

# 指令1

- 1. CISC指复杂指令集，有丰富的指令并有专用指令处理特殊任务（乘法、浮点计算等），在这些特殊任务上性能较高，指令也较为简单，但为实现复杂指令往往需要更多晶体管、更复杂的电路和更长的指令周期
- 2. RISC指精简指令集，指令较少且每个指令只执行一项操作，并且执行速度更快，但对于复杂指令则需要多个简单指令组合执行，效率可能较低，通常需要少量晶体管、更简单的电路和较短的指令周期。总的来说，CISC架构通常用于桌面和服务器的计算机，而RISC架构通常用于嵌入式系统和移动设备等资源有限的系统。
- 3. 执行时间=时钟周期数x时钟周期，而时钟周期数=程序指令数x每条指令平均周期数（CPI），时钟周期=1/CPU主频

4.

影响因素	影响	如何影响
算法	指令数、CPI	算法决定了源程序执行指令的数目，从而决定了CPU执行指令的数目。算法通过选用较快或者较慢的指令影响CPI。例如，当算法使用较多的除法运算时，会导致CPI增大
程序设计语言	指令数、CPI	编程语言的语句需要翻译为指令，因此不同的编程语言会影响指令数。编程语言影响CPI，例如Java语言充分支持数据抽象，因此在间接调用时，会使用CPI较高的指令
编译器	指令数、CPI	编译器决定了源程序到计算机指令是如何翻译的。不同的编译器会导致翻译后的指令数不同。编译器影响CPI的方式比较复杂
指令集体系架构	指令数、CPI、时钟周期	指令集体系架构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
硬件实现	CPI、时钟周期	处理器时钟周期与硬件的具体实现相关。CPI的计算公式中包括每个指令所需要的周期，因此也与硬件实现有关。

- 5.
- 1.  $CPI = (500 \times 10 + 100 \times 3 + 800 \times 1) / (800 + 500 + 100) = 4.36$
  - 2. 减少20%算术指令相当于减少  $(800 \times 0.2 \times 1) / (500 \times 10 + 100 \times 3 + 800 \times 1) = 0.6\%$  的执行时间  
而  $99.4\% / 90\% > 1$ ，即增加后CPU处理时间变长，不该增加这些指令
  - 3. 加速为2倍，则该指令  $CPI = 5$ ， $CPI' = (500 \times 5 + 100 \times 3 + 800 \times 1) / (800 + 500 + 100) = 3600$   
 $CPI / CPI' = 1.6944$ ，即速度变为原来的1.6944倍；若加速10倍，则该指令  $CPI = 1$   
 $CPI'' = (500 \times 1 + 100 \times 3 + 800 \times 1) / (800 + 500 + 100) = 1600$ ， $CPI / CPI'' = 3.8125$ ，速度变为原来的3.8125倍

