

Homework 11 — December 4

Lecturer: Hu Weiwu

Completed by: 2022K8009929010 Zhang Jiawei

11.1

UCA 是一种集中式共享结构, 多个处理器核通过总线或者交叉开关连接 LLC, 所有处理器核到 LLC 的访问延迟相同。作为集中式的共享 LLC, 很容易随着处理器核数目的增加成为瓶颈。另外, UCA 结构由于使用总线或者交叉开关互连, 可扩展性受限。

NUCA 是一种分布式共享结构, 每个处理器核拥有本地的 LLC, 并通过片上互连访问其他处理器核的 LLC。在 NUCA 结构中, 处理器核可以访问所有的 LLC, 但是不同位置的 LLC 具有不同的访问延迟。当工作集较小时, 处理器核的本地 Cache 足够容纳工作集, 处理器核只使用本地 Cache; 当工作集较大时, 本地 Cache 中放不下的数据可以放到远地 Cache 中。NUCA 结构需要高效 Cache 查找和替换算法, 使得在使用远地 Cache 时不影响性能。NUCA 结构中通常采用可扩展的片上互连 (如 Mesh 片上网络等), 采用基于目录的 Cache 一致性协议, 具有良好的可扩展性, 可以有效支持较多数目的处理器核。

11.2

对两个线程的内容进行编号:

```
1  # P1
2  X = 1; # 1
3  print Y; # 2
4
5  # P2
6  Y = 1; # 3
7  print X; # 4
```

在顺序一致性下, 两个线程的执行顺序有以下几种:

- 1-2-3-4, 此时输出为 0-1
- 3-4-1-2, 此时输出为 0-1
- 1-3-2-4, 此时输出为 1-1
- 1-3-4-2, 此时输出为 1-1
- 3-1-2-4, 此时输出为 1-1
- 3-1-4-2, 此时输出为 1-1

在弱一致性下, 两个线程中的指令可以乱序执行, 总共有 $4! = 24$ 种执行顺序, 除以上 6 种外, 还有其他 18 种执行顺序:

- 1-2-4-3, 此时输出为 1-0

- 1-4-2-3, 此时输出为 1-0
- 1-4-3-2, 此时输出为 1-1
- 2-1-3-4, 此时输出为 0-1
- 2-1-4-3, 此时输出为 0-1
- 2-3-1-4, 此时输出为 0-1
- 2-3-4-1, 此时输出为 0-0
- 2-4-1-3, 此时输出为 0-0
- 2-4-3-1, 此时输出为 0-0
- 3-2-1-4, 此时输出为 1-1
- 3-2-4-1, 此时输出为 1-0
- 3-4-2-1, 此时输出为 0-1
- 4-1-2-3, 此时输出为 0-0
- 4-1-3-2, 此时输出为 0-1
- 4-2-1-3, 此时输出为 0-0
- 4-2-3-1, 此时输出为 0-0
- 4-3-1-2, 此时输出为 0-1
- 4-3-2-1, 此时输出为 0-1

11.3

与 ESI 协议相比, 增加一个 Modified 状态的优点是减少了 Cache 到内存的数据传输次数, Cache 只需要将 Modified 状态的 Cache 行写回内存。

11.4

使用 Fetch_and_Increment 指令实现自旋锁的伪代码如下:

```
1  int lock = 0, i = 0;
2  void lock(int *lock) {
3      while (lock != i);
4      Fetch_and_Increment(&lock);
5      critical_section();
6  }
7
8  void unlock() {
9      i++;
10 }
```

使用 Compare_and_Swap 指令实现自旋锁的伪代码如下：

```
1  int lock = 0, i = 0;
2  void lock(int *lock) {
3      while (Compare_and_Swap(&lock, 0, 1) != 0);
4      critical_section();
5  }
6
7  void unlock() {
8      lock = 0;
9  }
```

可能的性能改进措施包括引入短暂的延迟来减少忙等待的 CPU 开销、改用队列锁、添加超时机制避免无线自旋等。

11.5

程序员应该确保不同线程访问的数据结构在内存中对齐到不同的缓存行, 以减少伪共享的影响; 或者在共享数据结构中插入填充(padding)字段, 使得不同线程访问的数据位于不同的缓存行。

11.6

虚通道是片上网络路由器中一种逻辑上的数据通路管理机制。它通过在一个物理通道上划分多个逻辑通道, 允许多个数据流共享同一个物理链路, 而不会因为竞争而发生死锁或性能下降。其主要好处包括: 防止死锁、提高网络吞吐、减少头部阻塞、增强网络的公平性、支持多种路由策略。

11.7

Femi GPU 的存储层次图如下：

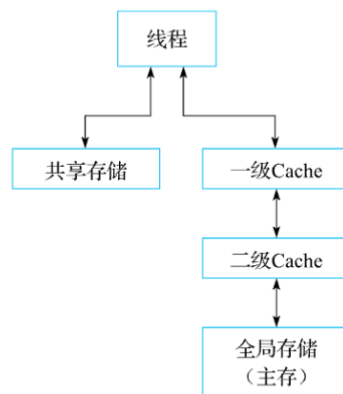


图 11.1. Femi GPU 存储层次图

寄存器文件带宽极高、延迟极低、每个线程私有; 共享内存/一级 cache 带宽高、延迟低、所有线程共享; 二级 cache 带宽中等、延迟中等、所有线程共享; 全局内存带宽较低、延迟高、所有线程共享。