

全国计算机等级考试四级复习纲要七[3] PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/138/2021_2022__E5_85_A8_E5_9B_BD_E8_AE_A1_E7_c98_138620.htm 1.指令集发展的两个

途径（1）CISC途径CISC是Complex Instruction Set Computer（复杂指令集计算机）的缩写。它的基本思想是:进一步增强原有指令的功能，用更为复杂的新指令取代原先由软件子程序完成的功能，实现软件功能的硬化。这是一种传统的发展方向，早在50年代就已采用。这种途径必然导致机器的指令系统越来越庞大而复杂。事实上，目前使用的绝大多数计算机都属于CISC类型。（2）RISC途径RISC是Reduced

Instruction Set Computer（精简指令集计算机）的缩写。其基本思想是:通过减少指令总数和简化指令功能，降低硬件设计的复杂度，使指令能单周期执行，并通过优化编译，提高指令的执行速度。这是一种革新的发展方向，在70年代末才开始兴起。这种途径必然导致机器的指令系统进一步精炼而简单。

2.面向目标代码的优化 为了提高目标程序的实现效率，人们对大量的机器语言目标代码及其执行情况进行了统计。对程序中出现的各种指令以及指令串进行统计得到的百分比称为静态使用频度。在程序执行过程中对出现的各种指令以及指令串进行统计得到的百分比称为动态使用频度。按静态使用频度来改进目标代码可减少目标程序所占的存储空间。按动态使用频度来改进目标代码可减少目标程序运行的执行时间。大量统计表明，动态和静态使用频度两者非常接近，最常用的指令是存、取、条件转移等。对它们加以优化，既可以减少程序所需的存储空间，又可以提高程序的执行速度

。 3.面向高级语言的优化 面向高级语言的优化就是尽可能缩小高级语言与机器语言之间的语义差距，以利于支持高级语言编译系统，缩短编译程序的长度和编译所需的时间。 4.面向操作系统的优化 面向操作系统的优化就是进一步缩小操作系统与体系结构之间的语义差距，以利于减少操作系统运行所需的辅助时间，节省操作系统软件所占用的存储空间。操作系统的实现依赖于体系结构对它的支持。许多传统机器指令例如算术逻辑指令、字符编辑指令、移位指令、控制转移指令等，都可用于操作系统的实现。此外，还有相当一部分指令是专门为实现操作系统的各种功能而设计的。（四）指令集的精简化 1.RISC的开拓性工作（1）1967年Seymour Cray在研制CDC-6600时，事实上他就已经采用了精简指令集的思想。因此，我们说他的工作是RISC思想的先驱。（2）IBM801项目1975年IBM的沃森研究中心开始了对复杂指令系统是否合理的研究，称为801项目。到1979年研制成32位的IBM801小型计算机，该机只有120条指令，速度达到10MIPS，这可以说是世界上最早利用精简指令思想研制成的计算机。（3）伯克利的RISC项目1980年美国加州大学伯克利分校的David Patterson教授领导了RISC项目。RISC这一缩写正是Patterson在伯克利讲课时首先使用的。（4）斯坦福的MIPS项目1981年美国斯坦福大学的John Hennessy教授开始了MIPS项目。以上四个著名的项目可以说是RISC技术的四个主要技术来源，它们为RISC技术奠定了基础。 100Test 下载频道开通，各类考试题目直接下载。详细请访问

www.100test.com