

SoPC FPGA 云平台软硬件协同交互框架^①

赵 然^{②*} 常轶松^{**} 刘 波^{***} 刘超伟^{***} 陈明宇^{**} 张 科^{③*}

(* 中国科学院计算技术研究所 北京 100190)

(** 中国科学院大学 北京 100049)

(*** 北京控制工程研究所 北京 100190)

摘 要 针对现有商用现场可编程门阵列(FPGA)云服务(FaaS)平台部署密度低、成本高的问题,提出一种基于可编程片上系统(SoPC)FPGA 软硬件协同框架部署云平台的方法。该框架在满足一定性能需求的前提下,实现高密度、低成本、方便管理的 SoPC FPGA 云平台结构。该框架充分利用包含通用硬核处理器的 SoPC 软件编程性为大数据处理平台体系结构引入新维度的异构计算能力;同时通过在 SoPC 上进行软-硬件协同的系统级设计,为 FPGA 可编程硬件提供灵活、安全、可控的资源管理和配置环境,有效提升云平台加速节点的通用计算及数据管理能力;并结合 SoPC 和 FPGA 自身的低功耗特性,提升云平台加速节点的计算能效。通过基于该框架的原型平台上部署深度学习与存储阵列加速实例,验证了该云平台具备快速部署以及加速应用的能力,并具有高部署密度、低成本等特点。

关键词 现场可编程门阵列(FPGA)云服务(FaaS);深度学习;加速器;存储阵列

0 引 言

随着云计算、大数据、物联网等概念的兴起,在特定应用场景下(如深度神经网络处理等),基于图形处理器(graphics processing unit, GPU)、定制专用集成电路(application specific integrated circuit, ASIC)、现场可编程门阵列(field-programmable gate array, FPGA)等芯片的异构计算系统相比于单一的中央处理器(central processing unit, CPU)系统通常能够达到几十倍的性能提升^[1],异构计算已成为行业发展的重要趋势之一。然而对于普通用户来讲,开发与部署一个异构计算系统往往需要高成本和长周期的投入。因此,为了追求经济与效益的最优化,

由大型互联网公司搭建异构计算云平台并对普通用户提供云资源的服务应运而生^[2]。FPGA 云平台就是常见的异构计算云平台之一^[3,4]。

目前商业 FPGA 云平台大多由若干台 x86 服务器组成, x86 处理器与 FPGA 节点板卡通过 PCIe(peripheral component interconnect express)接口进行管理和数据交互^[5]。受限于接口形式, FPGA 节点板卡需部署在 x86 服务器机箱内,而由于单台 x86 服务器机箱尺寸及主板 PCIe 接口数量的限制,单台服务器内可部署的 FPGA 节点数量通常不会超过 8 个,如亚马逊的 FPGA EC2 实例 F1^[6]。因此在现有框架下, FPGA 节点很难进行更高密度的部署(即较低的纵向可扩展性);如果需要更大规模的横向可扩展性,则需要大量 x86 服务器组成并行集群结构,

① 国家重点研发计划(2016YFB1000401),中国科学院 C 类战略先导科技专项(XDC05030400),国家自然科学基金(61702485),中国科学院青年创新促进会(2017143),中国科学院计算所创新课题(20196010)和计算机体系结构国家重点实验室创新(CARCH4410)资助项目。

② 男,1990 年生,硕士,工程师;研究方向:FPGA 异构加速系统的设计与实现;E-mail: zhaoran@ict.ac.cn

③ 通信作者, E-mail: zhangke@ict.ac.cn
(收稿日期:2019-04-23)

这将导致更高的部署成本和占用更多的机架物理空间。

此外,普通用户使用 FPGA 云平台的目的通常仅使用 FPGA 可编程逻辑资源进行相关的逻辑设计和验证,而对 x86 服务器的运算能力并无过多需求。x86 服务器中的 CPU 处理器仅作为加速逻辑资源节点管理之用,存在较大性能浪费。

综上所述,为解决 FPGA 云平台存在的部署密度低、管理困难、成本高及计算能效低等问题,FPGA 加速节点需要具备独立管理、低功耗、低成本等特点。值得注意的是,近年来出现了面向边缘计算(edge computing, EC)的 FPGA 器件(如 Intel Stratix SoC、Xilinx Zynq MPSoC 等系列)。这一类型产品内部同时含有处理器硬核和 FPGA 可编程逻辑资源,是一种软硬件全可编程的可编程片上系统(system on programmable chip, SoPC)FPGA 芯片。如果使用该器件作为云平台的加速节点,每个节点将由各自的处理器单独管理并脱离 x86 平台的限制,这将使得云平台的部署更自由且更紧密。同时,免除 x86 服务器也使得云平台的成本与功耗更低,计算能效更高。因此本文提出采用基于 SoPC 的加速节点设计并研制高密度、低部署成本、低功耗、管理灵活的 SoPC FPGA 云平台,并结合深度神经网络加速处理应用针对此平台中的软硬件协同交互框架进行详细介绍。

1 相关工作

2016 年底,美国亚马逊公司第一次正式对外提出 FPGA 云平台商业概念及公测服务。在其 AWS 平台的 FPGA 云服务 F1 实例中,每个实例计算池上配置 8 个 FPGA,可应用于数据分析、视频、安全、机器学习等应用^[6]。国内互联网企业在 2017 年年初相继提供类似产品,如百度的 FPGA 应用主要是用于自身的机器学习,也包括语音识别、自动驾驶等应用^[7]。腾讯云为开发者提供 IP 商店腾讯云服务市场,IP 开发者和 IP 提供商可以通过它为客户提供 IP 和对应的测试程序^[8]。

这些商业 FPGA 云平台的基本架构均为 x86 通

过 PCIe 与 FPGA 进行交互,存在上述所提的部署密度低、管理困难、成本高、计算能效低等问题,因此本文拟基于 SoPC FPGA 构建下一代 FPGA 云平台。

2 SoPC FPGA 云平台

本文提出的基于 SoPC FPGA 构建的云平台,其内部核心为若干个含有处理器和 FPGA 逻辑资源的 SoPC 板卡节点,且节点之间依靠标准互连网络组成集群结构。该 SoPC FPGA 云平台在传统 FPGA 云平台具有的大容量可编程硬件逻辑基础上,为 FPGA 加速节点提供更为灵活的软件可编程特性。首先,充分利用包含通用硬核处理器的 SoPC 软件编程性为大数据处理平台体系结构引入新维度的异构计算能力;其次,通过在 SoPC 上进行软-硬件协同的系统级设计,为 FPGA 可编程硬件提供灵活、安全、可控的资源管理和配置环境,有效提升云平台加速节点的通用计算及数据管理能力;再次,结合 SoPC 和 FPGA 自身的低功耗特性,提升云平台加速节点的计算能效。

如图 1 所示,SoPC FPGA 云平台主要由若干 SoPC 加速节点和标准以太网交换机组成。在云平台与用户之间,由前端管理服务器为用户提供远程接入和管理等服务。用户通过前端管理服务器申请 SoPC FPGA 加速资源。前端服务器根据平台实时运行情况分配具体的 SoPC 加速节点给来自远程连接的用户,用户可以按需部署使用 FPGA 云加速服务和 FPGA 逻辑资源。

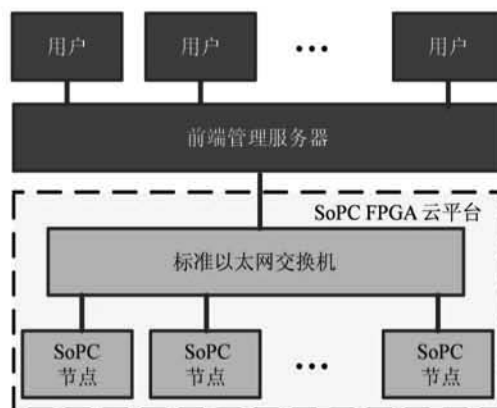


图 1 SoPC FPGA 云平台连接结构示意图

SoPC FPGA 云平台内部按照功能可分为管理配置框架、调试框架、用户服务框架以及软硬件协同交互框架等,本文重点介绍云平台软硬件协同交互框架。该框架是 SoPC FPGA 的核心框架,是处理器与 FPGA 加速器之间沟通的桥梁,负责处理器与 FPGA 之间所有控制流与数据流的交互。用户在操作系统下执行应用,并将计算部分交由 FPGA 进行加速。处理器与 FPGA 之间使用高带宽的高级微控制器总线结构(advanced microcontroller bus architecture, AMBA)总线互连^[9],通过该总线处理器可以快速访问 FPGA 逻辑,而 FPGA 逻辑也能快速访问处理器内存,实现软硬件协同交互。

如图 2 所示,该框架按照抽象层次分为 4 层,由高到低分别是应用框架层(支持 TensorFlow、Caffe、DPDK 等)、应用程序编程接口(application programming interface, API)层、内核空间的设备驱动层以及 FPGA 硬件设备层。



图 2 SoPC FPGA 平台软硬件协同交互框架结构

在 SoPC FPGA 平台软硬件协同交互框架中,基于标准 C 语言的应用程序编程接口层为不同应用提供统一的 API 接口,使该平台可以支持多种应用;内核设备驱动层的功能是抽象不同硬件算子的通用逻辑、隐藏不同厂商芯片及板卡的差异以及对 FPGA 硬件资源的通用管理;FPGA 硬件设备层则实现了一套通用的加速应用逻辑,该部分逻辑包含一些基本的数据传输功能,核心计算部件则需要用户自行生成添加。

2.1 硬件逻辑

本文将 FPGA 按逻辑功能划分为静态区域和动态区域,如图 3 所示。静态区域负责加速逻辑与对

处理器内存进行数据搬移,包括直接存储器访问(direct memory access, DMA)引擎及双倍数据速率(double data rate, DDR)内存控制器等基本组件。动态功能区包含各类运算单元,即硬件算子。通过与 FPGA 开发工具(如 Xilinx VIVADO)的深度融合,可支持动态区逻辑的部分可重构。

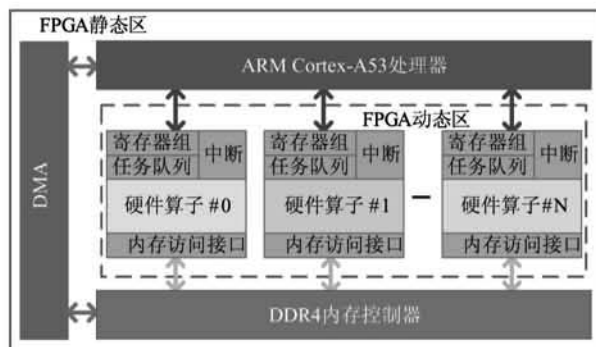


图 3 SoPC FPGA 平台软硬件协同交互框架硬件部分逻辑

为了使处理器端无差异地使用这些算子,本文提取了硬件算子的通用功能逻辑。首先,内存访问接口基于标准片内互连协议,负责读写 FPGA 卡上的 DDR 内存,实现不同算子间、静态区与动态区间的数据交互;其次,将硬件算子的控制/状态寄存器、中断信号等通过标准片内互连协议进行封装,并与静态区的 DMA 接口控制器互连,允许处理器端软件对硬件算子进行控制,如启动运算、查看状态等。此外,本文还提出并实现了基于任务队列的硬件算子工作机制,并在设备驱动层进行逻辑抽象,实现处理器端多线程或多进程应用程序并行使用 FPGA 计算资源。

2.2 软件接口

在软硬件协同交互框架中,软件接口用于为在 FPGA 云平台上做应用的开发者提供服务。其主要功能是对底层硬件逻辑进行抽象,并分为用户空间和内核空间两部分,如图 4 所示。

在用户空间,加速节点作为 udev 设备、DMA 通道设备及硬件算子设备被调用,开发者可以使用基本用户态 C 程序接口对 FPGA 逻辑资源进行初始化(FPGA_InitConfig())、调用 DMA 进行节点内数据传输(FPGA_CopyBufH2D())、FPGA_CopyBufD2H())、以及管理算子任务的分发和执行(FPGA_CommitTask())等,将计算部分移交给硬件加速器

进行处理。内核空间包含 FPGA 云服务 (FaaS) 中的 FPGA 驱动框架。与硬件逻辑框架对应, FPGA 驱动框架分为静态区平台驱动、动态区算子驱动和 DMA 系统驱动 3 个部分。其中, 在动态区算子驱动中包含有算子抽象数据结构的描述, 记录了算子的数量及各个算子的寄存器空间; 在静态区的平台驱动中包含着任务队列抽象数据结构的描述, 记录了任务编号及对应任务的寄存器读写操作; DMA 驱动负责给用户提供服务注册及中断处理等服务。

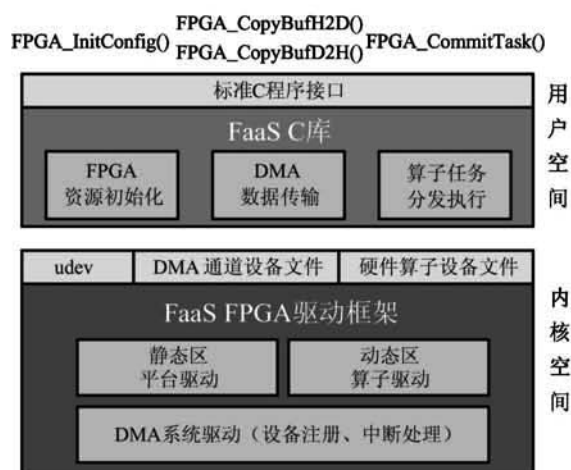


图4 SoPC FPGA 平台软硬件协同交互框架软件栈结构

用户空间和内核空间使用通用字符设备驱动程序接口进行转换, 用户程序通过该接口进入内核空间, 读取相关算子的配置文件通过驱动层完成底层硬件设备上硬件模块的执行, 进而完成整个计算过程。

3 平台应用部署实例

本文实现了一套基于 Xilinx Zynq UltraScale + MPSoC 芯片的 SoPC FPGA 云平台原型, 并实现神经网络加速器和 NVMe 存储 2 个应用实例来对框架性能进行评估。

3.1 环境部署

在此平台中, 每个异构加速平台节点板载 1.2 GHz 4 核 ARM Cortex-A53 处理器与 154K FPGA 逻辑资源 (Logic Cells)。处理器配有独立的 2133M DDR4 内存, 可运行完整的 Linux 内核版本操作系统, 文件系统为 Linaro/Debian。经过 Xilinx FPGA 开发工具 VIVADO 2017.2 的综合与实现, 硬件部分

静态区占用 FPGA 芯片整体的逻辑量 (LUTs) 不足 5%, 工作频率可达 266.667 MHz, 满足 FPGA 云平台的运行需求。

3.2 深度学习加速器实现

深度神经网络 (deep neural networks, DNN) 是一种计算密集型的学习模型。DNN 由卷积、池化、点积、归一化、激活等多层组成^[10], 并包含了大量的矩阵运算 (DNN 中最大计算量为卷积层中卷积核与数据的矩阵乘法), 因此适合使用 FPGA 进行加速计算^[11,12]。使用 SoPC FPGA 云平台又可进一步降低异构计算平台搭建所需的时间与经济成本, 从而实现快速部署。

本文实现了基于 SoPC FPGA 云平台的 DNN 加速器, 并集成至 TensorFlow 的运行环境中, 支持 AlexNet、VGG、LeNet、Cifar10 等多种模型处理。如图 5 所示, 在 FPGA 硬件框架中的动态区加载了用户编写的计算单元 (processing element, PE) 阵列, 实现了卷积运算中基本的乘累加操作。

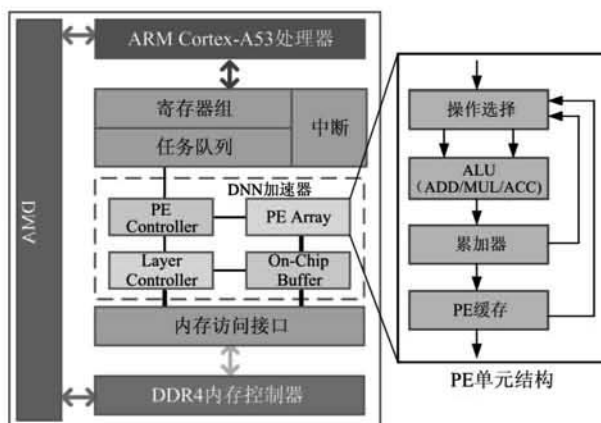


图5 部署 DNN 加速器逻辑的硬件结构

用户连接云平台后, 前端管理服务器会给用户分配 SoPC FPGA 云平台中的空闲加速节点资源。在用户使用完这些资源后, 由前端管理服务器释放, 再次作为云平台的空闲资源供下次使用。为了比对和验证 DNN 加速器计算结果的正确性, 在实验中首先单独使用 CPU 进行计算并保存每一层的计算结果作为参考。在进行 DNN 加速器实现时, 用户的应用程序运行在基于 ARM 处理器的操作系统中, 用户程序启动后的执行流程如图 6 所示。

首先由处理器配置 FPGA 逻辑①, 将待处理数

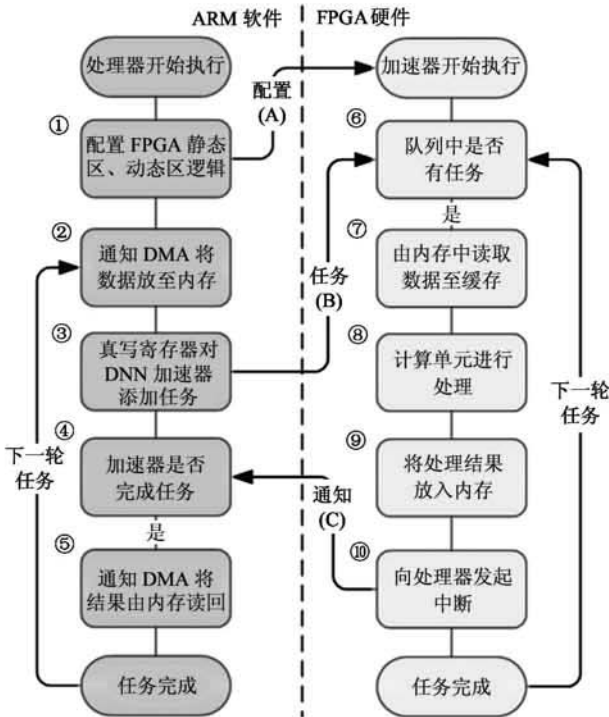


图 6 DNN 加速器实例执行流程及软硬件交互示意图

据存至内存中②,填写寄存器发起加速任务③,随后等待加速器完成④。收到加速器中断通知(C)后,调度 DMA 读回结果⑤,此次加速完成,可以根据情况选择是否执行下一轮加速任务②。FPGA 逻辑被配置好(A)之后,加速器便开始查询队列中的任务⑥,收到任务(B)后读取内存相应位置数据至缓存⑦,随后进行处理⑧。完成后写回内存⑨,并通知处理器⑩,随后继续查询队列中是否有下一轮任务⑥。

本文使用 LeNet 作为测试用例进行实验,在加速流程结束后,读取 DNN 加速器每一层的计算结果与 ARM 处理器的计算结果分别进行比较,发现数据完全一致,并分别统计总体运行时间结果如表 1 所示。由此可以证明 DNN 加速器计算正确,SoPC FPGA 异构加速方案可以顺利执行,且具有一定的加速能力。

表 1 LeNet 模型运行时间

目标平台	平均计算用时(μ s)
ARM Cortex-A53	128 109
FPGA DNN 加速器	3 791

3.3 NVMe-SSD 存储应用及软硬件交互性能测试

随着大数据时代的到来,传统存储服务器及相应的资源服务已逐渐出现性能瓶颈。如何充分释放如固态硬盘(solid state drive,SSD)等新型存储介质以及如 NVMe(non-volatile memory express)等新型存储协议和接口的性能优势,从而为云计算环境提供更低延迟、更高带宽的存储服务,受到广泛关注^[13]。

本文基于云平台软硬件协同框架构建了 NVMe 的数据中心存储应用。通过该应用可以评估框架在真实应用下进行大量数据交互时的通路性能。因此本文在该平台下对本地读写 NVMe 硬盘进行了带宽测试,并与 x86 平台进行对比。测试所使用的硬盘为 Samsung SSD 970 EVO 500 GB,操作系统为 Linux Kernel 5.0-rc4,测试工具为 Fio,进行 8 MB 块大小的顺序读写测试,测试结果如图 7 所示。

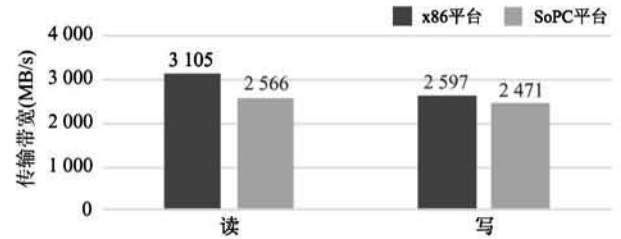


图 7 NVMe-SSD 本地存储性能测试

可以看出,在 x86 平台上读写数据性能接近固态硬盘标称带宽^[14],而在 SoPC 平台上性能稍有下降,结果表明该平台的瓶颈为处理器与 FPGA 之间的数据通路。该性能则代表该框架中软硬件交互的带宽,约为 2 500 MB/s。相较于 x86 平台,SoPC 平台的 ARM 处理器与 AMBA 总线处理性能略低,影响数据传输带宽,但该平台具有较高的成本效益。

搭建存储阵列可以充分利用 SoPC 的软件可编程能力对存储资源进行资源管理、调度和分配;同时该平台可以支持专用硬件逻辑对存储相关协议栈及处理算法进行加速,提升存储系统整体处理性能。针对上述目标,后续将进一步开展存储阵列的基础设施与加速服务工作。

4 结 论

本文分析了目前商用 FPGA 云平台存在的部署密度低、高成本、计算能效低等问题,提出在满足一

定性能需求的前提下,实现高密度、低成本、方便管理的 SoPC FPGA 云平台结构,并介绍了 SoPC FPGA 云平台中最为关键的软硬件协同交互框架。通过部署 DNN 加速器及 NVMe-SSD 存储阵列加速实例,描述了用户应用在该框架下的执行流程和软硬件之间数据交互流程,验证了该平台具备快速部署以及加速应用的能力。

下一步,本文将对 SoPC FPGA 软硬件协同交互框架和关键路径进行优化,提升数据带宽和处理速度,并开发支持更多功能以适配更多样、更复杂的加速应用;同时开展 FPGA 虚拟化等功能进一步提高资源使用率并降低部署成本。

参考文献

- [1] Zhang C, Li P, Sun G, et al. Optimizing FPGA-based accelerator design for deep convolutional neural networks [C] // Proceedings of the 2015 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, New York, USA, 2015: 161-170
- [2] Serrano N, Gallardo G, Hernantes J. Infrastructure as a service and cloud technologies [J]. *IEEE Software*, 2015, 32(2): 30-36
- [3] Chen F, Shan Y, Zhang Y, et al. Enabling FPGAs in the cloud [C] // Proceedings of the 11th ACM Conference on Computing Frontiers, Cagliari, Italy, 2014: 1-10
- [4] Yanovskaya O, Yanovsky M, Kharchenko V. The concept of green cloud infrastructure based on distributed computing and hardware accelerator within FPGA as a Service [C] // Proceedings of IEEE East-West Design & Test Symposium (EWDTS 2014), Kiev, Ukraine, 2014: 1-4
- [5] Caulfield A, Chung E, Putnam A, et al. A cloud-scale acceleration architecture [C] // The 49th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO), Taipei, China, 2017: 1-13
- [6] Amazon Web Services. Amazon EC2 F1 instances [EB/OL]. <https://aws.amazon.com/cn/ec2/instance-types/f1/>; Amazon, 2019
- [7] 百度云. FPGA 云服务器 [EB/OL]. <https://cloud.baidu.com/product/fpga.html>; 百度, 2019
- [8] 腾讯. FPGA 云服务器——腾讯云 [EB/OL]. <https://cloud.tencent.com/product/fpga>; 腾讯, 2019
- [9] Xilinx. Zynq UltraScale + MPSoC Data Sheet: Overview [EB/OL]. https://www.xilinx.com/support/documentation/data_sheets/ds891-zynq-ultrascale-plus-overview.pdf; Xilinx, 2019
- [10] Sharma H, Park J, Mahajan D, et al. From high-level deep neural models to FPGAs [C] // Proceedings of the 2016 IEEE/ACM International Symposium on Microarchitecture, Piscataway, USA, 2016: 1-12
- [11] Nurvitadhi E, Subhaschandra S, Boudoukh G, et al. Can FPGAs beat GPUs in accelerating next-generation deep neural networks? [C] // Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, New York, USA, 2017: 5-14
- [12] Zhao J, Chang Y, Li D, et al. On retargeting the AI programming framework to new hardware [C] // Proceedings of the 2018 IFIP International Conference on Network and Parallel Computing, Muroran, Japan, 2018: 39-51
- [13] Xu Q, Siyamwala H, Ghosh M, et al. Performance analysis of NVMe SSDs and their implication on real world databases [C] // Proceedings of the 8th ACM International Systems and Storage Conference, Haifa, Israel, 2015: 6
- [14] Samsung. 970 EVO NVMe M.2 固态硬盘 [EB/OL]. <https://www.samsung.com/cn/memory-storage/970-evo-nvme-m-2-ssd/MZ-V7E500BW/#specs>; Samsung, 2019

SoPC-enabled FPGA cloud platform and its hardware-software collaborative interaction framework

Zhao Ran^{***}, Chang Yisong^{***}, Liu Bo^{***}, Liu Chaowei^{***}, Chen Mingyu^{***}, Zhang Ke^{***}

(^{*} Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190)

(^{**} University of Chinese Academy of Sciences, Beijing 100049)

(^{***} Beijing Institute of Control Engineering, Beijing 100190)

Abstract

In order to tackle the problem of low computing deployment density and energy efficiency in mainstream commercial field-programmable gate array (FPGA) cloud, a hardware-software collaboration framework is proposed. Firstly, the framework makes fully leverage hardware-software-SW programmability of the system on programmable chip (SoPC)-enabled FPGA cloud platform without involvement of commodity expensive CPU-based servers. Secondly, through the system-level design of software-hardware cooperation on SoPC, it provides a flexible and controllable resource management and configuration environment for the programmable hardware of FPGA, effectively improving the general computing and data management capabilities of cloud platform accelerating nodes. Thirdly, through combining the low power characteristics of SoPC and FPGA, the framework can improve the computing energy efficiency of cloud platform accelerating nodes. Finally, the accelerating instantiation results demonstrate the feasibility and efficiency of the proposed framework.

Key words: field-programmable gate array (FPGA)-as-a-service (FaaS), deep learning, accelerator, storage array