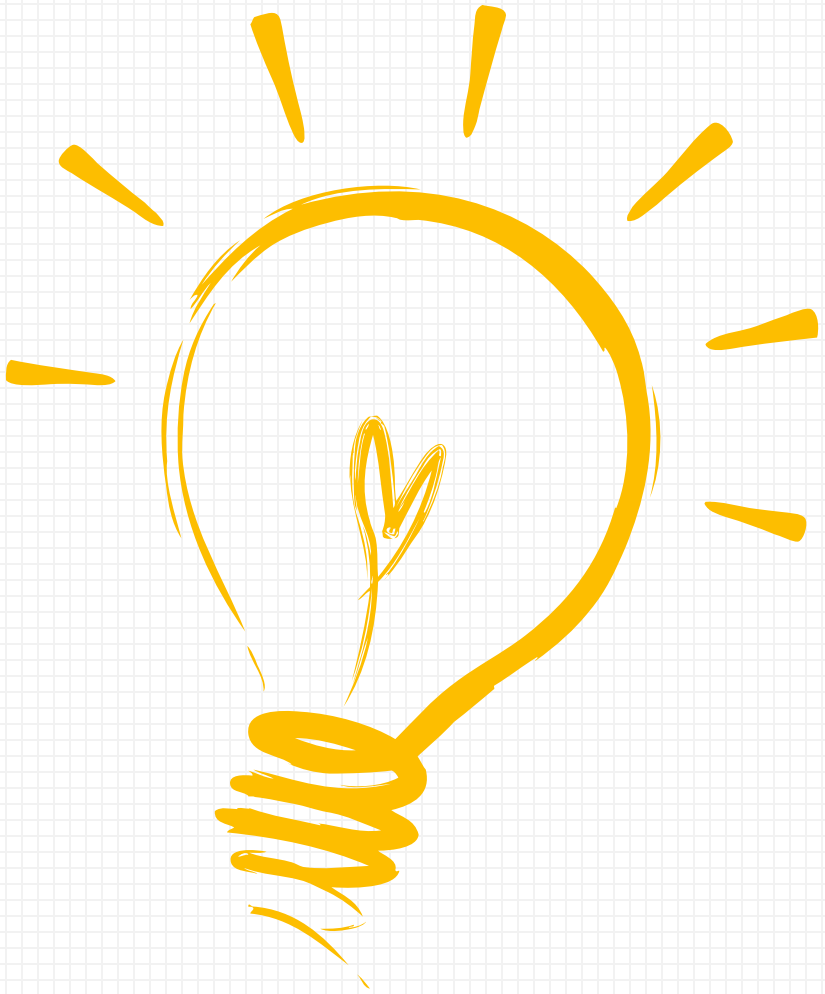
寄存器和本地内存

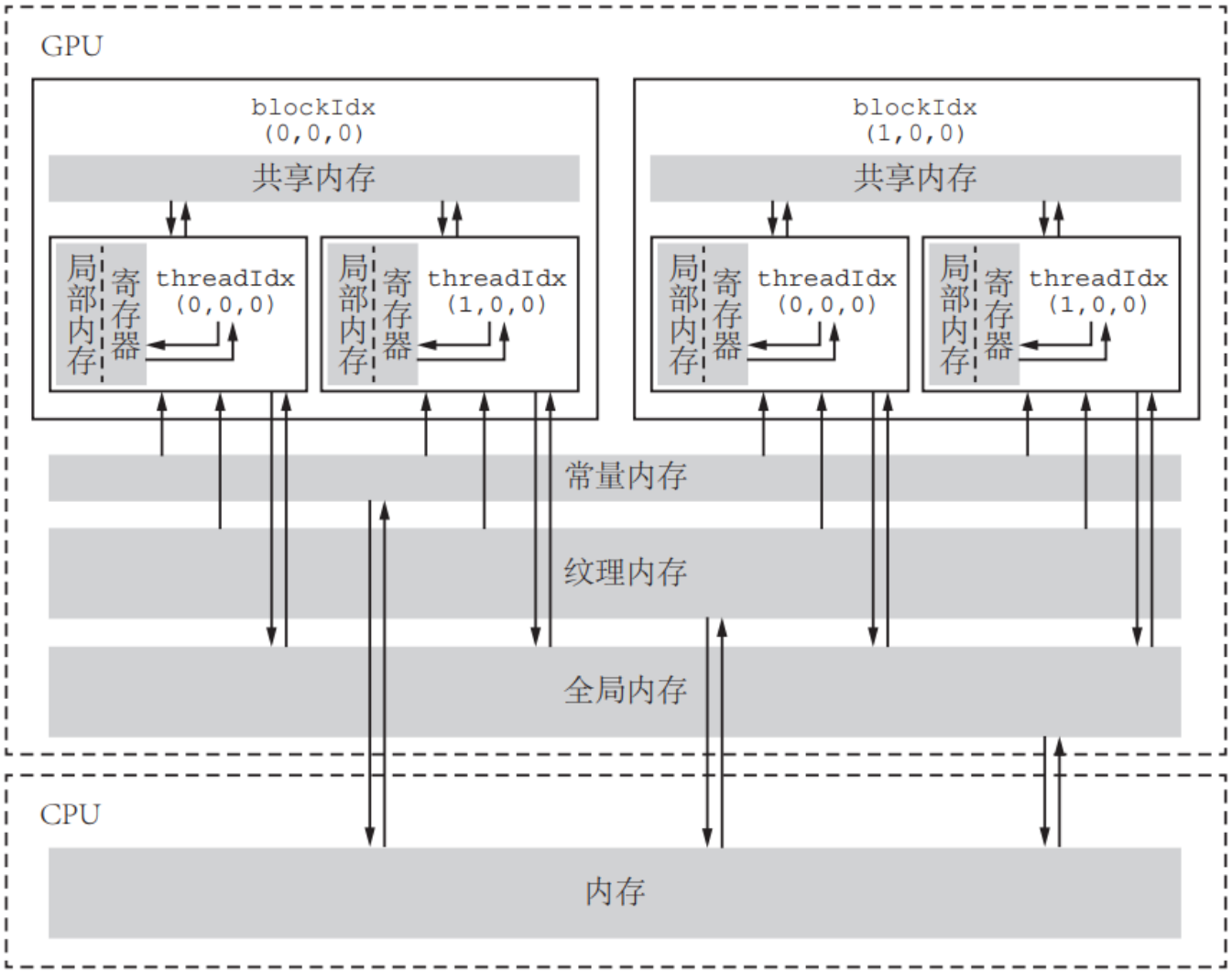
CUDA并行编程系列课程

主讲：权双

CONTENTS



寄存器



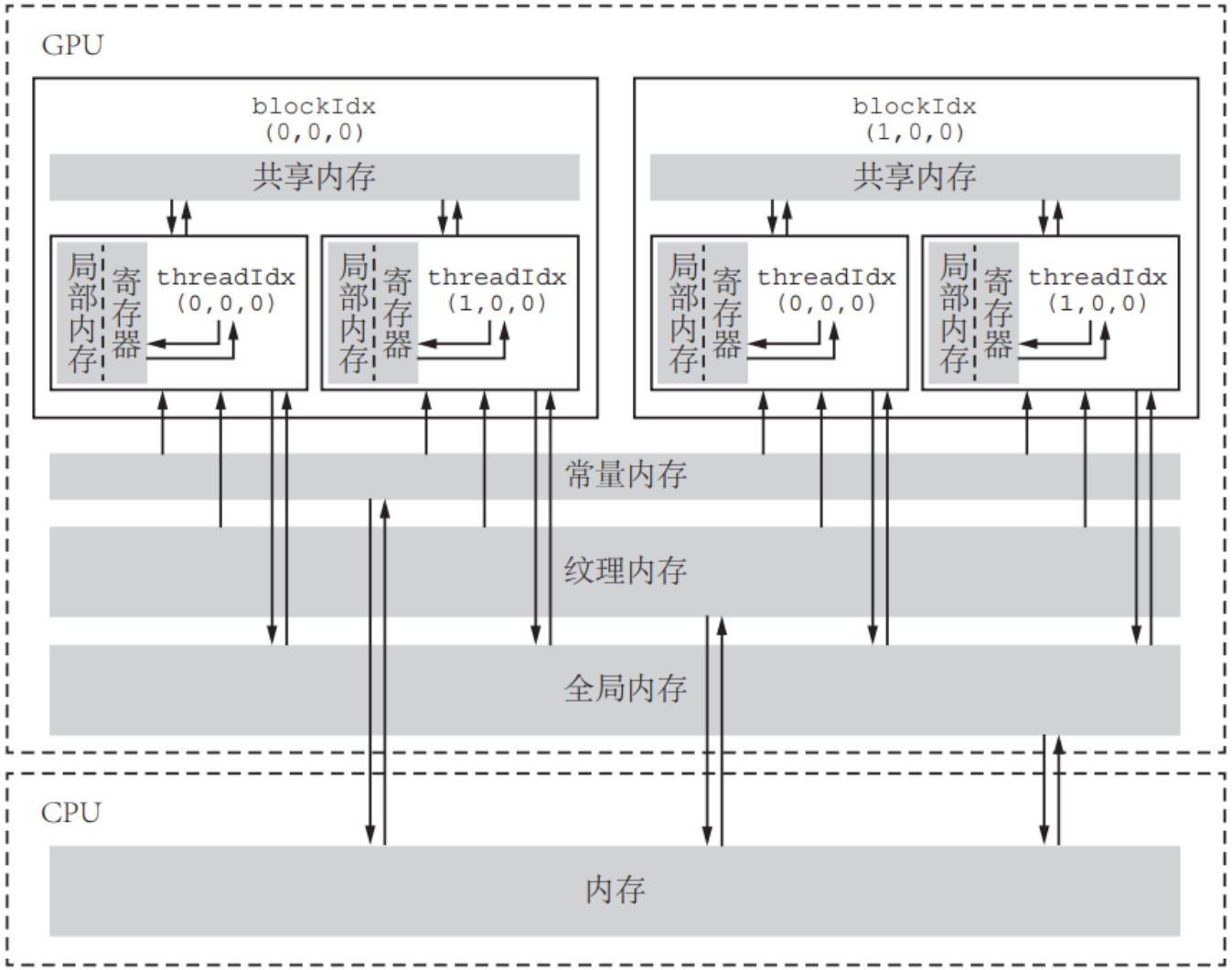
- 寄存器内存在片上（on-chip），具有GPU上最快的访问速度，但是数量有限，属于GPU的稀缺资源；
- 寄存器仅可在线程内可见，生命周期也与所属线程一致；
- 核函数中定义的不加任何限定符的变量一般存放在寄存器中；
- 内建变量存放于寄存器中，如gridDim、blockDim、blockIdx等；
- 核函数中定义的不加任何限定符的数组有可能存在于寄存器中，但也有可能存在于本地内存中；

寄存器

	Compute Capability													
Technical Specifications	5.0	5.2	5.3	6.0	6.1	6.2	7.0	7.2	7.5	8.0	8.6	8.7	8.9	9.0
Number of 32-bit registers per SM	64 K													
Maximum number of 32-bit registers per thread block	64 K		32 K	64 K		32 K	64 K							
Maximum number of 32-bit registers per thread	255													

- ★ 寄存器都是32位的，保存1个double类型的数据需要两个寄存器，寄存器保存在SM的寄存器文件；
- ★ 计算能力5.0~9.0的GPU，每个SM中都是64K的寄存器数量，Fermi架构只有32K；
- ★ 每个线程块使用的最大数量不同架构是不同的，计算能力6.1是64K；
- ★ 每个线程的最大寄存器数量是255个，Fermi架构是63个；

本地内存



★ 寄存器放不下的内存会存放在本地内存：

1、索引值不能在编译时确定的数组存放于本地内存；

2、可能占用大量寄存器空间的较大本地结构体和数组；

3、任何不满足核函数寄存器限定条件的变量。

本地内存

★ 每个线程最多高达可使用512KB的本地内存

★ 本地内存从硬件角度看只是全局内存的一部分，延迟也很高，本地内存的过多使用，会减低程序的性能。

★ 对于计算能力2.0以上的设备，本地内存的数据存储在每个SM的一级缓存和设备的二级缓存中

	Compute Capability													
Technical Specifications	5.0	5.2	5.3	6.0	6.1	6.2	7.0	7.2	7.5	8.0	8.6	8.7	8.9	9.0
Maximum amount of local memory per thread	512 KB													

寄存器溢出

★ 核函数所需的寄存器数量超出硬件设备支持，数据则会保存到本地内存（local memory）中：

1、一个SM运行并行运行多个线程块/线程束，总的需求寄存器容量大于64KB；

2、单个线程运行所需寄存器数量个255个；

★ 寄存器溢出会降低程序运行性能：

1、本地内存只是全局内存的一部分，延迟较高；

2、寄存器溢出的部分也可进入GPU的缓存中；

THANKS

谢谢聆听