



微机原理和接口技术

第六讲 指令系统与汇编程序6

提 纲

13. 编程语言及汇编语言编程风格

19. 子程序设计概述

14. 汇编程序设计中的伪指令

20. 子程序设计举例

15. 汇编与调试过程

16. 汇编语言程序设计概述

17. 程序设计的结构化

18. 基本程序设计

提 纲

13. 编程语言及汇编语言编程风格



汇编语言编程风格

• 1. 编程语言

- 计算机的工作过程就是执行程序的过程，程序是完成某一运算或功能的若干指令的有序集合。
- 用于程序设计的语言可分为三种：机器语言、汇编语言和高级语言。

➤ 机器语言

计算机能够识别的、用二进制代码表示的语言。

➤ 汇编语言

用助记符表示的编程语言称为汇编语言，用汇编语言编写的程序称为汇编语言程序，或称为源程序。

➤ 高级语言

高级语言是面向过程和问题的程序设计语言，是一种接近人们自然语言和常用数学表达的计算机语言。



汇编语言编程风格

- 2. 编程语言的特点

- 机器语言的特点

- 程序不通用、不易读、易出错、难以维护，目前几乎不用机器语言编写程序。

- 汇编语言的特点

- 1) 指令与机器码一一对应，程序效率高，占用存储空间小，运行速度快，且能反映计算机的实际运行情况，所以用汇编语言能编写出最优化的程序。
- 2) 汇编语言能直接管理和控制硬件设备，如访问存储器、I/O接口，处理中断等。
- 3) 汇编语言是“面向机器”的语言，编程比高级语言困难。
- 4) 汇编语言通用性差，面向具体的机器，不同的微控制器具有不同的指令系统，不能通用。



汇编语言编程风格

➤ 高级语言的特点

- 1) 近似于人们日常用语的语言，直观、易学、易懂，通用性强，易于移植到不同类型的机器中去。如BASIC、PASCAL、C、C++等。
- 2) 计算机不能直接识别和执行高级语言，需要用编译程序将高级语言转换成机器语言。
- 3) 高级语言不受具体机器的限制，而且使用了许多数学公式和习惯用语，从而简化了程序设计的过程。

8051微控制器的高级编程语言，称为C51



汇编语言编程风格

完成同一个任务所使用的方法或程序不是唯一的。程序设计的质量将直接影响到计算机系统的工作效率、运行可靠性。良好的编程风格对于实现高质量的程序具有重要意义。

汇编程序设计时，应关注以下几点。

1. 注释

- 注释是程序设计中的重要内容之一，汇编指令的抽象特性更要重视注释的作用。
- 注释内容用“;”与助记符指令隔离，注释内容长度不限，换行时，头部仍要标注“;”。

2. 标号的使用

- 标号由不多于8个ASCII字符组成，第一个字符必须是字母，不能使用汇编语言已定义的符号，如助记符，寄存器名等。
- 同一个标号在一个独立的程序中只能定义一次。
- 标号通常代表地址，标号名的选取应能够表达所代表的目的地址。



汇编语言编程风格

3. 子程序的使用

- 将大而复杂的任务划分为若干个小而简单的任务，这些小任务通过子程序的形式完成。
- 每个子程序最好都有对应的注释块，在注释块中说明子程序的出入口参数、功能等。

4. 堆栈的使用

- 用于寄存器、SFR以及内存数据的保护和恢复。

5. 伪指令的使用

- 起始汇编伪指令和结束汇编伪指令不可缺少。
- 充分利用其他伪指令，如赋值、定义字节、定义字等，增加程序的可维护性和可读性。

Thank you!

