HZ BOOKS
华章IT

HPC领域资深专家推荐，中国最大OpenACC技术社区创建者撰写，不可多得的OpenACC技术专著

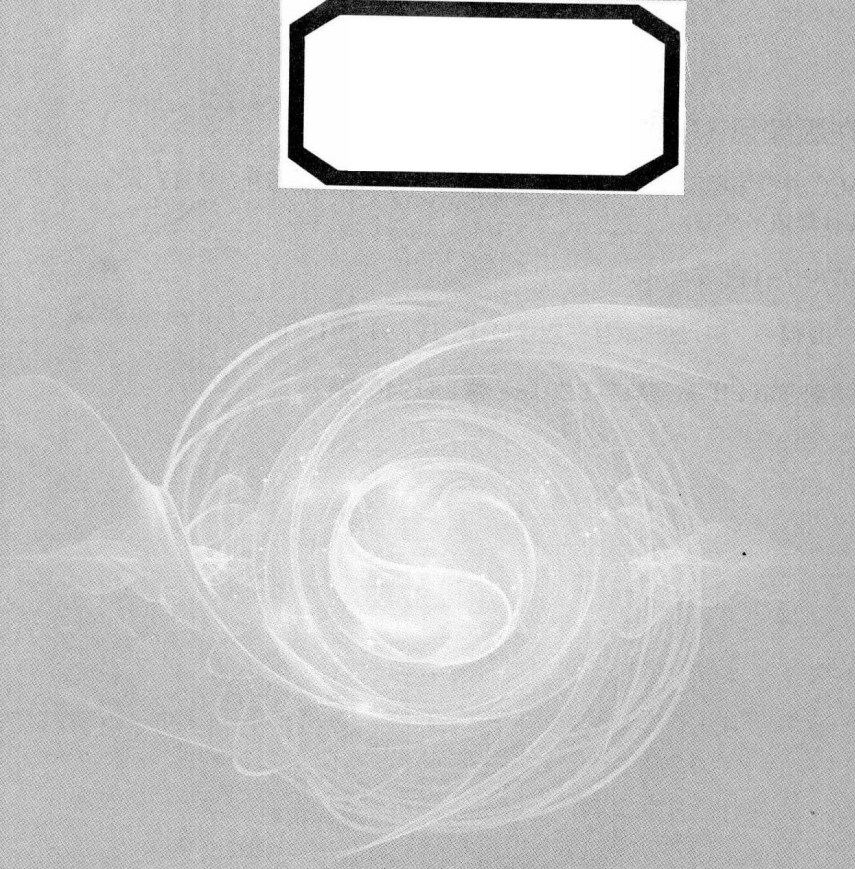
全面讲解OpenACC编程规范、语法的行为机理与设计动机，160个完整示例覆盖众多并行编程场景

OpenACC Parallel Programming

OpenACC 并行编程实战

何沧平 著

机械工业出版社
China Machine Press



OpenACC Parallel Programming

OpenACC 并行编程实战

何沧平 著

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

OpenACC 并行编程实战 / 何沧平著 . —北京：机械工业出版社，2017.1
（高性能计算技术丛书）

ISBN 978-7-111-54965-9

I. O… II. 何… III. 图形软件—程序设计 IV. TP391.41

中国版本图书馆 CIP 数据核字（2016）第 234940 号

OpenACC 并行编程实战

出版发行：机械工业出版社（北京市西城区百万庄大街 22 号　邮政编码：100037）
责任编辑：高婧雅　　责任校对：殷　虹
印　　刷：北京诚信伟业印刷有限公司　　版　　次：2017 年 1 月第 1 版第 1 次印刷
开　　本：186mm × 240mm　1/16　　印　　张：17.25
书　　号：ISBN 978-7-111-54965-9　　定　　价：69.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88379426　88361066　　投稿热线：（010）88379604
购书热线：（010）68326294　88379649　68995259　　读者信箱：hzit@hzbook.com

Preface 序

随着计算机科学技术的飞速发展，作为计算机的核心——处理器的体系结构也经历了从单核、多核到众核的革命性跨越。如今，“异构 + 众核”已成为超级计算机主流的体系结构，并将引领未来 E 级计算的进一步发展。

然而，这种较为“前卫”的结构设计，也给编程者带来了很大的挑战：他们不得不面对更为复杂的底层结构和更多的存储层次，以往的算法设计和程序代码也不得不随之调整。在这种情况下，诸如 CUDA、OpenCL、OpenACC 等新的异构并行编程语言应运而生。其中，OpenACC 是一种极具发展前景的编程模型，其具有使用方便、代码改动小、平台适用范围广的特性，将为新时代的编程者们带来极大的帮助。

OpenACC 虽然有着易学易用的特点，但是要想全面掌握其丰富的语法特性和使用技巧，以编写高效的程序，还是需要一本有权威性、实用性的技术书籍来指导。本书作者何沧平博士是华为高级系统设计工程师，也是 OpenACC QQ 群的群主，具有多年的程序开发经验，一直致力于 OpenACC 的发展与推广，在并行计算应用领域有着很深的造诣和独到的见解。

对于想要掌握 OpenACC 使用技巧和编程精髓的读者来说，本书是一本非常具有参考价值的学习教程。与传统的技术书籍相比，本书更加注重内容的可读性和易用性，逻辑清晰，内容全面准确，且更加注重编程实践，有大量 C/C++/Fortran 的完整代码实例，便于读者学习和实践。作为第一本中文的 OpenACC 技术书籍，可谓为国内的编程学习者带来了福音。

特别值得一提的是，本书首次引入了 OpenACC 在“神威・太湖之光”超级计算机上应用情况的章节。神威・太湖之光是 2016 年全球 TOP500 排名第一的超级计算机，配备了完全由国人自主研发的异构众核处理器。其超强的计算速度成为了高性能计算应用的强力助推。结合应用的特点和处理器独特的结构设计，系统对 OpenACC 进行了扩展。目前，该系统完成了气候气象、航空航天、船舶工程、药物设计等十多个领域的大型应用课题，其中三个高性能计算应用入围“戈登贝尔奖”。这是我国近 30 年来首次入围该奖项。在这些应用的开发过程中，OpenACC 起到了关键作用。

漆锋滨

国家并行计算机工程技术研究中心

前　言 *Forword*

2010年以来，中国超级计算机建设突飞猛进，欣欣向荣。一个原因是国力强盛，大力投资高新科技；另一个原因是整体科技水平提高，需求旺盛。天气预报、石油物探、工程仿真、基因测序等传统应用对计算资源的需求持续增长，以深度学习为代表的人工智能大爆发，资金雄厚的互联网公司对计算能力极度渴求。超级计算机的建设、应用主战场正在从教育科研单位转向科技企业。

为什么要写这本书

面对浩如烟海的数据，CPU已经力不从心，因此世界领先的超级计算机都装备大量的加速器或者众核处理器。

目前主流加速器产品是NVIDIA GPU、AMD GPU和Intel至强Phi协处理器。三种加速器使用的编程语言分别为CUDA C/CUDA Fortran、OpenCL和MIC导语。加速器计算有4个困难。

一是CUDA/OpenCL等低级语言编程难度大，且需要深入了解加速器的硬件结构。而大部分的用户不是专业编程人员，学习一门新的编程技术将耗费大量时间。

二是加速器的计算模型与CPU差别很大，移植旧程序需要几乎完全重写。大量的旧程序在性能优化上已经千锤百炼，稳定性上也久经考验，完全重写是不可完成的任务。

三是低级编程语言开发的程序与硬件结构密切相关，硬件升级时必须升级软件，否则将损失性能。而硬件每隔两三年就升级一次，频繁的软件升级将给用户带来巨大负担。

四是投资方向难以选择。三种加速器均有自己独特的编程语言，且互不兼容。用户在投资建设硬件平台、选择软件开发语言时就会陷入困境，不知三种设备中哪个会在竞争中胜出。如果所选加速器将来落败，将会带来巨大损失；而犹豫不决又将错过技术变革的历史机遇。

OpenACC应运而生，可以克服这4个困难。OpenACC的编程机制是，程序员只在原程序中添加少量编译标识，编译器根据作者的意图自动产生低级语言代码。无须学习新的

编程语言和加速器硬件知识，便能迅速掌握。只添加少量编译标识，不破坏原代码，开发速度快，既可并行执行又可恢复串行执行。在硬件更新时，重新编译一次代码即可，不必手工修改代码。OpenACC 标准制定时就考虑了目前及将来的多种加速器产品，同一份代码可以在多种加速器设备上编译、运行，无成本切换硬件平台。掌握 OpenACC 后，编写程序省时、省力、省心。

本书特色

笔者学习超算技术时有过苦泪：教材一上来就讲技术细节，只能机械地学习，不清楚这些算法、语法要解决什么问题，花费巨大精力后却发现解决不了自己的难题；新技术的资料往往是英文的，而且零碎，还不一定准确，收集、学习成本很高，求助无门；科学计算领域的祖师语言 Fortran 现在有点小众，新技术资料更少。为节省读者时间，本书特别重视易读性、易用性，具体有如下特点。

- 第一本中文 OpenACC 技术书籍，方便国内读者系统阅读，快速掌握。
- 全面、准确：本书覆盖 OpenACC 的所有特性，重要特性结合例子详细讲解，不常用的特性列出标准定义，方便查找。对特性、功能的描述均来自 OpenACC 规范，并用实例验证，确保准确。
- 大量的例子：书中的示例代码超过 160 个，不是代码片断，而是完整代码，拿来即用。全部由笔者编写、测试，运行无报错。
- 兼顾 C/C++ 和 Fortran：以 C/C++ 为主来讲解，但所有的例子都有 Fortran 版的代码，Fortran 专用特性会详细讲解。物理、气象等领域积累的优秀 Fortran 代码可以借此移植到新硬件上，重焕生机。
- 阅读轻松：按照并行程序的编写、调优顺序，先交代各阶段面临的问题，然后自然引入 OpenACC 提供的解决办法，最后实测验证效果。一切顺理成章，读者不会有云里雾里的感觉。

读者对象

- 科学家：迅速改造旧程序，快速编写新的原型程序验证算法、理论，将更多的时间投入到高价值的科研创新中去。
- 企业程序员：一套程序适配多种运行平台，维护简单；性能敏感的部分用 CUDA 等低层语言编写，性能不敏感部分用 OpenACC 编写，相互配合，兼得程序性能和开发效率。
- 本科生、研究生：花最少的时间掌握一门计算工具，省出时间学习更多的知识。

如何阅读本书

第 1 章介绍超级计算技术的发展趋势和并行编程概况，可以从中了解 OpenACC 的作用。没有 CUDA C 基础的读者能够掌握基本概念，便于深入理解 OpenACC 的并行化技术。第 2 章介绍 OpenACC 语言的设计思路。第 3～4 章是本书的核心，将计算部分并行化，并将数据传递时间减到最少。至此，读者已经能够编写性能良好的 OpenACC 程序。第 5～7 章介绍高级并行技术，以进行极致性能优化，以及与 CUDA C/CUDA Fortran 和各类库的混合编程。第 8 章给出 OpenACC 规定的所有运行时例程，不用细读，用到时再参考。第 9 章指导部署开发环境，以便快速上手。

勘误和支持

本人水平有限，错误与疏漏在所难免，恳请批评指正。联系笔者请发送电子邮件至 hpcfan@foxmail.com。更多技术资源请访问 www.gpujisuan.com。技术交流 QQ 群 284876008（将满）、564520462，欢迎加入。

致谢

感谢国家并行计算机工程技术研究中心漆锋滨老师撰写第 10 章，并为本书作序。感谢 PGI 美女工程师王珍、帅气工程师仰珊歆、Daniel Tian、资深专家 Mathew Colgrove 的技术支持。感谢机械工业出版社的策划编辑高婧雅尽心协调与支持。

何沧平

Contents 目　录

第 1 章 Chapter 1

并行编程概览

对绝大多数人而言，编程语言只是一个工具，讲究简单高效。科学家的主要精力应该用在科研创新活动上，编程工作仅仅是用来验证创新的理论，编程水平再高也不可能获得诺贝尔奖。对学生和企业程序员而言，技术无穷尽，永远学不完，不用即忘，应该认清技术发展方向，学习有前途的技术，不浪费青春年华。

OpenACC 语言专为超级计算机设计，因此读者需要了解超级计算机的技术演进方向，特别是主流加速器的体系架构、编程模型，看清 OpenACC 的应用场景，有的放矢。普通读者虽然不会用到大型机群，但小型机群甚至单台服务器、普通显卡的计算模式都是相同的。

最近几年的著名超级计算机（见附录 A）均采用加速器作为主要计算部件，可预见未来几年的上层应用仍将围绕加速器展开。

1.1 加速器产品

超级计算机的加速器历史上有多种，本节只介绍当前流行的两种：英伟达 GPU 和英特尔融核处理器。加速器的物理形态是 PCIe 板卡，样子大致如图 1.1 所示，图 1.2 是拆掉外壳后的样子，正中央的是 GPU 芯片，芯片周围的小黑块是显存颗粒，金黄色的边缘处是与 PCIe 连接的金手指，通过 PCIe 插槽与 CPU 相连。图 1.3 中的机架式服务器左下部装有 4 块 GPU 卡，图 1.4 是服务器的主板俯视图，箭头处就是 4 个 PCIe 插槽。

从逻辑关系上来看，目前市场在售的英特尔服务器 CPU 中已经集成了内存控制器和 PCIe 通道。2 颗 CPU 通过 1 个或 2 个 QPI 接口连接，CPU 通过内置的内存控制器连接 DDR3

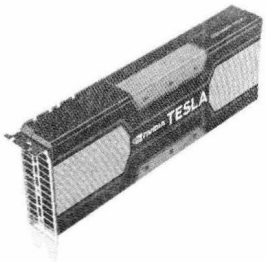


图 1.1 英伟达 Tesla GPU[⊖]

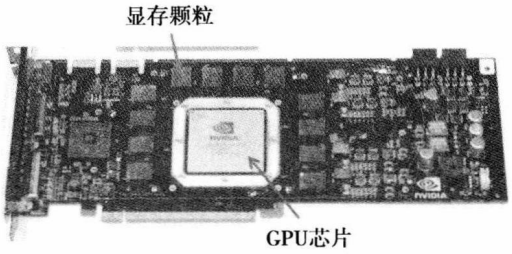


图 1.2 英伟达 GPU 内部构造

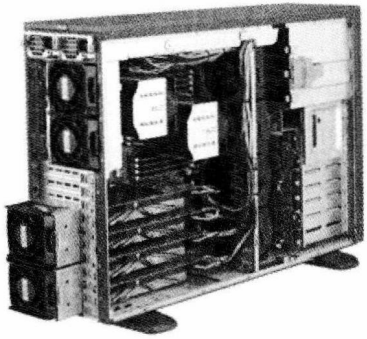


图 1.3 GPU 服务器[⊜]

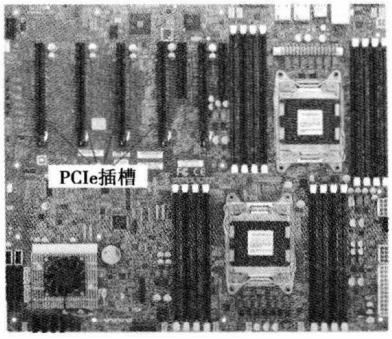


图 1.4 GPU 服务器的主板

或 DDR4 内存条，GPU 加速器通过 PCIe 总线与 CPU 相连（图 1.5）。2016 年 6 月主流的英特尔至强 E5-2600 v3 和 E5-2600 v4 CPU 每颗拥有 40 个 PCIe Lan，而每块 GPU 的接口需要 16 个 PCIe Lan，因此每颗 CPU 上最多全速挂载 2 块 GPU。有些服务器的 PCIe ×16 插槽上只安排 ×8，甚至 ×4 的信息速率，可以挂载更多的 GPU。

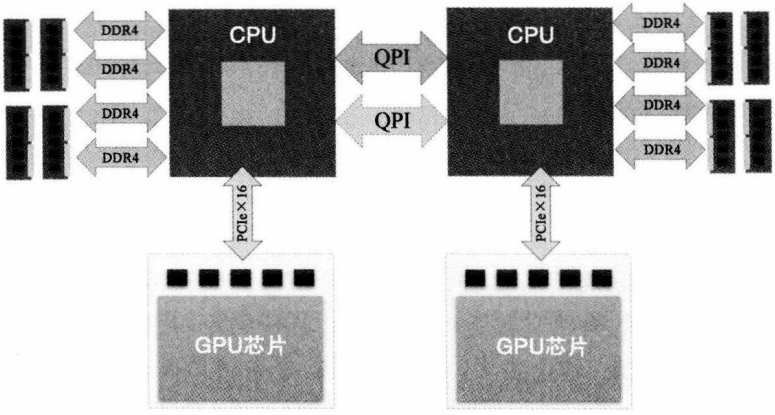


图 1.5 加速器在服务器上的逻辑位置

⊖ 图片来自 www.nvidia.com。

⊜ 图片来自 www.supermicro.org.cn。

英特尔融核处理器在服务器上的位置与 GPU 一样，不再赘述。

1.1.1　英伟达 GPU

通用计算 GPU 是英伟达公司发明的，每隔 2～3 年就会升级硬件架构。通用计算 GPU 的第一代架构代号是 G80，每 8 个核心封装在一起称为一个流式多处理器（Streaming Multiprocessor，SM），2 个流多处理器共用一块 L1 缓存，所有的核心共用 L2 缓存，任意核心都能访问芯片外部的 GPU 显存（图 1.6 中的 FB）。从图 1.6 中可以看出，G80 架构 GPU 产品最多可以有 128 个核心。

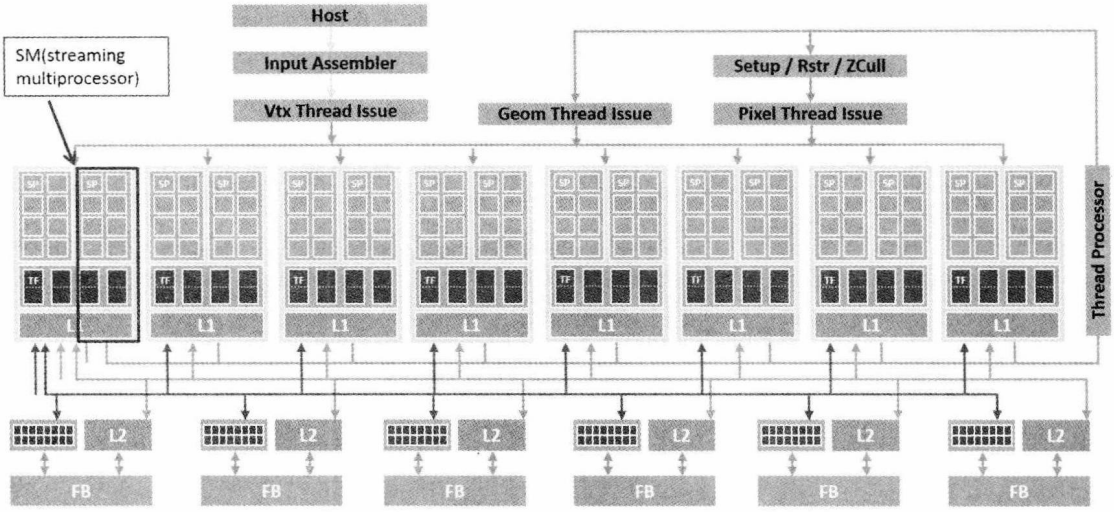


图 1.6　通用计算 GPU 的 G80 架构

接下来的架构代号是 GT200，然后是 Fermi。Fermi 架构（图 1.7）最多包含 14 个流式多处理器，每个流式多处理器包含 32 个核心。6 个 DRAM 接口，1 个 PCIe 接口（图 1.16 中的 Host Interface）。

流式多处理器内部组件也相当多（图 1.8）：1 个指令缓存（Instruction Cache），2 组 Warp 调度器（Warp Scheduler）、分发单元（Dispatch Unit），一堆寄存器；最重要的是 32 个核心，每个核心拥有一个单精度浮点单元和一个整数单元；16 个 Load/Store 单元；4 个特殊功能单元（Special Function Unit），负责计算双精度浮点数、三角函数、超越函数、倒数、平方根等。共享内存 +L1 缓存共 64KB，可以灵活配置。

接下来是 Kepler 架构（图 1.9），这一代架构最大可以包含 15 个流式多处理器（Streamong Multiprocessor extreme，SMX）和 6 个 64 位内存控制器。不同的产品型号将使用不同的配置，可以包含 13 或 14 个 SMX，多种参数都有升级和更改。Kepler 架构的主要设计目标是提高用电效率，台积电的 28nm 制造工艺在降低功耗方面起着重要的作用。Kepler 架构还提高了双精度计算能力。

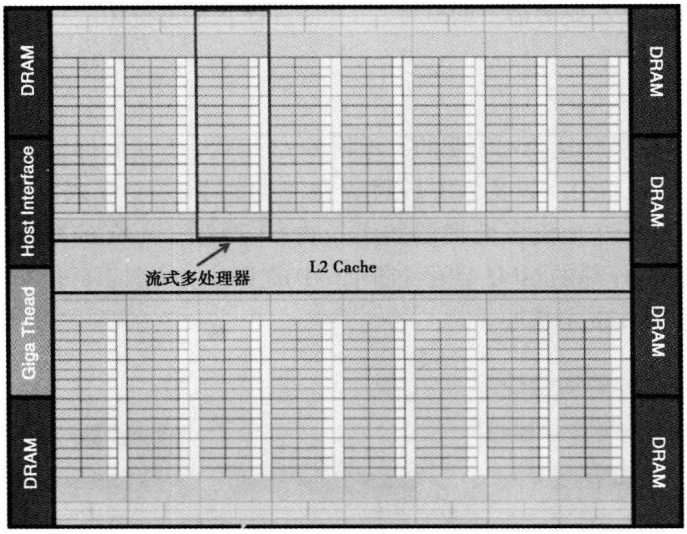


图 1.7 Fermi 架构图

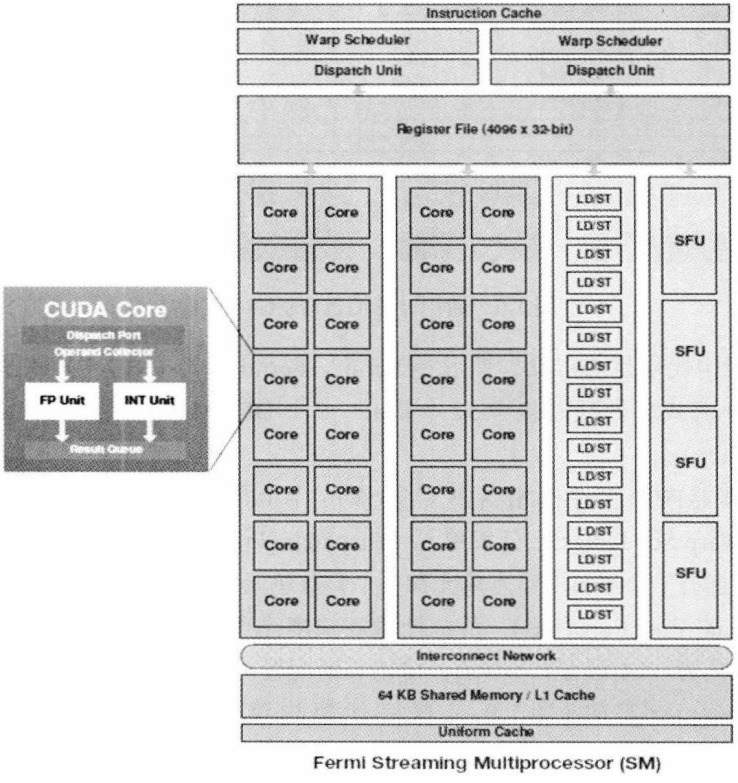


图 1.8 Fermi 架构的流式多处理器⊖

⊖ 图片来自 www.nvidia.com。

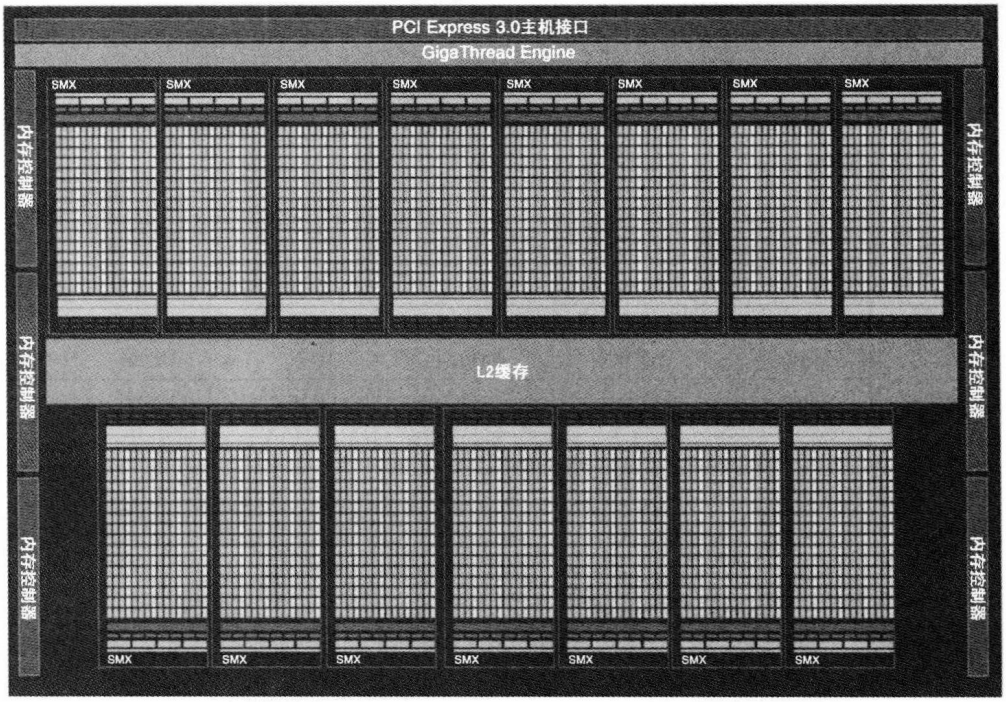


图 1.9　Kepler 架构[⊖]

流式多处理器 SMX（图 1.10）包含 192 个单精度核心、64 个双精度单元、32 个特殊功能单元（SFU）和 32 个加载 / 存储单元（LD/ST）、3 个 Warp 调度器、6 个分发器、一大堆寄存器。每个核心由 1 个浮点计算单元和 1 个整数算术逻辑单元组成，支持融加（FMA）运算。每个 SMX 拥有 64KB 的片上存储器，可配置为 48KB 的共享存储器和 16KB 的 L1 缓存，或配置为 16KB 的共享内存和 48KB 的 L1 缓存。

这里简要介绍市面的主力 GPU 产品型号，见表 1.1。

表 1.1　2014～2016 年市场主力 GPU 产品规格

特性 \ 型号	Tesla K80	Tesla K40	Tesla K20X	Tesla K20	Tesla K10
GPU 数量类型	2×Kepler GK210	1 个 GK110	1 个 Kepler GK110		2 个 Kepler GK104
双精浮点峰值（基础 / 提速）	1.87 Tflops 2.91 Tflops	1.43 Tflops 1.66 Tflops	1.31 Tflops	1.17 Tflops	190（2×95）Gflops
单精浮点峰值（基础 / 提速）	5.6 Tflops 8.74 Tflops	4.29 Tflops 5 Tflops	3.95 Tflops	3.52 Tflops	4576（2×2288）Gflops
存储器带宽（ECC 关闭）	480GB/s	288GB/s	250GB/s	208GB/s	320（2×160）GB/s

⊖　图片来自《英伟达 ™（NVIDIA ®）下一代 CUDA TM 计算架构：Kepler TM GK110》。

（续）

型号 / 特性	Tesla K80	Tesla K40	Tesla K20X	Tesla K20	Tesla K10
存储器容量（GDDR5）	24GB（2×12GB）/s	12GB/s	6GB/s	5GB/s	8GB（2×4GB）/s
CUDA 核心数	4992	2880	2688	2496	3072（2×1536）
核心时钟	562MHz	745MHz	732MHz	706MHz	745MHz
超频时钟	875MHz	810/875MHz	不支持	不支持	不支持
存储器时钟	5GHz DDR5	6GHz GDDR5	5.2GHz DDR5	5.2GHz DDR5	5GHz DDR5
最大功耗	300W	235W	235W	225W	225W
制程	28nm	28nm	28nm	28nm	28nm

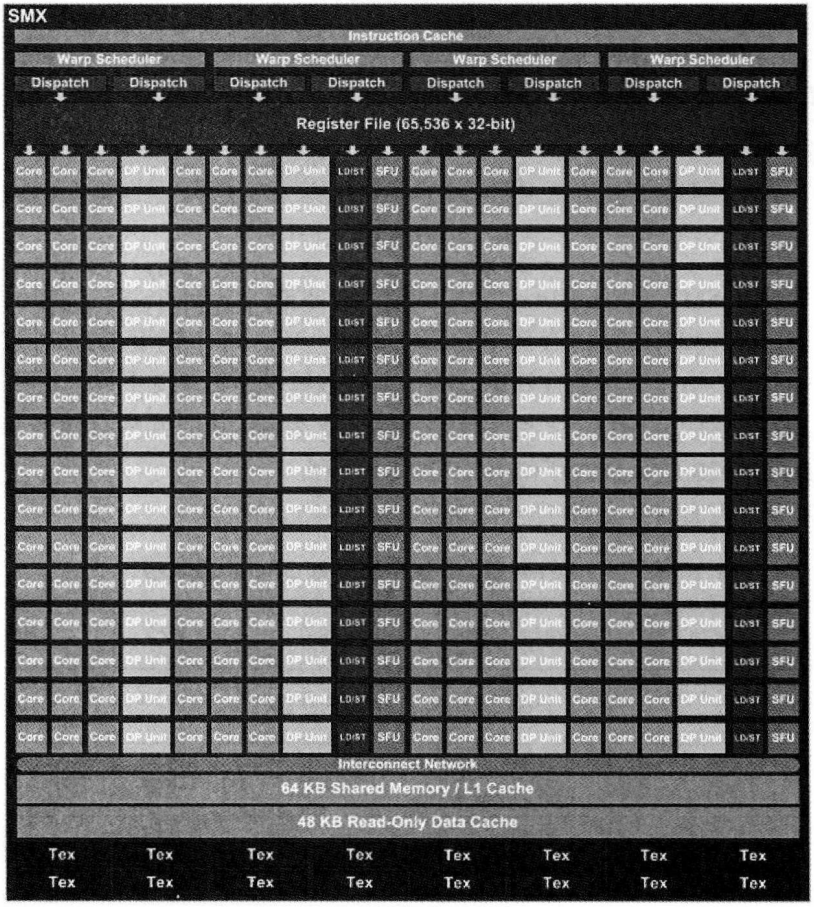


图 1.10 kepler 架构中的流式多处理 SMX ⊖

⊖ 图片来自《英伟达™（NVIDIA®）下一代 CUDA TM 计算架构：Kepler TM GK110》。

2016 年 4 月，英伟达在 GPU 技术会议（GPU Technology Conference）上发布了新一代 Pascal 架构和旗舰产品 Tesla P100，引入一些新特性。

- 极致性能：为高性能计算、深度学习等计算领域设计。双精度浮点峰值 5.3 Tflops，单精度浮点峰值 10.6 Tflops，专为深度学习设计的半精度浮点峰值达到惊人的 21.2 Tflops。按双精度峰值对比，Tesla P100 是同期主力高端 CPU 英特尔至强 E5-2680 v4 的 10 倍。按照深度学习应用性能对比，Tesla P100 是 E5-2680 v4 的 20 倍。
- NVLink：为应用扩展性全新设计的高速、高带宽互连协议。
- HBM2：快速、大容量、高效片上堆叠式内存。
- 统一内存：用统一的代码来管理主机 CPU 内存和 GPU 内存，方便开发代码。

Pascal 最强劲的是 GP100 硬件架构（图 1.11），包含 60 个 Pascal 流式多处理器和 8 个 512 位内存控制器（共 4096 位）。每个流式多处理器拥有 64 个 CUDA 核心和 4 个纹理单元。GP100 共有 3840 个单精度核和 240 个纹理单元。每个内存控制器附带 512KB 的 L2 缓存，每个 HBM2 堆叠内存由一对内存控制器管理。L2 缓存共计 4096KB。Tesla P100 共配置有 56 个流式多处理器。

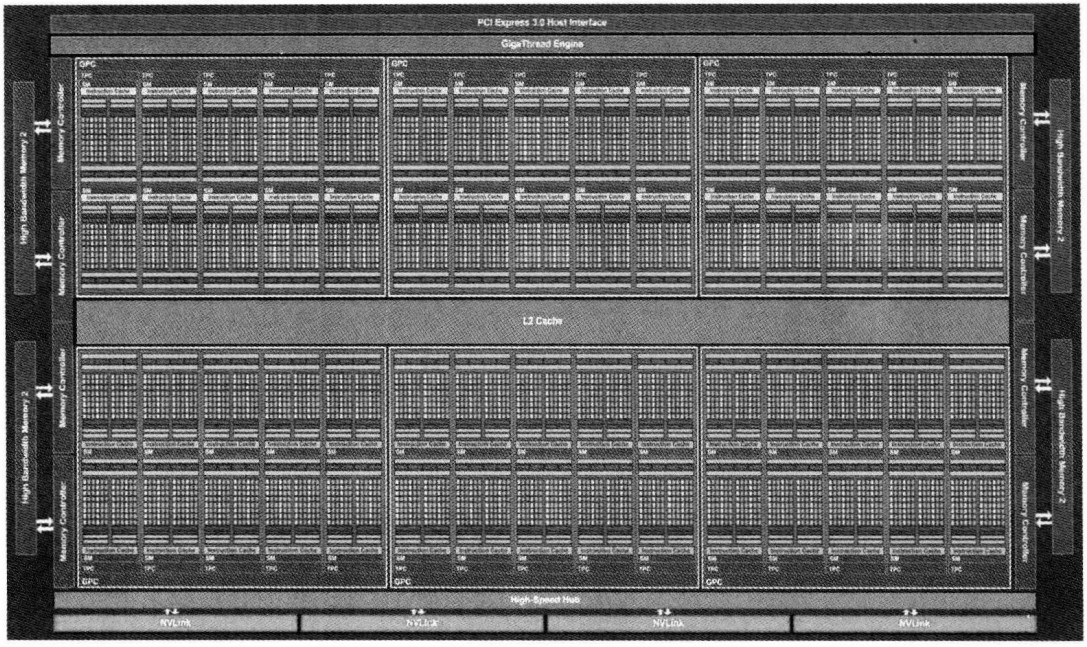


图 1.11　GP100 架构全景图[⊖]

GP100 的第 6 代流式多处理器架构提升了 CUDA 核心的利用率和能源效率，可以运行在更高的频率上。每个流多处理器包含 64 个单精度（FP32）CUDA 核心，分成两部分，每

⊖ 图片来自《Pascal architecture whitepaper》。