

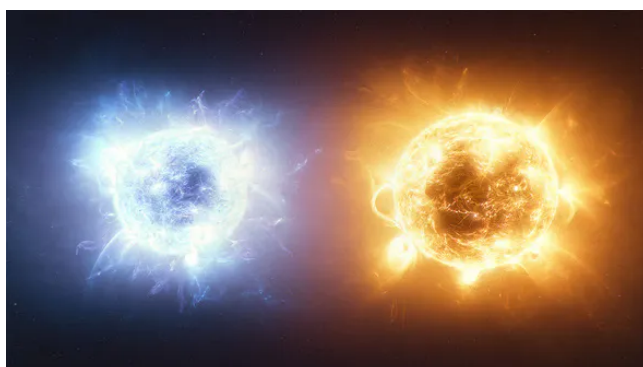
“龙芯杯”第六届全国大学生计算机系统能力培养大赛

重庆大学“所以延迟槽会消失对不队”队

---

## CDIM 项目 决赛设计报告

---



陈泱宇

cyy@cyysf.name

李燕琴

maxpicca@qq.com

王梓宇

925136384@qq.com

张翀

20194159@cqu.edu.cn

2022 年 8 月

# 目录

<b>第 1 章 概述</b>	<b>1</b>
1.1 项目背景 . . . . .	1
1.2 名词解释 . . . . .	1
1.3 项目概述 . . . . .	1
<b>第 2 章 CPU 设计方案</b>	<b>2</b>
2.1 双发策略 . . . . .	2
2.2 数据通路设计 . . . . .	2
2.3 冲突处理 . . . . .	2
2.4 CP0 寄存器设计 . . . . .	2
2.5 中断和异常 . . . . .	2
2.6 缓存设计 . . . . .	2
<b>第 3 章 仿真验证框架</b>	<b>3</b>
<b>第 4 章 操作系统支持</b>	<b>4</b>
<b>第 5 章 Soc 设计</b>	<b>5</b>

# 第 1 章 概述

## 1.1 项目背景

本项目依托于龙芯杯提供的 FPGA 实验平台、Soc 工程环境以及基准测试程序，设计并实现了一个部分兼容 MIPS32 体系结构的小端序 CPU，名为 CDIM（CQU Dual Issue Machine），其能成功通过龙芯杯提供的功能测试、性能测试、系统测试，具有较完善的运算处理、AXI 访问、异常处理、中断响应等功能，并能够运行 u-boot 引导程序、uCore 操作系统和 Linux 操作系统等。

## 1.2 名词解释

本项目中可能用到的一些名词缩写及其解释如表 1.1 所示。

表 1.1: 名词缩写和解释

名词缩写	全称	解释
MIPS	Microprocessor without Interlocked Pipeline Stages	无内部互锁流水级的微处理器
SOC	System On a Chip	片上系统
MIPS	Microprocessor without Interlocked Pipeline Stages	无内部互锁流水级的微处理器
SOC	System On a Chip	片上系统
CPU	Central Processing Unit	中央处理器
ALU	Arithmetic Logic Unit	算术逻辑单元
GPR	General Purpose Register	通用寄存器
CP0	Co-Processor 0	协处理器 0
BRAM	Block Random Access Memory	块随机访问存储器
FIFO	First In First Out	先进先出
RAW	Read After Write	写后读
WAW	Write After Write	写后写
WAR	Write After Read	读后写

## 1.3 项目概述

CDIM（CQU Dual Issue Machine），采用对称双发射五级顺序流水线的设计，支持指令 FIFO、分支预测、指令缓存和数据缓存等特殊单元，以提升系统性能。其中双发射，采用对称双发逻辑以充分保证双发率；五级顺序流水线由取指（Instruction Fetch）、译码（Instruction Decode）、执行（Excute）、访存（Memory access），写回（Write Back）五个阶段组成；指令 FIFO 可以隔离取指阶段和后续阶段，以实现高效取指的作用；指令缓存和数据缓存均采用二路组相联和突发传输的设计，单路均为 4KB 以匹配 TLB 页面要求，其中指令缓存一行为 64bit 以适应双发取指要求，数据缓存一行为 32bit。**此部分尚需完善**。此外，CDIM 还支持 U-Boot 引导程序，并基于该引导程序，成功运行 uCoreh 和 Linux 操作系统。

## 第 2 章 CPU 设计方案

CDIM 采用对称双发射顺序执行，共有 5 级流水。CDIM 的数据通路示意图如下，其中红线部分为 master path，蓝线部分为 slave path。在对称双发射中，master 和 slave 的主要区别在于前者 PC 小于后者，在处理异常、提交等事宜时会被优先处理。当然，上述的“对称”双发射，只是相对于只支持 ALU 指令双发的非对称逻辑而言，是对称的，因为 slave path 支持的指令更多。但严格来讲，并不是绝对对称，在处理跳转指令、例外指令、写 TLB 指令等会刷新流水线的指令时，需要优先在 master 处理，这在后续的双发策略中会详细说明。

### 2.1 双发策略

### 2.2 数据通路设计

#### 2.2.1 取值阶段

#### 2.2.2 译码阶段

#### 2.2.3 执行阶段

#### 2.2.4 访存阶段

#### 2.2.5 写回阶段

### 2.3 冲突处理

#### 2.3.1 数据冲突

#### 2.3.2 结构冲突

#### 2.3.3 控制冲突

### 2.4 CP0 寄存器设计

### 2.5 中断和异常

### 2.6 缓存设计

## 第 3 章 仿真验证框架

## 第 4 章 操作系统支持

## 第 5 章 Soc 设计

## 参考文献

- [1] MIPS® Architecture For Programmers I, II, III. Imagination Technologies LTD.
- [2] 计算机组成与设计: 硬件/软件接口. David A.Patterson