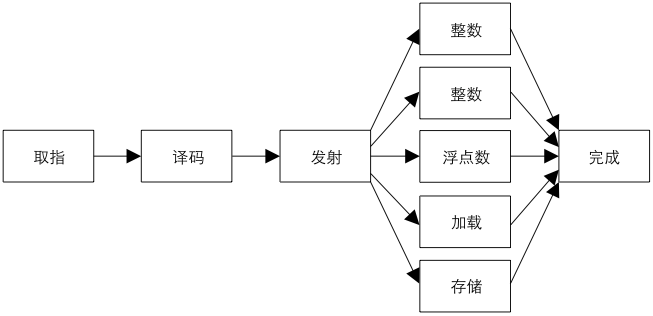
**一、指令集并行**

超标量处理器：同时可发射2至6条指令



**二、片内多线程**

可共享系统资源(缓存、内存、I/O)的轻量级进程称作线程

**1实现背景**

·多进程并发程序的调度(PC、寄存器、中断控制器等资源的交换)，造成流水线暂停

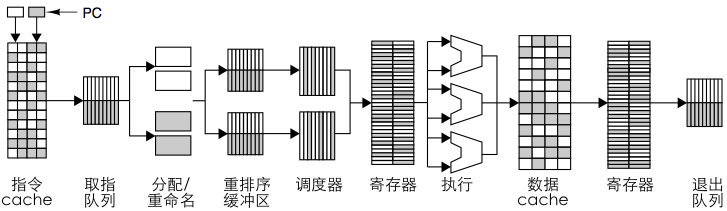
单核单线程，一线程调出后，流水线暂停，直到另一进程调入

单核双线程，一线程调出后，另一线程继续保持流水线满负荷执行

·多进程并发程序的资源(缓冲、内存、I/O等)相同，导致资源浪费

**2片内多线程的设计**

设计：扩展物理内核，使其可模拟两个逻辑内核(并发执行两线程)



逻辑内核的实现：

·双倍的资源：两逻辑内核始终使用的资源，用双倍资源避免并发时产生调度

PC计数器、相关专用寄存器、中断控制器等

·共享的资源：逻辑内核共享的资源，如cache、通用寄存器等

·平分独占的资源：逻辑内核使用1/2的物理核心资源，如流水线上各种队列

·动态独占的资源：使用阀值(最大100%)的物理核心资源

调度器，保证每个线程都能发射指令到执行单元

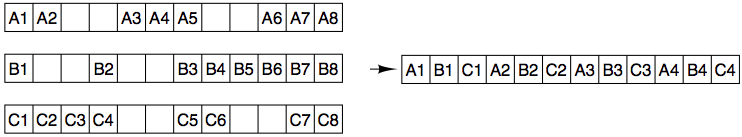
**3细颗粒度多线程调度**

指令被延迟的周期：

·cache调页时的延迟周期、

·条件转移排空流水线的延迟周期

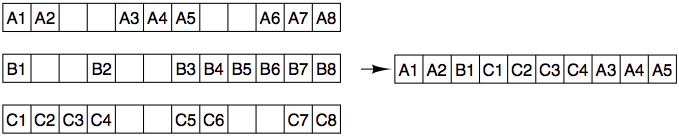
实现：单核心提供上述最大延迟周期的并发线程，通过循环调度线程来屏蔽上述延迟



问题：单核心提供的线程数由硬件确定，通常为2个

**4粗颗粒度多线程调度**

实现：在译码时若遇可导致暂停的指令(load、store、branch)，立即进行调度



区别细颗粒度：解决了细颗粒度单核心需支持多线程的情况

**5并发多线程**

实现：粗颗粒度线程调度运行在超标量计算机上

**三、单片多核心**



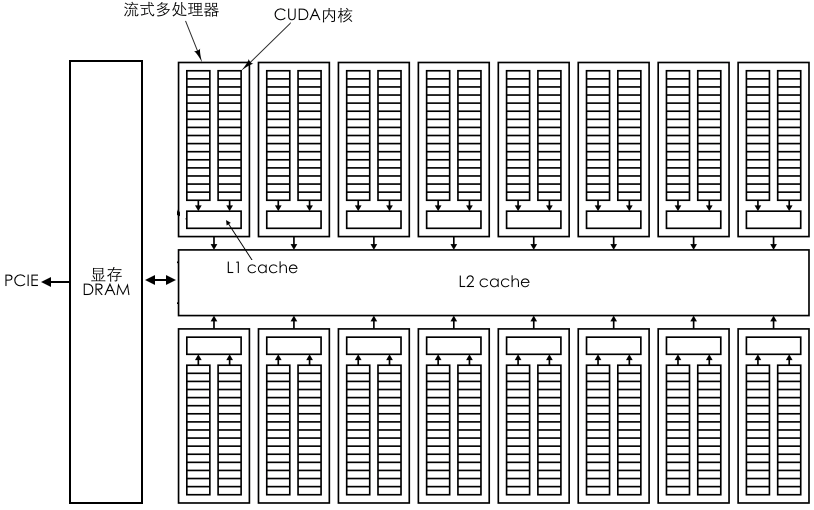
L2、L3采用环形网络交互。L2调页的数据可能来自其它L2或L3

**四、协助处理器**

**1网络处理器**

略

**2图形处理器**



流式处理器(SM)：每周期译码一条指令，并由32个CUDA执行(指令相同操作数不同)

单指令多数据(SIMD)就是形容SM的特性

CUDA内核：支持单精度整数和浮点数计算的简单处理器

GPU体系：包含16个SM，因此具有同时执行16个操作(16个不同指令)的能力

**五、共享内存的多处理器**

作用和效果与单片多核心类似

