



计算机科学

COMPUTER SCIENCE

众核处理器研究技术综述和分析

宋立国, 胡承秀, 王亮

引用本文

宋立国, 胡承秀, 王亮. 众核处理器研究技术综述和分析[J]. 计算机科学, 2022, 49(11A): 211000012-7.

SONG Li-guo, HU Cheng-xiu, WANG Liang. [Summary and Analysis of Research on ManyCore Processor Technologies](#) [J]. Computer Science, 2022, 49(11A): 211000012-7.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[基于混合混沌大爆炸算法的三维片上网络低功耗映射](#)

Low-power Mapping Method for Three-dimensional Network on Chip Based on Hybrid Chaotic Big Bang-big Crunch

计算机科学, 2019, 46(8): 100-105. <https://doi.org/10.11896/j.issn.1002-137X.2019.08.016>

[USN安全研究与设计](#)

计算机科学, 2003, 30(9): 107-110.

[多级存储结构视频点播系统中的接纳控制算法研究](#)

计算机科学, 2002, 29(6): 70-72.

[多节点分级网络RAID存储结构研究](#)

计算机科学, 2002, 29(11): 22-23.

[基于多级存储结构的海量信息管理系统的研究](#)

计算机科学, 2000, 27(9): 8-11.

众核处理器研究技术综述和分析

宋立国¹ 胡承秀² 王亮¹

1 北京微电子技术研究所 北京 100076

2 北京宇航系统工程研究所 北京 100076

摘要 处理器正在由单核处理器向众核处理器发展,文章首先介绍了目前众核处理器的发展状况;然后重点从能效、性能和可靠性 3 个方面,分体系结构、片上存储和软件等不同层次综合分析国外众核处理器最新研究成果;结合后摩尔时代集成电路发展趋势,指出自适应技术和三维集成技术将是众核处理器发展的重点。文章最后认为,众核处理器未来发展将是不同拓扑结构、软件编程与硬件定义、经典设计与新器件、新工艺的创新融合。

关键词: 众核处理器;片上网络;存储结构;软件调度

中图分类号 TP368

Summary and Analysis of Research on ManyCore Processor Technologies

SONG Li-guo¹, HU Cheng-xiu² and WANG Liang¹

1 Beijing Microelectronics Technology Institute, Beijing 100076, China

2 Beijing Institute of Aerospace Systems Engineering, Beijing 100076, China

Abstract Processors have been developing from single-core to manycore. The latest research results abroad on manycore are comprehensively analyzed. The development status of many-core processors is first introduced, and then the related recent papers are summarized and retrieved from three aspects: architecture, on-chip storage and software. The main contributions and basic ideas of these papers are analyzed from the perspectives of energy efficiency, performance and reliability. Finally, combined with the development trend of integrated circuits in the post Moore era, two main technical direction are expounded which are the emerging adaptive architecture technology and three-dimensional integration technology of manycore processors.

Keywords Manycore processor, Network-on-chip, Memory-on-chip, Software scheduling

1 引言

众核处理器芯片上集成了数十甚至更多的处理器核,在保持单个处理器核工作频率基本不变的情况下,处理器的理论计算性能随核数的增加而提高,既能够适应应用需求对高性能处理器的需求,又能够适应后摩尔时代延续摩尔和超越摩尔的技术发展趋势。在众核处理器发展的同时,也面临着一些挑战:1)高能效设计,无论在高性能计算系统,还是在移动设备、嵌入式系统,其能耗问题都受到越来越多的关注,高能效设计也成为这些系统的首要设计需求之一;2)高性能是衡量众核处理器优劣的关键指标,但随着处理器核数量的增加,存储墙的问题也愈加严重,计算、存储、I/O 的速度越来越不匹配,众核处理器性能的提升也越来越困难,这一直是研究的重点;3)众核处理器制造工艺尺寸缩小至纳米水平,芯片的故障率急剧提高,面临的可靠性威胁日益严峻,如果按照可靠性最坏情况进行保守设计,虽然能够满足可靠性的要求但是开销极大,会部分抵消众核处理器性能优势,因此,研究基于体系结构和运行方式的可靠性设计成为热点。针对众核处理器面临的高能效、高性能和可靠性需求,本文重点从能效、性能和可靠性 3 个方面,分体系结构、片上存储和软件等不同层次综合分析国外众核处理器最新研究成果。

2 众核处理器发展

根据计算核心的结构复杂度和组织方式,可将众核处理器分为基于通用处理器核和基于计算簇的众核处理器两大类。

(1)基于通用处理核的众核处理器可以看作是多核结构处理器的进一步延伸。此时集成的处理器核一般由通用处理器简化而来,所有核心功能齐全,并保留了通用处理器中的多级缓存(Cache)存储结构。典型代表产品包括: Intel 公司的 Ice Lake-SP 架构的至强系列处理器、AMD 公司的 Zen 架构的锐龙系列处理器、Kalray 公司的 MPPA 系列处理器、富士通公司的 A64FX 处理器、我国飞腾公司基于自研 FTC662 和 FTC663 内核的 FT-2000+/64 和 S2500,产品综合指标对比如表 1 所列。

表 1 基于通用处理器核的众核处理器对比

Table 1 Comparison of manycore based on general processor core

处理器型号	工艺/nm	核数/GHz	性能	功耗/W	发布时间
至强 8360	10	36	2.4	250	2021 年
锐龙 3990X	7	64	2.9	280	2020 年
MPPA3	16	80	1.2/4 TFLOPS	20	2019 年
A64FX	7	52	2.2/3.3 TFLOPS	—	2019 年
S2500	16	64	2.2	150	2020 年

(2)基于计算簇的众核处理器,旨在通过简单运算单元的聚合提供超高计算性能。多个运算单元以组或簇的形式进行组织,计算簇内所有计算核心共用指令发射单元,共享一级 Cache 等存储资源;计算簇间则共享二级 Cache 和主存等。典型代表产品主要包括:NVIDIA 公司的 Ampere、Turing 架构系列 GPGPU,AMD 公司的 RDNA2 架构 GPU,日本 PEZY 公司的 PEZY-SC 系列处理器,我国的申威 26010 处理器,产品综合指标对比如表 2 所列。

表 2 基于计算簇的众核处理器对比

Table 2 Comparison of manycore based on calculation cluster

处理器型号	工艺/nm	核数/GHz	性能	功耗/W	发布时间
RTX3090	7	10496	285TFLOPS (FP16)	350	2021 年
GA100	7	6912	624TOPS (INT8)	400	2021 年
6800	7	3840	35.6TFLOPS (FP16)	250	2021 年
PEZY-SC3	7	8192	43.6TFLOPS	400	2019 年
SW26010	28	260	3TFLOPS (FP32)	—	2016 年

从表 1、表 2 中可以看出,随着工艺的进步,芯片中集成处理器核数量明显有不断增加的趋势,核数已经成为衡量处理器性能的关键指标,众核架构是处理器发展的必然。

3 众核处理器重点优化方向分析

以图灵机理论为基础的冯·诺依曼体系结构是串行执行模型,而众核处理器则属于分布式并行结构,如何在解决二者不匹配问题的同时,降低处理器性能墙、功耗墙、存储墙、应用墙的制约,是众核处理器研究的关键,本节从能效、性能和可靠性 3 方面对近期众核相关研究进行综合分析。

3.1 众核处理器能效优化技术

众核处理器面临严峻的功耗墙挑战,随着 Dennard 缩放比例定律的失效,虽然晶体管的密度仍然会随工艺的进步不断提升,但是每代晶体管能量优化的速率在快速降低。因此,众核处理器必须依据系统目标和关键应用需求,针对功耗问题,从多个层次上进行优化设计。

3.1.1 体系结构级能效优化

众核处理器体系结构主要包括处理器核、片上网络(NoC)两个主要部分,因此,众核处理器体系结构级能效优化主要围绕这两部分展开。

(1)处理器核能效优化

处理器的高能效技术按照作用的层次可分为系统级、结构级、电路级和工艺级。系统级主要通过软硬协同的方式,根据负载情况进行能耗管理,实现运行、休眠等不同运行等级状态的切换;结构级通过选择面向能效优化的算法和编码,在保证一定性能的前提下,控制芯片的峰值功耗和运行功耗;电路级主要是针对确定功能的部件,选择能效最优的电路实现;工艺级需要密切结合工艺情况,采用合适的晶体管和逻辑器件,优化后端设计流程,以降低功耗。动态电压频率调节技术(DVFS)是能耗管理中常用的一种技术,它根据程序特征的实时变化,自适应地调节处理器核的电压和频率,在功耗和性能之间取得折中。对处理器核能效优化,通常采用两种策略:1)将功耗控制和功耗分配解耦合,使两者独立地根据各自的目标进行优化,降低复杂度;2)建立频率和功耗的关系模型,

通过反馈方式指导 DVFS 调节,并通过在线调整模型参数,提高众核处理器面对不同特征应用时的能效适应能力。文献[1-4]应用 DVFS 技术来降低众核处理器功耗。

(2)片上网络能效优化

片上网络的功耗在处理器总功耗中具有很高的比重,如麻省理工学院的 RAW 处理器(16 核)^[5]片上网络功耗为 7.2 W,占整个处理器功耗的 36%;Intel 公司的“万亿级芯片”(80 核)片上网络功耗占总体功耗的 40%^[6]。片上网络能效优化主要包括拓扑结构能效优化和路由单元能效优化。

片上网络的拓扑结构定义了网络内节点与链路的布局和互连方式,对网络的功耗、时延、面积等有至关重要的影响。Wang 等利用拓扑能耗模型^[7],以二维网格环绕、高维网格环绕等为例,探讨了不同制造工艺对片上网络拓扑能耗的影响,指出不同的制造工艺对应的能耗最优的网络拓扑也不同。Pinto 等研究了分簇对二维网格(Mesh)、二维环绕、胖树、蝶网等网络拓扑的影响^[8],指出通过流量局部化技术,可减少 20%~40%的能耗。Souza 等对具有分布式和共享缓存机制的不同片上网络拓扑结构开展设计空间探索和优化,能够降低 38%的功耗^[9]。

路由单元是片上网络的重要组成部分,目前对路由单元功耗的研究涵盖了系统级和电路级两个层次:系统级主要是基于历史信息的动态调整电压/频率;电路级主要是缩减单元内部缓冲容量和修改交换开关结构。Doppa 等针对路由单元中的虚拟通道和链路开展细粒度控制^[10],Lee 等在路由单元中增加多功能自适应通道^[2],根据数据流量动态分配链路和路由单元缓冲空间。Bokhari 等提出 SUPERNET 结构^[11],采用双电压/频率设计两套路由链路,分别工作于不同的电压和频率,任务运行的过程中,能够根据应用需求,选择不同的路由链路传递信息。Scionti 等提出 SDNoC 结构^[12],路由单元设计有正常、旁路和电源门控 3 种工作模式,通过动态配置,融合二维网络结构和环形网络结构,有效降低单元功耗。

3.1.2 片上存储能效优化

片上缓存利用程序、数据的空间局部性和时间局部性,缓解主存与 CPU 处理速度不匹配的问题,其因容量大、速度快、访问频繁,成为处理器功耗的主要来源,约占总功耗的 30%~60%^[13-14]。在众核处理器中,L1 Cache 与处理器核紧密耦合,组织方式单一,设计的重点是高速;而 L2 及更高层次 Cache 则多为大容量 Cache,组织方式多样^[15],因此 Cache 资源管理的研究多集中在 L2 及更高层次的 Cache 上。Titosgil 等基于路(Way)组合思想,提出缓存一致性目录结构^[16],为每个缓存条目设计了一个指针;对于那些需要多个指针的地址,可以从同一组(Set)中其他空路处获得额外的指针,在最大限度减少存储开销的同时,又不丧失适应多个共享度的灵活性,提高了缓存能效。

相较于 Cache 优化,集成软件控制的 SPM(便签存储器)是一个更为理想的选择。这是因为:1)从面积角度看,同 Cache 相比,SPM 由于缺少了用来存储地址的 TAG 存储器和比较地址的 TAG 比较器,硬件实现更加简单,在相同的制造工艺下,SPM 所占面积更小,约为 Cache 的 66%^[17];2)从能耗角度看,SPM 访问能耗更低,相同工艺条件下约为 Cache 的 60%^[18];3)从指令执行角度看,使用 Cache 无法预测实际最差工作情况,而 SPM 由于程序员可见,并受其直接控制,因此行为更加确定。正是由于 SPM 在面积、能耗、实时性方面

的优势, Alovarez 等针对众核共享存储结构, 采用运行时库 (Runtime Library)^[19], 允许编译器自动产生管理 SPM 的代码, 使得存储器访问操作能够自动转到有效数据地址空间中。Kim 等设计了 SRAM 与 NVM(非易失性新型存储器件) 存储混合的末级缓存结构^[20], 在缓存之间、片上存储之间进行优化设计, 达到写访问的均衡性, 填补片上缓存与片外存储访问之间不断扩大的鸿沟。

3.1.3 软件任务调度能效优化

并行任务调度算法的好坏和处理器的能效密切相关。Wang 等提出一个可扩展的任务调度框架(WAANSO)^[21], 基于小波聚类方法, 根据运行给定应用程序的内核数量, 将任务集自动映射到处理器核上, 与蚁群优化算法、粒子群优化算法相比, WAANSO 能够提高的能效在 19%~65% 之间。Capotodni 等对众核处理器上的计算簇进行空间划分^[22], 让它们分别负责运行 OpenCL 和 OpenMP 编写的核心程序, 通过获取必要的硬件信息, 建立运行时调度系统, 实现对众核上运行程序的能效优化。Le 等基于流最小代价启发式算法^[23], 通过任务映射和优化, 最小化不同任务间信息传递的路由单元数量。Huang 等提出以共享资源为中心的调度策略^[24], 计算任务首先向共享资源请求数据, 直到请求操作挂起后, 映射任务的处理器核才执行计算, 以此提高任务执行的能效。

3.2 众核处理器性能优化技术

众核处理器核心数量的增长、性能的提高, 对片上存储层次、片上互连和一致性协议的扩展性都提出了更高的要求。目前的研究集中在异构集成、片上网络拓扑、片上缓存和软件编译与调度几个方面。

3.2.1 体系结构级性能优化

众核处理器体系结构级性能优化主要是从异构集成、资源动态组合和片上网络方面开展优化设计。

(1) 异构集成与资源动态组合性能优化

异构集成是将不同类型处理器核心集成在一个芯片内, 分别处理程序中具有不同特征的代码段, 包括集成少量强大的管理核心以及集成众多面向计算开发的精简运算核心, 高效处理线程级并行, 大幅提高芯片性能。Davidson 等在芯片结构中集成 3 种内核^[25], 分别是 5 个支持操作系统的高性能 RISC-V 内核、496 个支持大规模并行运算的精简 RISC-V 内核, 以及支持人工智能算法的神经网络加速引擎。其性能相比 NVIDIA 公司的 Jetson 系列嵌入式 GPU 提高了 28 倍。

利用众核内部丰富的处理器核资源, 针对特定应用, 构成高性能虚拟核或者虚拟加速单元, 实现资源动态组合。Capotodni 等提出多编程模型运行时系统 MPM-TRS^[22], 对处理器核进行组合, 分别负责运行 OpenCL 和 OpenMP 编写的虚拟加速事件, 达到加速运行性能的目的。Wang 等建立基于 PyTorch 学习库的密集和稀疏张量处理框架^[21], 框架由 3 部分构成: 1) 局部 DAE 机制(访问/执行解耦), 芯片内处理器核划分为访问内核和执行内核, 在访问内核与执行内核间通过 SPM 建立软件队列; 2) 脉动 DAE 机制, 利用数据重用, 允许多个执行内核共享一个访问内核; 3) 集成硬件访问加速模块, 提高数据吞吐量。访问/执行的解耦和硬件访问加速机制, 使得处理器内部访问与执行达到了并行化、流水化, 从而提高处理器性能。

(2) 网络拓朴性能优化

片上网络关系着同步、通信及访存等开销, 选择和设计合适

的片上网络拓朴结构, 是片上网络研究中的关键。Ddehwal 等基于数据驱动模型和工程师的经验知识, 提出一种用于片上网络拓朴结构的多目标优化搜索框架^[26], 该框架是重复迭代的两阶段优化算法, 在每次迭代中, 首先基于数据驱动树模型, 自动选择目标参数; 然后, 从选定的起始解开始执行局部搜索, 并利用得到的帕累托最优组来更新数据驱动树模型, 从仿真结果上看, 片上网络性能可以提高 20%。

Mesh 网络拓朴简单、寻径方便、可扩展性好, 因为成为最常用的片上网络互连结构, Abbas 等详细分析了二维网格及其扩展结构的片上互连性能^[27], 分析结果表明, Mesh 网格中所有结点在某一个方向(水平或垂直)上实际是一个线性阵列, 因此在较大规模网络连接中网络直径较大, 传输延时也较大。改善片上网络的平均延时, 能够在很大程度上提高系统的性能。Ogras 等在 Mesh 网络中增加专用全局连线^[28], 以降低某些长距离全局通信的延时。

在路由节点添加流量控制器, 建立传输反馈机制, 从而降低网络拥塞, 减小网络延时。Benoî 等基于确定性网络演算技术, 为保证 NoC 服务质量, 从全局的角度, 针对路径划分问题, 将网络流量和突发度参数解耦, 借助网络演算中的到达曲线和节点路由器的服务曲线, 获得更好的网络流量和端到端传递延时边界^[30]。

(3) 路由单元性能优化

路由单元由数据通路和控制通路两部分组成, 降低数据通路延时, 并且减少控制通路给数据流带来的停顿, 是提高路由单元性能的关键因素。研究表明, 当路由单元的流水线级数从 1 级提高到 5 级时, 众核处理器的整体性能将下降约 10%^[31]。

Bokhari 等提出双层路由链路^[11]来分别传递读、写数据包, 降低数据传递过程中的冲突, 提高传递效率。Doppa 等在路由单元中增加 SMART 电路^[10], 允许跨路由、单周期多跳传递。Li 等通过 3 种策略改进路由单元结构^[32]: 1) 在路由单元中增加一条新的传递路径, 使得处理器核与路由单元间存在两条传递路径; 2) 在路由单元中增加智能流量控制模块, 降低全局通信竞争概率; 3) 改进路由单元虚拟通道状态表、路由计算模式、虚拟通道分配算法、交叉条结构和分配算法, 以降低局部通信竞争概率; 改进后的路由单元的片上网络吞吐量会提高 51%, 传递延时降低 38%。

3.2.2 片上缓存性能优化

众核处理器由于不均匀的数据访问延时和同一数据在多个处理器核上的不同拷贝, 会导致严重的存储一致性问题。侦听与目录是存储一致性协议的两种重要的实现方式。由于侦听协议的可扩展性差, 具有更高可扩展性的目录协议得到了广泛的使用, 并且日趋成熟。

在此情况下, 提高片上存储访问性能成为研究热点。Han 等提出了一种可重构缓存系统^[33], 该系统能够根据需要将缓存行(Cache Line)配置为消息传递缓存区, 提高片上存储的利用率。Masing 等开展基于区域可配置的缓存一致性研究^[34], 保证了区域内缓存的强一致性, 区域间缓存的弱一致性, 简化并加速核间一致性消息传递, 从而提高整体性能。Burgio 等采用可预测执行模型(PRAM)^[35], 基于内存感知方法, 将任务分为存储和计算两个阶段, 在存储阶段, 显式控制从内存中检索数据, 并将其复制到内核的本地缓存中, 通过这种机制, 降低内存争用而导致的延时可变性。Chen 等为了

发挥出 3D 堆叠 DRAM 的带宽,提出了历史辅助自适应粒度缓存机制^[36],该机制能够根据存储访问的历史地址信息,对缓存访问进行前期预测,并对缓存粒度提供弹性映射,提升缓存的命中率,有效降低存储访问延时。Nguyen 等提出 MORC 缓存压缩结构^[37],采用行间压缩技术,利用日志的方式将压缩后的数据组织在一起,在结构中增加行映射表,实现缓存到日志的灵活映射。

3.2.3 软件编译与任务调度性能优化

固定的缓存策略难以适应程序中数据访存模式的多样性,容易造成缓存抖动,从而影响性能。为此,众核处理器动态地为程序中的数据对象分配 Cache 段,并且动态改变段容量、段内相联度、块大小,从而适应访存模式的多样性。为避免增加程序员的负担,众核缓存的软件管理工作主要由编译器来承担。Tang 等提出一种编译器缓存优化框架^[38],采用代码重组和计算调度方法,在考虑缓存、内存中 BANK 数量的基础上,最大化缓存并行性(Cache Level Parallel, CLP)和内存并行性(Memory Level Parallel, MLP),自动优化 MLP 和 CLP 间的平衡,降低末级缓存未命中带来的延时,获得最佳的应用性能。Kisla 等针对众核处理器中非均匀内存访问架构提出的编译策略^[39-40],考虑了众核处理器内部内核、末级缓存、存储控制器相对位置对访存延迟的影响,实现计算任务在处理器内核上的优化映射。

应用程序划分以及任务映射对网络流量起着决定性作用,将多个计算任务映射到处理器核上进行并行计算的过程,是一个 NP 完全问题。相较于事先确定的负载静态调度,运行时测量负载并划分的动态调度,能够获得 30%~40% 的性能提高^[41]。因此,为众核处理器提供运行时支持,能够实现应用程序的自动优化和软件到硬件的动态映射,达到降低众核系统编程难度、提高性能的目的。Miomandre 等提出运行时管理器 SPIDER^[42],其在计算簇上动态部署可重构数据流图,支持在非统一内存访问体系结构中执行可重构数据流图,运行时系统通过对计算内核和分配任务的调度和映射,达到提高性能的目的。Zhang 等提出快速检测数据流处理瓶颈的方法^[43],通过细粒度运行时管理,快速响应程序工作负载和资源(分配的内核数量)的变化,自适应扩展和收缩参与流处理的内核数量,提高流处理的吞吐量。

3.3 众核处理器可靠性优化技术

众核处理器可靠性的问题非常突出,包括软错误和芯片老化两方面。

3.3.1 软错误容错优化技术

随着半导体工艺特征尺寸缩小、工作电压降低,电路翻转所需要的临界电荷也在不断降低,Borkar 等和 Sivakumar 等分别指出每代工艺的进步会使数据位的软错误率(SER)增加 8%,导致众核处理器的错误率急剧增大^[44-45]。根据软错误发生的位置,可分为处理器核错误和路由单元错误。研究有效的众核处理器容错技术,可应对软错误常态化挑战。

在体系结构方面,可通过添加超过系统性能需求的资源,提高系统的复杂度来提高系统的容错能力。在这样的冗余设计上,一个部分出现了错误,不会影响整个系统的工作。有的系统具有重构功能,错误的冗余部分可以得到恢复从而恢复系统的容错能力,主要包括运算核重布局技术^[46]、自适应路由技术^[47]、冗余传输策略技术^[48]。Bokhari 等在路由单元数据路径中增加纠错模块^[11],能够对 64 位微片纠正 8 位错、

检测 16 位错;在控制路径中,通过双模锁步模式提高传输的可靠性。Doppa 等在路由单元中增加自我监测和配置电路^[10]来防止死锁现象的发生。Wang 等在路由单元中增加自适应错误纠正/检测模块和重新传输控制机制^[49],以提高可靠性。

在软件方面,Balboni 等针对故障的检测和规避,对受故障影响的区域进行界定,基于 OSRLite 运行时系统,动态重构路由功能^[50]:1)利用令牌快速建立、断开链路功能;2)建立不同网络隧道功能,将发生故障的路由所承担的功能,快速且可控地切换到新的路由上,依靠令牌和隧道传输,实现新旧路由间的信息交换。

3.3.2 老化效应防护技术

器件老化导致的可靠性问题主要包括:经时击穿(TD-DB)、热载流子注入(HCI)、负栅压温度不稳定性(NBTI)、电迁移(EM)。同时,功耗密度的增加也使温度成为加剧器件故障的重要因素。Haselman 等指出,在纳米工艺下,芯片中晶体管 and 连线发生故障的概率将超过 15%^[51]。工艺进步加剧了芯片老化。为了维持芯片的正常使用寿命,需要对电路进行抗老化设计,常用的方法有基于电路拓扑结构重构的老化防护技术^[52-53]、基于输入向量控制(IVC)和内部节点控制(INC)的集成电路老化防护技术^[54-56]以及基于动态调整技术的集成电路老化防护^[57-60]。

为降低电迁移对可靠性的影响,Kim 等建立等效电流模型,提出了一种系统级动态可靠性管理技术^[61],通过寻找最佳处理器核的电压和开/关状态,提高在接近阈值运行的低功耗众核微处理器的可靠性。Taeyoung 等将自适应强化学习方法和混合整数线性规划方法相结合^[4],对处理器核上的计算任务进行实时调整或者迁移,结合 DVFS 技术,达到延长众核处理器寿命的目的。

为降低 NBTI 对可靠性的影响,Rathore 等提出一种面向性能的约束感知任务映射技术^[62],此技术由两部分构成:芯片设计阶段,采用阈值接受模拟退火算法达到帕累托最优,以最大化延长平均故障时间;在芯片运行阶段,通过工作负载的分配,降低最脆弱的处理器核的工作负载。与性能贪婪和温度感知任务映射技术相比,此方法能够将芯片寿命提高 54%。

为降低功率密度增加导致的热力问题对众核可靠性的影响,Rathore 等为众核处理器设计了资源管理策略^[63]:采用分层划分方法,利用暗核技术进行热缓解;使用强化学习技术降低芯片的老化效应,此资源管理策略可以根据应用程序的负载数量和处理器核的数量进行扩展和动态改变。

4 众核处理器未来研究重点分析

当前,微电子发展处于技术变革的重要时期,延续摩尔定律和超越摩尔定律是其发展的两个重要方面,到 2025 年,传统 CMOS 微缩面临终结,新原理、新结构将登上舞台。众核处理器未来关注的重点是体系结构的自适应性和三维集成的高效性。

4.1 众核处理器自适应性技术

众核处理器的设计理念与传统的单核和多核处理器不同,它不再过度追求单个处理器核的计算能力,而是追求并行计算能力和处理器整体的计算能力。为适应变化的任务需求,众核处理器应具备软、硬件结构和计算模式的可变性以及

可编程性,研究众核处理器的自适应性成为必然。

现在的众核处理器很难高效适应不同的计算模式,为提高众核处理器适应性,Doppa等提出了一种众核自适应体系结构^[10],自适应性体现在应用(任务分配和调度)、处理器核(DVFS和电源门控)和互连网络(路由)多个方面。特别是在路由单元设计中,提出可配置中继器的概念,能够根据需要将多个短距离链路通过中继器连接起来,创建单周期长距离链路,允许单周期跨多个跃点传递数据。Scionti等开展可扩展的软件定义片上网络(SDNoC)体系结构研究^[12],在保持二维网络拓扑固定的基础上,融合环形拓扑和动态配置能力,允许映射不同类型的拓扑;路由单元为每条链路配置计数器,用于跟踪数据传递的统计值,以动态调整链路资源分配,允许软件层控制并监控片上网络。Zheng等提出一种自适应NoC架构^[64],能够为并发运行的程序动态分配多个不相交的子网,每个子网区域能够配置成最适合映射程序特征的拓扑结构,如网格、环形或树形,以提高性能。Nazarian等提出了一个自优化、自适应的众核系统框架^[65],由代表计算模型、连接模型、内存模型和存储模型的4层图模式组成,该框架通过实时调整这些模型,将硬件木马和侧信道等攻击的影响降到最低。

除了众核片上网络的自适应能力,自适应性还体现在处理器核上,在任务执行过程中可以实时选取相应数量的物理处理器核动态组成与任务需求匹配的虚拟处理器核,实现针对不同任务、不同需求的资源最优匹配。因此,众核处理器的处理过程是一种不断“动态异构”的过程,具有面对不同应用的适应性、组织性和自主性。Cao等通过对任务的拆分与合并,动态构建虚拟处理器核^[66],实现核资源的优化、隔离和访问。

4.2 众核处理器三维集成技术

受到电路性能改善的驱动,众核处理器向三维集成方向发展。三维结构能有效减小全局和半全局互连线长度,在提高芯片集成度的同时,使互连延时和互连功耗明显降低。尽管三维集成具有上述优点,但也存在一些问题,热管理就是最严重的问题之一。这是因为垂直堆叠的方式增加了芯片的功率密度,而且层间材料的热导率相对于硅片和金属来说很小,导致处理器内部产生的热量很难散失。为了使三维处理器不会因为温度过高而失效,必须控制其温度在允许的范围之内。Lee等和Musavvir等对单片三维众核处理器的性能和热平衡问题进行了研究^[2,67],他们建立热力学模型,提出了功耗-性能-热平衡的众核处理器架构设计原则,探索达到性能和热平衡的单片三维众核处理器架构。Chatterjee等基于系统计算和通信特征,考虑三维工艺的影响,提出了一种基于模拟学习算法的电源管理策略^[3],为各处理器核提供适当的电压/频率等级,以降低处理器核功耗,达到约束温度的目的。

结束语 综合来看,众核处理器的未来发展主流是融合创新。首先是多种拓扑网络的融合,不同拓扑结构决定了网络延时、带宽、吞吐率和系统功耗、芯片面积和容错能力上的不同,通过融合拓扑网络,能够优化出最适合应用的低直径拓扑、低功耗路由单元、容错路由算法及网络管理方式;其次是软件技术和硬件技术的创新融合,软件定义的范围和影响力将继续拓展,不仅可实现面向应用的处理器上资源的调度和管理,还能针对系统架构、存储架构的个性化需求,实现软硬解耦和资源灵活配置,达到与算法或框架深度融合的硬件动态专用定制;再次,是经典设计与新器件、新工艺的融合,为了

解决存储墙、I/O带宽墙和功耗墙的问题,同时由于非易失存储、3D堆叠等新技术的不断发展,众核处理器体系结构正发生着存储向计算靠近的融合结构变革。

参考文献

- [1] SOURAV D, JANARDHAN R D, PARTHA P P. Energy-Efficient and Reliable 3D Network-on-Chip (NoC): Architectures and Optimization Algorithms[C]// 2016 IEEE/ACM international conference on Computer-Aided Design (ICCAD). 2016:1-6.
- [2] LEE D J, DAS S, DOPPA J R. Performance and Thermal Tradeoffs for Energy-Efficient Monolithic 3D Network-on-Chip [J]. ACM Transactions on Design Automation of Electronic Systems, 2018, 23(5):1-25.
- [3] CHATTERJEE A, KIM R G, DOPPA J R. Power Management of Monolithic 3D Manycore Chips with Inter-tier Process Variations[J]. ACM Journal on Emerging Technologies in Computing Systems, 2021, 17(2):1-19.
- [4] TAEYOUNG K, SUN Z Y, CHEN H B. Energy and Lifetime Optimizations for Dark Silicon Manycore Microprocessor Considering Both Hard and Soft Errors[J]. IEEE Transactions on Very Large Scale Integration Systems, 2017, 25(9):2561-2574.
- [5] KIM J S, TAYLOR M B, MILLER J, et al. Energy characterization of a tiled architecture processor with on-chip networks [C]// Proceedings of the International Symposium on Low Power Electronics and Design. 2003:424-427.
- [6] Intel Corp. From a Few Cores to Many: A Terascale Computing Research Overview [EB/OL]. <https://www.intel.com/content/dam/www/public/us/en/documents/technology-briefs/intel-labs-tera-scale-research-paper.pdf>
- [7] WANG H, PEH L S, MALIK S. A Technology-Aware and Energy-Oriented Topology Exploration for On-Chip Networks [C]// Proceedings of Design, Automation and Test in Europe. 2005:1238-1243.
- [8] PINTO A, CARLONI L P, SANGIOVANNI VINCENTELLI A L. Efficient Synthesis of Networks on Chip [C]// Proceedings of the 21st International Conference on Computer Design. 2003:146-150.
- [9] SOUZA M A, FREITAS H C, MEHAUT J F. Design Space Exploration of Energy Efficient NoC and Cache-Based Many-Core Architecture Using Distributed L2 and Adaptive L3 Caches [C]// 2018 30th International Symposium on Computer Architecture and High Performance Computing. 2018:402-409.
- [10] DOPPA J R, KIM R G. Adaptive Manycore Architectures for Big Data Computing [C]// 2017 Eleventh IEEE/ACM International Symposium on Networks On-Chip. 2017:1-8.
- [11] BOKHARI H, JAVAID H, SHAFIQUE M. SuperNet: Multi-mode Interconnect Architecture for Manycore Chips [C]// Proceeding of the 52nd Annual Design Automation Conference. 2015:1-6.
- [12] SCIONTI A, MAZUMDAR S, PORTERO A. Software Defined Network-on-Chip for Scalable CMPs [C]// 2016 International Conference on High Performance Computing & Simulation (HPCS). 2016:112-115.
- [13] GONZALEZ R, HOROW I. Energy dissipation in general purpose microprocessors [J]. IEEE Journal of Solid State Circuits, 1996, 31(9):1277-1284.

- [14] ISHMANN V, IRWINMJ K, IRWIN M J, et al. Energy-driven integrated hardware-software optimizations using simple-power [C]//Proceedings of the 27th Annual International Symposium on Computer Architecture. 2000;95-106.
- [15] SUBRAMANIAN R, SMARAGDAKIS Y, LOH G H. Adaptive caches; effective shaping of cache behavior to workloads [C]// Proceedings of the 39th Annual IEEE/ACM Int Symp on Microarchitecture. 2006;385-396.
- [16] TITOSGIL R, FLORES A, FERNANDEZ-PASCUAL R. Way-Combining Directory: An Adaptive and Scalable Low-Cost Coherence Directory[C]//Proceedings of the International Conference on Supercomputing. 2017;1-10.
- [17] BANAKAR R, STEINKE S, LEE B, et al. Scratchpad Memory: A Designed Alternative for Cache On-chip memory in Embedded System[C]//Proceedings of the Tenth International Symposium on Hardware/Software Codesign(CODES 2002). 2002;73-78.
- [18] BANAKAR R, STEINKE S, LEE B, et al. Comparison of cache and scratchpad-based memory systems with respect to performance, area and energy consumption[R]. Fakultät für Informatik, TU Dortmund, 2001.
- [19] ALVAREZ L, VILANOVA L, MORETO M. Coherence Protocol for Transparent Management of Scratchpad Memories in Shared Memory Manycore Architectures[C]// Proceedings of the 42nd Annual International Symposium on Computing Architecture. 2015;720-732.
- [20] KIM N, AHN J, CHOI K. Benzene: An Energy-Efficient Distributed Hybrid Cache Architecture for Manycore Systems [J]. ACM Transactions on Architecture and Code Optimization, 2018, 15(1):1-23.
- [21] WANG X Q, XI J J, WANG Y H. An Efficient Task Mapping for Manycore Systems[C]// 2020 IEEE International Symposium on Circuits and Systems. 2020;1-4.
- [22] CAPOTONDI A, HAUGOU G, MARONGIU A, et al. Runtime Support for Multiple Offload-Based Programming Models on Embedded Manycore Accelerators[C]//Proceedings of the 2015 International Workshop on Code Optimisation for Multi and Manycores. 2015;1-10.
- [23] LE T T, ZHAO D, WU H Y, et al. Optimizing the Heterogeneous Network On-Chip Design in Manycore Architectures[C]// 2017 30th IEEE International System-on-Chip Conference. 2017;184-189.
- [24] HUANG W H, CHEN J J, REINEKE J. MIRROR: Symmetric timing analysis for real-time tasks on multicore platforms with shared resources[C]// Proceedings of the 53rd Annual Design Automation Conference. 2016;1-6.
- [25] DAVIDSON S, XIE S, TORNG C, et al. The Celerity Open-Source 511-Core RISC-V Tiered Accelerator Fabric: Fast Architectures and Design Methodologies for Fast Chips[J]. IEEE Micro, 2018(3/4):30-41.
- [26] DESHWAL A, JAYAKODI N K, et al. MOOS: A Multi-Objective Design Space Exploration and Optimization Framework for NoC Enabled Manycore Systems[J]. ACM Transaction on Embedded Computing Systems, 2019, 18(5):1-23.
- [27] ABBAS E K, AXEL J, LU Z H, et al. Mathematical formalisms for performance evaluation network on chip [J]. ACM computing Surveys, 2013, 45(3):1-41.
- [28] OGRAS U Y, MARCULESCU R. Application-specific network-on-chip architecture customization via long-range link insertion [C]//Proceedings of the 2005 International Conference on Computer Aided Design (ICCAD 05). Washington, DC: IEEE Computer Society, 2005;246-253.
- [29] WANG W, QIAO L, YANG G W, et al. Performance Analysis of the 2-D Networks-on-Chip[J]. Journal of Computer Research and Development, 2009, 46(10):1601-1611.
- [30] BENOÎT D, GRAILLAT A. Network-on-Chip Service Guarantees on the Kalray MPPA-256 Bostan Processor[C]// Proceedings of the 2nd International Workshop on Advanced Interconnect Solutions and Technologies for Emerging Computing Systems. 2017;35-40.
- [31] JERGER N E, PEH L S, LIPASTI M. Circuit-Switched Coherence[C]//Proceedings of the the 2nd International Symposium on Networks-on-Chip. 2008;193-202.
- [32] LI Y, LOU R A. ALPHA: A Learning-Enabled High-Performance Network-on-Chip Router Design for Heterogeneous Manycore Architectures[J]. IEEE Transactions on Sustainable Computing, 2021, 6(2):274-288.
- [33] HAN X, FU Y, JIANG J. Reconfigurable MPB Combined with Cache Coherence Protocol in Many-core[C]// 2016 IEEE Advanced Information Management, Electronic and Automation Control Conference. 2016;385-388.
- [34] MASING L, KRE F, SRIVATSA A, et al. In-NoC circuits for low-latency cache coherence in distributed shared-memory architectures[C]//2018 IEEE 12th International Symposium on Embedded Multicore/Many-core Systems-on-Chip. 2018;138-145.
- [35] BURGIO P, MARONGIU A, VALENTE P, et al. A memory-centric approach to enable timing-predictability within embedded many-core accelerators[C]//2015 CSI Symposium on Real-Time and Embedded Systems and Technologies. 2015;1-8.
- [36] CHEN K, LI S, AHN J H, et al. History-Assisted Adaptive-Granularity Caches (HAAG \$) for High Performance 3D DRAM Architectures[C]//Proceedings of the 29th ACM on International Conference on Supercomputing. 2015;251-261.
- [37] NGUYEN T M, WENTZLAFF D. MORC: A Manycore-Oriented Compressed Cache[C]// Proceedings of the 48th International Symposium on Microarchitecture. 2015;76-88.
- [38] TANG X L, KANDEMIR M T, KARAKOY M, et al. Co-optimizing Memory-Level Parallelism and Cache-Level Parallelism [C]// Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation. 2019;935-945.
- [39] KISLAL O, KOTRA J, TANG X L, et al. POSTER: Location-Aware Computation Mapping for Manycore Processors[C]// 2017 26th International Conference on Parallel Architectures and Compilation Techniques. 2017;138-139.
- [40] KANDEMIR M T, TANG X L, ZHAO H, et al. Distance-in-Time versus Distance-in-Space [C]// Proceedings of the 42nd ACM SIGPLAN Conference on Programming Language Design and Implementation. 2021;665-680.
- [41] KAMEDA H, FA E S, RYU I, et al. A performance comparison of dynamic vs. static load balancing policies in a mainframe-personal computer network model [C]// Proceedings of the 39th IEEE Conference on Decision and Control, Sydney, Australia. 2000;1415-1420.

- [42] MIOMANDRE H G, HASCOET J L, DESNOS K, et al. Embedded Runtime for Reconfigurable Dataflow Graphs on Manycore Architectures[C]// Proceedings of the 9th Workshop and 7th Workshop on Parallel Programming and RunTime Management Techniques for Manycore Architectures and Design Tools and Architectures for Multicore Embedded Computing Platforms. 2018;51-56.
- [43] ZHANG X, JAVAID H, SHAFIQUE M, et al. ADAPT: An ADaptive Manycore Methodology for Software Pipelined Applications[C]// The 20th Asic and South Pacific Design Automation Conference. 2015;701-706.
- [44] BORKAR S. Designing reliable systems from unreliable components: the challenges of transistor variability and degradation [J]. *Micro IEEE*, 2005, 25(6):10-16.
- [45] SIVAKUMAR P, KISTLER M, KECKLER S W, et al. Modeling the effect of technology trends on soft error rate of combinatorial logic [C]// Proceedings International Conference on Dependable Systems and Networks. 2002;389-398.
- [46] WONG R, LI J, FU A, et al. (α, k) -Anonymous data publishing [J]. *Journal of Intelligent Information Systems*, 2009, 33(2):209-234.
- [47] YANG G M, YANG J, ZHANG J P. Semi-supervised clustering-based anonymous data publishing[J]. *Journal of Harbin Engineering University*, 2011, 33(11):1489-1495.
- [48] TENG J F, ZHONG C. Clustering-based sensitive attribute-diversity anonymization algorithms[J]. *Computer Engineering and Design*, 2010, 31(20):4378-4381.
- [49] WANG K, LOURI A, KARANTH A, et al. IntelliNoC: A Holistic Design Framework for Energy-Efficient and Reliable On-Chip Communication for Manycores[C]// Proceedings of the 46th International Symposium on Computing Architecture. 2019:589-600.
- [50] BALBONI M, BERTOZZI D, FLICH J, et al. Synergistic Use of Multiple On-Chip Networks for Ultra-Low Latency and Scalable Distributed Routing Reconfiguration[C]// 2015 Design, Automation & Test in Europe Conference & Exhibition. 2015;806-811.
- [51] HASELMAN M, HAUCK S. The Future of Integrated Circuits; A Survey of Nanoelectronics [C]// Proceeding of the IEEE. 2010;11-38.
- [52] KAI-CHIANG W, MARCULESCU D. Joint logic restructuring and pin reordering against NBTI-induced performance degradation[C]// Proceedings Design, Automation & Test in Europe Conference & Exhibition(DATE'09). 2009;75-80.
- [53] BUTZEN P F, BEM V D, REIS A I, et al. Transistor network restructuring against NBTI degradation [J]. *Microelectronics Reliability*, 2010, 50(9/10/11):1298-1303.
- [54] ABELLA J, VERA X, GONZALEZ A. Penelope: The NBTI-Aware Processor [C]// 40th Annual IEEE/ACM International Symposium on Proc. Microarchitecture (MICRO 2007). 2007:85-96.
- [55] SONG J, YINHE H, LEI Z, et al. M-IVC: Using Multiple Input Vectors to Minimize Aging-Induced Delay [C]// Proceedings Asian Test Symposium(ATS'09). 2009;437-442.
- [56] WANG Y, LUO H, HE K, et al. Temperature-Aware NBTI Modeling and the Impact of Standby Leakage Reduction Techniques on Circuit Performance Degradation[J]. *IEEE Transactions on Dependable and Secure Computing*, 2011, 8(5):756-769.
- [57] MINTARNO E, SKAF J, RUI Z, et al. Self-Tuning for Maximized Lifetime Energy-Efficiency in the Presence of Circuit Aging[J]. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2011, 30(5):760-773.
- [58] LIDE Z, DICK R P. Scheduled voltage scaling for increasing lifetime in the presence of NBTI[C]// Proceedings Design Automation Conference, 2009. Asia and South Pacific, 2009;492-497.
- [59] BASOGLU M, ORSHANSKY M, EREZ M. NBTI-aware DVFS: A new approach to saving energy and increasing processor lifetime[C]// 2010 ACM/IEEE International Symposium on Proc. Low-Power Electronics and Design (ISLPED). 2010:253-258.
- [60] KUMAR S V, KIM C H, SAPATNEKAR S S. Adaptive Techniques for Overcoming Performance Degradation Due to Aging in CMOS Circuits[J]. *IEEE Transactions on Very Large Scale Integration(VLSI) Systems*, 2011, 19(4):603-614.
- [61] KIM T, SUN Z, COOK C, et al. Dynamic reliability management for near-threshold dark silicon processors[C]// Proceedings of the 35th International Conference on Computer-Aided Design. 2016:1-7.
- [62] RATHORE V, CHATURVEDI V, SRIKANTHAN T. Performance Constraint-Aware Task Mapping to Optimize Lifetime Reliability of Manycore Systems[C]// Proceedings of the 26th edition on Great Lakes Symposium on VLSI. 2016;377-380.
- [63] RATHORE V, CHATURVEDI V, SINGH A K, et al. Towards Scalable Lifetime Reliability Management for Dark Silicon Manycore Systems[C]// 2019 IEEE 25th International Symposium on On-Line Testing and Robust System Design. 2019:204-207.
- [64] ZHENG H, WANG K, LOURI A. Adapt-NoC: A Flexible Network-on-Chip Design for Heterogeneous Manycore Architectures[C]// 2021 IEEE International Symposium on High-Performance Computer Architecture. 2021;723-735.
- [65] NAZARIAN S, BOGDAN P. S4oC: A Self-Optimizing, Self-Adapting Secure System-on-Chip Design Framework to Tackle Unknown Threats — A Network Theoretic, Learning Approach [C]// 2020 IEEE International Symposium on Circuits and Systems. 2020;1-8.
- [66] CAO Y J, QIAN D P, WU W G, et al. Adaptive Scheduling Algorithm Based on Dynamic Core-Resource Partitions for Many-Core Processor Systems[J]. *Journal of Software*, 2012, 23(2):240-252.
- [67] MUSAVVIR S, CHATTERJEE A, KIM R G, et al. Power, Performance, and Thermal Trade-offs in M3D-enabled Manycore Chips[C]// 2020 Design Automation & Test in Europe Conference & Exhibition. 2020;1752-1757.



SONG Li-guo, born in 1973, Ph.D researcher. His main research interests include high-performance computing and many-core processor architecture.