

汇编语言的主要特点 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/287/2021\\_2022\\_\\_E6\\_B1\\_87\\_E7\\_BC\\_96\\_E8\\_AF\\_AD\\_E8\\_c98\\_287051.htm](https://www.100test.com/kao_ti2020/287/2021_2022__E6_B1_87_E7_BC_96_E8_AF_AD_E8_c98_287051.htm) 一方面，汇编语言指令是用一些具有相应含义的助忆符来表达的，所以，它要比机器语言容易掌握和运用，但另一方面，它要直接使用CPU的资源，相对高级程序设计语言来说，它又显得难掌握。汇编语言程序归纳起来大概有以下几个主要特性。

- 1、与机器相关性 汇编语言指令是机器指令的一种符号表示，而不同类型的CPU有不同的机器指令系统，也就有不同的汇编语言，所以，汇编语言程序与机器有着密切的关系。由于汇编语言程序与机器的相关性，所以，除了同系列、不同型号CPU之间的汇编语言程序有一定程度的可移植性之外，其它不同类型(如：小型机和微机等)CPU之间的汇编语言程序是无法移植的，也就是说，汇编语言程序的通用性和可移植性要比高级语言程序低。
- 2、执行的高效率 正因为汇编语言有“与机器相关性”的特性，程序员用汇编语言编写程序时，可充分发挥自己的聪明才智，对机器内部的各种资源进行合理的安排，让它们始终处于最佳的使用状态，这样做的最终效果就是：程序的执行代码短，执行速度快。现在，高级语言的编译程序在进行寄存器分配和目标代码生成时，也都有一定程度的优化(在后续课程《编译原理》的有关章节会有详细介绍)，但由于所使用的“优化策略”要适应各种不同的情况，所以，这些优化策略只能在宏观上，不可能在微观上、细节上进行优化。而用汇编语言编写程序几乎是程序员直接在写执行代码，程序员可以在程序的每个具体细节上进行优化，

这也是汇编语言程序执行高效率的原因之一。

### 3、编写程序的复杂性

汇编语言是一种面向机器的语言，其汇编指令与机器指令基本上一一对应，所以，汇编指令也同机器指令一样具有功能单一、具体的特点。要想完成某件工作(如计算：A B C等)，就必须安排CPU的每步工作(如：先计算A B，再把C加到前者的结果上)。另外，在编写汇编语言程序时，还要考虑机器资源的限制、汇编指令的细节和限制等等。由于汇编语言程序要安排运算的每一个细节，这就使得编写汇编语言程序比较繁琐、复杂。一个简单的计算公式或计算方法，也要用一系列汇编指令一步一步来实现。

### 4、调试的复杂性

在通常情况下，调试汇编语言程序要比调试高级语言程序困难，其主要原因有四：汇编语言指令涉及到机器资源的细节，在调试过程中，要清楚每个资源的变化情况；程序员在编写汇编语言程序时，为了提高资源的利用率，可以使用各种实现技巧，而这些技巧完全有可能破坏程序的可读性。这样，在调试过程中，除了要知道每条指令的执行功能，还要清楚它在整个解题过程中的作用；高级语言程序几乎不显式地使用“转移语句”，但汇编语言程序要用到大量的、各类转移指令，这些跳转指令大大地增加了调试程序的难度。如果在汇编语言程序中也强调不使用“转移指令”，那么，汇编语言程序就会变成功能单调的顺序程序，这显然是不现实的；调试工具落后，高级语言程序可以在源程序级进行符号跟踪，而汇编语言程序只能跟踪机器指令。不过，现在这方面也有所改善，CV(CodeView)、TD(Turbo Debug)等软件也可在源程序级进行符号跟踪了。

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](http://www.100test.com)