

其他部分略，这里主要讲性能那部分

1.性能的定义与度量

性能=1/执行时间（只是用来比较两个计算机的性能）

时钟周期=1/时钟频率

2.CPU 的性能及其因素

性能计算

一个程序的 CPU 执行时间=一个程序的 CPU 时钟周期数*时钟周期时间=一个程序的 CPU 时钟周期数/时钟频率

CPU 时钟周期数=程序的指令数*每条指令的平均时钟周期

CPI：执行每条指令所需的时钟周期数的平均值（注意是时钟周期数，是数）

总的时钟周期数=指令数*CPI

现在我们可以用指令数（instruction count）、CPI 和时钟周期时间来写出基本的性能公式：

CPU 时间 = 指令数 × CPI × 时钟周期时间

或

CPU 时间 = 指令数 × CPI / 时钟频率

这些公式特别有用，因为它们把性能分解为三个关键因素。如果知道实现方案或替代方案如何影响这三个参数，我们可用这些公式来比较不同的实现方案或评估某个设计的替代方案。

☞ 指令数：执行某程序所需的总指令数量。

执行时间 = 秒/程序 = 指令数/程序 × 时钟周期数/指令 × 秒/时钟周期

永远记住，唯一能够被完全可靠测量的计算机性能指标是时间。例如，对指令集减少指令数目的改进可能降低时钟周期时间或提高 CPI，从而抵消了改进的效果。类似地，CPI 与执行的指令类型相关，执行指令数最少的代码其执行速度未必是最快的。

性能的分量	测量单位
程序的CPU执行时间	程序执行的执行时间，以秒为单位
指令数目	程序执行的指令数目
指令的平均执行时钟周期(CPI)	每条指令的平均执行的时钟周期数
时钟周期时间	每个时钟周期的长度，以秒为单位

影响因素

01 理解程序性能 程序的性能与算法、编程语言、编译程序、体系结构以及实际的硬件有关。下表概括了这些成分是如何影响 CPU 性能公式中的各种因素的。

硬件或软件指标	影响什么	如何影响
算法	指令数，可能的 CPI	算法决定源程序执行指令的数目，从而也决定了 CPU 执行指令的数目。算法也可能通过使用较快或较慢的指令影响 CPI。例如，当算法使用更多的除法运算时，将会导致 CPI 增大
编程语言	指令数，CPI	编程语言显然会影响指令数，因为编程语言中的语句必须翻译为指令，从而决定了指令数。编程语言也可影响 CPI，例如，Java 语言充分支持数据抽象，因此将进行间接调用，需要使用较高的 CPI 指令
编译程序	指令数，CPI	因为编译程序决定了源程序到计算机指令的翻译过程，所以编译程序的效率既影响指令数又影响 CPI。编译器会以复杂的方式影响 CPI
指令集体系结构	指令数，CPI 时钟频率	指令集体系结构影响 CPU 性能的所有 3 个方面，因为它影响完成某功能所需的指令数、每条指令的周期数以及处理器的时钟频率