

在有些情况中,被引用的存储器位置实际上是存储在磁盘存储器上的。此时,硬件会产生一个缺页(page fault)异常信号。同其他异常一样,这个异常会导致处理器调用操作系统的异常处理程序代码。然后这段代码会发起一个从磁盘到主存的传送操作。一旦完成,操作系统会返回到原来的程序,而导致缺页的指令会被重新执行。这次,存储器引用将成功,虽然可能会导致高速缓存不命中。让硬件调用操作系统例程,然后操作系统例程又会将控制返回给硬件,这就使得硬件和系统软件在处理缺页时能协同工作。因为访问磁盘需要数百万个时钟周期,OS缺页中断处理程序执行的处理所需的几百个时钟周期对性能的影响可以忽略不计。

从处理器的角度来看,将用暂停来处理短时间的高速缓存不命中和用异常处理来处理长时间的缺页结合起来,能够顾及到存储器访问时由于存储器层次结构引起的所有不可预测性。

### 旁注 当前的微处理器设计

一个五阶段流水线,例如已经讲过的 PIPE 处理器,代表了 20 世纪 80 年代中期的处理器设计水平。Berkeley 的 Patterson 研究组开发的 RISC 处理器原型是第一个 SPARC 处理器的基础,它是 Sun Microsystems 在 1987 年开发的。Stanford 的 Hennessy 研究组开发的处理器由 MIPS Technologies(一个由 Hennessy 成立的公司)在 1986 年商业化了。这两种处理器都使用的是五阶段流水线。Intel 的 i486 处理器用的也是五阶段流水线,只不过阶段之间的职责划分不太一样,它有两个译码阶段和一个合并的执行/访存阶段[27]。

这些流水线化的设计的吞吐量都限制在最多一个时钟周期一条指令。4.5.9 小节中描述的 CPI(Cycles Per Instruction, 每指令周期)测量值不可能小于 1.0。不同的阶段一次只能处理一条指令。较新的处理器支持超标量(superscalar)操作,意味着它们通过并行地取指、译码和执行多条指令,可以实现小于 1.0 的 CPI。当超标量处理器已经广泛使用时,性能测量标准已经从 CPI 转化成了它的倒数——每周期执行指令的平均数,即 IPC。对超标量处理器来说,IPC 可以大于 1.0。最先进的设计使用了一种称为乱序(out-of-order)执行的技术来并行地执行多条指令,执行的顺序也可能完全不同于它们在程序中出现的顺序,但是保留了顺序 ISA 模型蕴含的整体行为。作为对程序优化的讨论的一部分,我们将会在第 5 章中讨论这种形式的执行。

不过,流水线化的处理器并不只有传统的用途。现在出售的大部分处理器都用在嵌入式系统中,控制着汽车运行、消费产品,以及其他一些系统用户不能直接看到处理器的设备。在这些应用中,与性能较高的模型相比,流水线化的处理器的简单性(比如说像我们在本章中讨论的这样)会降低成本和功耗需求。

最近,随着多核处理器受到追捧,有些人声称通过在一个芯片上集成许多简单的处理器,比使用少量更复杂的处理器能获得更多的整体计算能力。这种策略有时被称为“多核”处理器[10]。

## 4.6 小结

我们已经看到,指令集体系结构,即 ISA,在处理器行为(就指令集合及其编码而言)和如何实现处理器之间提供了一层抽象。ISA 提供了程序执行的一种顺序说明,也就是一条指令执行完了,下一条指令才会开始。

从 IA32 指令开始,大大简化数据类型、地址模式和指令编码,我们定义了 Y86-64 指令集。得到的