CUDA编程模型

编程结构

- Host: CPU和CPU内存
- Device: GPU和GPU内存
- Unified memory (CUDA 6)

内存管理

- cudaMalloc
- cudaFree
- cudaMemcpy(阻塞CPU线程)
- cudaMemset

线程组织

- Grid: 一个kernel启动的所有线程，共享global memory，以多个block的形式组织起来
- Block: 一组线程，可以通过同步或者shared memory进行合作

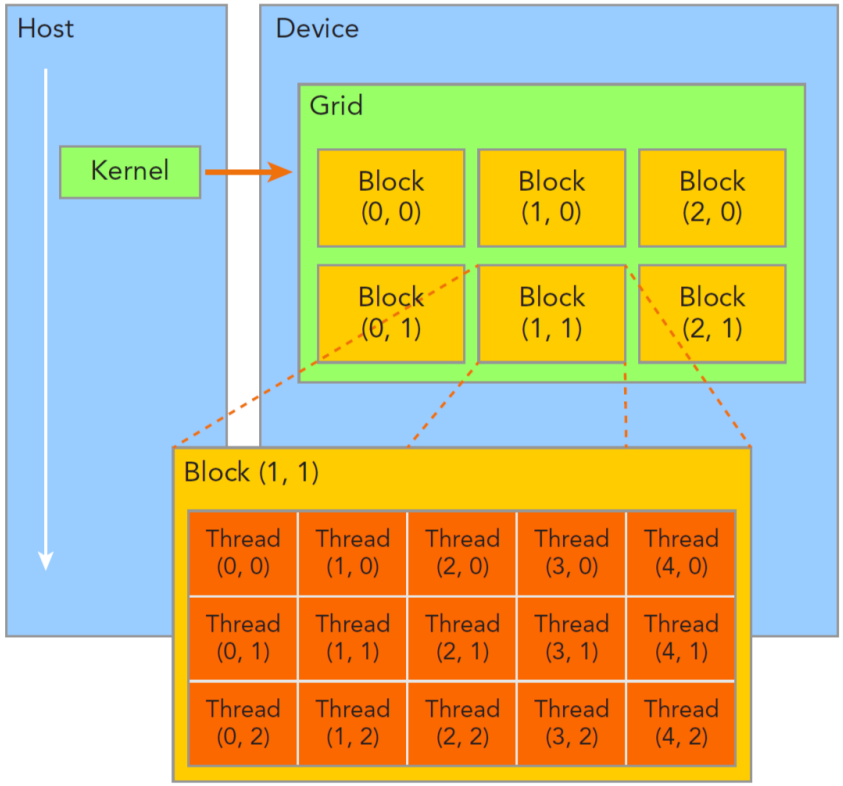

FIGURE 2-5

调用kernel函数

- kernel_name<<<grid,block>>>(argument list)
- 与CPU线程异步
  - cudaDeviceSynchronize()

函数限定符

- __global__
  - Device上执行，由host或device调用，返回void类型
- __device__
  - Device上执行，由device调用
- __host__
  - 可省略，host上执行，由host调用

CUDA API错误处理

- 返回类型cudaError_t
  - cudaSuccess, cudaErrorMemoryAllocation…
- char* cudaGetErrorString(cudaError_t error);

性能分析

- $ nvprof [nvprof_args] <application> [application args]