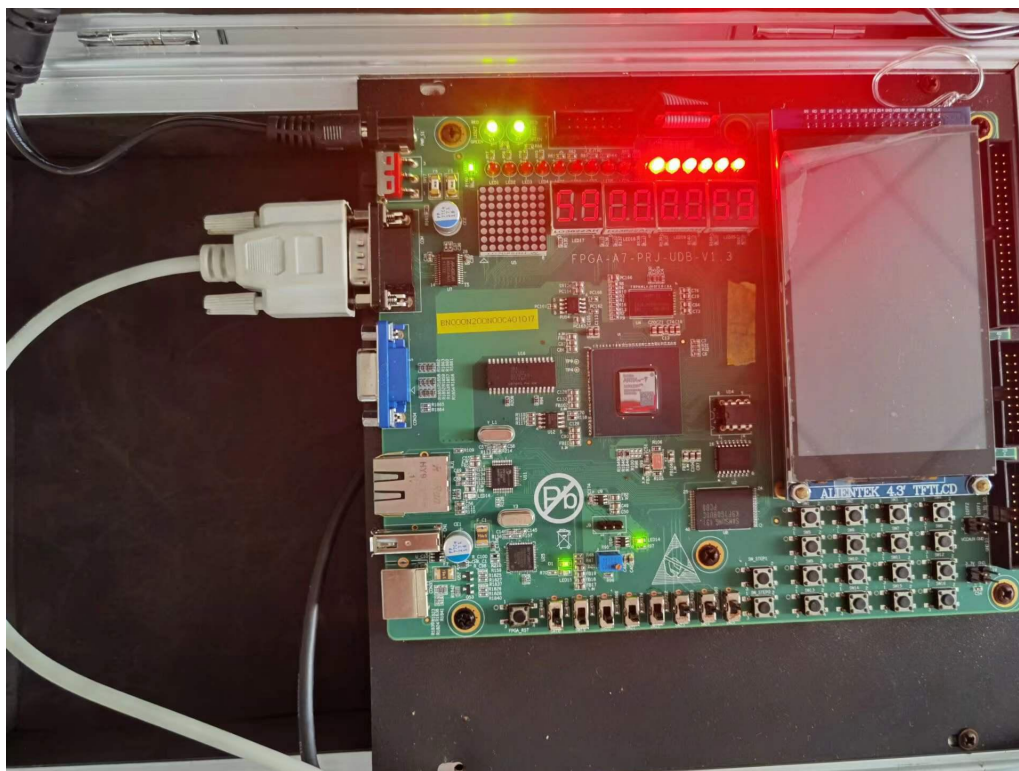


图 3、上板演示

#### 3.2 设计演示结果

---

本 CPU 通过了 AXI 接口下的 89 个功能测试点、通过了记忆游戏测试、通过了 10 个性能测试点、通过了系统测试。

## 4、参考设计说明

设计中使用的 IP 核：

AXI crossbar (2.1) —— Xilinx

Bank RAM —— Xilinx

## 5、参考文献

- [1] MIPS32 Architecture For Programmers Volume I: Introduction to the MIPS32™ Architecture
- [2] MIPS32 Architecture For Programmers Volume II: The MIPS32™ Instruction Set
- [3] MIPS32 Architecture For Programmers Volume III: The MIPS32™ Privileged Resource Architecture
- [4] 姚永斌 . 超标量处理器设计. 清华大学出版社, 2014.
- [5] 汪文祥, 邢金璋. CPU 设计实战[M]. 北京: 机械工业出版社, 2021.