

典型多核处理器的组织结构，其中微处理器芯片有4个CPU核，每个核都有自己的L1和L2高速缓存，其中的L1高速缓存分为两个部分——一个保存最近取到的指令，另一个存放数据。这些核共享更高层次的高速缓存，以及到主存的接口。工业界的专家预言他们能够将几十个、最终会是上百个核做到一个芯片上。

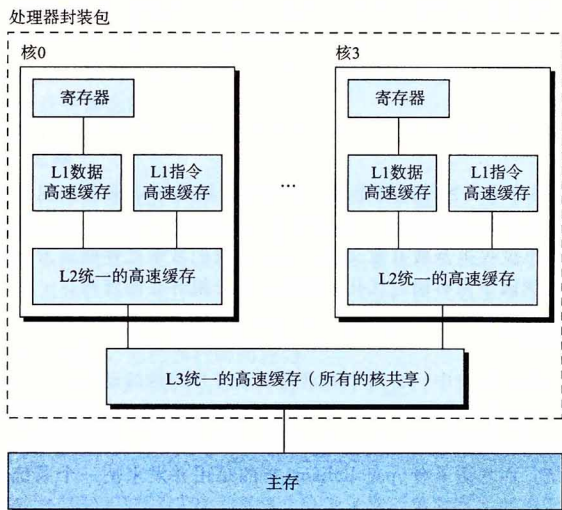


图 1-17 多核处理器的组织结构。4个处理器核集成在一个芯片上

超线程，有时称为同时多线程(simultaneous multi-threading)，是一项允许一个CPU执行多个控制流的技术。它涉及CPU某些硬件有多个备份，比如程序计数器 and 寄存器文件，而其他的硬件部分只有一份，比如执行浮点算术运算的单元。常规的处理器需要大约20 000个时钟周期做不同线程间的转换，而超线程的处理器可以在单个周期的基础上决定要执行哪一个线程。这使得CPU能够更好地利用它的处理资源。比如，假设一个线程必须等到某些数据被装载到高速缓存中，那CPU就可以继续去执行另一个线程。举例来说，Intel Core i7处理器可以让每个核执行两个线程，所以一个4核的系统实际上可以并行地执行8个线程。

多处理器的使用可以从两方面提高系统性能。首先，它减少了在执行多个任务时模拟并发的需要。正如前面提到的，即使是只有一个用户使用的个人计算机也需要并发地执行多个活动。其次，它可以使应用程序运行得更快，当然，这必须要求程序是以多线程方式来书写的，这些线程可以并行地高效执行。因此，虽然并发原理的形成和研究已经超过50年的时间了，但是多核和超线程系统的出现才极大地激发了一种愿望，即找到书写应用程序的方法利用硬件开发线程级并行性。第12章会更深入地探讨并发，以及使用并发来提供处理器资源的共享，使程序的执行允许有更多的并行。

## 2. 指令级并行

在较低的抽象层次上，现代处理器可以同时执行多条指令的属性称为指令级并行。早期的微处理器，如1978年的Intel 8086，需要多个(通常是3~10个)时钟周期来执行一条指令。最近的处理器可以保持每个时钟周期2~4条指令的执行速率。其实每条指令从开