

面向多数据流处理的时空并行可重构计算阵列结构

李萌¹, 解超¹, 牛兵²

(1. 中电长城科技有限公司, 北京 100190; 2. 山东融汇物产集团有限公司, 山东 济宁 272000)

摘要: 面向高性能智算系统中网络流式数据的高效处理需求, 提出了一种时空并行可重构计算阵列结构, 并将其应用于智能芯片设计。该架构融合多上下文处理单元与存储增强型微处理单元, 支持动态上下文切换、近存计算与流水线并行。通过计算与存储的深度耦合突破存储墙, 以时分复用方式提升资源利用率, 相较于传统方案, 其处理单元利用率提升30%以上, 时延降低25%以上, 能效得到显著提升。文中详细阐述该架构的并行机制, 分析传统架构存在的问题、主流技术路径及所提方案的创新点, 并将该关键技术应用于可重构智能网卡芯片设计, 为智算中心、AI训练与推理等场景的网络传输提供新的国产化方案。

关键词: 高性能智算系统; 流式数据; 可重构计算阵列; 可重构智能网卡

中图分类号: TP393

文献标志码: A

DOI: 10.11959/j.issn.2096-109x.AQ26028

Spatiotemporally parallel reconfigurable computing array architecture for multi-data stream processing

Li Meng¹, Xie Chao¹, Niu Bing²

1. China Electronics Great Wall Technology Co., Ltd., Beijing 100190, China

2. Shandong Ronghui Products Group Co., Ltd., Jining 272000, China

Abstract: To address the efficient processing requirements of network streaming data in high-performance intelligent computing systems, a spatiotemporally parallel reconfigurable computing array architecture was proposed and applied to the design of intelligent chips. In this architecture, multi-context processing units and storage-enhanced micro-processing units were integrated, and dynamic context switching, near-memory computing, and pipeline parallelism were supported. The memory wall was broken through the deep coupling of computation and storage, and the resource utilization was improved by means of time-division multiplexing. Compared with traditional solutions, the utilization rate of processing units was increased by more than 30%, the latency was reduced by more than 25%, and the energy efficiency was significantly enhanced. In this paper, the parallel mechanism of the proposed architecture was elaborated in detail, the problems existing in traditional architectures, the mainstream technical approaches, and the innovations of this scheme were analyzed, and this key technology was further applied to the design of reconfigurable smart NIC chips. A new domestic solution for network transmission in scenarios such as intelligent computing centers and AI training and inference was thereby provided.

Key words: high-performance intelligent computing system, streaming data, reconfigurable computing array, reconfigurable intelligent network interface card

0 引言

随着人工智能、高速网络等领域技术的快速

发展, 高性能智算系统对算力的需求急剧增长, 算力、能效、灵活性三者相互制约的权衡矛盾日

益凸显。通用图形处理器 (graphics processing unit, GPU) 存在数据搬运能耗高、专用场景效率不足的问题; 专用集成电路 (application-specific integrated circuit, ASIC) 虽然能效表现优异, 但是算法适配性差, 很难适配快速迭代的各类异构计算任务。基于软件定义硬件特性的可重构计算架构, 逐渐成为兼顾计算灵活性与高能效的主要研究方向。

当前国内外可重构阵列相关技术方案性能差异显著, 普遍存在多项缺陷。传统架构过多关注单一数据流处理, 忽视了多数据流场景下互联开销大、配置时延高等问题; 采用粗粒度可重构架构 (coarse-grained reconfigurable architecture, CGRA) 虽然能够提升芯片能效, 但在面向多数据流并行计算场景时, 仍存在处理单元利用效率低、片上存储带宽不足的问题^[1], 不能满足高性能智算系统这类多模态数据密集型业务的落地需求。

行业主流方案多基于中央处理器 (central processing unit, CPU)、现场可编程门阵列 (field programmable gate array, FPGA) 承载网络处理负载。面对网络处理的复杂数据和控制依赖、低资源利用率问题, CPU 与 FPGA 两种主流执行方案都存在明显短板, 核心矛盾在于吞吐需求与芯片能效、硬件并行能力的失衡^[2-3]。

CPU 基于指令流串行执行报文处理流程, 需要逐个消除数据依赖和控制依赖约束。在 400 Gbit/s 带宽、6 000 万 PPS 小包转发场景下, 串行执行很容易造成处理队列阻塞^[4]。即使开启多核调度, 也会因为上下文切换产生巨大开销, 无法达到线速吞吐指标。

FPGA 可以通过硬件流水线并行处理多数据流报文, 将协议解析、报文加密、数据转发等任务拆分至独立逻辑单元执行, 从而有效解决串行执行带来的数据与控制依赖阻塞问题。不过, FPGA 的核心架构由可编程逻辑块与可编程互联网络构成, 依靠硬件配置实现差异化数字电路功能, 因此其内部可编程逻辑单元、触发器等硬件资源的利用率会受业务数据复杂度的限制。

CPU 串行执行方式很难解决高吞吐场景下的依赖阻塞问题, FPGA 的并行架构虽然可以突破

CPU 的串行吞吐瓶颈, 但是存在重构配置时延高、逻辑资源规模受限的短板。两种方式都不能同时满足复杂依赖化解与资源高效利用的双重需求, 其根本原因在于硬件架构的刚性与网络处理的动态性、依赖性不匹配。

针对算力、能效、灵活性三者相互制约的核心矛盾, 以及现有传统方案存在的各类缺陷, 本文提出一种面向多数据流处理的时空并行可重构阵列架构智能网卡芯片设计思路, 主要解决如下三类关键问题。

1) 突破现有架构的性能瓶颈。本文所提时空并行设计, 通过时域调制与空域分区协同工作, 化解传统架构高通量处理与可重构能力之间的固有矛盾, 提升处理单元利用率, 同时突破环节互联开销过大、配置时延过高的桎梏。

2) 多异构场景下的适配需求。通过动态重构计算通路、灵活切换并行处理策略, 适配各类不同异构任务, 兼顾低功耗与高并行数据处理需求。

3) 保障技术自主能力。本文所提架构具备计算阵列规模化扩展能力, 可为国产智能网卡芯片的研发、实现进口替代提供核心硬件架构与技术支撑。

通过对比域内主流先进方案, 本文的主要研究内容与创新点归纳如下。

1) 在智能网卡芯片架构层面, 开展多数据流时空并行可重构计算阵列研究。空域上, 设计异构可重构处理单元阵列, 支持多流任务细粒度并行映射; 时域上, 采用流水线式动态重构调度机制, 实现任务级与指令级双层时空协同调度体系。现有基于传统技术架构设计的芯片架构, 并行处理模式固化, 仅针对单一数据流进行优化, 在多数据流并发场景下很容易出现硬件资源竞争、调度失衡等问题, 很难同时兼顾高吞吐率与重构带来的额外开销。

2) 在多数据流处理逻辑上, 开展可动态适配多流业务特征与差异化计算粒度的研究。通过构建分层柔性互联网络与分布式局部缓存, 减少多流数据冲突、降低全局存储访问时延, 并建立硬件资源分配理论模型, 实现多流负载均衡调度与硬件利用率最大化。与领域内现有先进方案相

比, 本设计突破单一时域/空域并行局限, 在保证低重构开销的同时提升多数据流并发处理能力, 在多数据流场景下, 芯片的吞吐率、资源利用率与数据实时性具有显著优势。

1 相关工作

时空并行是可重构计算阵列突破芯片算力瓶颈的重要创新思路。空间并行主要依托多处理单元的同步运算提升数据处理的并行度; 时间并行主要借助流水线调度、指令复用等方式来优化时序运行效率。目前, 基于时空并行可重构计算阵列芯片的相关研究, 大多围绕传统架构固有缺陷设计新型时空并行机制以实现性能跃升, 相关方案也逐步从理论研究走向工程化落地, 适配边缘人工智能 (artificial intelligence, AI)、数据中心、光通信等多元化业务场景。

德国柏林工业大学研究团队提出了Pasithea-1顺序可重构阵列架构。针对传统粗粒度可重构阵列存在的可编程性和能效失衡、微架构与指令集耦合过紧问题^[5], 该研究团队设计了融合顺序控制流与空间数据流的架构形式, 以实现时空并行的协同优化。其大致设计思路是, 采用 2×4 瓦片 (tile) 阵列, 每瓦片集成16个处理单元 (processing element, PE), 通过重构S总线与M总线构建互连网络, 从而实现7种不同拓扑结构的灵活配置, 以适配不同的并行场景。在时间维度上, 通过片段实例管理与透明换入/换出机制, 实现类似于精简指令集计算机 (reduced instruction set computer, RISC) 指令的跨子程序复用, 减少指令取指过程的开销; 在空间维度上, 通过PE阵列同步运算, 结合指令集与微架构的解耦设计, 在硬件可配置的同时仍然保持二进制兼容性。Pasithea-1顺序可重构阵列架构虽然能有效控制能效, 但是在大规模扩展时, 互连拓扑切换的开销呈指数级增长, 且固定PE密度难以适配稀疏/稠密混合的算力需求。

珠海芯动力团队提出了可重构并行处理器 (reconfigurable parallel processor, RPP) 架构并在此基础上设计开发了R8芯片。该方案旨在改进传统CGRA架构, 提出准静态可重构阵列与单指令多线程 (single instruction multiple threads, SIMT) 编程模型, 用来解决动态可重构阵列调度复杂、难

以大规模扩展的问题^[6], 重点加强边缘AI场景的时空并行能力。同时, 引入多线程SIMT编程模型并兼容统一计算设备架构 (compute unified device architecture, CUDA) 语言, 支持数据并行算法的高效映射, 通过空间阵列的大规模同步运算 (空间并行) 与线程流水线调度 (时间并行), 提升了边缘AI任务的吞吐量。RPP架构的准静态重构设计虽然简化了调度过程, 但动态自适应能力不足, 在多任务交错场景下资源利用率差, 关键路径时延过高, 难以匹配不规则数据驱动的隐式相位变化。同时, 片上计算吞吐量提升后, 片外内存带宽不足的问题会凸显出来, 造成计算与存储性能脱节。

国内高校和研究团队围绕可重构架构开展了多项探索: 清华大学提出可重构数据自适应脉动阵列 (reconfigurable data-adaptable systolic array, Re-Das) 架构, 设计了轻量级脉动阵列和环形数据通路, 通过相邻PE短连接优化数据局部性问题, 但是多数据流协同处理能力有限^[7]; 寒武纪系列芯片采用异构计算架构, 偏重人工智能推理与训练任务加速, 但是通用计算的适应性较弱^[8-9]; 某研究所研发的“魂芯”系列可重构处理器通过粗粒度处理单元提升空间并行效率, 但时间并行调度机制不够灵活^[10]。

综合分析现有国内外研究成果, 当前各类方案普遍存在时空并行协同调度能力弱、多数据流场景适配性差、重构硬件开销与计算性能难以均衡等问题, 亟须构建一种能够实现时空资源动态优化、高效适配多类型数据流的可重构计算阵列结构。

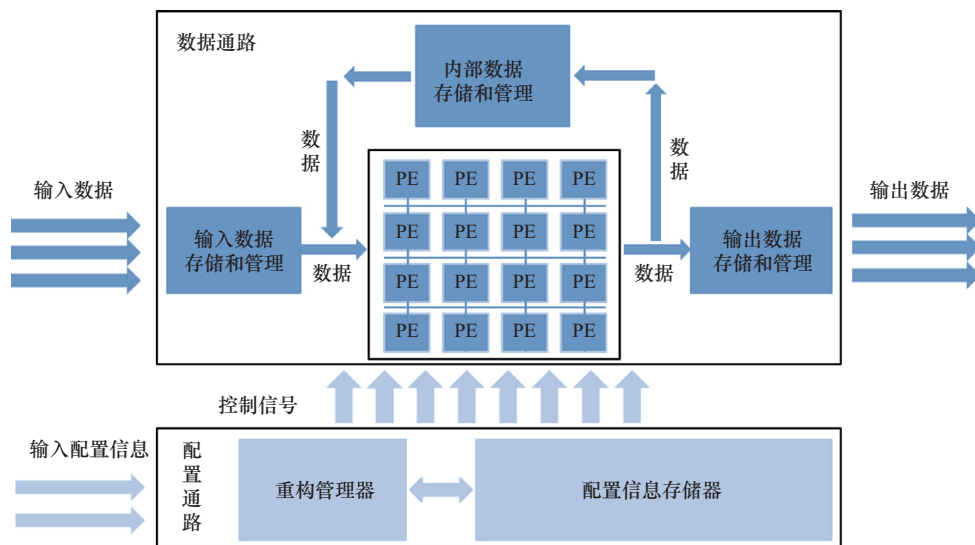
2 技术分析与理论基础

2.1 软件定义芯片技术

软件定义芯片是采用时空联合映射技术、配置流与数据流双驱动架构实现的新型芯片, 可完成计算任务空间并行执行与硬件资源时分复用。其核心思想为: 在空间维度将计算任务拆分至多组PE并行运算, 在时间维度通过动态配置切换让同一个硬件资源用于多个任务^[11]。软件定义芯片技术原理框图如图1所示。

(1) 核心架构基础

软件定义芯片的本质特征是软硬件的深度解耦, 硬件功能由软件配置指令定义^[12], 而非固



化硬件电路约束，芯片功能可在全芯片生命周期内完成动态重配置。该架构以软件配置替代指令驱动形式：配置流负责定义 PE 单元运算功能与互联拓扑，数据流则触发实际计算流程。芯片主体由大量可配置 PE 和可编程互连网络组成，支持从细粒度到粗粒度的资源重组。

(2) 空间并行的实现机理

任务分解与空间映射流程由 4 部分组成，分别是任务划分、资源分配、互联配置、并行执行。首先，将复杂的计算任务分解成多个独立的或者弱相关的子任务，挖掘数据的并行性和任务的并行性；其次，根据各子任务计算特性，将其映射到 PE 阵列的不同区域，构建空间隔离的并行执行区域；随后，通过配置流设定 PE 之间的互联关系，构建专属数据传输通路，支持并行数据的传输；最后，数据流驱动全部 PE 同步运算，省去通用架构的指令取指或译码阶段开销，芯片整体能效可大幅度提升。

支撑空间并行机制的关键技术主要包括脉动阵列映射、分布式存储架构、动态负载均衡^[13]。其中，脉动阵列映射针对矩阵运算等规则计算，将任务映射为 PE 阵列的流水线执行，实现数据流式并行处理。分布式存储架构指采用多交互数据存储技术，解决多个并行阵列访问同一存储器引发的冲突问题，可以最大程度地实现数据重用。动态负载均衡是在运行时监测各 PE 的负载，通过配置调整任务分布，从而避免资源闲置。

(3) 硬件资源时分复用的实现机理

硬件资源时分复用的核心原理是将硬件资源的使用时间划分为多个时隙，不同的计算任务在各自时隙内独占资源，通过快速配置切换实现多任务共享同一硬件的目的。主要实现流程包括时间片划分、配置预加载、上下文管理以及配置切换。时间片划分是指调度器根据任务的优先级和资源需求，将时间划分为固定的或者动态长度的时隙。配置预加载通过双缓冲或者多缓冲机制，在当前任务执行时预加载下一个任务的配置信息，以减少切换产生的时延。上下文管理是为每个任务保存独立的状态信息，在时隙切换时快速恢复上下文。配置切换涉及 3 种不同粒度的时分复用方式，其中细粒度时分是指同一 PE 在不同时钟周期执行不同的操作，通过配置快速切换功能实现。粗粒度时分通过部分重构技术切换 PE 阵列的功能区域，以支持较大规模的任务切换。混合粒度时分则结合细粒度和粗粒度的优势，以平衡切换开销和灵活性^[14]。

(4) 时空协调调度机制

该机制主要通过编译器开展运行时的协同优化。编译器分析任务的特性，确定最佳时空映射策略，生成配置序列和执行时间表。运行时，调度器根据系统负载的变化，实时调整时空资源的配置，优化资源利用率。专用硬件单元主要用于辅助调度决策，如预测计算任务执行时间、监测资源冲突等。

2.2 近存计算技术

近存计算技术打破了冯·诺依曼架构“存储-计算”分离的瓶颈，将PE从CPU/GPU内核迁移至靠近存储介质的位置，从物理层面缩短数据传输的路径，同时减少数据在存储和PE之间的搬运次数，最终实现数据传输过程能耗的大幅度降低。3D动态随机存取存储器（dynamic random access memory, DRAM）架构分层示意如图2所示，其中NoC指片上网络。图2展示了存储层与逻辑层的垂直集成关系，其核心在于将PE垂直集成于存储裸片旁的逻辑层，通过硅通孔（through-silicon via, TSV）技术实现超短距数据交互，从而突破内存墙瓶颈。

近存计算技术主要通过减少数据搬运次数和缩短传输路径的方式实现能耗的大幅度降低，具体机理主要包含以下几个方面。

从物理层面上来看，近存计算技术主要通过缩短数据传输路径来降低单位数据传输的功耗。根据电磁传输理论，数据在互联电路中的能耗与传输距离的平方近似呈正比关系。在传统的架构中，数据从内存到CPU核心的传输距离可达数厘米，传输链路阻抗大，必然导致能耗增

加。在近存架构中，存算距离缩短至毫米，甚至纳米级别，互联链路的寄生电阻、电容大幅度降低，单位数据的传输能耗可实现数量级的降低。

从数据层面上来看，近存计算技术主要通过减少数据搬运次数来降低传输功耗。该技术的核心原理是，在靠近存储单元的位置完成部分计算任务，从而减少需传输至远端处理计算核心的数据量。从系统架构设计的角度来看，该技术主要通过降低访存时延来减少计算资源闲置过程中产生的能耗。近存芯片通过提升内存带宽、降低访存时延等手段，可以使PE的等待时间减少80%以上，从而大幅度减少闲置能耗。与此同时，高带宽互联的能效比远高于传统总线，可以进一步降低系统能耗。

3 设计与实现

3.1 架构设计与实现

本文所提架构的核心设计思路是通过多上下文PE与存储增强型微处理单元（micro processing element, mPE）的深度融合，实现计算与存储的协同优化及功能互补适配，并基于此构建时空并行可重构计算阵列^[15-17]，如图3所示，其中ALU指算术逻辑单元。在阵列构建设计层面，整

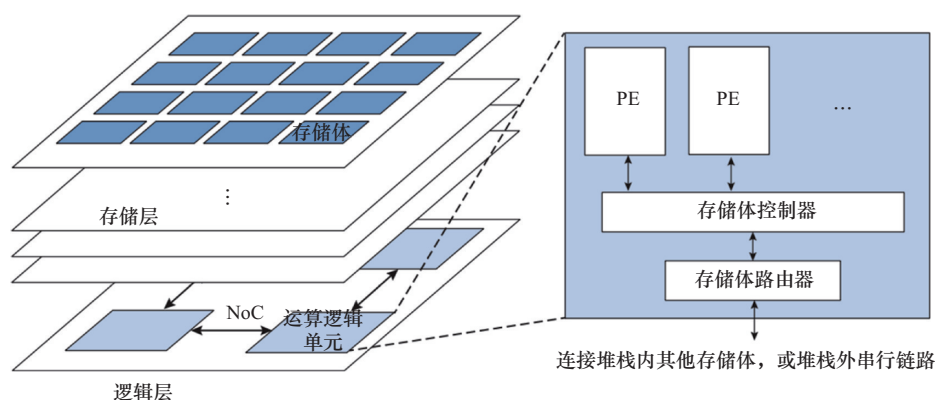


图2 3D DRAM架构分层示意

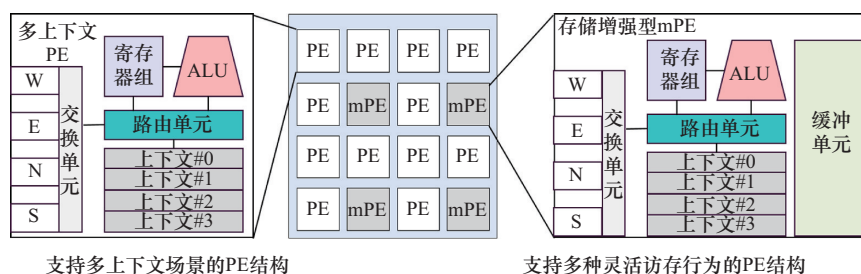


图3 多数据流时空并行的可重构计算阵列结构

合多上下文 PE 阵列和存储增强型 mPE 阵列。其中, 上下文 PE 负责基础逻辑运算, 存储增强型 mPE 通过缓冲单元 (buffer) 与 3D DRAM 存储体深度耦合, 支撑近存计算。同时, 依托存储分区路由 (vault router) 实现堆栈内存存储体间并行互联及堆栈外高速串行链路对接, 保障数据高速传输。在多上下文动态调度的设计上, 每个 PE 配置 4 组独立的配置上下文 (context), 通过交换单元 (switch) 与路由单元 (router) 实现多任务并行处理, 适配智能网卡多队列、多租户需求, 动态切换上下文以调度不同网络任务, 提升资源利用率。在动态重构机制设计上, 重构管理器与配置存储器协同工作, 实时调整 PE/mPE 阵列的互联方式与运算模式, 可在智算中心、AI 训练/推理、网络安全等多场景间快速切换, 以适配不同场景需求。

3.1.1 融合架构整体设计: 奠定时空并行基础

本文提出的基于时空并行可重构计算阵列结构的核心思想是采用异构混合部署和全连接可配置的互联结构, 为两种不同 PE 的协同工作提供硬件层面的支撑。

在粗粒度可重构计算阵列的设计中, 多上下文 PE 和存储增强型 mPE 的最优配置比例通常由计算访存特征、硬件资源利用率均衡理论、流水线无阻塞设计原则及性能、功耗、面积 (power performance area, PPA) 权衡后共同决定。结合本文拟开展的研究及具体应用场景, 如网络报文处理、开源虚拟交换机 (open vSwitch, OVS) 硬件卸载等典型负载, 其运算操作与访存操作的天然比例接近 3:1^[18]。为避免计算资源闲置或访存瓶颈导致的队列阻塞, PE 与 mPE 的配置数量应与计算访存比保持一致。同时, 依据阿姆达尔定律与流水线平衡理论, mPE 因集成片上存储、地址生成与多端口访问逻辑, 其硬件开销约为普通 PE 的 3 倍。采用 3:1 的配比可使 PE 与访存单元同时达到较高利用率, 避免资源失衡造成流水线停顿或数据饥饿; 同时, 该比例可实现计算资源与存储资源在面积与功耗上的均衡^[19]。

在芯片内部处理模块设计上, 该架构根据多种不同类型数据流, 按照 3:1 的比例混合部署多上下文 PE 和存储增强型 mPE, 具备支持并发任

务处理及灵活访存支撑的能力。在互联逻辑上, 采用胖树与交叉开关混合互联的方式, 支持不同 PE 间的高速数据交互, 并且集成了智能路由调度单元, 能够根据数据流类型、任务特征等动态分配数据传输通道, 避免数据堆叠和等候。此外, 该架构支持阵列重构逻辑的统一管理, 可在 1 ms 内实现全局重构与部分重构, 并根据业务需求动态调整两种 PE 的功能分工与协作模式。

3.1.2 核心单元设计: 各司其职, 精准适配需求

基于多线程硬件上下文切换理论、控制依赖化解机制、实时任务调度模型以及硬件多线程 (hardware multi-threading, HMT) 架构与时序约束设计原则, 通过为不同任务分配独立硬件上下文资源, 避免控制流冲突与指令重加载开销, 显著提升系统在复杂网络业务下的并发处理能力^[20]。

多上下文 PE 的设计旨在解决并发任务的高效调度问题, 以化解控制依赖、提高利用效率为重点, 适配智算系统中智能网卡应用的多任务并发场景。在 PE 内部设置 8 组独立上下文存储模块, 每组包含 256 条指令缓存、操作数地址映射表和状态寄存器, 能够缓存 8 套完整任务的控制信息^[21], 覆盖智能网卡应用的核心业务场景; 搭载硬件级快速切换控制器, 检测到控制依赖时, 能够在 12 ns 内完成上下文切换^[22], 流水线不会中断, 单元利用率比传统上下文单元提高 70%; 采用上下文本地化存储设计, 避免片外访存开销, 切换时访存时延降至 2 ns 以内, 为多数数据流并行传输释放互联带宽。

存储增强型 mPE 主要聚焦动态报文的灵活访存, 以适配动态报文和优化数据供给为重点, 针对性地解决智能网卡报文长度多变、访存模式多样的问题。其内置分级可配置数据缓存单元, L1 缓存适配短报文高速访问, L2 缓存适配长报文存储, 支持缓存容量动态调整; 支持先进先出 (first-in first-out, FIFO) 队列、后进先出 (last-in first-out, LIFO) 队列、随机访存、跨步访存 4 种核心模式, 通过软件配置可以适配不同的场景。FIFO 适配 RDMA 连续数据传输; LIFO 适配协议栈解析; 跨步访存适配向量运算类加速任务; 随机访存与计算协同调度模块, 实时接收运算单元任务反馈, 自动调整预取策略与访存模式, 使访

存带宽与运算带宽精准匹配,避免计算等待访存。

基于时空并行的可重构计算阵列结构设计时,存储增强型mPE为多上下文PE提供底层支撑。通过集成分布式局部缓存与高效访存接口,将多上下文切换所需的配置数据、中间计算结果就近存储,避免出现全局访存时延,破解多上下文切换的资源开销与时延瓶颈,保障多上下文PE可快速切换并行策略,适配多数据流、多异构任务的动态需求。另外,多上下文PE发挥可重构优势,动态调度存储增强型mPE的访存带宽与存储粒度,根据不同数据流处理需求,灵活配置存储读写模式,实现计算任务与存储资源的精准匹配,既发挥存储增强型mPE的高带宽、低时延优势,又通过多上下文协同提升PE的利用率,二者深度融合,共同支撑网卡芯片实现多模态数据流的高效并行处理,兼顾高吞吐、低时延与灵活适配性^[23-24]。

该结构既破解了多任务控制依赖导致的资源闲置问题,又满足了智能网卡动态报文处理的灵活访存需求,实现了多数据流的高吞吐、低时延、高灵活处理,适配应用场景下的网络虚拟化、远程直接内存访问(remote direct memory access, RDMA)、协议加速等核心需求。

3.2 算法设计与实现

在算法设计上,主要围绕可重构网络处理、低时延数据传输卸载、流量调度与虚拟化隔离等模块开展工作,通过算法层面的协同,从软件上

实现100 Gbit/s的线速处理和纳秒级别的时延控制,同时要兼顾系统编程的灵活性。这些算法的设计和实现是提升网卡芯片综合性能的关键。

3.2.1 可重构网络处理核心算法

可重构网络处理算法是可重构计算阵列网卡芯片设计的核心支撑。在结构层面,以64核同构轻量级网络专用微引擎(network-oriented multi-context engine, NME)为可重构核心,通过网格状互联架构实现多核协同^[25-27]。每个核内置32 KB的私有指令缓存和64 KB的私有数据缓存,并共享2 MB的静态随机存储器(static random-access memory, SRAM)与最大16 GB的片外双倍数据速率(double data rate, DDR)内存,分别用于存储全局流量表、协议配置参数、缓存报文数据以及临时处理结果。支持循环流水线的可重构阵列配置方法及配置流程^[28]如图4所示。

算法设计的核心由可重构报文解析流水线与匹配动作流水线构成^[29],通过高优先级虚拟通路(high-priority virtual path, HPV)总线实现数据的高速交互。解析流水线采用配置有限状态机(finite state machine, FSM)架构,可灵活适配以太网、互联网协议第4/6版(Internet protocol version 4/6, IPv4/IPv6)、传输控制协议/用户数据报协议(transmission control protocol/user datagram protocol, TCP/UDP)、虚拟扩展局域网(virtual extensible LAN, VxLAN)、融合以太网远程直接内存访问第二版(RDMA over converged Ether-

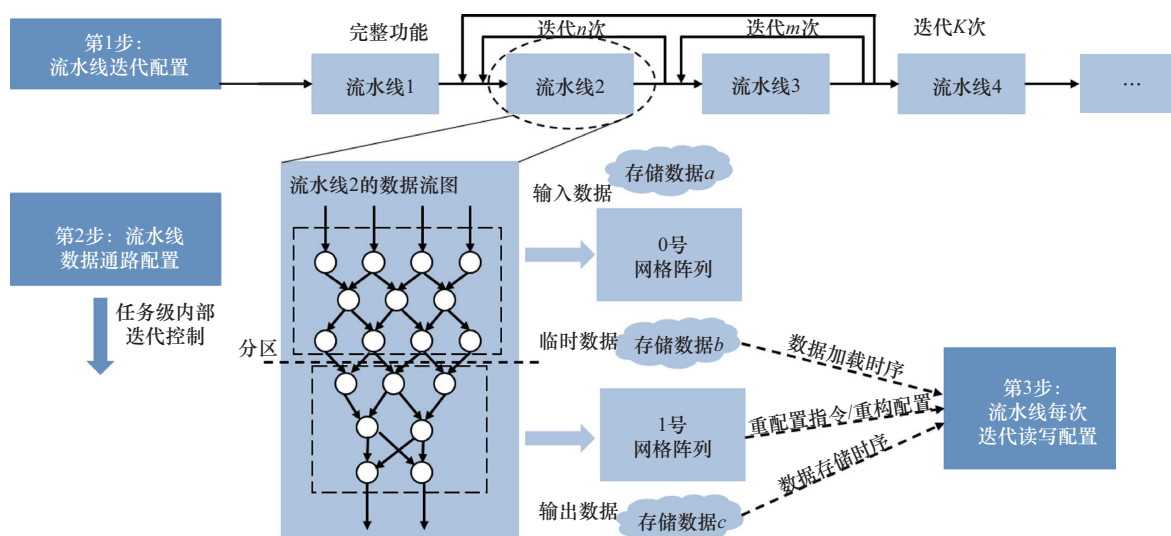


图4 支持循环流水线的可重构阵列配置方法及配置流程

net version 2, RoCEv2) 等多层标准协议, 同时支持自定义协议快速适配, 无须重新编译硬件, 仅通过更新 FSM 配置参数就可以完成解析逻辑的动态调整。该解析流水线支持 64 B 至 9 KB 可变长度报文的处理, 在预处理阶段完成报文去冗余、循环冗余校验 (cyclic redundancy check, CRC) 初步校验, 单路解析时延 < 500 ns, 并生成 64 B 固定长度的包头向量 (packet header vector, PHV), 将分散的协议字段标准化, 为后续处理提供统一接口。

匹配动作流水线采用多级并行架构, 支持精准匹配、最长前缀匹配 (longest prefix match, LPM) 等多种模式。LPM 采用硬件化前缀树实现, 查找时延 < 10 ns; 精准匹配通过哈希表硬件加速, 冲突率控制在 0.1% 以下, 可支持 1 Mbit/s+ 流表项硬件加速。动作引擎内置 16 个并行执行单元, 能够完成转发、字段修改、封装解封装等

多种操作。流表更新采用双缓冲机制, 支持在线动态更新的同时不中断数据传输。全流水线单包处理时延 < 1 μ s, 最终实现 2×100 Gbit/s 线速处理。

3.2.2 低时延数据传输卸载

低时延数据传输卸载主要面向高性能智算系统中高速数据交互的需求, 其设计核心是集成 RoCEv2 RDMA 协议全硬件卸载与可重构直接内存访问 (DMA) 引擎^[30], 以实现无须 CPU 干预的高速数据传输。在 RoCEv2 RDMA 卸载模块中, 每个物理功能 (physical function, PF) 支持 64 K 队列对 (queue pair, QP) 与 128 K 完成队列 (completion queue, CQ), 每对 QP 独立维护发送、接收及 CQ, 支持可靠、不可靠、数据包等 3 种传输模式, 可以完美适配不同业务的可靠性需求。OVS 管理层卸载实现机理流程如图 5 所示, 其中 netdev 指网络设备结构体, S-SPU 指安

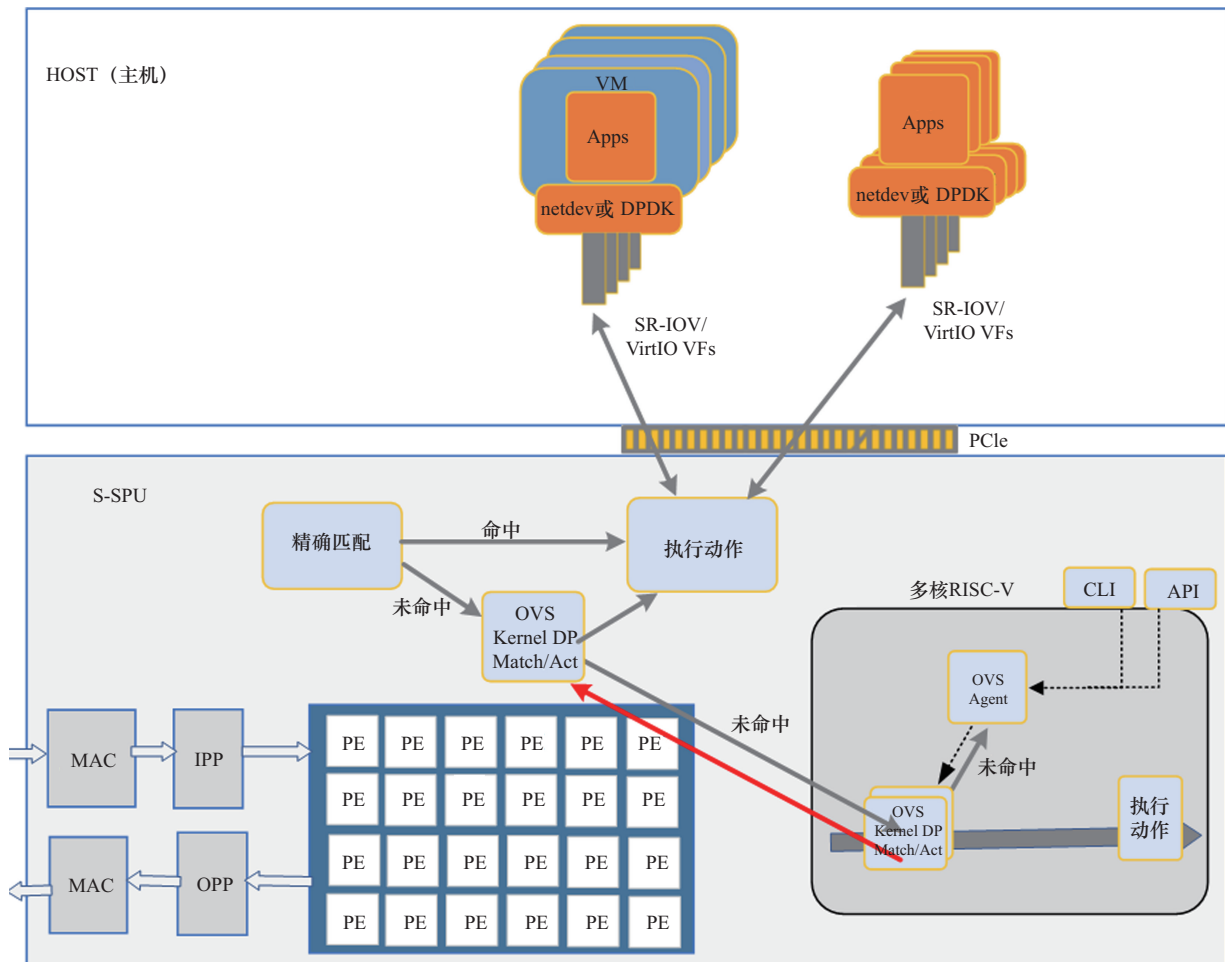


图 5 OVS 管理层卸载实现机理流程

全计算子单元，MAC指媒体访问控制，IPP指内存报文解析器，OPP指外层报文解析器，RISC-V指第5代精简指令集计算机，CLI指命令行界面，API指应用程序接口，OVS Agent指开放虚拟交换机代理，OVS Kernel DP Match/Act指开放虚拟交换机内核态数据通路的流匹配/动作执行模块，VirtIO VFs指虚拟化输入输出功能。

可重构DMA引擎采用描述符环与分散/聚集相结合的机制^[31]，支持非连续物理内存的高效传输，适配大页与用户态内存，实现数据零拷贝，避免内存拷贝带来的时延开销。该引擎集成TCP/IP全栈卸载（TCP offload engine, TOE）、TCP分段卸载/大接收卸载（TCP segmentation offload/large receive offload, TSO/LRO）分段重组及校验和硬件计算功能。TOE通过硬件化实现滑动窗口管理与重传机制；TSO与LRO分别实现大包分段与小包重组，减少CPU中断次数。基于PCIe 4.0×16接口，直接内存访问（DMA）双向带宽达32 Gbit/s，支持1 024个并发队列，平衡传输时延与CPU占用率。

3.2.3 流量调度与虚拟化隔离

流量调度与虚拟化隔离主要解决多业务、多租户场景下的资源分配和安全隔离问题，旨在实现流量的高效调度和租户资源的独立管控^[32]。流量调度采用端口、队列、流的3级架构，系统支持1 024个硬件队列，并通过严格优先级、加权轮询和加权公平队列等混合调度算法，根据业务优先级和带宽需求动态分配网络资源，优先保证高优先级任务的数据传输质量^[33-35]。

在拥塞管理方面，采用优先级流控方式，支持多达8个优先级的独立流控，运行过程中根据队列水位动态发送暂停帧，避免拥塞丢包。同时，集成显式拥塞通知标记功能，可以实时检测队列的拥塞状态并对IP头的标志位进行标记，触发端侧降速，实现拥塞的主动控制。虚拟化隔离的实现则是通过单根I/O虚拟化（single-root input/output virtualization, SR-IOV）技术，支持2个PF和256个虚拟功能（virtual function, VF）部署，实现VF Trust、Anti-Spoof（反地址欺骗）等硬件化隔离机制，防止租户间资源越权访问。同时，集成虚拟以太网桥/虚拟端口聚合器/嵌入

式硬件交换机（VEB/VEPA/ESwitch），支持多租户网络隔离、流量镜像与访问控制列表（access control list, ACL）访问控制，适配云数据中心多租户部署需求，且支持127 VF/PF的RDMA虚拟化，实现多租户RDMA资源的独立分配。

3.3 理论分析

借助融合协同机制，可以实现1+1>2的时空并行效能。两种单元并非独立工作，而是通过三重协同机制进行深度融合，核心在于根据多数据流特征、任务计算粒度及实时负载，灵活分配两种单元的数量与功能，兼顾可重构灵活性与数据处理效率，在最大限度上实现阵列的处理能力。

通过构建异构可重构计算阵列，满足了网络处理过程的差异化负载需求，并通过时分复用实现高吞吐与高计算资源利用率。硬件资源动态划分示意如图6所示。

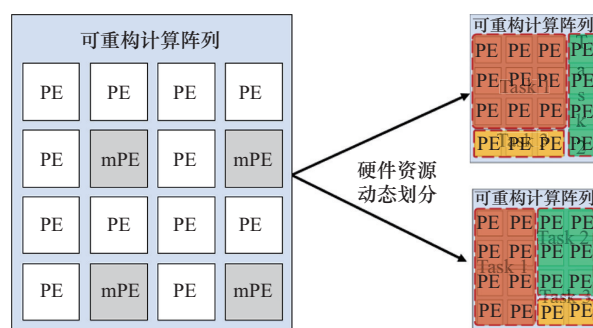


图6 硬件资源动态划分示意

3.3.1 三重协同机制深度融合，最大化阵列处理能力

通过时空维度协同，可使并行效率成倍提升。在时间维度上，PE的上下文切换与mPE的访存操作流水线重叠：PE切换上下文时，mPE同步完成数据预取，减少任务间隙的空闲时间。在空间维度上，阵列能够按任务类型进行动态分区，控制依赖型多任务被分配到PE集群，访存密集型任务分配到mPE集群，二者可同时处理且不产生冲突。

3.3.2 数据流调度协同，精准匹配资源

智能路由调度模块根据数据流特征动态分配处理单元：控制类数据流优先分配至PE，利用其快速切换能力处理多任务并发；业务类数据流

优先分配至 mPE，借助其灵活访存能力提升数据吞吐；混合数据流通过互联网络实现 PE 处理控制头和 mPE 处理数据块的流水线协作。

3.3.3 重构适配协同：灵活应对业务迭代

全局重构控制器支持两种单元的功能动态扩展。PE 可通过重构临时集成轻量级访存模块，适配突发的访存需求。mPE 可通过重构加载多上下文控制逻辑，临时承担多任务并发处理任务。在不同架构数据的处理逻辑上，基于动态异构多数据通路并行处理的模式具备明显优势。

在面对新协议、新业务时，无须重构整个阵列，仅需调整对应单元的配置逻辑，适配周期大大缩短，完全满足可重构智能网卡产品快速迭代的需求。

4 实验验证及分析

4.1 实验方法

充分借鉴国际公开可重构智能网卡及可重构计算阵列领域性能优异、技术先进的实验验证方案，在功能验证、性能对比、资源开销、可重构性等维度形成可直接落地的实验流程，与行业方案形成完整对比闭环。

针对本文方案的实验验证主要分为两个阶段，分别采用不同的验证方法。第一阶段开展技术方案的正确性验证，即流片前设计验证：首先通过高层次建模工具进行架构设计，然后利用可重构计算阵列建模与行为评估工具验证相关技术创新点，接着在 FPGA 上进行电路验证、通过新思（Synopsys）、Cadence 等芯片设计与验证工具进行版图设计，确保技术方案的正确性。第二阶段开展基于该技术创新的工程化验证，即流片后芯片测试与系统级验证：基于该关键技术完成芯片设计，完成芯片流片后通过网络分析仪、高性能示波器进行网络协议分析与高速信号定性测试，

进而在具体服务器及智算系统中开展定量测试。

4.1.1 测试平台及资源占用情况

建模仿真验证时采用面向 100 Gbit/s+ 高速网络与可重构计算的主流高端 FPGA 平台，该平台具备充足的逻辑单元、存储和高速接口资源，与本文研究的内容及定位完全匹配。验证平台与资源占用情况见表 1，其中 LUT 指查找表，REU 指寄存器，SerDes 指串行解串器。

4.1.2 流片前验证方法

流片前验证主要分为 3 个步骤：通过仿真环境全面测试芯片功能、协议、时序及物理设计的正确性，并通过 FPGA 平台进行原型验证。流片前验证过程采用的软件工具及环境如图 7 所示。

（1）RTL 功能与协议验证方法

采用通用验证方法学（universal verification methodology, UVM）搭建标准化验证环境，对芯片全功能模块进行测试。编写覆盖全场景的测试用例，包括正常工作场景、边界条件及异常场景（如链路中断、数据错误、协议异常），全面验证芯片的逻辑功能及 PCIe 4.0、100 Gbit/s 以太网、RDMA、SR-IOV、SM2/IPSec 等协议一致性。通过功能覆盖率、代码覆盖率、断言覆盖率三维分析，确保验证无遗漏。当各项覆盖率达到预设阈值时，完成该阶段验证，确保寄存器传输级（register transfer level, RTL）设计逻辑正确。

（2）时序与物理验证方法

时序验证采用静态时序分析（static timing analysis, STA），整理芯片关键路径，验证建立时间、保持时间、最大工作频率及路径时延，确保 100 Gbit/s 线速传输下的时序收敛。功耗验证结合动态功耗仿真与静态功耗分析，优化电源域划分与时钟树设计，将芯片功耗控制在预设范围内。物理验证通过设计规则检查（design rule check, DRC）、版图与原理图一致性检查（layout

表 1 验证平台与资源占用情况

项目	资源情况说明
FPGA 型号	Xilinx Virtex UltraScale+VU9P
逻辑资源	LUT 占 45%~55%，REG 占 40%~50%（含 100 Gbit/s 高速接口、可重构计算阵列、多上下文管理模块）
Block RAM	占 60%~70%（用于 8 组独立上下文存储、操作数地址映射表与状态寄存器缓存）
DSP48	占 55%~65%（支撑高速报文处理、加解密加速与硬件快速切换控制器逻辑）
高速接口	集成 2×100 Gbit/s/40 Gbit/s/25 Gbit/s 可配置 SerDes，PCIe 4.0×16 接口资源完整占用

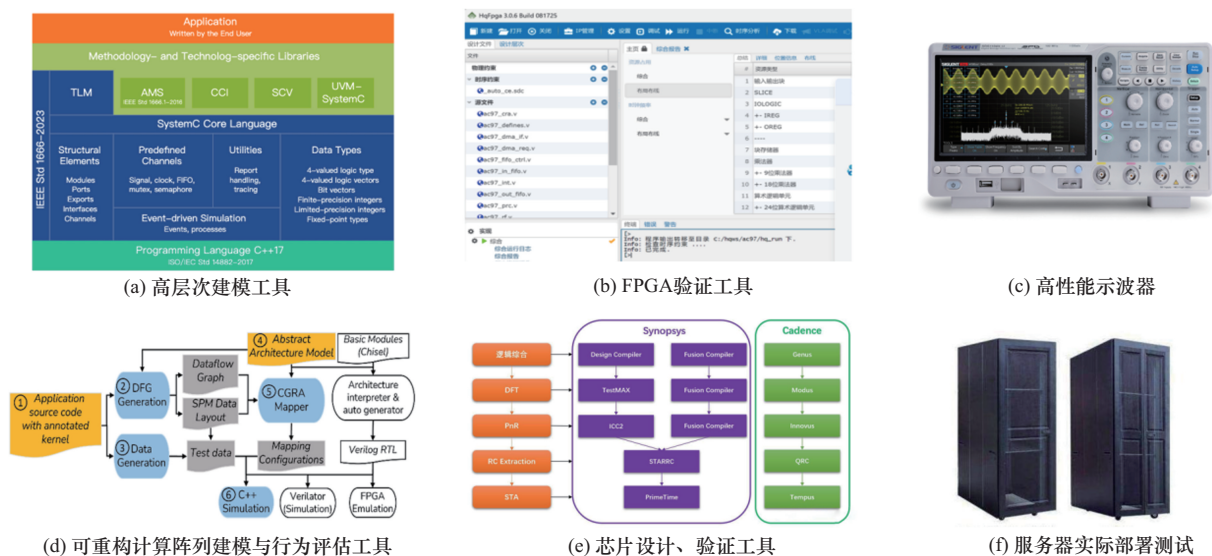


图7 流片前验证过程采用的软件工具及环境示意

versus schematic, LVS)、电气规则检查 (electrical rule check, ERC) 等手段, 排查版图设计问题, 确保版图设计符合制造要求。

(3) FPGA原型验证方法

将RTL代码综合至FPGA验证平台, 搭建软硬件协同的验证环境。验证驱动与芯片的交互逻辑, 包括寄存器访问、中断响应、DMA数据通路传输等; 测试系统级功能, 如100 Gbit/s线速转发(核心评估包长: 64 B小包; 典型验证包长: 1 518 B大包)、RDMA低时延通信、SR-IOV隔离等; 实测芯片吞吐量、端到端时延、丢包率等性能指标, 检验芯片设计在接近真实应用场景下的可行性与稳定性, 为流片提供数据支撑。

4.1.3 流片后的芯片测试和系统级的验证

(1) 晶圆级测试方法

采用探针测试技术, 对芯片进行参数测试和结构测试。参数测试重点检测电源电压、I/O电平、漏电流、静态电流、时钟抖动等电气参数, 确保其满足设计要求; 结构测试通过扫描链及内置自测试(built-in self-test, BIST)技术, 筛查芯片制造过程中产生的逻辑缺陷与存储器故障, 剔除不合格晶圆, 降低封装成本。同时完成基础寄存器读写、复位、时钟等功能点快速验证, 确保芯片基础工作正常。

(2) 封装成品测试方法

对封装后的成品芯片进行全功能与全性能测试。全功能测试覆盖PCIe链路训练、100 Gbit/s

光口协商、MAC/PHY功能、匹配-动作表、DMA传输、硬件加解密、RDMA、SR-IOV等所有核心功能, 确保功能正确与协议兼容。性能测试实测100 Gbit/s线速吞吐量、小包转发能力(packets per second, PPS)、端到端时延, 以及加解密与RDMA带宽; 可靠性测试通过电压/温度拉偏、电源噪声、信号完整性、静电放电(electro-static discharge, ESD)、闩锁效应测试, 验证芯片在复杂环境下的工作稳定性。

(3) 板级与整机兼容性测试方法

将网卡芯片焊接至自研网卡载板上, 并插入国产架构服务器, 测试其与国产主流操作系统、虚拟化平台的兼容性。通过7×24小时满负载压力测试, 监测芯片错误率、平均无故障工作时间(mean time between failures, MTBF), 检验其工作稳定性; 测试与主流交换机、智能网卡及存储设备的互操作性, 检验芯片在实际系统中的适配性, 实现即插即用。

(4) 应用场景针对性验证方法

结合实际情况开展针对性验证。在数据中心场景下, 测试云网络与容器网络东西向流量转发性能; 在AI场景下, 验证RDMA低时延通信与分布式训练网络的适配性。通过实际测试确保芯片设计贴合实际应用场景。

目前, 已经完成芯片的架构设计、功能设计及原型芯片设计验证, 即将进入芯片的流片阶段。本文重点介绍第一阶段(即技术方案的正确

性验证)过程,工程应用情况验证将作为流片后的重点工作在芯片流片后持续开展。

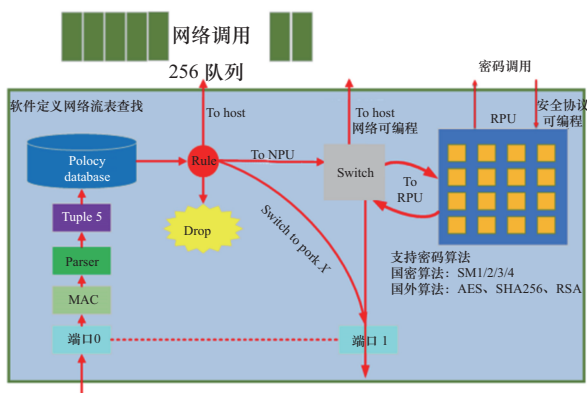
4.2 实验过程及结论

本节主要通过搭建针对性的实验环境,开展仿真测试与硬件实测。在充分完成软件仿真的基础上,将RTL代码综合至FPGA验证平台,通过软硬件协同验证环境完成两大核心目标的验证:一方面,验证高性能可重构网络处理模块的功能完整性与可靠性,确保自研固件能准确实现网络包解析、分片、封装及抗重放等核心网络编程功能;另一方面,验证模块对 100 Gbit/s 网络处理能力的支持程度,通过软件的仿真分析与FPGA硬件测试,确认模块在 100 Gbit/s 带宽场景下的处理性能达标,为后续

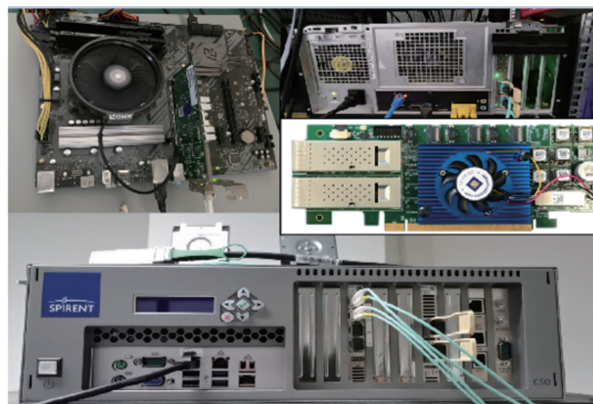
智能网卡集成、设计应用等提供数据支撑。

(1) 自研固件验证网络包的解析、分片、封装、抗重放等网络编程功能。

主要的实验方法是搭建 100 Gbit/s 网络点对点链路的硬件测试环境,将FPGA开发板通过 100 Gbit/s 光模块与网络性能分析仪及服务器进行互联,如图 8 所示。首先通过 ModelSim 进行应用场景的系统仿真,输入自定义测试数据包,验证各功能逻辑的正确性;然后将固件烧录至FPGA开发板,建立实际网络链路,通过真实数据包传输,检验网络传输功能在硬件环境中的准确性与实时性。验证指标包括功能实现准确率、数据包处理时延、可重构配置响应时间等。



(a) 数据流逻辑图



(b) FPGA仿真测试环境

图 8 智能网卡的数据流逻辑图及FPGA仿真测试环境

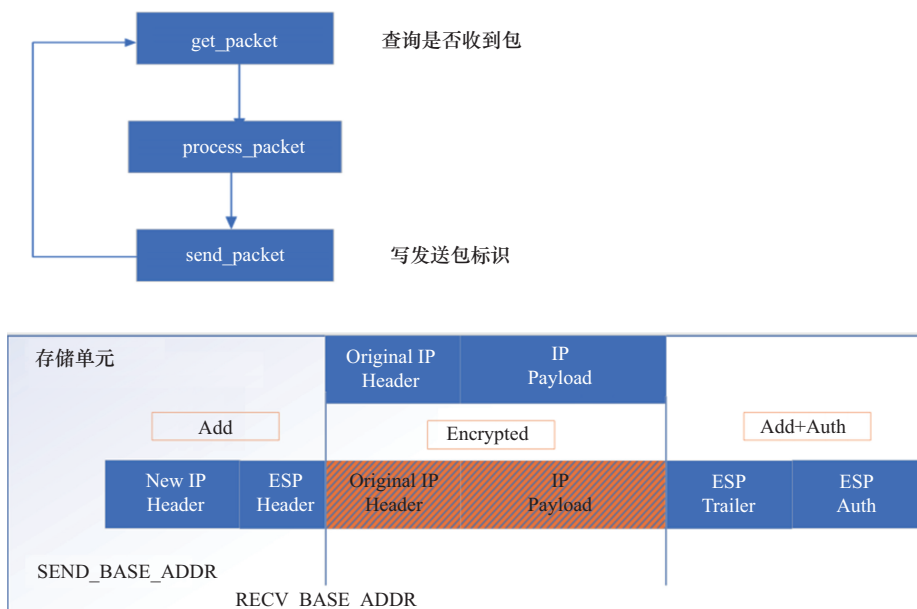


图 9 process_packet流程

编程结构采用 process_packet 流程，如图9所示，在同一段内存中，将数据包进行加密操作，并在加密内存前后添加两个数据段，用于存放封装安全载荷（encapsulating security payload, ESP）包的其他部分。此过程不涉及原数据包的拷贝操作，而是在数据包拷贝到内存时即已分配好数据空间，为后续插入数据做好准备。这样就以零拷贝的方式，完成了一个标准 ESP 包的封装。这也是本编程方法的一个优势：直接在数据包原有内存段上进行处理，最大限度减少内存处理过程中的拷贝操作。

功能测试结果见表2，其中 ICMP 指互联网控制报文协议，ARP 指地址解析协议，PFC 指无损以太网核心机制，iperf 为网络带宽测试工具。

表2 功能测试结果

测试内容	测试结果
IPv4 性能测试	通过
IPv6 性能测试	通过
ICMP 测试	通过
ARP 测试	通过
pause 帧测试	通过
带优先级的流控 PFC 测试	通过
iperf 测试	通过

经过软件模拟仿真和实际硬件环境测试，高性能可重构网络处理模块的各项核心功能均可通过测试并且功能的准确率达到100%，数据处理平均时延不超过2 μs。测试结果表明，时域调制与空域分区协同设计，可以有效解决传统架构高通量与可重构性的矛盾，且可重构配置响应迅速，可完全满足高性能、可灵活适配不同应用场

景的设计需求。

(2) 100 Gbit/s 网络数据传输与处理性能检验

通过软件模型仿真分析和FPGA硬件模拟测试，检验高性能可重构网络处理模块在100 Gbit/s 带宽场景下的吞吐量、丢包率、处理时延等关键性能指标，验证模块稳定支撑100 Gbit/s 网络的数据传输与处理能力。仿真验证采用 ModelSim 仿真环境结合 System Verilog 搭建100 Gbit/s 以太网传输模型，模拟100 Gbit/s 带宽流量场景；硬件测试搭建100 Gbit/s 全链路测试环境，通过网络性能分析仪生成100 Gbit/s 带宽流量，实测模块处理能力。

网络数据转发测试结果见表3。

表3数据表明，采用时空并行可重构架构的网络处理模块在100 Gbit/s 带宽场景下，吞吐量、丢包率、时延等指标均能够满足相关网络传输标准，达到预定设计构想。模块具备稳定支持100 Gbit/s 网络数据传输与处理能力，系统在运行过程中稳定可靠，能够满足高带宽网络应用场景的需求。

(3) 围绕网络加速、AI推理加速、存储虚拟化等核心场景验证，形成异构计算协同验证体系

多模态异构任务验证体系见表4，其中RoCE v2指第2代融合以太网RDMA，NVMe-oF指基于远程直接内存访问的网络NUME存储协议。能效比测试以纯CPU软件处理方案为基准，即只用服务器CPU通过通用协议栈与软件库的形式完成系统异构任务的处理。测试平台基准参数见表5，其中TDP为热设计功耗。

表3 网络数据转发情况测试结果

总测试次数/次	合格试验次数/次	帧长/B	短期负载	输入负载	聚合吞吐量/ (Mbit·s ⁻¹)	聚合理论最大带宽/ (Mbit·s ⁻¹)
1	1	64	40.445%	40.445%	40 445.314	100 000
1	1	128	65.195%	65.195%	65 195.314	100 000
1	1	256	93.813%	93.813%	93 812.503	100 000
1	1	512	100%	100%	100 000	100 000
1	1	1 024	100%	100%	100 000	100 000
1	1	1 280	100%	100%	100 000	100 000
1	1	1 518	100%	100%	100 000	100 000

表 4 异构任务测试验证体系

任务类别	典型负载/子任务	对应硬件模块	测试目的
网络协议卸载	TCP/IP 协议栈解析、VxLAN/GRE 隧道封装/解封装、RDMA RoCE v2 协议处理	64 核网络可编程引擎、硬件流表引擎	验证高并发网络任务调度效率，测试 100 Gbit/s 线速下的任务切换性能
AI 推理加速	多模态特征提取（图像/文本/语音）、大模型推理数据预处理、向量检索加速	可重构计算阵列、专用张量处理单元	验证异构算力对多模态 AI 任务的适配性与调度成功率
存储虚拟化	NVMe-oF RDMA 传输、块存储 I/O 卸载、多租户存储隔离	PCIe 4.0×16 主机接口、存储队列管理单元	验证存储与网络异构任务的协同调度，提升系统整体能效

表 5 测试平台基准参数

基准维度	具体参数	说明
基准平台	飞腾 64 核服务器（2.1 GHz）+ 普通千兆/万兆网卡	与可重构芯片测试平台保持 CPU 架构一致，排除平台差异干扰
基准功耗	基准平台总功耗=CPU 满载功耗（约 700 W，双路，单路 TDP 功耗 350 W）+ 普通网卡功耗（约 10 W）+ 内存/主板等基础功耗（约 50 W）=760 W	包含完成异构任务所需的全部硬件功耗，覆盖全链路
基准性能	基准平台在多模态异构任务下的总吞吐量与任务完成率	作为对比基准，确保测试公平性

测试指标包括任务调度成功率、能效比和性能损耗。

任务调度成功率：测试覆盖网络转发、安全卸载、数据处理等多类型并发任务，通过自研调度监控工具，实时统计总任务发起数与成功调度数。

能效比：通过功率计采集网卡在高并行负载下的实时功耗，通过网络测试仪统计网卡的任务吞吐量，并计算能效比。

性能损耗：分别在正常模式和低功耗模式下，以相同负载测试网卡的吞吐量，并计算性能损耗。

可重构架构与固定架构的 3 项测试指标对比见表 6。

在多模态异构任务的测试中，本文方案可动态重构计算通路，任务调度成功率在 99% 以上，在高并行处理场景下的能效比较基准方案提升 35% 以上，在低功耗模式下的性能损耗控制在 10% 以内。实验结果表明，本文设计的可重构架构可适配各类异构任务，同时兼顾低功耗与高并行处理需求，并为多模态数据处理适配提供了可行的解决方案。

（4）相同硬件资源下时空并行可重构架构的对比测试

基于标准测试平台与测试流程，针对 PE 利用率、互联冲突率、全局访存时延等核心性能指标进行对比测试，进一步验证采用时空并行可重构架构设计的优越性。

PE 利用率测试：以固定架构为基准，在相同负载下进行对比测试。通过芯片内置性能计数器，实时采集 PE 的活跃周期数与总周期数，计算 PE 利用率。

互联冲突率测试：采用全片上网络多点同时注入流量的方式，覆盖 PE、DMA 控制器、片上存储、PCIe 接口等关键节点，模拟高并发数据传输场景。通过性能检测工具采集网络的冲突数与总请求数，计算互联冲突率。

全局访存时延测试：由 100 Gbit/s 网络测试仪发送测试数据包，记录数据包到达网卡 MAC 接口的时间戳及成功写入主机内存的时间戳，计算端到端时延。

可重构架构与固定架构的 3 项核心性能指标对比见表 7。

表 6 可重构架构与固定架构的 3 项测试指标对比

测试指标	测试基准（固定架构）	可重构架构	提升情况
任务调度成功率	95.2%~96.8%（均值 96.0%）	99.1%~99.7%（均值 99.4%）	99% 以上
能效比	2.8~3.2（均值 3.0）	4.0~4.3（均值 4.1）	提升 35% 以上
低功耗模式下性能损耗	14.5%~16.2%（均值 15.3%）	7.8%~9.5%（均值 8.6%）	控制在 10% 以内

表7 可重构架构与固定架构的3项核心性能指标对比

测试指标	测试基准（固定架构）	可重构架构	提升情况
PE利用率	45%~55%（典型场景均值50%）	75%~85%（典型场景均值80%）	提升30%以上
片上互联冲突率	18%~25%（高并发场景均值22%）	9%~13%（高并发场景均值11%）	降低40%左右
全局访存时延	1.8~2.2 μs （均值2.0 μs ）	1.3~1.5 μs （均值1.4 μs ）	缩短25%左右

在同等硬件资源的情况下，相较于传统单一时域/空域并行架构，本文方案的PE利用率可以提升30%以上，互联冲突率降低约40%，全局访存时延缩短约25%，吞吐率得到明显提升。

综上，通过对多数据流处理的时空并行可重构架构模块进行功能验证与性能验证两个层次的实验，从两个维度全面验证了可重构网络处理模块功能的有效性和可靠性。测试结果表明，该架构不仅能准确、高效地实现网络包解析、分片、封装、抗重放等核心功能，而且具备良好的可重构性和灵活性；同时能稳定支撑100 Gbit/s网络数据处理，各项性能指标满足高速网络数据传输要求。实验结果表明，该架构具备良好的可实现性与规模化扩展能力，可为国产智能网卡芯片研制提供可靠的技术基础，助力弥补国内可重构架构的技术短板，并为后续基于时空并行的可重构阵列国产网卡芯片设计的持续优化、智能网卡集成设计以及高性能智算系统实际工程应用提供可靠依据。

5 技术应用

基于多数据流并行的可重构计算阵列架构设计的网卡芯片，为国产智能网卡设计提供了新的思路。该芯片可将传统网卡的数据搬运功能升级为包含网络传输、协议卸载、存储加速于一体的智能数据处理器，在多个核心场景中发挥技术优势，提供全方位的硬件加速能力。从具体场景来看，在智算中心AI训练集群场景，可以为大模型训练、AI算力集群等应用提供低时延、高带宽的节点间互联，支撑RDMA高速通信与GPU直传；在数据中心云网络虚拟化卸载场景中，支持云数据中心的虚拟交换机卸载与多租户流表加速，有效解决传统CPU软转发的性能瓶颈；尤为重要的是，在信息技术应用创新国产化替代等应用场景中，可完全替代海外中高端智能网卡，实现网络设备的国产自主可控。

目前，该项目已完成关键技术的落地验证。基于可重构技术研发的国产化网卡芯片已完成RTL设计、综合布局布线及时序收敛，并通过了FPGA原型验证与性能测试，各项指标均达到预定设计目标。同时，芯片与高性能智算系统的联调适配已成功完成，可稳定实现网络协议处理、算力卸载等核心功能，验证了技术应用的可行性与可靠性。基于可重构技术开发的网卡芯片的前期设计工作已基本完成，即将进入流片环节。流片完成后，将为智算系统智能网卡创新设计提供可落地、可推广的实践方案。

6 展望

面向多数据流的时空并行可重构计算阵列结构网卡芯片设计，作为应对智算系统算力需求的重要技术方向，未来将在可重构智能网卡芯片的架构设计、性能优化与生态建设等方面逐步实现突破并走向工程应用。通过精细化时空并行调度、异构融合可重构单元、存算一体等多种技术的创新应用，可重构计算阵列的并行效率、能效比和灵活性得到显著提升。未来，面向多数据流的时空并行可重构计算阵列将成为构建高效、灵活、低功耗计算系统的重要支撑，为国产可重构智能网卡的发展及应用提供强大的技术动力。

参考文献：

- [1] 曲彤洲,戴紫彬,陈琳,等.一种面向序列密码的混合粒度并行运算单元[J].电子与信息学报,2023,45(1):78-86.
Qu T Z, Dai Z B, Chen L, et al. A hybrid granularity parallel arithmetical unit for stream cipher[J]. Journal of Electronics & Information Technology, 2023, 45(1): 78-86.
- [2] 丁博.面向异构资源的FPGA动态重构系统自动化设计方法[D].合肥:中国科学技术大学,2024.
Ding B. Automated design method for FPGA dynamic reconfiguration system oriented to heterogeneous resources[D]. Hefei: University of Science and Technology of China, 2024.
- [3] 冷静文,过敏意,曾德泽,等.数据流芯片的发展现状、趋势与挑战[J].

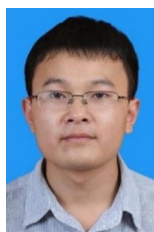
- 中国科学: 信息科学, 2025, 55(3): 452-463.
- Leng J W, Guo M Yi, Zeng D Z, et al. Development status, trends and challenges of dataflow chips[J]. Science China: Information Sciences, 2025, 58(3): 452-463.
- [4] 张俊也. 可编程数据中心网络的 RDMA 传输韧性增强机制[D]. 北京: 北京邮电大学, 2025.
- Zhang J Y. Resilience enhancement mechanism for RDMA transmission in programmable data center networks[D]. Beijing: Beijing University of Posts and Telecommunications, 2025.
- [5] Kaiser T, Gottschalk E, Biethahn K, et al. Pasithea-1: an energy-efficient sequential reconfigurable array with CPU-like programmability[J]. IEEE Open Journal of Circuits and Systems, 2025, 6: 1-13.
- [6] 朱志成. 基于 SIMT 架构的智能网卡片外访存优化[D]. 武汉: 华中科技大学, 2025.
- Zhu Z C. Optimization of off-chip memory access for smart NIC based on SIMT architecture[D]. Wuhan: Huazhong University of Science and Technology, 2025.
- [7] Han M, Wang L, Xiao L M, et al. ReDas: a lightweight architecture for supporting fine-grained reshaping and multiple dataflows on systolic array[J]. IEEE Transactions on Computers, 2014, 73(8): 1997-2011.
- [8] 付炫钰, 李颖颖. LLM-GEMM: 异构计算架构下通用矩阵乘法的自动化代码转换框架[J]. 软件导刊, 2026.
- Fu X Y, Li Y Y. LLM-GEMM: an automatic code transformation framework for general matrix multiplication under heterogeneous computing architectures[J]. Software Guide, 2026.
- [9] 中科寒武纪科技股份有限公司. 思元 590 (MLU590) 芯片技术规格说明书[Z]. 2025.
- Cambricon Technologies Co., Ltd., Specification manual of MLU590 (SiYuan 590) Chip[Z]. 2025.
- [10] 张多利, 陈楠, 汪杨, 等. 粗粒度多核系统任务级多线程调度研究[J]. 微电子学与计算机, 2020, 37 (1): 46-52.
- Zhang D L, Chen N, Wang Y, et al. Research on task-level multi-thread scheduling for coarse-grained multi-core systems[J]. Microelectronics & Computer, 2020, 37(1): 46-52.
- [11] Nowatzki T, Gangadhar V, Sankaralingam K, et al. Pushing the limits of accelerator efficiency while retaining programmability[C]//Proceedings of the 2016 IEEE International Symposium on High Performance Computer Architecture (HPCA). Piscataway: IEEE Press, 2016: 27-39.
- [12] Liu D J, Liu T, Mo X Y, et al. Polyhedral-based pipelining of imperfectly-nested loop for CGRAs[C]//Proceedings of the 2021 IEEE/ACM International Conference on Computer-Aided Design. Newark, USA: IEEE, 2021:1-9.
- [13] 王彦伟, 李仁刚, 徐冉, 等. 基于可重构架构的数据中心异构加速软硬件系统级平台[J]. 计算机研究与发展, 2025, 62 (4): 963-977.
- Wang Y W, Li R G, Xu R, et al. Hardware and software system-level platform for heterogeneous acceleration in data centers based on reconfigurable architecture[J]. Journal of Computer Research and Development, 2025, 62(4): 963-977.
- [14] 郭岩松, 刘雷波. 一种面向分组密码的粗粒度可重构阵列及 AES 算法映射[J]. 微电子学与计算机, 2015, 32 (9): 1-5.
- Guo Y S, Liu L B. A block cipher oriented coarse-grained reconfigurable array and AES algorithm mapping[J]. Microelectronics & Computer, 2015, 32 (9): 1-5.
- [15] Wen T H, Huang J, Huang W, et al. Fusion of memristor and digital compute-in-memory processing for energy-efficient edge computing[J]. Science, 2024, 384(6693): 325-332.
- [16] 洪途, 景乃锋. 存算解耦合的粗粒度可重构阵列访存结构设计[J]. 计算机工程, 2021, 47(2): 239-245.
- Hong T, Jing N F. Design of memory access structure for coarse-grained reconfigurable array with decoupled access execute[J]. Computer Engineering, 2021, 47 (2): 239-245.
- [17] 魏少军, 李兆石, 朱建峰, 等. 可重构计算: 软件可定义的计算引擎[J]. 中国科学: 信息科学, 2020, 50 (9): 1407-1426.
- Wei S J, Li Z S, Zhu J F, et al. Reconfigurable computing: toward software defined chips [J]. Science China Information Sciences, 2020, 63(9): 190401.
- [18] Prabhakar R, Zhang Y Q, Koeplinger D, et al. Plasticine: a reconfigurable architecture for parallel patterns[C]//Proceedings of the 2017 ACM/IEEE 44th Annual International Symposium on Computer Architecture (ISCA). Piscataway: IEEE Press, 2017: 389-402.
- [19] 窦勇, 邵贵明, 徐进辉, 等. 支持循环自动流水线的粗粒度可重构阵列体系结构[J]. 中国科学(E辑), 2008(4): 579-591.
- Dou Y, Wu G M, Xu J H, et al. Coarse-grained reconfigurable array architecture supporting cyclic automatic pipeline[J]. Science in China (Series E), 2008(4): 579-591.
- [20] Hennessy J L, Patterson D A, Kozyrakis C. Computer architecture: a quantitative approach [M]. 7th ed. Cambridge: Morgan Kaufmann, 2025.
- [21] Chacko V, Anderson J H. Power, performance and area consequences of multi-context support in CGRAs[C]//Proceedings of the 32nd IEEE International Conference on Application-specific Systems, Architectures and Processors. Newark, USA: IEEE, 2021: 53-58.
- [22] 常立博, 武丹妮, 杜慧敏, 等. 可重构 CNN 处理器的高效能自适应映射策略[J]. 计算机辅助设计与图形学学报, 2025, 37 (7): 1157-1169.
- Chang L B, Wu D N, Du H M, et al. Adaptive mapping for data orchestration in reconfigurable CNN processors via deep reinforcement learning[J]. Journal of Computer-Aided Design & Computer Graphics, 2025, 37(7): 1157-1169.
- [23] 常立博, 武丹妮, 杜慧敏, 张盛兵, 郝鹏, 蔡秀霞. 可重构 CNN 处理器的高效能自适应映射策略[J]. 计算机辅助设计与图形学学报, 2025, 37 (7): 1157-1169.
- Chang Libo, Wu Danni, Du Huimin, Zhang , et al. Adaptive Mapping for Data Orchestration in Reconfigurable CNN Processors via Deep Reinforcement Learning[J]. Journal of Computer-Aided Design & Computer Graphics, 2025, 37(7): 1157-1169.
- [24] 张骁, 周清雷, 李斌. 基于 HRCA 的可重构 SM4 密码算法研究与实现[J]. 网络与信息安全学报, 2020, 6(5): 101-109.
- Zhang X, Zhou Q L, Li B. Research and implementation of reconfigurable SM4 cryptographic algorithm based on HRCA[J]. Chinese Journal of Network and Information Security, 2020, 6(5): 101-109.
- [25] 朱威, 王友祥, 杨文聪, 等. 面向算网融合的数据处理单元(DPU)技术探索与实践[J]. 邮电设计技术, 2025 (10): 76-81.

- Zhu W, Wang Y X, Yang W C, et al. Exploration and practice of data processing Unit (DPU) technology for computing and network convergence [J]. Posts & Telecommunications Design Technology, 2025 (10):76-81.
- [26] 李兆石. 高灵活可重构处理器的编程模型和硬件架构关键技术研究[D]. 北京: 清华大学, 2018.
- Li Z S. Research on programming model and key hardware architecture technologies of highly flexible reconfigurable processors[D]. Beijing: Tsinghua University, 2018.
- [27] 张丽果, 吴凯, 王文哲, 等. 高性能可重构网络协议解析器的设计与实现[J]. 西安邮电大学学报, 2025, 30(3): 58-67..
- Zhang L G, Wu K, Wang W Z, et al. Design and implementation of high-performance reconfigurable network protocol parser[J]. Journal of Xi'an University of Posts and Telecommunications, 2025, 30(3): 58-67.
- [28] 浙江大学. 一种自动支持循环迭代流水线的粗粒度可重构阵列及其配置方法: 201710712367. X[P]. 2018-06-19.
- Zhejiang University. A coarse-grained reconfigurable array with automatic support for cyclic iterative pipelines and its configuration method:201710712367.X[P].2018-06-19.
- [29] 曹壮. 基于可重构的网络报文处理关键技术及快速生成方法研究[D]. 长沙: 国防科技大学, 2020.
- Cao Z. Research on key technologies and rapid generation method of reconfigurable network packet processing[D]. Changsha: National University of Defense Technology, 2020.
- [30] 张杰. 面向现代数据中心的智能网卡软硬件协同技术研究[D]. 杭州: 浙江大学, 2025.
- Zhang J. Research on software-hardware coordination technology of smart NIC for modern data centers[D]. Hangzhou: Zhejiang University, 2025.
- [31] 刘玉. 可重构网络包处理转发引擎的设计与实现[D]. 西安: 西安电子科技大学, 2024.
- Liu Y. Design and implementation of reconfigurable network packet processing forwarding engine[D]. Xi'an: Xidian University, 2024.
- [32] 朱博. 基于可编程网络的多租户虚拟资源管理与优化技术研究[D]. 武汉: 武汉大学, 2019.
- Zhu B. Research on multi-tenant virtual resource management and optimization technology based on programmable network[D]. Wuhan: Wuhan University, 2019.
- [33] 赵佳欣, 王竞舟, 孙玉娥, 等. 可重配置数据中心的高效重配置与流量调度算法[J]. 计算机研究与发展, 2026: 1-16.
- Zhao J X, Wang J Z, Sun Y E, et al. Efficient reconfiguration and traffic scheduling algorithm for reconfigurable data centers[J]. Journal of Computer Research and Development, 2026: 1-16.
- [34] 陈杰, 王俊昌, 胡嘉晨, 等. 面向多样化网络需求的优先级队列流量调度算法[J]. 计算机工程, 2025, 51 (11): 152-159.
- Chen J, Wang J C, Hu J C, et al. Priority queue traffic scheduling algorithm for diverse network requirements[J]. Computer Engineering, 2025, 51(11): 152-159.
- [35] 李宏慧, 李炜东, 付学良. 基于多层虚拟拓扑节能的SDN数据中心网络流量调度算法[J]. 计算机应用与软件, 2021, 38 (4): 116-122.
- Li H H, Li W D, Fu X L. Energy-efficient traffic scheduling algorithm for SDN data center networks based on multi-layer virtual topology[J]. Computer Applications and Software, 2021, 38(4): 116-122.

[作者简介]



李萌 (1980-), 男, 清华大学博士生, 中电长城科技有限公司高级工程师, 主要研究方向为数据科学与先进计算。



解超 (1986-), 男, 硕士, 中电长城科技有限公司高级工程师, 主要研究方向为数据科学与先进计算。



牛兵 (1980-) 男, 山东融汇物产集团有限公司工程技术人员, 主要研究方向为物联网、天地一体化信息网络、卫星通信、网络安全等新技术的跟踪及其在矿山方面的智能化应用。