

## 第三章-指令系统

### (一)指令格式

指令：要求计算机进行基本操作的命令。一台计算机所能执行的全部指令的集合称为该计算机的指令系统或指令集。

一条指令中至少含 3 方面的信息：要执行的操作、操作数的来源、操作结果的去向。

指令的基本格式：由两部分组成，**操作码**和**地址码**。



操作码：描述指令的操作功能，必不可少的部分。

地址码：描述与操作数有关的信息，可以是操作数本身也可以是存放操作数的地址。

指令的地址码：

根据指令中地址码部分显式给出的地址的个数，可将指令格式分为以下 5 种：

1. 零地址指令 Op
2. 一地址指令  $A \leftarrow \text{Op}(A)$  或  $\text{AC}[\text{累加器}] \leftarrow (\text{AC})\text{Op}(A)$
3. 二地址指令  $A1 \leftarrow (A1)\text{Op}(A2)$
4. 三地址指令  $A3 \leftarrow (A1)\text{Op}(A2)$
5. 多地址指令

指令的操作码：指令系统中的每一条指令都有唯一确定的操作码，不同指令的操作码是不相同的。

固定长度操作码：所有指令的操作码长度固定，集中在一个字段内。

可变长度操作码：指令系统中的操作码长度有多种，不同指令的操作码的长度不完全相同。

拓展操作码技术：将操作码设计为几种不同的固定长度，且相互之间按某种规则进行拓展。

### (二)数据类型

数据表示：计算机硬件能够直接识别、指令系统可以直接调用的数据类型。

数据结构：指由软件进行处理和实现的各种数据类型。

操作数的大小：操作数的位数或字节数。主要的大小有：

字节(Byte, 8 bits)、半字(16 bits)、字(32 bits)、双字(64 bits)。

### (三)寻址方式

一些概念：

形式地址：指令的地址码部分给出的地址。

有效地址：形式地址经过地址变换后才能得到操作数的真正地址。

寻址方式：确定指令操作数有效地址的方法。

1. 直接寻址：在指令的地址码字段直接给出操作数所在的主存单元的地址。
2. 间接寻址：指令地址码字段给出的是操作数所在内存单元地址的地址。【个人认为类似于 C 中的\*\*】
3. 立即寻址：地址码字段直接给出操作数，而非地址。
4. 寄存器直接寻址与间接寻址：地址码字段为一个寄存器的编号，里面为操作数或操作数的地址。
5. 隐含寻址：指令中没有操作数的地址，约定好操作数在某个特定的寄存器或堆栈中。
6. 相对寻址：将程序计数器 PC 的内容与指令中给出的形式地址（也叫偏移量）相加，形成有效地址。
7. 变址寻址：将指令中的形式地址与变址寄存器  $R_x$ （可以是专用寄存器也可以是通用寄存器）中的内容相加形成有效地址。

8. 基址寻址：将指令中给出的形式地址与基址寄存器  $R_B$  的内容相加，形成操作数的有效地址。
9. “基址+变址”寻址：有效地址  $= R_B + R_X + D$

#### (四)指令类型与功能

1. 数据传输类指令：  
将数据从一个地方传送到另一个地方。如 8086 指令中的 MOV AL, BL。主要实现主存和主存之间、主存和寄存器之间、寄存器和寄存器之间的数据传送。
2. 数据运算类指令：  
用来实现数据的算术运算，逻辑运算和移位运算。
3. 程序控制类指令：  
主要用来控制程序的执行顺序和方向。包含转移指令、子程序调用和返回指令、自陷指令等。
4. 输入输出指令（简称 I/O 指令）：  
主要用于实现主机与外部设备之间的信息交换。