同濟大學
TONGJI UNIVERSITY

《计算机系统结构》研究学习报告

RISC-V 处理器架构技术发展和性能分析
——以玄铁 C906 处理器为例

成　员　林继申、朱昱瑾、陈君、奥泉瑞

学　号　2250758、2250693、2250420、2250695

学　院　软件学院

专　业　软件工程

班　级　42036803

教　师　王冬青

二〇二四 年 五 月 三十一 日

小组成员

姓名	学号	电子邮箱	班级	工作量
林继申	2250758	2250758@tongji.edu.cn	42036803	25%
朱昱瑾	2250693	2250693@tongji.edu.cn	42036803	25%
陈君	2250420	2250420@tongji.edu.cn	42036803	25%
奥泉瑞	2250695	2250695@tongji.edu.cn	42036803	25%

摘要

随着科技的飞速发展，特别是在物联网、人工智能和高性能计算等领域，对处理器性能和灵活性的需求不断增加。RISC-V 作为一种开源指令集架构（ISA），因其高度的可定制性和开放性，近年来备受关注。RISC-V 的开放性允许任何组织或个人自由地使用和修改其架构，极大地促进了全球范围内的技术创新和知识共享。本文以玄铁 C906 处理器为例，详细分析了 RISC-V 处理器架构的技术发展、核心架构及其性能表现，探讨了其技术创新、优势以及在现代计算环境中的应用潜力。

文章首先介绍了 RISC-V 架构的历史背景和发展历程，指出其源自 1980 年代初期的 RISC 理念，并经过多年的发展和演变，最终由伯克利加州大学的学者团队在 2010 年推出。RISC-V 架构的简洁高效设计支持 32 位、64 位及 128 位处理器设计，具有灵活的子集扩展方式，适用于多种应用场景。随后，文章深入探讨了 RISC-V 架构的技术特点，包括其模块化的指令集设计、寄存器集和存储器访问方式等。

在具体实例分析中，本文选取了玄铁 C906 处理器进行详细介绍。这款由阿里巴巴旗下的平头哥半导体公司设计的处理器，采用 RISC-V 架构，具备高性能和低功耗的特点。通过技术规格分析，文章揭示了玄铁 C906 在计算效率、功耗管理和扩展能力等方面的优势。此外，本文还将玄铁 C906 与其他 RISC-V 处理器进行了比较，探讨了其在技术规格、性能和应用场景方面的优势和劣势。

最后，本文展望了 RISC-V 架构的未来发展和应用前景。随着越来越多的公司和研究机构加入 RISC-V 生态系统，这一架构有望在更广泛的应用领域中得到推广和应用，特别是在物联网和人工智能领域，RISC-V 处理器凭借其高效、低功耗和灵活的特点，具有巨大的市场潜力。总之，RISC-V 以其开放创新的特性，为未来处理器的发展和应用提供了新的方向和可能性。

关键词：RISC-V；玄铁 C906；处理器架构；处理器性能

目录

第 1 章 引言

1.1 研究背景与意义

随着科技的飞速发展，尤其是在物联网（IoT）、人工智能（AI）和高性能计算（HPC）领域，对处理器性能和灵活性的需求不断增长。RISC-V 作为一种开源指令集架构（ISA），因其高度的可定制性和开放性，近年来在全球范围内受到了广泛关注。RISC-V 架构允许任何组织或个人自由使用和修改其设计，极大地促进了技术创新和知识共享。这种开放性不仅降低了研发成本，还加快了产品上市速度，使得 RISC-V 在学术界和工业界都展现出巨大的潜力。

RISC-V 的开源特性使其在各种应用领域中得到了广泛的应用，从微控制器到高性能服务器，无一不涉及。其模块化设计和灵活的子集扩展方式，使得 RISC-V 能够适应快速变化的技术需求，特别是在需要高度定制硬件解决方案的领域。本文将以玄铁 C906 处理器为例，详细分析 RISC-V 架构的技术发展、核心架构及其性能表现，探讨其技术创新、优势以及在现代计算环境中的应用潜力。

1.2 研究目的与问题

本研究致力于深入分析 RISC-V 处理器架构的技术发展，以及通过具体实例——玄铁 C906 处理器，探讨其技术规格、核心架构以及性能表现。本研究将解答以下核心问题：

1. 技术发展与创新：RISC-V 架构的技术发展历程是什么？它的主要技术创新点有哪些？
2. 核心技术与性能：玄铁 C906 处理器的核心技术规格和性能如何？它在处理效率和功耗管理方面表现如何？
3. 竞争优势与劣势：相比其他 RISC-V 处理器及传统架构处理器，玄铁 C906 的优势和劣势具体包括哪些方面？
4. 性能评估：在当前的技术环境下，RISC-V 架构处理器的性能如何？它们在实际应用中能够满足的需求有哪些？

通过对这些问题的深入探讨，我们期望能够全面理解 RISC-V 架构的技术优势和应用潜力，并具体评估玄铁 C906 在此架构下的实现效率和性能。这不仅有助于指导当前的技术开发者，也为未来的处理器设计和应用提供了宝贵的参考和启示。

1.3 论文构成

本文的结构安排如下：

- 第二章将介绍 RISC-V 架构的技术发展，包含其历史背景、发展历程及技术特点。
- 第三章将详细论述玄铁 C906 处理器，包括其核心结构、子系统及应用领域。
- 第四章将对 RISC-V 架构进行性能分析，具体介绍测试环境、测试方法、测试结果及性能对比分析。
- 第五章总结全文，并提出优化建议及未来发展展望。

通过以上内容的系统阐述，本研究将全面呈现 RISC-V 架构及其在玄铁 C906 处理器上的实际应用表现，为相关领域的研究与开发提供有力支持。

第 2 章 RISC-V 架构技术发展

2.1 RISC-V 简介

RISC-V 是基于精简指令集计算（RISC）原理建立的开放指令集架构（ISA），V 表示为第五代 RISC（精简指令集计算机），表示此前已经有四代 RISC 处理器原型芯片。每一代 RISC 处理器都是在同一人带领下完成，那就是加州大学伯克利分校的大卫·帕特森教授。与大多数 ISA 相反，RISC-V ISA 可以免费地用于所有希望的设备中，允许任何人设计、制造和销售 RISC-V 芯片和软件。它虽然不是第一个开源的指令集（ISA），但它很重要，因为它是第一个被设计成可以根据具体场景、可以选择适合的指令集的指令集架构。基于 RISC-V 指令集架构可以设计服务器 CPU，家用电器 cpu，工控 cpu 和用在比指头小的传感器中的 cpu。

2.2 RISC-V 架构的历史发展

RISC 的历史源远流长，始于 1980 年代初期，当时的学术界已经认识到简单而高效的计算机架构的潜在价值，但具体的设计原则尚未广泛讨论。随着时间的推移，这一概念逐渐成熟，直至 1990 年，为了出版《计算机体系结构：量化研究方法》一书，学者们设计了基于 RISC 原则的 DLX 指令集。该指令集由包括大卫·帕特森在内的伯克利学者团队提出，尽管仅用于教育目的，但它对后续的指令集开发产生了深远影响。

此后，RISC 架构的开发出现了新的动向。例如，OpenRISC 项目继承了 DLX 的设计理念，并在开源社区中得到了广泛应用，尤其是在 Linux 系统中。此外，早期的 ARM 处理器版本也采用了类似的开源策略，尽管这些设计并未在商业领域广泛使用。

到了 2010 年，随着开源硬件的兴起，伯克利加州大学的 Krste Asanović 教授和 David Patterson 教授等人发起了 RISC-V 项目，意图创建一个完全开源的指令集，不仅适用于学术研究，还能满足工业界的需求。RISC-V 项目得到了 DARPA 的部分资助，并很快吸引了包括 Google、华为、IBM 等多家大公司的支持和参与。

随着 RISC-V 的快速发展，2019 年，David Patterson 在瑞士宣布成立 RISC-V 国际开源实验室，旨在推动全球 CPU 产业的战略新方向，并以深圳为核心建立一个全球创新网络。

RISC-V 架构的一个主要优点是其简洁高效的设计，支持 32 位、64 位及 128 位的处理器设计，并提供了灵活的子集扩展方式。这些设计特性使其在嵌入式系统、个人电脑以及超级计算机等多种应用场景下都表现出色。同时，其开放的设

计理念也使得任何人都可以使用和改进这一指令集，极大地降低了软件和硬件的开发成本，促进了硬件生态系统的多样化发展。

2.3 RISC-V 架构的特点

2.3.1 完全开源

RISC-V 基金会采用宽松的 BSD 协议，无需高额授权费，企业可以自由使用且可添加私有指令集扩展，不必开放共享。

2.3.2 架构简单

与复杂的 x86 和 ARM 架构不同，RISC-V 架构摒弃过时定义，具有更少的指令和更简洁的文档（基础指令集 40 多条，规范文档 145 页），降低了新操作系统或应用的开发门槛。

2.3.3 易于移植

RISC-V 明确区分了特权级和用户级指令，提供详尽的指令规范，便于开发者移植 Linux 和 Unix 系统至 RISC-V 平台。

2.3.4 模块化设计

RISC-V 支持模块化指令集组合，适应不同应用场景需求，如嵌入式场景可选 RV32IC，高性能场景可选 RV32IMFDC，实现定制化设备设计。

2.3.5 工具链支持

RISC-V 社区提供完整工具链支持，降低 CPU 设计的复杂度。主流工具如 GCC 编译器和 QEMU 仿真器已支持 RISC-V，简化开发流程。

2.2 RISC-V 架构的技术特点与创新

2.2.1 指令子集

RISC-V 采用模块化设计的指令集，分为基本指令集和可选的扩展指令集。这些指令集是由科技产业、研究机构和学术界共同开发的。基本指令集涵盖了基础的指令编码、控制流、寄存器数量及其长度、存储器访问方式和逻辑运算等，足以支持一个简单的通用电脑系统。所有的基本和扩展指令集都设计为相互兼容，以确保在扩展使用时不会发生冲突。

名称	类别	说明
RV32I	基础指令	整数指令：包含算法、分支、逻辑、访存指令，有32个32位寄存器。能寻址32位地址空间
RV64I	基础指令	整数指令：包含算法、分支、逻辑、访存指令，有32个64位寄存器。能寻址64位地址空间
RV128I	基础指令	整数指令：包含算法、分支、逻辑、访存指令，有32个128位寄存器。能寻址128位地址空间
RV32E	基础指令	与RV32I一样，只不过只使用前16个（0~15）32位寄存器
M	扩展指令	包含乘法、除法、取模求余指令
F	扩展指令	单精度浮点指令
D	扩展指令	双精度浮点指令
Q	扩展指令	四倍精度浮点指令
A	扩展指令	原子操作指令：比如比较并交换，读改写等指令
C	扩展指令	压缩指令：单指令长度为16位，主要用于改善程序大小
P	扩展指令	单指令多数据（Packed-SIMD）指令
B	扩展指令	位操作指令
H	扩展指令	支持（Hypervisor）管理指令
J	扩展指令	支持动态翻译语言指令
L	扩展指令	十进制浮点指令
N	扩展指令	用户中断指令
G	通用指令	包含I、M、A、F、D指令

图 2.1 RISC-V 指令子集

2.2.2 寄存器集

RISC-V 指令集拥有 32 个整数寄存器，而在嵌入式版本中则减少为 16 个。若实现浮点扩展，则会有额外的 32 个浮点寄存器。与存储器访问指令不同，常规指令仅能访问寄存器，不直接访问存储器。在这些寄存器中，第一个整数寄存器被设定为“零寄存器”，其特性是写入无效且读取总是返回 0，这简化了指令集的设计。例如，可以通过将某个寄存器与零寄存器相加来复制数值到另一寄存器。此外，虽然 RISC-V 设有控制寄存器和状态寄存器，但用户模式下的程序只

能访问与性能测量和浮点管理相关的部分。RISC-V 没有提供用于上下文切换、中断处理或函数调用时保存和恢复多个寄存器的指令，认为这样的设计不必要、复杂且可能降低效率。

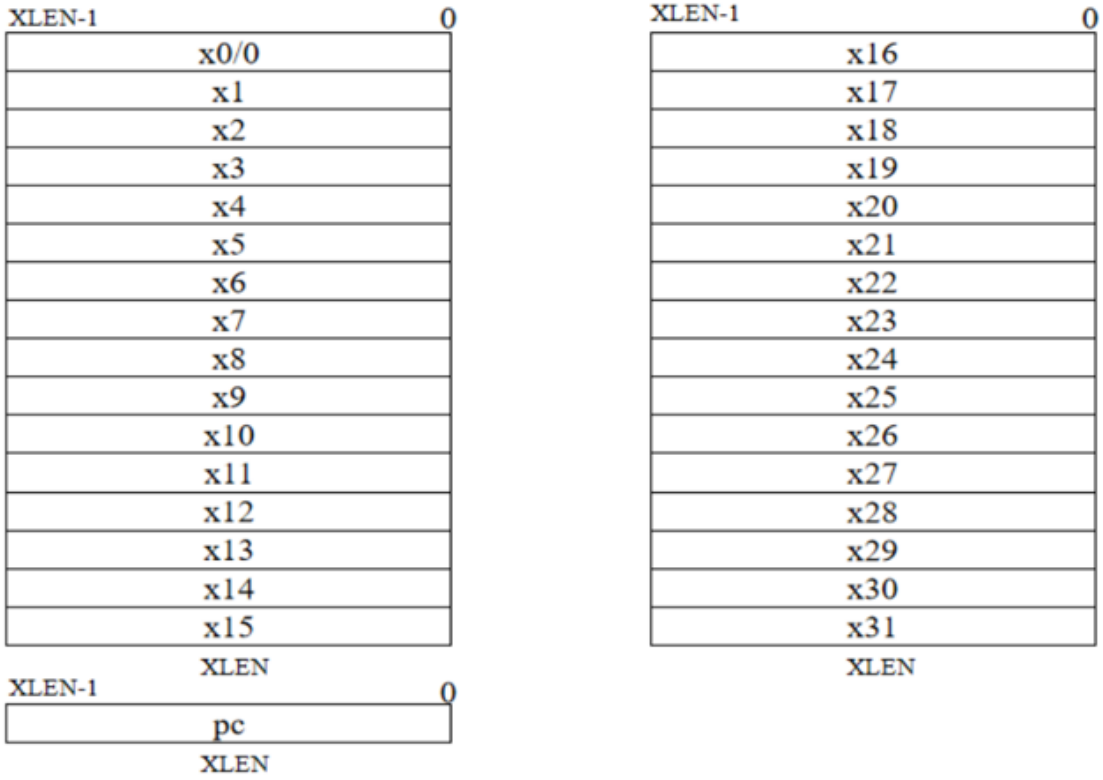


图 2.2 RISC-V 整数寄存器

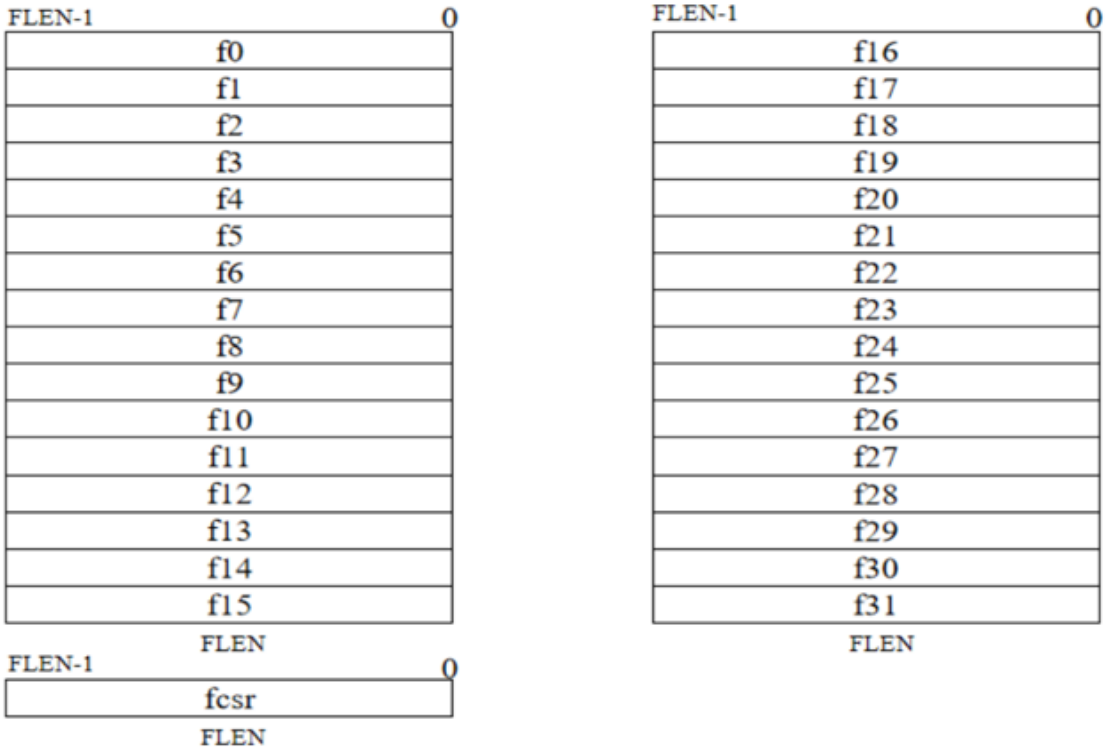


图 2.3 RISC-V 浮点寄存器

2.2.3 存储器访问

RISC-V 是一种加载-存储架构，其设计中仅有 load 和 store 指令直接与存储器进行交互，从而将处理器的运算与存储器访问分离，简化了指令集并提升了执行效率。这些指令通过使用基底寄存器与 12-bit 的有符号偏移量来访问内存，偏移范围为±2KB。如果基底寄存器为 0，能够方便地访问如 ROM 的固定存储内容。RISC-V 默认使用 little-endian 数据格式，尽管也支持 big-endian，但主流实现仍优先使用 little-endian。此架构允许非对齐的存储器访问，虽然可能触发对齐异常，但对齐访问仍可提升性能并减少硬件需求。此外，FENCE 指令在多线程环境中确保内存操作的顺序和可见性，是数据共享和设备 I/O 操作中不可或缺的。与 MIPS 或 PowerPC 的 16-bit 位移不同，RISC-V 的 Load/Store 指令使用 12-bit 位移，使指令更紧凑并支持灵活的编码。

2.2.4 立即数

在 RISC-V 架构中，读取 32-bit 常量和地址主要通过使用上位 20-bit 的指令实现。LUI（Load Upper Immediate）指令用于将 20-bit 立即数加载到寄存器的上位（31~12 位）。AUIPC（Add Upper Immediate to PC）指令则将这 20-bit 立即数加上程序计数器（PC）的值，存储到一个基底寄存器，以支持相对于代码位置的地址计算。这可以结合 12-bit 的偏移量在 Load 和 Store 指令中使用，以实现完整的 32-bit 地址访问。在 64-bit 系统中，LUI 和 AUIPC 操作的结果将自动扩展至 64-bit。此外，为了提高执行效率，高速 CPU 有时会将 LUI 和 AUIPC 指令与 Load/Store 指令融合成单一操作。

2.2.5 函数调用、跳跃和分支

RISC-V 中，函数调用通过 JAL（Jump and Link）指令实现，该指令将当前程序计数器（PC，即返回地址）保存到指定寄存器并跳转到目标地址，从而通过减少存储器访问提高执行速度。JAL 指令采用 20-bit 的有符号位移，通过左移一位后加到 PC，计算新的指令地址。另一指令 JALR（Jump and Link Register）则基于寄存器跳转，添加 12-bit 的位移到基底寄存器的值以计算跳转地址，支持复杂的跳转逻辑。

RISC-V 使用寄存器值比较来实现条件分支，支持等于、不等于、小于等比较类型，并包含 12-bit 的有符号位移，允许±4KB 范围内跳转。分支预测默认向后跳跃（如循环）预测发生，向前跳跃（如 if-else 的 else）预测不发生。

JAL 和 JALR 同样用于无条件跳跃，当目标寄存器是零寄存器（x0）时，跳转不保存返回地址，实现纯跳转功能。

2.2.6 算术和逻辑集

RISC-V 的算术和逻辑操作主要集中在一个小的 I 子集中,包括基本的加法、减法、位移、位操作和比较分支。除了原子操作外,这些操作可以通过软件模拟。RISC-V 没有实现某些特殊位操作如计算前导零等。整数乘法和除法操作被归类在 M 子集中。对于浮点运算，F 子集支持单精度操作，需要额外的 32 个浮点寄存器。D 子集提供双精度操作，而 Q 子集支持 128 位的四精度浮点运算。在没有硬件浮点支持的情况下，也可以使用软件库执行浮点运算。运算错误如溢出或除以零不会引发异常，而是生成默认的合理数值，并通过状态位来标示错误。

2.2.7 原子内存操作

RISC-V 支持在多 CPU 和多线程环境中进行存储器共享，采用“释放一致”模型来管理内存操作的顺序。这包括设置读操作为“获取”和写操作为“释放”，以控制操作的执行顺序。FENCE 指令提供基本的存储器访问顺序保证。原子操作子集（A 子集）包括 load-reserved（lr）和 store-conditional（sc）指令，用于处理地址的保留和条件存储。AMO（Atomic Memory Operation）指令组实现读-改-写操作，支持在存储器屏障的帮助下组合多个操作，提高操作效率。

第 3 章 玄铁 C906 处理器详解

3.1 玄铁 C906 处理器结构及子系统

3.1.1 核心结构

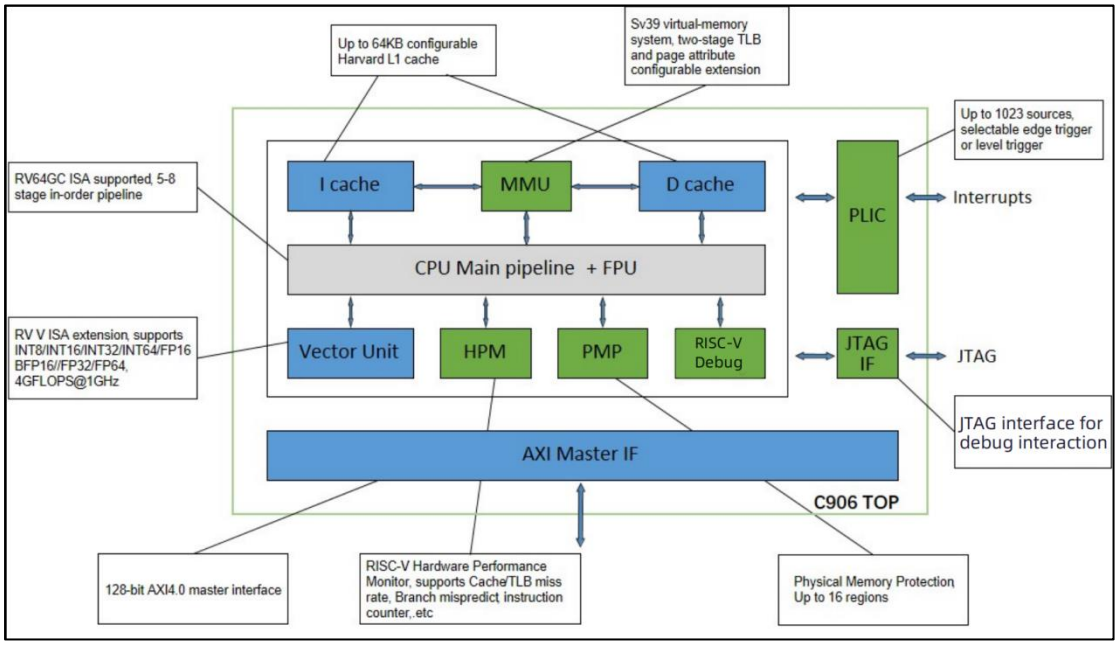


图 3.1 玄铁 C906 处理器核心结构

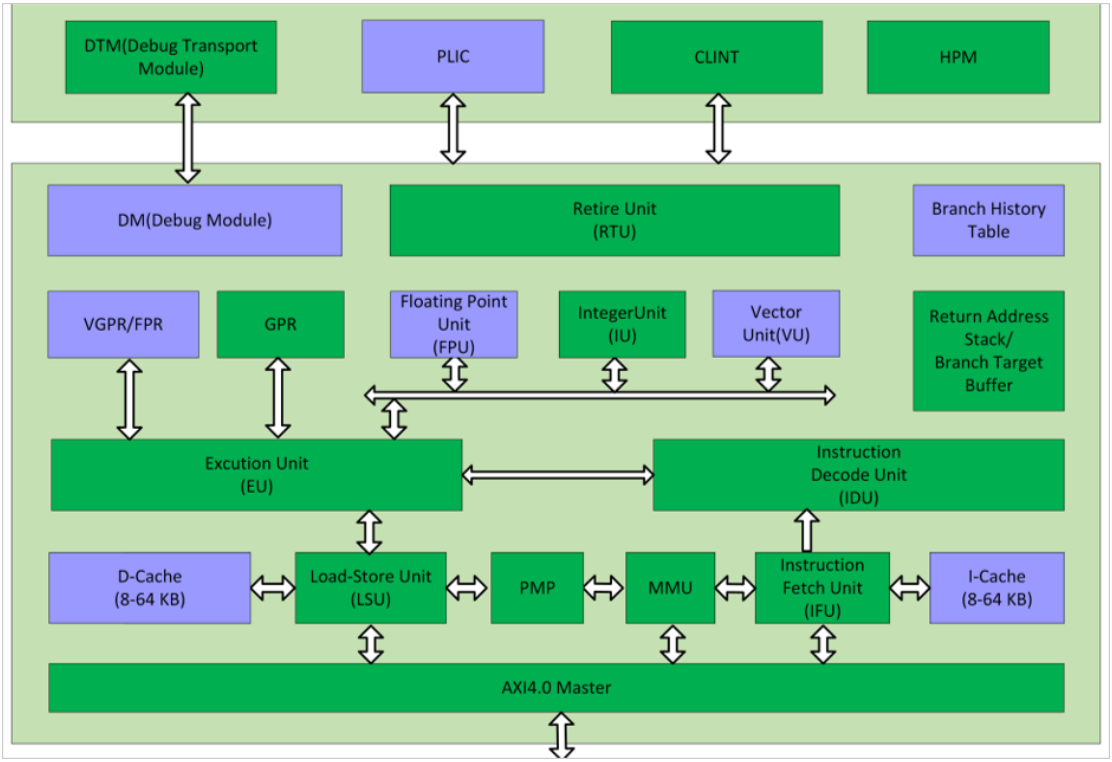


图 3.2 玄铁 C906 处理器核内子系统

玄铁 C906 是一款基于 RISC-V 指令架构的 64 位单核超高能效处理器，同时兼容 RV64IMA[F]C[V]指令架构。其核心结构框图如图 3.1 所示， 其核内子系统结构图如图 3.2 所示。

作为一款超高能效、低成本的单核处理器，其核内子系统主要包含：

- 核心处理器单元：包括指令提取单元（IFU）、指令译码单元（IDU）、整型执行单元（IU）、可配置的浮点执行单元（FPU）、可配的矢量执行单元（VPU），以及 5-8 级单发按序执行流水线。
- 内存子系统：包括可配的 I Cache 和 D Cache、存储载入单元（LSU）以及虚拟内存管理单元（MMU）和物理内存保护单元（PMP）。
- 中断子系统：包括处理器核局部中断控制器 CLINT 和中断控制器 PLIC。
- I/O 子系统：主要为主设备接口单元（AXI Master IF）。

3.1.2 核心处理器单元

核心处理器单元式处理器的核心部分，负责执行指令、进行算术逻辑运算等基本操作。

玄铁 C906 的指令提取单元（IFU）一次最多可以提取两条指令并进行处理。为了提高指令提取效率，可以配备高速缓存，当高速缓存缺失时采用关键指令优先获取技术；此外为了降低功耗，提高分支预测的准确率，配备指令暂存器缓存预取指令，并且采用低成本高准确率的 Gshared 分支预测器。

指令译码单元（IDU）负责单条指令的译码和数据相关性检测。

玄铁 C906 在整型运算时，为了减少指令执行过程中的数据相关，ALU 采用操作数前馈的方法，即在指令执行过程中提前将需要的操作数提供给 ALU；为了加速处理器性能，BJU 在单周期内完成对分支预测错误的处理，减少分支带来的延迟。

玄铁 C906 为了在保证处理器性能的同时，尽可能降低设计的复杂度和功耗，提高处理器的可靠性，采用 5-8 级的单发按序执行流水线技术。一方面采用多级流水结构可以提高处理器的吞吐量和执行效率，另一方面相对于更深的流水线，5-8 级的流水线的设计更为简洁，提高了处理器的可维护性和可靠性，且较短的流水线意味着更少的逻辑层级和时钟周期，从而降低了处理器的功耗。

3.1.3 内存子系统

内存子系统负责存储器的管理以及处理器与存储器之间的数据传输。

玄铁 C906 采用哈佛结构的一级高速缓存，包含独立的指令高速缓存和数据高速缓存。一级高速缓存位于处理器核心内部，与核心之间的数据传输速度非常快，可以显著降低访存延迟，提高指令执行的效率。同时 L1 指令高速缓存和 L1

数据高速缓存分别支持指令预取功能和数据预取功能来提高指令级并行度。L1 指令高速缓存中还使用分支历史表和 GShare 预测器作为预测机制，提高分支预测的准确率，减少控制冲突带来的制约。

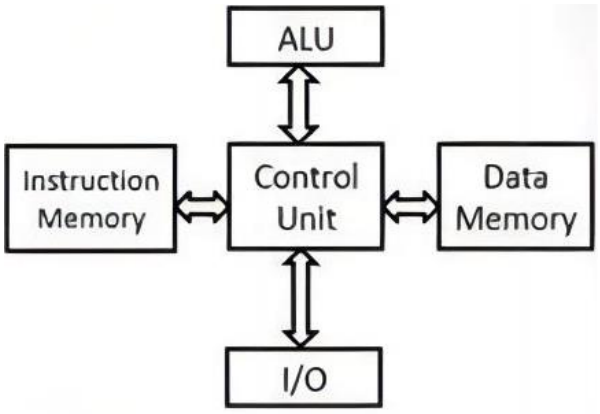


图 3.3 哈佛结构

玄铁 C906 虚拟内存管理 MMU 负责地址转换、页面保护和页面属性管理。MMU 主要利用译址后置缓冲器（TLB）来实现上述功能，TLB 存储量虚拟地址到物理地址的映射关系，可以快速将虚拟地址转换为物理地址，避免了频繁访问页表的开销，从而提高了地址转换的效率，降低了处理器的功耗。

玄铁 C906 的物理内存保护单元 PMP 负责对系统内存（包括存储器系统和外围设备）进行细粒度的访问控制和保护，防止未经授权的访问和恶意攻击，提高了系统的安全性和稳定性。

3.1.4 中断子系统

中断子系统是处理器处理中断请求和异常情况的核心组成部分。玄铁 C906 实现了处理器核局部中断控制器 CLINT，用于处理软件中断和计时器中断，同时使用中断控制器 PLIC 对外部中断源进行采样、优先级仲裁和分发。

与全局中断控制器相比，CLINT 减少了系统中断信号的传递和处理，可以更快地响应处理器核中的中断请求，减少了中断响应的延迟，从而既降低了处理器的开销又提高了系统的响应速度和性能。

PLIC 可以在处理器核和外部中断源之间起到缓冲和调度的作用，避免了处理器核不必要的中断处理负载。这种设计可以提高系统的性能和效率，减少处理器核在处理中断时的竞争和冲突。

3.1.5 I/O 子系统

I/O 子系统负责处理输入输出操作，玄铁 C906 的主设备接口支持 AMBA 4.0 AXI 总线协议，实现处理器与外部设备之间的高速数据传输和通信。

主设备接口支持 128 位总线带宽，可以满足处理器与外部设备之间大量数据

的传输需求。

采用 AXI 协议可以实现低延迟的数据交换，支持不同的数据传输模式和操作类型，并且具有良好的错误检测和纠正机制，能够保证数据传输的可靠性和数据完整性。

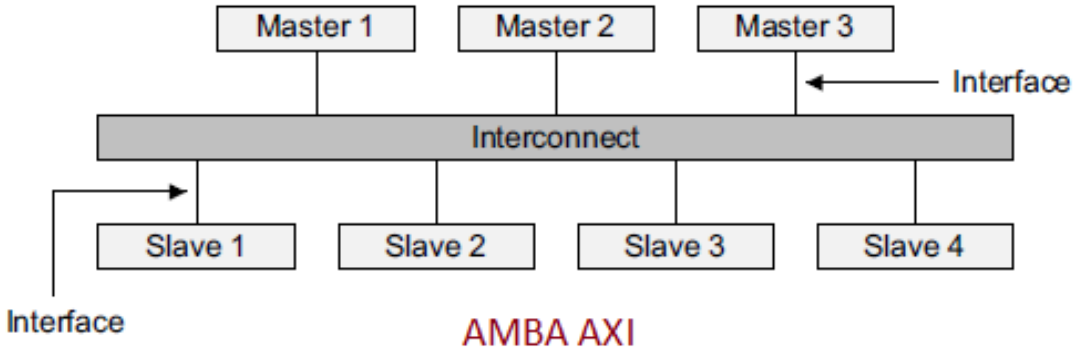


图 3.4 AMBA AXI

3.2 应用领域

玄铁 C906 凭借其高性能、低功耗的优势，可以广泛应用于人工智能和机器学习、消费电子等领域。

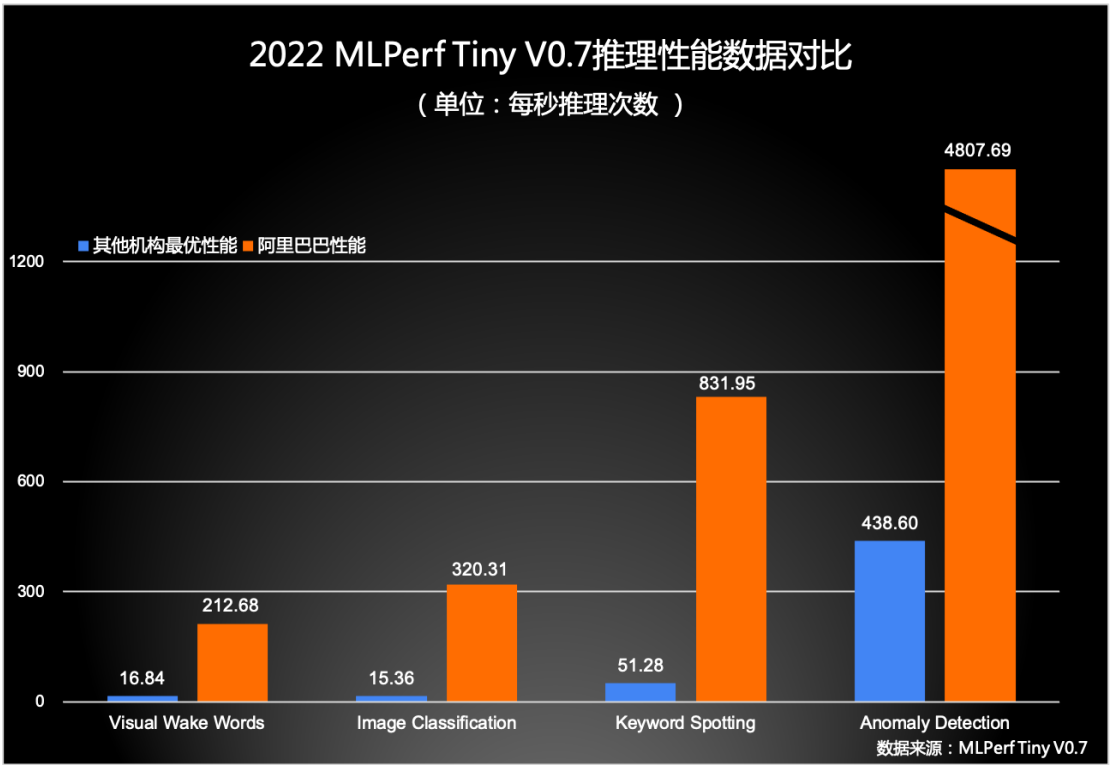
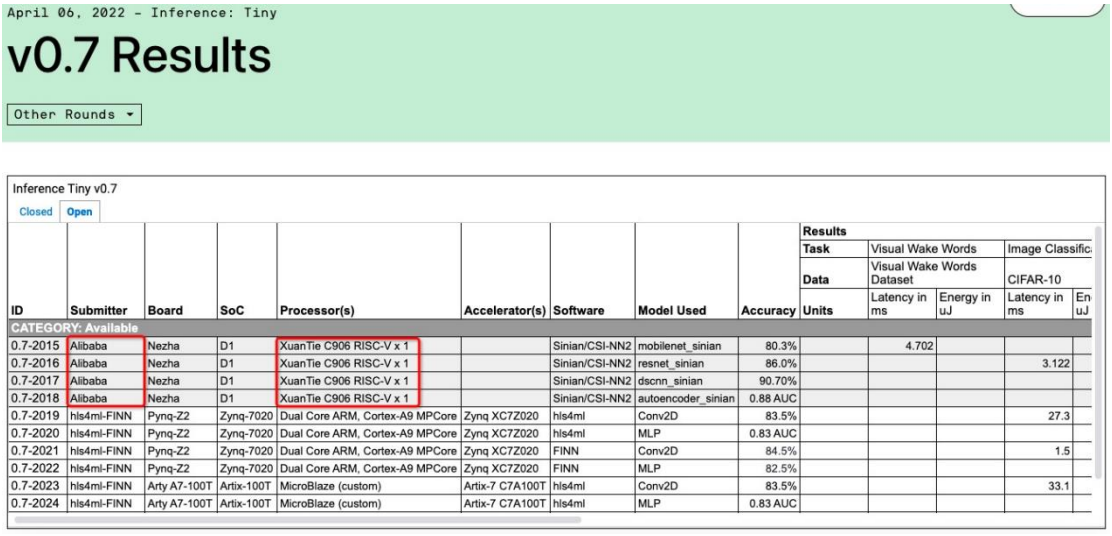


图 3.5 2022 MLPerf Tiny V0.7 推理性能数据对比

- 人工智能和机器学习：2022 年 4 月 7 日，全球权威 AI 基准测试 MLPerf

发布最新榜单，在聚焦低功耗、高能效的 IoT 领域 Tiny v0.7 榜单中，基于平头哥玄铁 RISC-V C906 处理器的软硬件联合优化方案，取得了全部 4 个指标的第一，并且达到了其他竞品同类最优性能的 10 倍以上。

- 玄铁 C906 不仅可以提供高性能的整型和浮点运算，还支持矢量指令集，可以高效地执行计算任务，对于神经网络的训练非常重要。
- 玄铁 C906 高性能的内存子系统和可配备的高速缓存使得数据在处理器和内存之间高效传输，适合需要大量数据处理的 AI 应用。


April 06, 2022 - Inference: Tiny

v0.7 Results

Other Rounds

Inference Tiny v0.7

Closed Open

ID	Submitter	Board	SoC	Processor(s)	Accelerator(s)	Software	Model Used	Accuracy	Results				
									Task	Visual Wake Words		Image Classific	
									Data	Visual Wake Words Dataset		CIFAR-10	
									Units	Latency in ms	Energy in uJ	Latency in ms	En uJ
CATEGORY: Available													
0.7-2015	Alibaba	Nezha	D1	XuanTie C906 RISC-V x 1		Sinian/CSI-NN2	mobilenet_sinian	80.3%		4.702			
0.7-2016	Alibaba	Nezha	D1	XuanTie C906 RISC-V x 1		Sinian/CSI-NN2	resnet_sinian	86.0%				3.122	
0.7-2017	Alibaba	Nezha	D1	XuanTie C906 RISC-V x 1		Sinian/CSI-NN2	dscnn_sinian	90.70%					
0.7-2018	Alibaba	Nezha	D1	XuanTie C906 RISC-V x 1		Sinian/CSI-NN2	autoencoder_sinian	0.88 AUC					
0.7-2019	hls4ml-FINN	Pynq-Z2	Zynq-7020	Dual Core ARM, Cortex-A9 MPCore	Zynq XC7Z020	hls4ml	Conv2D	83.5%				27.3	
0.7-2020	hls4ml-FINN	Pynq-Z2	Zynq-7020	Dual Core ARM, Cortex-A9 MPCore	Zynq XC7Z020	hls4ml	MLP	0.83 AUC					
0.7-2021	hls4ml-FINN	Pynq-Z2	Zynq-7020	Dual Core ARM, Cortex-A9 MPCore	Zynq XC7Z020	FINN	Conv2D	84.5%				1.5	
0.7-2022	hls4ml-FINN	Pynq-Z2	Zynq-7020	Dual Core ARM, Cortex-A9 MPCore	Zynq XC7Z020	FINN	MLP	82.5%					
0.7-2023	hls4ml-FINN	Arty A7-100T	Artix-100T	MicroBlaze (custom)	Artix-7 C7A100T	hls4ml	Conv2D	83.5%				33.1	
0.7-2024	hls4ml-FINN	Arty A7-100T	Artix-100T	MicroBlaze (custom)	Artix-7 C7A100T	hls4ml	MLP	0.83 AUC					

图 3.6 Tiny v0.7 榜单

- 电子消费领域
  - 玄铁 C906 的高性能计算能力适合需要处理大量的多媒体数据的消费电子产品（如智能电视、游戏机）。
  - 玄铁 C906 高能效和低功耗的特性可以使需要在有限的电池容量下长时间运行的消费电子产品有更好的表现。
  - 玄铁 C906 基于 RISC-V 的开放架构使其具有很高的灵活性和可扩展性，支持多种外设和接口，使得消费电子产品的设计更为简洁，有助于缩短产品的开发周期。

第 4 章 RISC-V 架构性能分析

4.1 测试环境与工具

在评估 RISC-V 架构尤其是玄铁 C906 处理器的性能时，选择合适的测试环境和工具至关重要。本文详细描述了测试所使用的硬件和软件环境、测试工具及方法，并展示了测试结果和对比分析。

4.1.1 硬件环境

为了进行全面的性能评估，我们在以下硬件环境中进行了测试：

- RISC-V 64 CPU：玄铁 C906，运行频率 1.0GHz。
- X86_64 CPU：
  - Intel Core i7-4770HQ:基准频率 2.2GHz,全核睿频稳定在 3.2GHz。用。
  - UnixBench 和大规模矩阵计算程序的测试。
  - Intel Core i5-8500：基准频率 3.0GHz，睿频至 4.10GHz。用于 microbench 的测试。
- 内存（RAM）：DDR3 512MB。
- 只读存储器（ROM）：Nor Flash 128MB。
- 网络：100MB。
- 音频：3.5mm CTIA。
- 按键：fel 1 + LRADC OK1。
- 调试接口：UART + ADB USB。
- 电源：USB Type-C 5V-2A。
- PCB 板层：6 层板。

4.1.2 软件环境

- 操作系统：Debian 11，Linux 内核版本 5.4。
- 交叉编译工具链：riscv64-linux-gnu-gcc (GCC) 11.2.0。

4.1.3 测试工具

在测试中，我们使用了以下工具：

- Microbench：由泰晓科技制作，基于指令集的性能评测工具。
- 大规模矩阵计算程序：使用 C/C++编写，基于 Eigen-3.4.0 数学库进行矩阵运算，用于测试单核整数和浮点数性能。

- UnixBench：一个经典的 Unix 系统性能基准测试工具。

4.2 测试方法

4.2.1 Microbench 和 UnixBench

我们首先在 X86_64 平台上对 microbench 和 UnixBench 进行了移植和交叉编译，然后在 RISC-V 64 平台上运行这些测试工具，并记录测试所得的分数。

4.2.2 大规模矩阵计算程序

该程序最初用 MATLAB 编写，后用 C/C++重写以提高性能，并使用 Eigen-3.4.0 库进行矩阵运算。程序在 X86_64 平台上完成开发和初步测试，然后交叉编译移植到 RISC-V 64 平台。测试程序在 RISC-V 平台上运行，记录运行时间，以评估处理器的性能。

4.3 测试结果

4.3.1 Microbench 测试结果

通过 microbench 测试，我们可以评估处理器在执行基本指令集操作时的性能。测试结果显示，与 X86_64 平台相比，RISC-V 64 处理器在基础指令性能上存在显著差距。

BM_nop	3.08 ns	2.99 ns	233889145
BM_ub	3.10 ns	2.99 ns	233893313
BM_bnez	12.4 ns	12.0 ns	58507883
BM_beqz	12.3 ns	12.0 ns	58468358
BM_load_bnez	12.2 ns	12.0 ns	58395094
BM_load_beqz	12.1 ns	12.0 ns	58380949
BM_cache_miss_load_bnez	12.2 ns	5.99 ns	116951654
BM_cache_miss_load_beqz	12.4 ns	5.99 ns	116876961
BM_branch_miss_load_bnez	8.56 ns	4.00 ns	175331742
BM_branch_miss_load_beqz	8.15 ns	3.99 ns	175418777
BM_cache_branch_miss_load_bnez	9.79 ns	4.82 ns	141039686
BM_cache_branch_miss_load_beqz	10.3 ns	4.96 ns	127321246
BM_inc	10.2 ns	9.97 ns	70122305
BM_dec	11.2 ns	11.0 ns	63613111
BM_mul	12.2 ns	12.0 ns	58461644
BM_div	11.1 ns	11.0 ns	63466231
BM_float_inc	18.5 ns	18.0 ns	39003335
BM_float_dec	18.5 ns	18.0 ns	39007247
BM_float_mul	18.5 ns	18.0 ns	38973395
BM_float_div	33.9 ns	32.9 ns	21276946
BM_and	11.2 ns	11.0 ns	63815846
BM_or	11.4 ns	11.0 ns	63767351
BM_not	11.3 ns	11.0 ns	63709582
BM_bits_and	11.3 ns	11.0 ns	63730270
BM_bits_or	12.2 ns	12.0 ns	58254510
BM_bits_nor	12.2 ns	12.0 ns	58446391
BM_bits_not	12.7 ns	12.0 ns	58380200
BM_bits_rshift	11.3 ns	11.0 ns	63750367
BM_bits_lshift	11.4 ns	11.0 ns	63743425
BM_for_loop	24.7 ns	24.0 ns	29241522
BM_while_loop	24.6 ns	23.9 ns	29267222
BM_do_while_loop	24.8 ns	23.9 ns	29262257
BM_bubble_sort	339 ns	328 ns	2160002
BM_std_sort	197 ns	192 ns	3683400
BM_calculate_pi	5182 ns	5073 ns	137929
BM_factorial	108 ns	106 ns	6612491

图 4.1 Microbench 测试结果

4.3.2 UnixBench 测试结果

UnixBench 是一个综合性基准测试工具，通过多项测试评估系统的整体性能。测试结果表明，玄铁 C906 在多个项目中均表现不佳，特别是在处理多任务和复杂计算时，其性能明显落后于 X86_64 处理器。

```
------unixbench----
------------------------------------------------------------------------
Benchmark Run: Wed Sep 07 2022 14:56:22 - 15:24:31
1 CPU in system; running 1 parallel copy of tests

Dhrystone 2 using register variables        3001049.4 lps   (10.0 s, 7 samples)
Double-Precision Whetstone                     1047.8 MWIPS (10.0 s, 7 samples)
Execl Throughput                                334.6 lps   (29.9 s, 2 samples)
File Copy 1024 bufsize 2000 maxblocks         42369.6 KBps  (30.0 s, 2 samples)
File Copy 256 bufsize 500 maxblocks           11763.5 KBps  (30.0 s, 2 samples)
File Copy 4096 bufsize 8000 maxblocks        120604.7 KBps  (30.0 s, 2 samples)
Pipe Throughput                              208931.1 lps   (10.0 s, 7 samples)
Pipe-based Context Switching                  30214.6 lps   (10.0 s, 7 samples)
Process Creation                                790.2 lps   (30.0 s, 2 samples)
Shell Scripts (1 concurrent)                    719.0 lpm   (60.0 s, 2 samples)
Shell Scripts (8 concurrent)                     92.6 lpm   (60.2 s, 2 samples)
System Call Overhead                         380766.8 lps   (10.0 s, 7 samples)

System Benchmarks Index Values               BASELINE       RESULT    INDEX
Dhrystone 2 using register variables         116700.0    3001049.4    257.2
Double-Precision Whetstone                       55.0       1047.8    190.5
Execl Throughput                                 43.0        334.6     77.8
File Copy 1024 bufsize 2000 maxblocks          3960.0      42369.6    107.0
File Copy 256 bufsize 500 maxblocks            1655.0      11763.5     71.1
File Copy 4096 bufsize 8000 maxblocks          5800.0     120604.7    207.9
Pipe Throughput                               12440.0     208931.1    168.0
Pipe-based Context Switching                   4000.0      30214.6     75.5
Process Creation                                126.0        790.2     62.7
Shell Scripts (1 concurrent)                     42.4        719.0    169.6
Shell Scripts (8 concurrent)                      6.0         92.6    154.4
System Call Overhead                          15000.0     380766.8    253.8
                                                                   ========
System Benchmarks Index Score                                       133.4
```

图 4.2 UnixBench 测试结果

4.3.3 大规模矩阵计算程序测试结果

在大规模矩阵计算测试中，玄铁 C906 处理器的运行时间显著长于 Intel Core i7-4770HQ 处理器。测试结果显示，与 X86_64 处理器相比，RISC-V 处理器在处理复杂计算任务时效率较低，性能差距大约为 39.25 倍。

```
Start time is: 1097
开始生成随机数矩阵
随机数矩阵生成完毕
开始生成稀疏矩阵
稀疏矩阵生成完毕
求得最小的函数值为: 0.151914
迭代次数为: 170
End time is: 20455667
Time consumption is: 20
```

图 4.3 大规模矩阵计算程序测试结果

4.4 性能对比分析

4.4.1 与 Intel Core i5-8500 的对比

在 microbench 测试中，玄铁 C906 与 Intel Core i5-8500 进行了对比。测

试结果显示，即使在单核性能上，RISC-V 平台处理器与 X86_64 平台处理器也存在显著差距。这种差距不仅仅是由于频率差异，更主要的是由于两种架构设计和实现工艺的不同。RISC-V 处理器的基础指令性能与 X86_64 处理器相比，差异在 4 到 10 倍之间，即使在同等主频下，性能差距仍然明显（约 2 到 5 倍）。

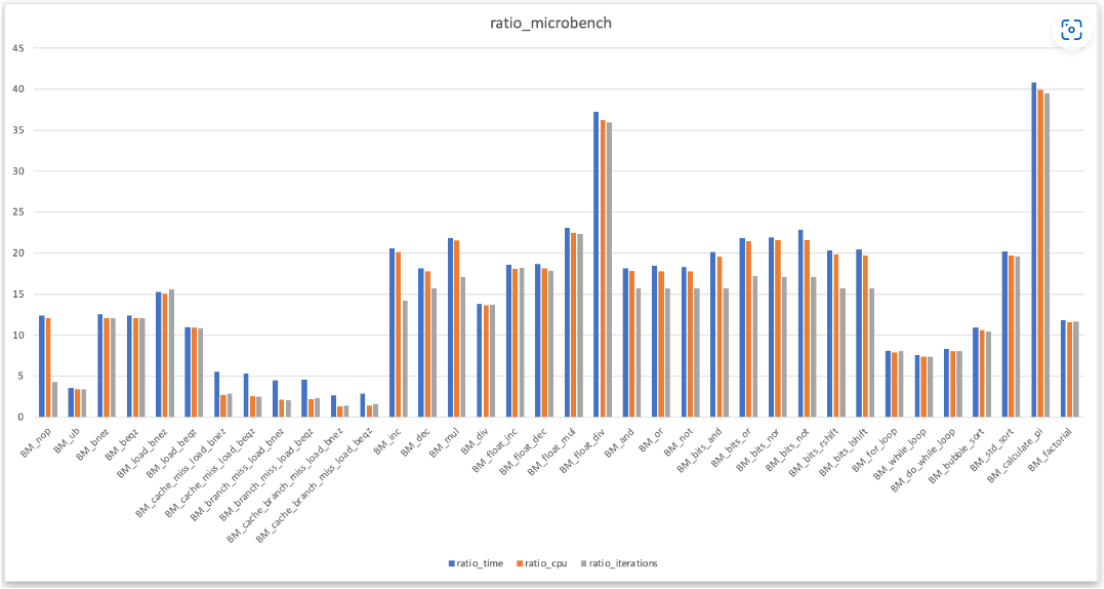


图 4.4 与 Intel Core i5-8500 的对比

4.4.2 与 Intel Core i7-4770HQ 的对比

UnixBench 和大规模矩阵计算程序的测试结果进一步证明了这一点。即使在多核和复杂任务的处理上，玄铁 C906 也无法与 Intel Core i7-4770HQ 媲美。性能差距主要体现在多任务处理、内存带宽和浮点计算能力上。

```
Benchmark Run: Wed Sep 07 2022 19:59:17 - 20:23:52
8 CPUs in system; running 1 parallel copy of tests

Dhrystone 2 using register variables       32343533.0 lps   (10.0 s, 7 samples)
Double-Precision Whetstone                     4533.0 MWIPS (10.5 s, 7 samples)
Execl Throughput                               3263.5 lps   (29.9 s, 2 samples)
File Copy 1024 bufsize 2000 maxblocks        635715.1 KBps  (30.0 s, 2 samples)
File Copy 256 bufsize 500 maxblocks          168269.6 KBps  (30.0 s, 2 samples)
File Copy 4096 bufsize 8000 maxblocks       2003899.3 KBps  (30.0 s, 2 samples)
Pipe Throughput                              677305.6 lps   (10.0 s, 7 samples)
Pipe-based Context Switching                 110291.0 lps   (10.0 s, 7 samples)
Process Creation                               8915.2 lps   (30.0 s, 2 samples)
Shell Scripts (1 concurrent)                   9925.1 lpm   (60.0 s, 2 samples)
Shell Scripts (8 concurrent)                   2371.0 lpm   (60.0 s, 2 samples)
System Call Overhead                         554765.5 lps   (10.0 s, 7 samples)

System Benchmarks Index Values               BASELINE       RESULT    INDEX
Dhrystone 2 using register variables         116700.0   32343533.0   2771.5
Double-Precision Whetstone                       55.0       4533.0    824.2
Execl Throughput                                 43.0       3263.5    759.0
File Copy 1024 bufsize 2000 maxblocks          3960.0     635715.1   1605.3
File Copy 256 bufsize 500 maxblocks            1655.0     168269.6   1016.7
File Copy 4096 bufsize 8000 maxblocks          5800.0    2003899.3   3455.0
Pipe Throughput                               12440.0     677305.6    544.5
Pipe-based Context Switching                   4000.0     110291.0    275.7
Process Creation                                126.0       8915.2    707.6
Shell Scripts (1 concurrent)                     42.4       9925.1   2340.8
Shell Scripts (8 concurrent)                      6.0       2371.0   3951.7
System Call Overhead                          15000.0     554765.5    369.8
                                                                     ========
System Benchmarks Index Score                                         1111.4

------------------------------------------------------------------------
```

图 4.5 和 i7-4770HQ 对比大规模矩阵运算程序(1)

```
Start time is: 1097
开始生成随机数矩阵
随机数矩阵生成完毕
开始生成稀疏矩阵
稀疏矩阵生成完毕
求得最小的函数值为：0.151914
迭代次数为：170
End time is: 20455667
Time consumption is: 20
```

图 4.6 和 i7-4770HQ 对比大规模矩阵运算程序(2)

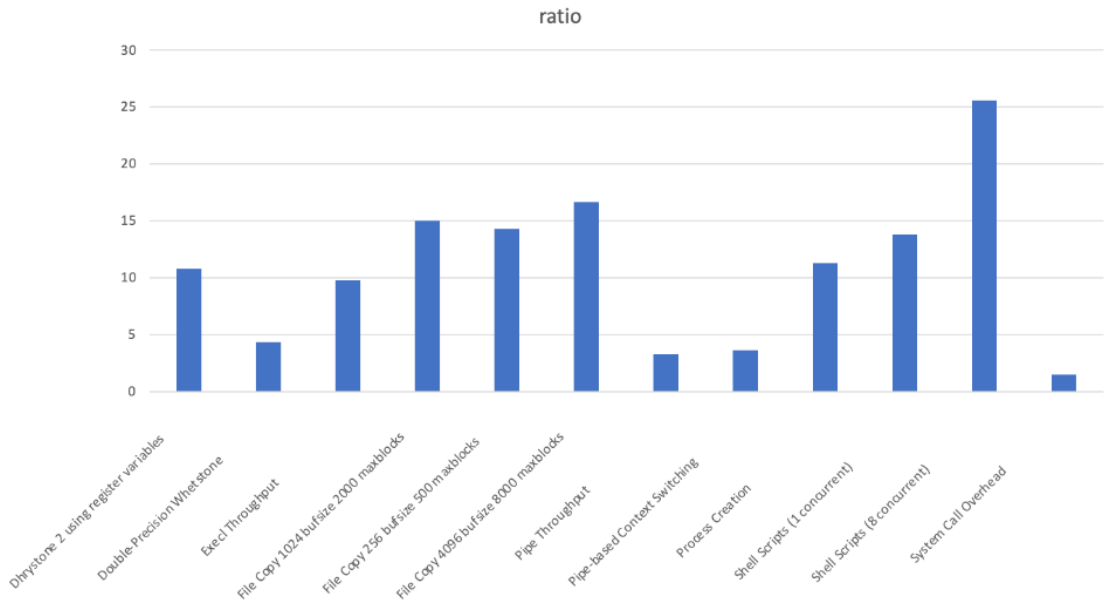


图 4.7 与 Intel Core i7-4770HQ 的对比

与 rv64 处理器相比，耗时差距大概为 39.25 倍

4.5 玄铁 C906 和 RISC-V 架构性能分析

4.5.1 玄铁 C906 处理器的优势和劣势

通过上述测试和分析，我们可以总结出玄铁 C906 处理器的优势和劣势。

玄铁 C906 处理器的优势如下：

1. 开源架构：玄铁 C906 基于 RISC-V 开源指令集，这为处理器的定制化和适应性提供了极大的灵活性。企业可以根据自身需求定制专有的指令集，优化特定应用的性能。
2. 成本效益：由于不需支付高昂的授权费用，以及采用开放的设计和生产生态系统，玄铁 C906 的生产成本相对较低。
3. 低功耗：RISC-V 处理器如玄铁 C906 通常设计为低功耗设备，适合在电源受限的环境中使用，如物联网设备。

玄铁 C906 处理器的劣势如下：

1. 性能差距：如测试结果所示，玄铁 C906 在处理高复杂度任务时的性能

与主流 x86 和 ARM 架构有明显差距，特别是在多任务处理和高性能计算方面。

2. 生态系统支持：尽管 RISC-V 社区正在快速发展，但与成熟的 x86 和 ARM 生态相比，其支持的软件和工具仍相对较少，这可能限制了其在某些高端应用的广泛采用。

4.5.2 玄铁 C906 和 RISC-V 架构性能分析

RISC-V 架构处理器在多个领域显示出强大的潜力，特别是在低功耗、高定制性的应用场景中表现突出。然而，对于需要高度计算性能的应用，如服务器和高性能计算平台，RISC-V 处理器当前的技术实现可能不足以满足这些需求。性能上的主要局限在于处理速度、多任务能力和复杂计算处理能力。

实际应用需求满足情况如下：

1. 物联网设备：RISC-V 处理器因其低功耗和高灵活性特点，非常适合用于物联网设备。在这些场景中，设备通常需要运行简单的应用，并且更注重能效比和成本效益。
2. 嵌入式系统：RISC-V 也广泛应用于嵌入式系统，如工业控制和汽车电子。其开源特性允许制造商根据具体需求定制芯片设计，优化性能和资源使用。
3. 教育和研究：在教育和学术研究领域，RISC-V 由于其开放性和易于理解的架构，成为研究处理器设计和计算机架构的理想工具。

总的来说，尽管 RISC-V 处理器在某些高性能计算任务中可能不足以与主流架构竞争，但其在成本效益、定制性和低功耗方面的优势使其在许多特定领域具有显著的应用潜力。随着技术的进步和生态系统的成熟，预期这些处理器将在更广泛的应用场景中发挥重要作用。

4.6 总结与优化建议

4.6.1 性能总结

通过上述测试和分析，我们可以得出以下结论：

1. 单核性能不足：RISC-V 处理器在单核性能上明显落后于 X86_64 处理器，这不仅是由于频率差异，更是因为指令集设计和实现工艺的差距。
2. 多任务处理能力有限：在处理多任务和复杂计算任务时，RISC-V 处理器的性能劣势更加明显。
3. 内存和缓存性能差距：RISC-V 处理器在内存带宽和缓存性能上也不如

X86_64 处理器，这对整体性能有较大影响。

4.6.2 优化建议

为了提高 RISC-V 处理器的性能，建议进行以下优化：

1. 编译器优化：通过优化编译器选项，如使用高优化级别（-O3）和特定的指令集扩展，可以提高处理器的执行效率。
2. 缓存优化：增加缓存容量或引入多级缓存，优化缓存配置，减少缓存未命中率。
3. 硬件加速：引入专用硬件加速器，如 AI 加速器，可以显著提升特定任务的处理性能。
4. 软件优化：针对具体应用程序进行优化，如多线程编程和算法优化，充分利用多核架构和指令集优势。
5. 功耗管理：通过动态电压和频率调节（DVFS）技术，在不同工作负载下调整处理器功耗，实现性能与能效的平衡。

第 5 章 总结

本文详细探讨了 RISC-V 处理器架构的发展历史、技术特点及其在具体实例——玄铁 C906 处理器中的应用和性能表现。

RISC-V 是一种开源指令集架构（ISA），其设计初衷是为了提供一种灵活、高效且易于扩展的指令集架构，以应对现代计算需求的多样化和复杂性。随着物联网、人工智能和高性能计算等领域的快速发展，对处理器的性能和灵活性提出了更高的要求。RISC-V 的开放性使其能够自由地进行定制和优化，促进了全球范围内的技术创新和知识共享。

RISC-V 架构具有简洁高效的设计，支持 32 位、64 位和 128 位的处理器设计，并提供了灵活的子集扩展方式。这些设计特性使其在嵌入式系统、个人电脑以及超级计算机等多种应用场景下都表现出色。其模块化的指令集设计允许开发者根据具体需求进行选择和定制，极大地增强了处理器的适应能力和优化空间。

本文通过分析玄铁 C906 处理器，展示了 RISC-V 在实际应用中的技术优势和性能表现。玄铁 C906 是由阿里巴巴旗下的平头哥半导体公司设计的高性能处理器，其采用 RISC-V 架构，具有出色的性能和低功耗特性。通过详细的技术规格分析，本文揭示了玄铁 C906 在计算效率、功耗管理和扩展能力等方面的优势。

本文还对玄铁 C906 与其他 RISC-V 处理器进行了比较分析，探讨了其在技术规格、性能和应用场景方面的优势和劣势。通过这种比较，进一步突显了玄铁 C906 在 RISC-V 生态系统中的独特地位和竞争力。

RISC-V 架构的开放性和可定制性为未来处理器设计和应用提供了广阔的空间。随着越来越多的公司和研究机构加入 RISC-V 生态系统，这一架构有望在更广泛的应用领域中得到推广和应用。特别是在物联网和人工智能领域，RISC-V 处理器凭借其高效、低功耗和灵活的特点，具有巨大的市场潜力。

总的来说，RISC-V 作为一种新兴的开源指令集架构，以其创新的设计理念和强大的技术优势，在全球范围内迅速崛起。玄铁 C906 处理器作为 RISC-V 架构的一个成功应用实例，充分展示了这一架构的潜力和未来发展方向。通过对 RISC-V 及其具体处理器实例的深入分析，本文为读者提供了对这一前沿技术的全面理解，并为未来处理器的设计和应用提供了有价值的参考。

RISC-V 的开放性和可定制性不仅降低了硬件和软件开发成本，也加快了产品的上市速度，推动了硬件生态系统的多样化发展。未来，随着更多创新和改进的出现，RISC-V 有望在更广泛的应用领域中发挥重要作用，成为推动科技进步的重要力量。

参考资料

1. Waterman A, Lee Y, Patterson D, et al. The RISC-V instruction set manual[J]. Volume I: User-Level ISA', version, 2014, 2: 1-79.
2. 刘畅, 武延军, 吴敬征, 等. Survey on RISC-V system architecture research[J]. Journal of Software, 2021, 32(12): 3992-4024.
3. 钱玉娟. 玄铁探路 RISC-V[N]. 经济观察报,2024-03-25(017).DOI:10.28421/n.cnki.njjgc.2024.000330.
4. 倪光南.以全球视野谋划和推动开源 RISC-V 生态发展[J].科技导报,2024,42(02):1-2.
5. 张学镇,汪西虎,董嗣万,等.五级流水线 RISC-V 微处理器的研究与设计[J/OL].计算机工程:1-8[2024-06-07].https://doi.org/10.19678/j.issn.1000-3428.0068146.
6. 邢明杰,宋威,张科,等.RISC-V 技术及生态专题前言[J].计算机系统应用,2023,32(11):1-2.DOI:10.15888/j.cnki.csa.009334.
7. 李金凤,于德明,郭瑞华.基于 RISC-V 指令集的处理器及其运行环境设计[J].南方农机,2023,54(15):34-39.
8. 高洁. 嵌入式 RISC-V 微处理器体系架构的研究[D].中南大学,2023.DOI:10.27661/d.cnki.gzhnu.2023.002830.
9. 邹和仕.RISC-V 多核处理器存储系统架构分析[J].自动化应用,2022(12):104-106+111.DOI:10.19769/j.zdhy.2022.12.027.
10. RISC-V 介绍（https://riscv.org/about）
11. RISC-V 维基百科（https://zh.wikipedia.org/wiki/RISC-V）
12. 玄铁处理器及芯片设计平台介绍（https://www.xrvm.cn/product/list/xuantie）
13. 玄铁 C906 处理器介绍（https://www.xrvm.cn/product/xuantie/C906）
14. 玄铁 C906 处理器云评估(https://www.xrvm.cn/community/laboratory/cloud-service?cpu=6）
15. 玄铁 C906 处理器 GitHub 仓库(https://github.com/XUANTIE-RV/openc906）
16. 玄铁 C906 集成手册(openc906)(https://github.com/XUANTIE-RV/openc906/blob/main/doc/%E7%8E%84%E9%93%81C906%E9%9B%86%E6%88%90%E6%89%8B%E5%86%8C.pdf）
17. 玄铁 C906R2S1 用户手册(xrvm)(https://occ-oss-prod.oss-cn-hangzhou.aliyuncs.com/resource//1713757789320/%E7%8E%84%E9%93%81C906R2S1%E7%

94%A8%E6%88%B7%E6%89%8B%E5%86%8C%EF%BC%88xrvm%EF%BC%89_20240422.pdf）

18. T-Head XuanTie C906 Processor Datasheet（https://occ-oss-prod.oss-cn-hangzhou.aliyuncs.com/resource//1686279877608/T-Head%20XuanTie%20C906%20Processor%20Datasheet.pdf）
19. Microbench 介绍（https://pypi.org/project/microbench）
20. UnixBench GitHub 仓库（https://github.com/kdlucas/byte-unixbench）