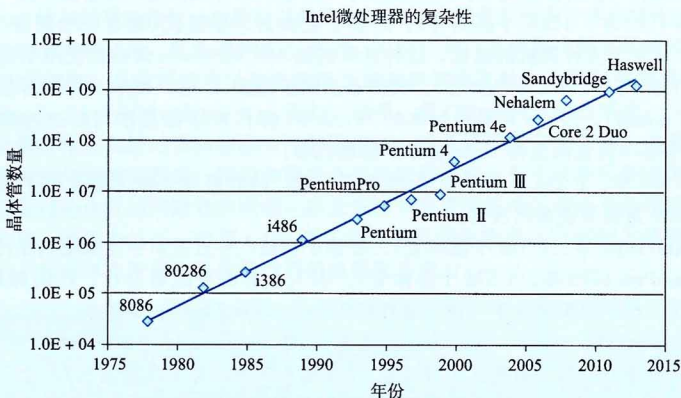


指令和指令格式。

每个后继处理器的设计都是后向兼容的——较早版本上编译的代码可以在较新的处理器上运行。正如我们看到的那样，为了保持这种进化传统，指令集中有许多非常奇怪的东西。Intel 处理器系列有好几个名字，包括 IA32，也就是“Intel 32 位体系结构 (Intel Architecture 32-bit)”，以及最新的 Intel64，即 IA32 的 64 位扩展，我们也称为 x86-64。最常用的名字是“x86”，我们用它指代整个系列，也反映了直到 i486 处理器命名的惯例。

#### 旁注 摩尔定律 (Moore's Law)

如果我们画出各种不同的 Intel 处理器中晶体管的数量与它们出现的年份之间的图 (y 轴为晶体管数量的对数值)，我们能够看出，增长是很显著的。画一条拟合这些数据的线，可以看到晶体管数量以每年大约 37% 的速率增加，也就是说，晶体管数量每 26 个月就会翻一番。在 x86 微处理器的历史上，这种增长已经持续了好几十年。



1965 年，Gordon Moore，Intel 公司的创始人，根据当时的芯片技术（那时他们能够在一个芯片上制造有大约 64 个晶体管的电路）做出推断，预测在未来 10 年，芯片上的晶体管数量每年都会翻一番。这个预测就称为摩尔定律。正如事实证明的那样，他的预测有点乐观，而且短视。在超过 50 年中，半导体工业一直能够使得晶体管数目每 18 个月翻一倍。

对计算机技术的其他方面，也有类似的呈指数增长的情况出现，比如磁盘和半导体存储器的存储容量。这些惊人的增长速度一直是计算机革命的主要驱动力。

这些年来，许多公司生产出了与 Intel 处理器兼容的处理器，能够运行完全相同的机器级程序。其中，领头的是 AMD。数年来，AMD 在技术上紧跟 Intel，执行的市场策略是：生产性能稍低但是价格更便宜的处理器。2002 年，AMD 的处理器变得更加有竞争力，它们率先突破了可商用微处理器的 1GHz 的时钟速度屏障，并且引入了广泛采用的 IA32 的 64 位扩展 x86-64。虽然我们讲的是 Intel 处理器，但是对于其竞争对手生产的与之兼容的处理器来说，这些表述也同样成立。

对于由 GCC 编译器产生的、在 Linux 操作系统平台上运行的程序，感兴趣的人大多并不关心 x86 的复杂性。最初的 8086 提供的内存模型和它在 80286 中的扩展，到 i386 的时候就已经过时了。原来的 x87 浮点指令到引入 SSE2 以后就过时了。虽然在 x86-64 程序中，我们能看到历史发展的痕迹，但 x86 中许多最晦涩难懂的特性已经不会出现了。