GPU基础知识分享

Ping Chen

2021/4/17

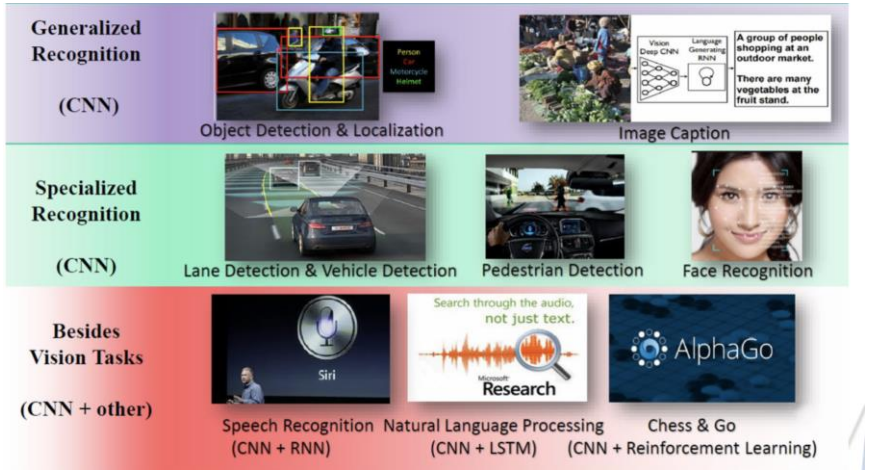
Zhejiang University

Outline

- **GPU发展概述**
- GPU计算部分
- GPU存储部分

GPU发展概述

用于处理图像以及AI训练/推理

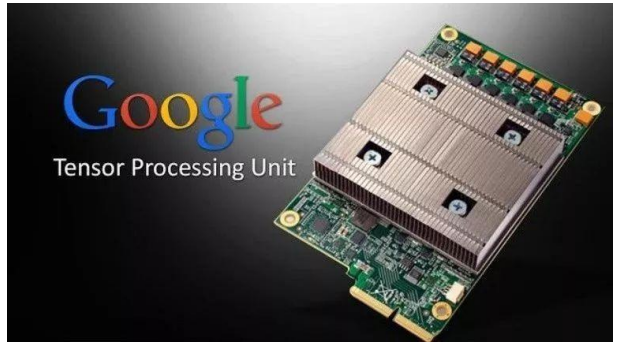


NVIDIA

世界上最大的独立显卡芯片生产销售商

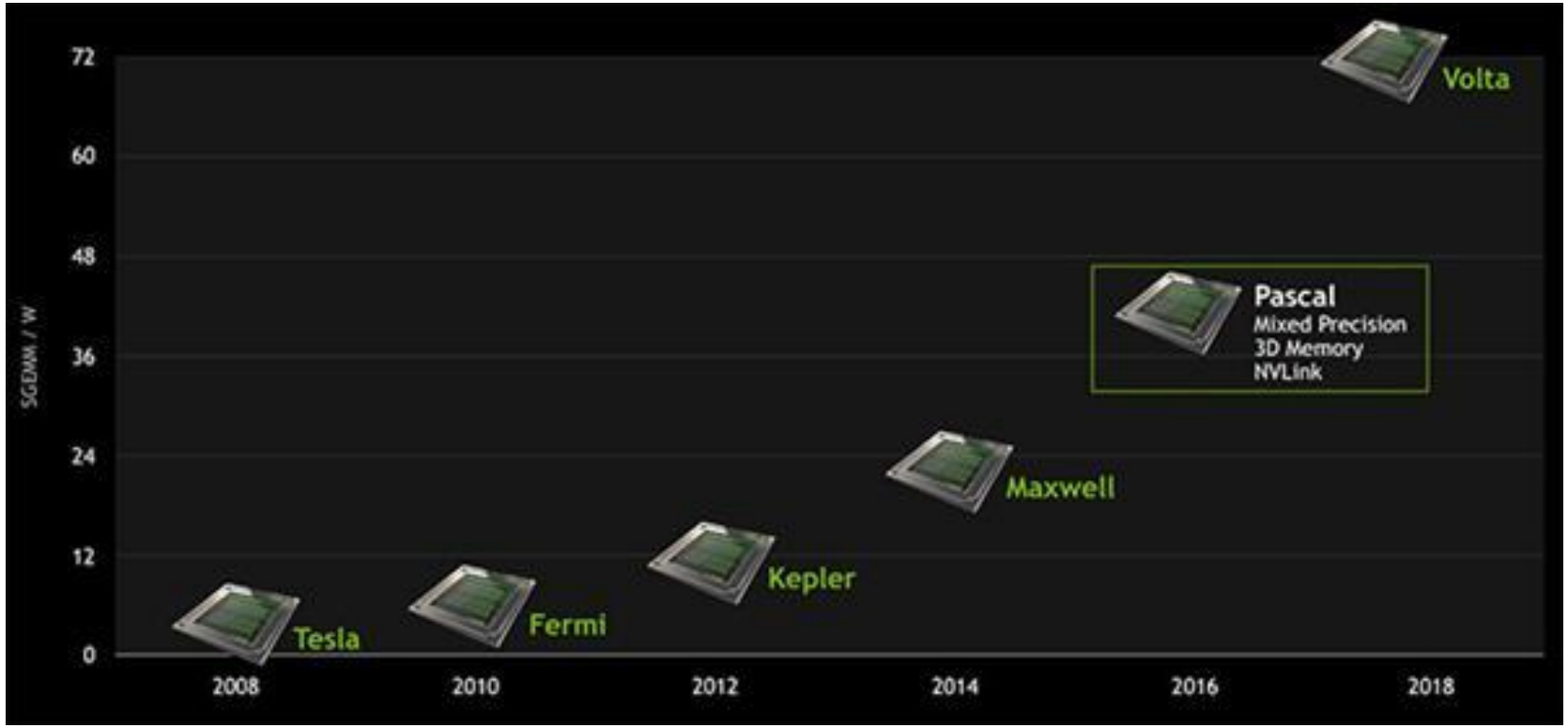
AMD

NVIDIA的主要竞争对手，不仅是独立显卡的经销商，也是CPU的经销商



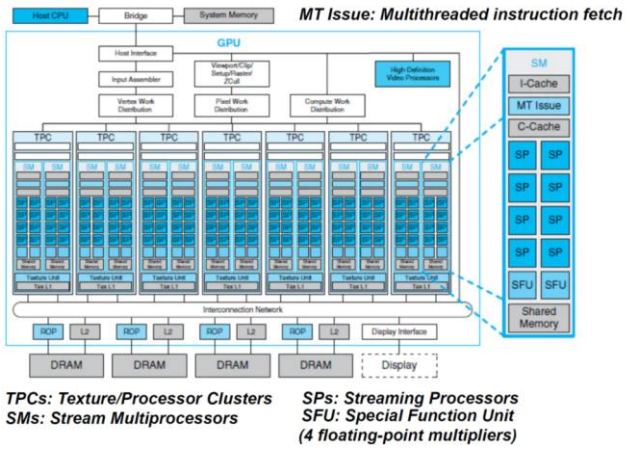
谷歌TPU

NVIDIA产品发展历程

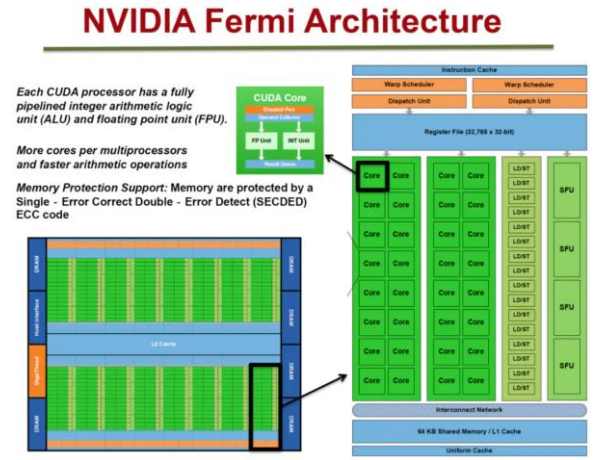


以及2017年的图灵架构（**Turing**）、2020年最新安培（**Ampere**）架构

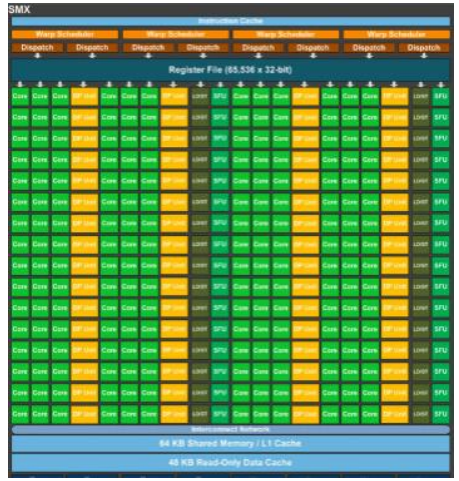
NVIDIA产品发展历程



Tesla最初是给计算处理单元使用的，应用于早期的CUDA系列显卡芯片。



Fermi是第一个完整的GPU计算架构。首款可支持与共享存储结合纯cache层次的GPU架构。



Kepler相较于Fermi更快，效率更高，性能更好。

Maxwell：对比Tesla架构来说，在处理单元上有了很大的提升。

Pascal 架构将处理器和数据集成在同一个程序包内，以实现更高的计算效率。

Volta 配备640 个Tensor 核心，每秒可提供超过100 兆次浮点运算(TFLOPS) 的深度学习效能，比前一代的Pascal 架构快5倍以上。

NVIDIA产品发展历程

Turing提供了64个fp32核心，64个int32核心和8个改进的混合精度Tensor Cores核心。这使得fp32和int32的操作可以同时执行。

参考：https://www.zhihu.com/question/21980949

Ampere架构，算力加强，最新推出的专业卡RTX A6000，与上一代产品相比，能够提供惊人的性能。

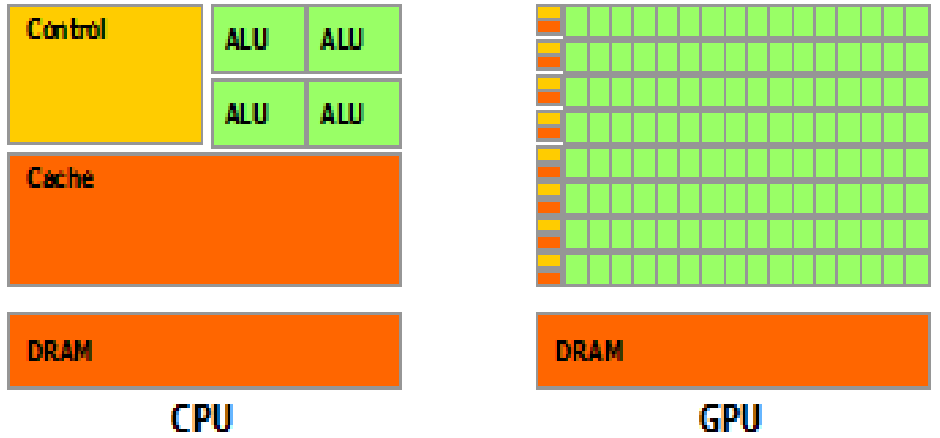
- 第二代RT Core：最高可提供2倍于上一代的吞吐量，以及并行光线追踪、着色和计算功能。
- 第三代Tensor Core：最高可提供5倍于上一代的吞吐量，并支持全新TF32和BF16数据格式，结合稀疏运算特性提供10倍加速性能。
- 全新CUDA Core：最高可提供2倍于上一代的FP32吞吐量，能够显著提高图形和计算能力。
- 48GB GPU内存：单个GPU可提供的最大内存，通过NVLink连接两个GPU可以扩展到96GB。
- 虚拟化：通过添加NVIDIA虚拟工作站等NVIDIA虚拟GPU软件，为远程用户提供针对图形工作负载和强大虚拟工作站实例的大规模支持，赋力高端设计、AI和计算工作负载的更大规模工作流程。
- PCIe Gen 4：提供2倍于上一代的带宽，加速Lenovo ThinkStation P620等PCIe Gen 4服务器和工作站中数据密集型工作负载（如数据科学、混合渲染和视频流）的GPU数据传输。

Outline

- GPU发展概述
- **GPU计算部分**
- GPU存储部分

GPU设计哲学

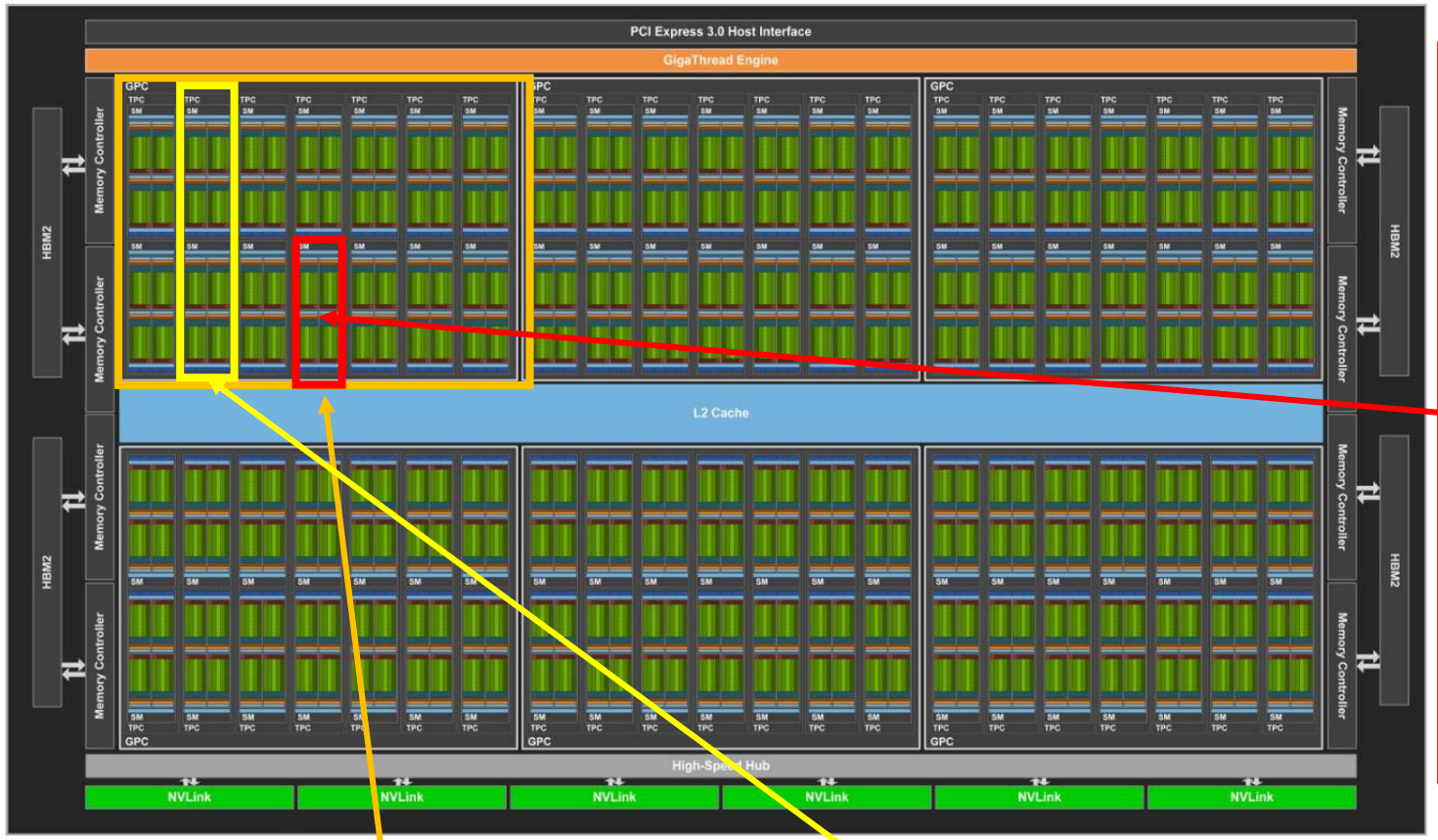
- CPU设计目标：低延迟
  - 高频率（少计算内核，超过5GHz）
  - 计算、逻辑、控制通用设计



- GPU设计目标：高吞吐
  - 多计算内核，低频率（几千个核心，低于1.5GHz）
  - 大量寄存器，保证大量线程切换

GPU计算部分

Volta V100 硬件结构

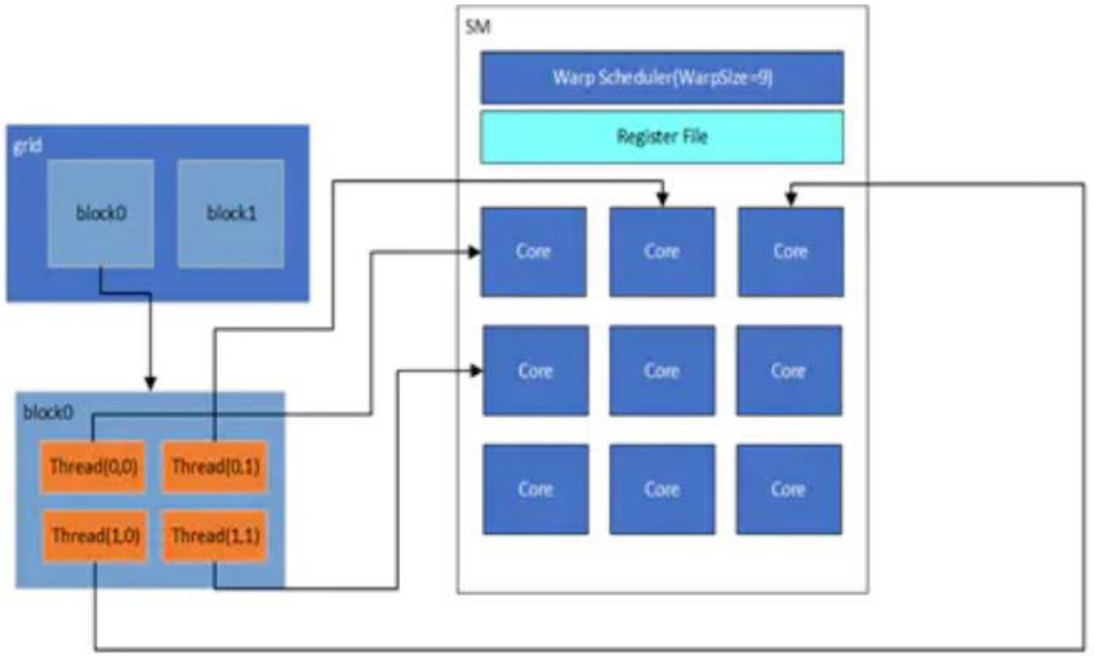


SM（Streaming Multiprocessors ）结构

纹理处理簇Texture Processing Clusters (TPCs)

形处理簇GPU Processing Clusters (GPCs)

GPU计算部分



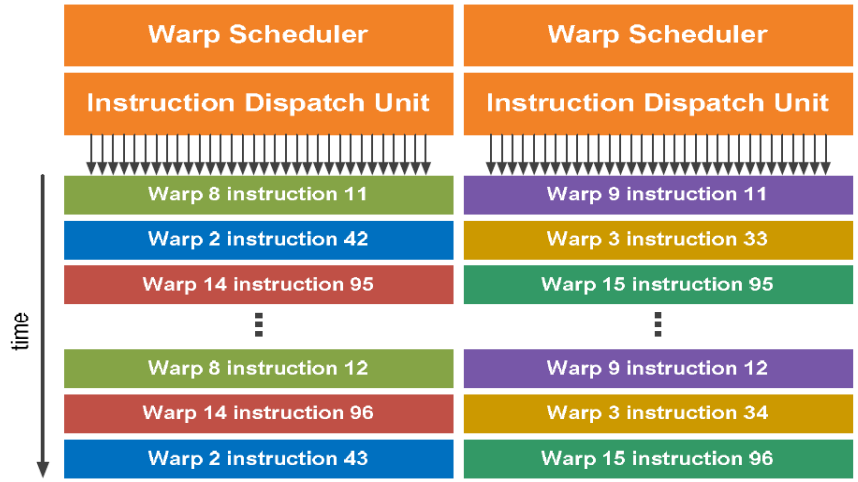
GPU计算单元部件**硬件结构**：①SM -> ②Core（V100中有80个SM，每个SM有如下Core）

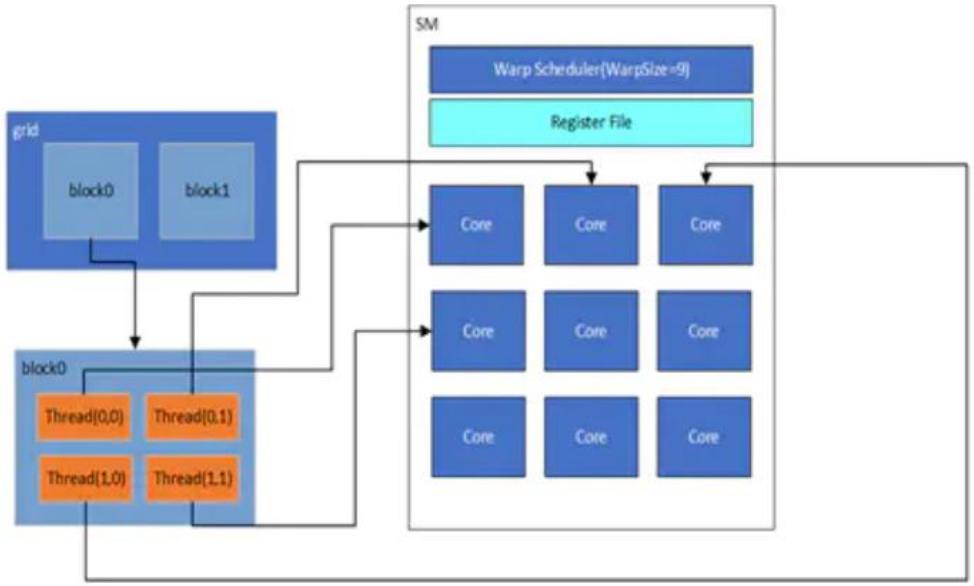
- 64 FP32 cores
- 64 INT32 cores
- 32 FP64 cores
- 8 Tensor Cores

软件结构：①Grid -> ②Block -> ③Thread

- 编程时，一个Kernel（function）对应一个Grid；
- 一个Grid包括多个Block（编程中的Gridsize）
- 一个Block包含多个Thread（编程中的Blocksize）

GPU计算部分



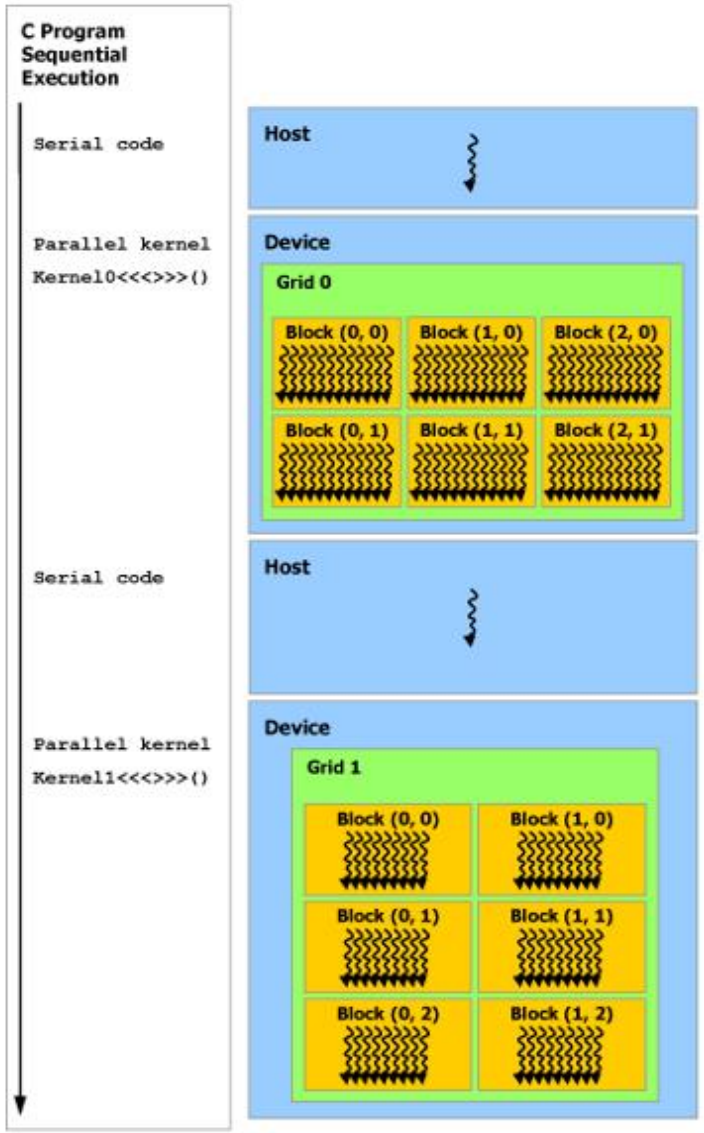


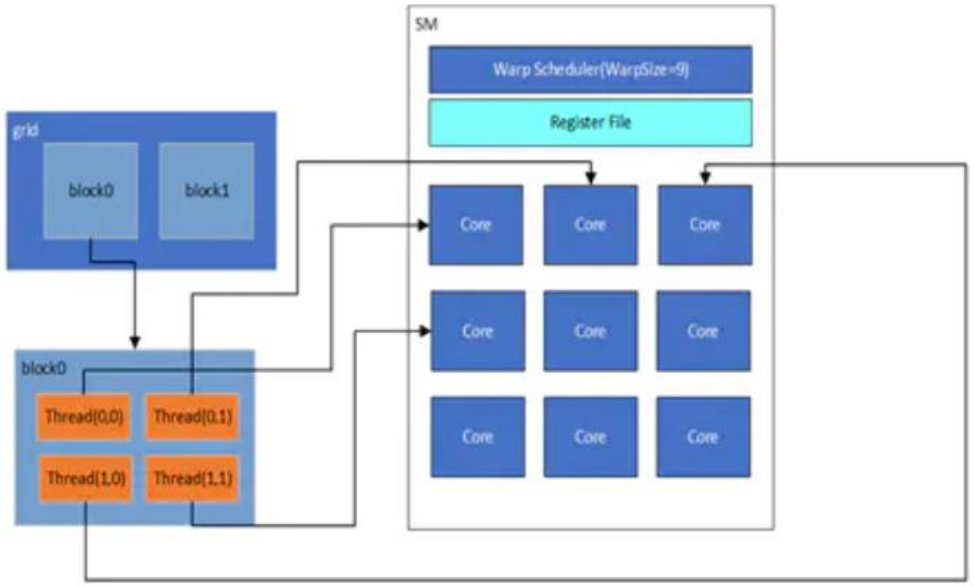
CUDA的调度单位: Warp(一组线程，一般为32个核)

- （Block <----> SM）一个逻辑Block的任务会放到一个物理SM中执行；
- （Thread <----> CUDA Core） Block中的线程会以32为粒度打包为Warp，在SM的Core中执行；
- 在SM内部有n个Warp scheduler（不同架构有不同数量，一般2~4个），一个SM内部真正能并行n*32个线程（有多少调度器就能并行多少warp），其余的在物理上其实是类似于串行（但是可以上下文切换：很像并发）。
- 一个block会在任务开始前将所有线程的的状态保存到寄存器中，上下文切换其实就是不断的切换寄存器即可，所以GPU内部的context switch非常快速。

https://stackoverflow.com/questions/6605581/what-is-the-context-switching-mechanism-in-gpu

GPU计算部分





简单优化规则：

- Grid一定要给足block
- Block内一定要给足线程(大量并发，隐藏延迟)
- Block内线程数目一定是Warp Size的倍数

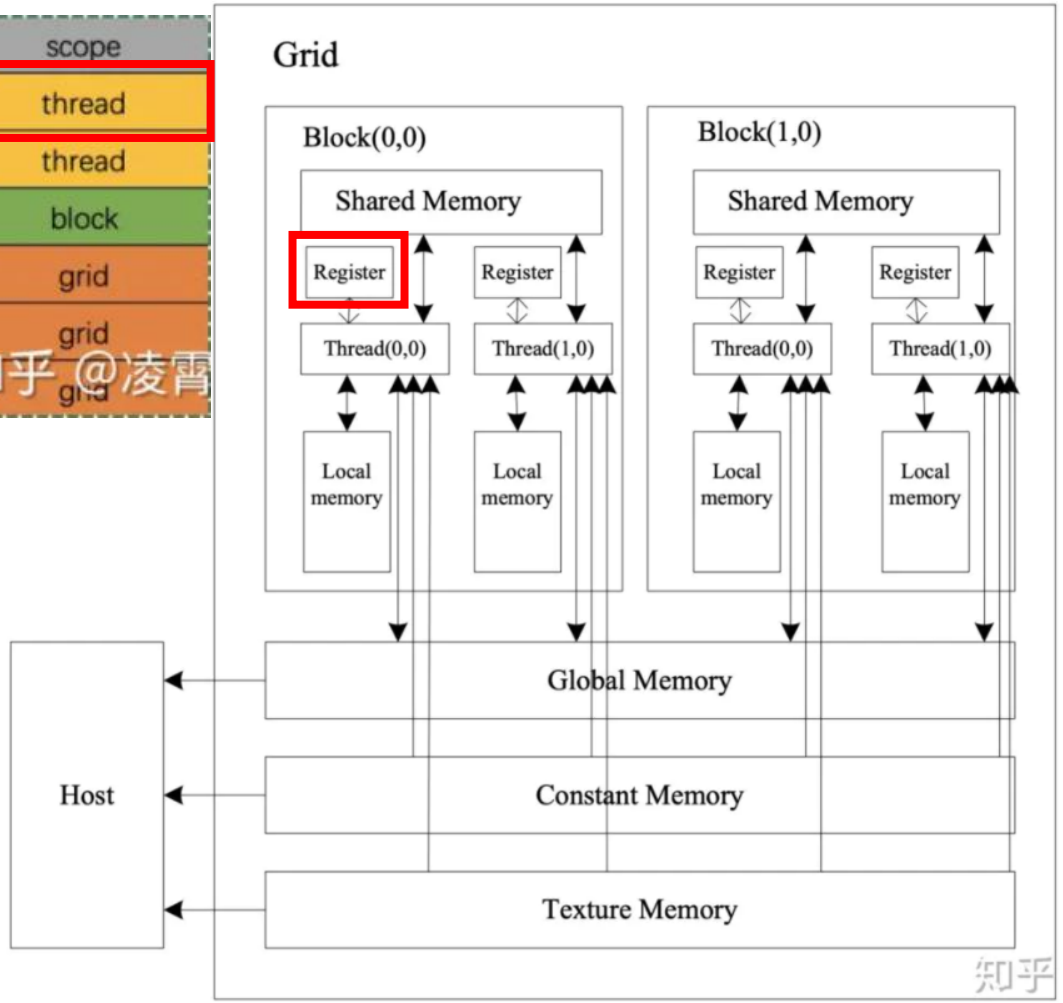
Outline

- GPU发展概述
- GPU计算部分
- **GPU存储部分**

GPU存储部分-Register

存储单元	位置	是否有cache	访问速度(时钟周期)	读/写	scope
register	on chip	N/A	0.19	R/W	thread
local memory	off chip	无	203	R/W	thread
shared memory	on chip	N/A	47	R/W	block
Constant Memory	off chip	有	110	R	grid
Global Memory	off chip	有	218	R/W	grid
Texture Memory	off chip	有	115	R	grid

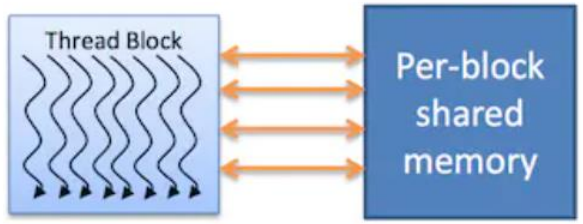
- 寄存器是速度最快的存储单元，位于GPU芯片的SM上，用于存储局部变量。
- 每个SM（SMX）上有成千上万（65536）的32位寄存器，当kernel函数启动后，这些寄存器被分配给指定的线程来使用。
- 由于不同kernel函数需要的寄存器数量也不相等，所以，也有一个规定一个线程的最大寄存器数量是256个。

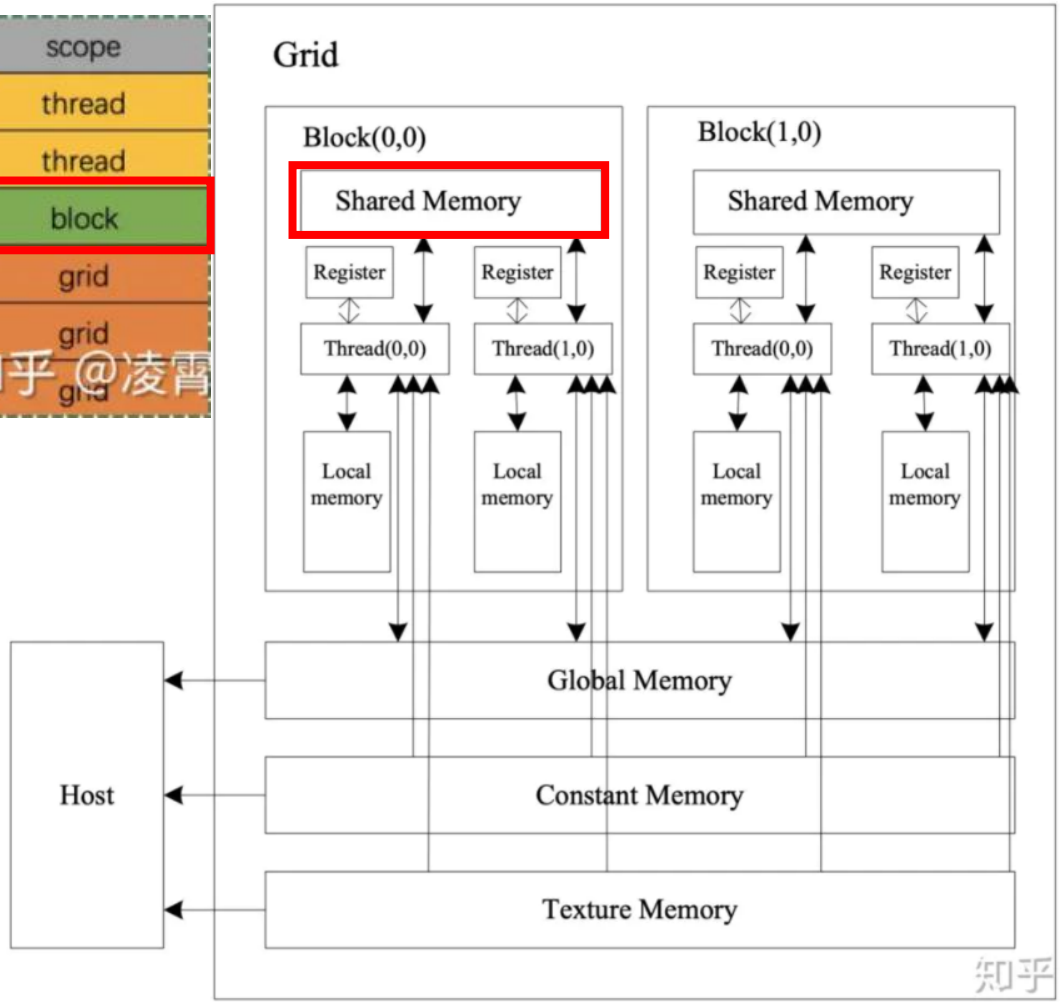


GPU存储部分-Shared Memory

存储单元	位置	是否有cache	访问速度(时钟周期)	读/写	scope
register	on chip	N/A	0.19	R/W	thread
local memory	off chip	无	203	R/W	thread
shared memory	on chip	N/A	47	R/W	block
Constant Memory	off chip	有	110	R	grid
Global Memory	off chip	有	218	R/W	grid
Texture Memory	off chip	有	115	R	grid

- Shared Memory位于GPU芯片上，访问延迟仅次于寄存器。
- Shared Memory是可以被一个Block中的所有Thread来进行访问的，可以实现Block内的线程间的低开销通信。
- V100 -> 96 KB/SM

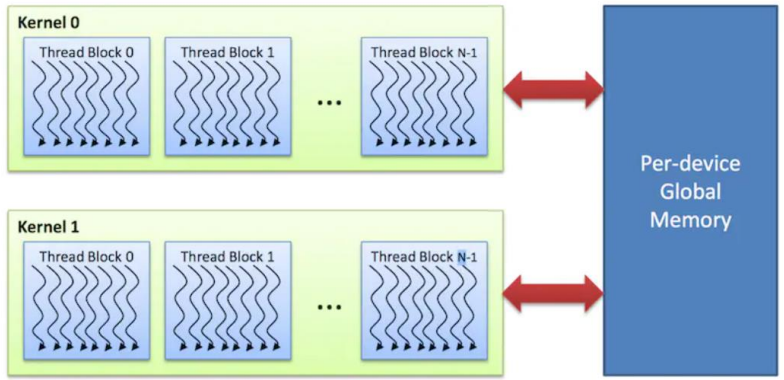


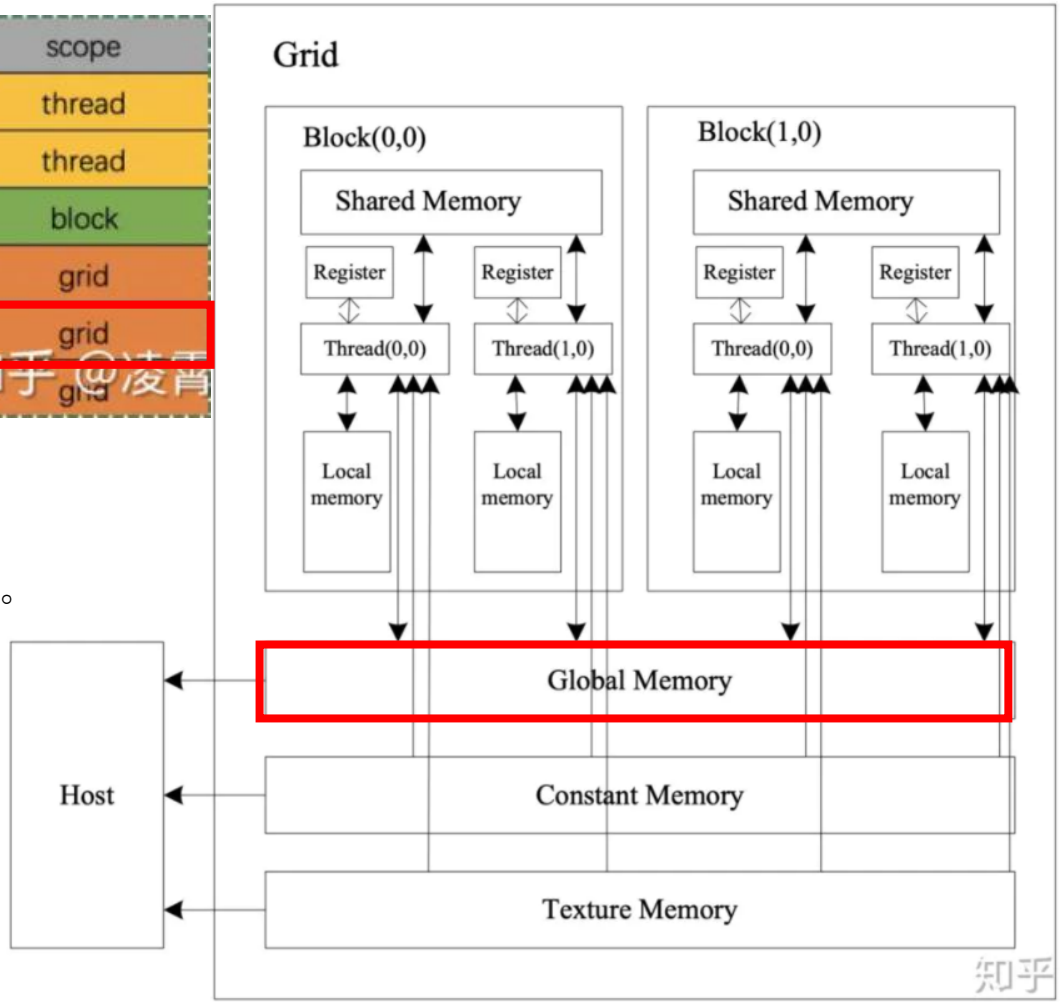


GPU存储部分-Global Memory

存储单元	位置	是否有cache	访问速度(时钟周期)	读/写	scope
register	on chip	N/A	0.19	R/W	thread
local memory	off chip	无	203	R/W	thread
shared memory	on chip	N/A	47	R/W	block
Constant Memory	off chip	有	110	R	grid
Global Memory	off chip	有	218	R/W	grid
Texture Memory	off chip	有	115	R	grid

- Global Memory是GPU中最大的存储单元，Host memory与GPU之间的数据交互均会通过Global Memory进行保存，它也是读取速度最慢的组件。GPU中所有计算单元均可以访问该存储单元。



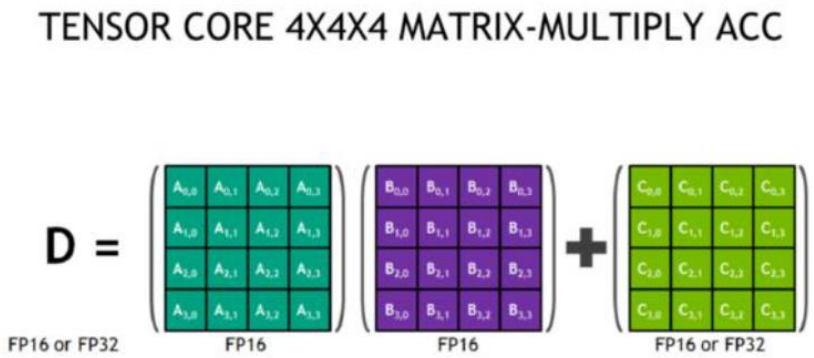


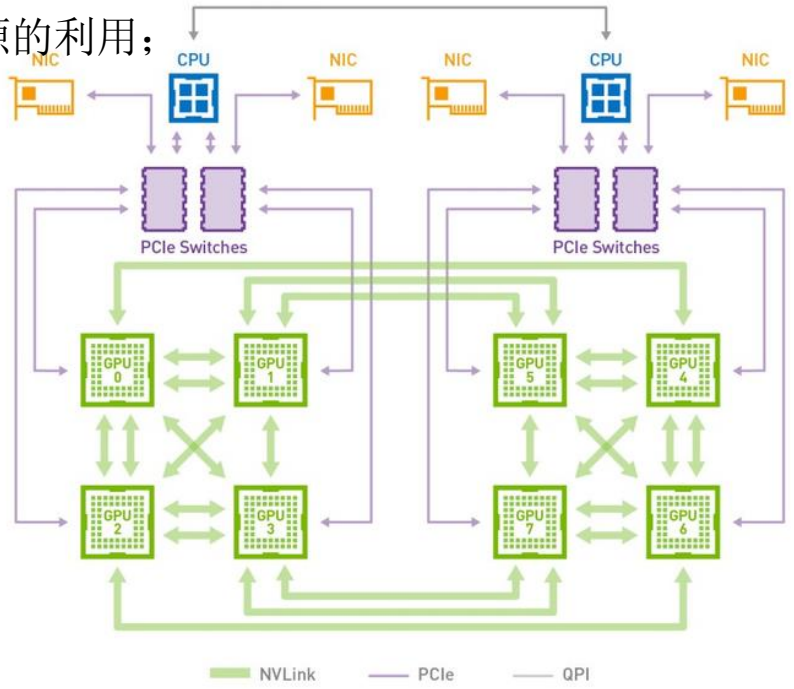
GPU高级组件-新特性

V100详细设备参数

Tesla Product	Tesla V100
GPU	GV100 (Volta)
SMs	80
TPCs	40
FP32 Cores / SM	64
FP32 Cores / GPU	5120
FP64 Cores / SM	32
FP64 Cores / GPU	2560
Tensor Cores / SM	8
Tensor Cores / GPU	640
GPU Boost Clock	1530 MHz
Peak FP32 TFLOPS[1]	15.7
Peak FP64 TFLOPS[1]	7.8
Peak Tensor TFLOPS[1]	125
Texture Units	320
Memory Interface	4096-bit HBM2
Memory Size	16 GB
L2 Cache Size	6144 KB
Shared Memory Size / SM	Configurable up to 96 KB
Register File Size / SM	256KB
Register File Size / GPU	20480 KB
TDP	300 Watts
Transistors	21.1 billion
GPU Die Size	815 mm^2
Manufacturing Process	12 nm FFN

- **1 NVLink：** 相比于PCI-E速度更快；
- **2 NVSwitch：** 类似于PCIe使用PCIe Switch用于拓扑的扩展，NVIDIA使用NVSwitch实现了NVLink的全连接；
- **3 Tensor Core：** 面向深度学习，混合精度计算，更高效；
- **4 GPUDirect Storage：** GPU直通SSD；
- **5 Volta MPS：** 在共享GPU场景下提高资源的利用；





总结

Thanks