

## 第十一章作业

钟赟 2016K8009915009

### 1、关于多核处理器的 cache 结构。请介绍 UCA 和 NUCA 的特点。

UCA：集中式共享结构，多个处理器核通过总线或交叉开关连接二级 Cache，所有处理器核对二级 Cache 的访问延迟相同，可扩展性有限，通常在核数较小的多核处理器中使用。

NUCA：分布式共享结构，每个处理器拥有本地的二级 Cache，通过片上互连访问其他处理器核的二级 Cache，处理器核访问本地 Cache 和远程 Cache 的延迟不同，具有良好的可扩展性，通常在核数较多的处理器中采用。

### 2、有两个并行执行的线程，在顺序一致性和弱一致性下，它各有几种正确的执行顺序？给出执行次序和最后的正确结果（假设 X、Y 的初始值均为 0）。

P1 X = 1;(1) print Y;(2)	P2 Y = 1;(3) print X;(4)
--------------------------------	--------------------------------

顺序一致性：在多处理机环境下的一个并行执行的结果等于同一程序在单处理机多进程环境下的一个执行的结果。

弱一致性：把同步操作和普通访存操作区分开来，只在同步点维护一致性由上述概念，鉴于 P1 和 P2 没有同步操作，为普通访存，只需要维护语句(1)。

在(2)之前执行，语句(3)在(4)之前执行，可能的执行顺序和对应的结果如下表：

执行顺序	打印结果
(1)(2)(3)(4)	Y=0 X=1
(1)(3)(2)(4)	Y=1 X=1
(1)(3)(4)(2)	X=1 Y=1
(3)(1)(2)(4)	Y=1 X=1
(3)(1)(4)(2)	X=1 Y=1
(3)(4)(1)(2)	X=0 Y=1

### 3、关于 cache 一致性协议，MESI 协议比 ESI 协议增加了 M 状态，请解释有什么好处。

M 状态表示当前 Cache 行被当前处理器核独占并且已经修改过，在处理器核替换该 Cache 行时需将该 Cache 行写回内存。

好处：相对于 ESI 一致性协议，仅需要写回 M 状态的 Cache 行，减少 Cache 到内存的写回的数据传输次数，故减少 Cache 到内存的访存量。

### 4、请分别采用 Fetch\_and\_Increment 和 Compare\_and\_Swap 原子指令编写实现自旋锁的代码，并分析可能的性能改进措施。

---

Fetch\_and\_Increment:

```
void lock_acquire(){
    while((l=Fetch_and_Increment(lock))!=lock);
    critical_section();
    ++lock;
}

void lock_release(){
    lock = 0;
}
```

Compare\_and\_Swap:

```
void lock_acquire(){
    while(Compare_and_Swap(lock, 0, 1)!=0);
    critical_section();
}

void lock_release(){
    lock = 0;
}
```

性能改进：降低循环访问共享变量的次数，可在循环中加入一定的延迟，减轻访存压力。

**5、在共享存储的多处理器中，经常会出现假共享现象。假共享是由于两个变量处于同一个 cache 行中引起的，会对性能造成损失。为了尽量减少假共享的发生，程序员在写程序时应该注意什么？**

- 1) 有的假共享可以由编译器优化避免；
- 2) 在线程访问全局或动态分配共享数据结构中注意变量放置位置，尽量放在不同缓存行中；
- 3) 也可以通过将需要使用的数据拷贝到线程本地，减少发生假共享的频率。

Source Web: <https://software.intel.com/zh-cn/articles/avoiding-and-identifying-false-sharing-among-threads/>

**6、请介绍片上网络路由器设计中的虚通道概念，并说明采用虚通道有什么好处。**

虚通道（VP）是一种通信电路，它可以在两个或多个端点之间传送 ATM 信元。端点之间的虚通道可以是用户到用户的链接，也可以是用户到网络的连接，还可以网强到网络的连接。虚通道可以用来分配流控单元在缓冲队列中的位置。

好处：在一条链路上可以建立多个虚通道，每条虚通路上传输的数据单元均在相同的物理线路上传输，且维持先后顺序，克服了分组交换中无序接收的缺点，保证了数据的连续性，更适合于多媒体数据的传输。

Source Web:<https://baike.baidu.com/item/虚通道连接/>

## 7、分析 Fermi GPU 的存储结构，指出不同层次存储结构的带宽、延迟，以及是否共享。

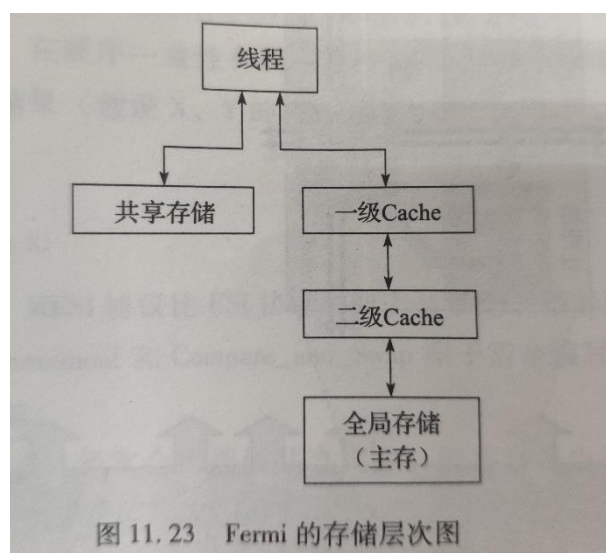
Fermi GPU 的存储层次由每个 SM 的寄存器堆、每个 SM 的一级 Cache、统一的二级 Cache 和全局存储组成，图为 Fermi GPU 的存储层次示意图。

1) 寄存器。每个 SM 有 32K 个 32 位寄存器，每个线程可以访问自己的私有寄存器，随线程的数目的不同，每个线程可访问的私有寄存器的数目在 21~63 之间变化。

2) 一级 Cache 和共享存储。每个 SM 有片上高速存储，主要用来缓存单线程的数据或者用于多线程间的数据共享，可以在一级 Cache 和共享存储之间进行配置。

3) 二级 Cache。768KB 统一的二级 cache 在 16 个 SM 之间共享，服务于全局内存中的 load/store 操作。

4) 全局存储。所有线程共享的片外存储。



Fermi 体系结构可以看作 GPU 和 CPU 融合的架构。以采用 Fermi 体系结构第 Ge-Force GTX 480 为例，它包含 480 主核，主频为 700MHz，单精度浮点峰值性能为 1.536TFLOPS，访存带宽为 177.4GB/s。