

# 标志位介绍

标签：[http](#) [io](#) [ar](#) 使用 [sp](#) [strong](#) [on](#) [数据](#) [div](#)

## 一、运算结果标志位

### 1、进位标志CF(Carry Flag)

进位标志CF主要用来反映运算是否产生进位或借位。如果运算结果的最高位产生了一个进位或借位，那么，其值为1，否则其值为0。

使用该标志位的情况有：多字(字节)数的加减运算，无符号数的大小比较运算，移位操作，字(字节)之间移位，专门改变CF值的指令等。

### 2、奇偶标志PF(Parity Flag)

奇偶标志PF用于反映运算结果中“1”的个数的奇偶性。如果“1”的个数为偶数，则PF的值为1，否则其值为0。

利用PF可进行奇偶校验检查，或产生奇偶校验位。在数据传送过程中，为了提供传送的可靠性，如果采用奇偶校验的方法，就可使用该标志位。

### 3、辅助进位标志AF(Auxiliary Carry Flag)

在发生下列情况时，辅助进位标志AF的值被置为1，否则其值为0：

- (1)、在字操作时，发生低字节向高字节进位或借位时；
- (2)、在字节操作时，发生低4位向高4位进位或借位时。

对以上6个运算结果标志位，在一般编程情况下，标志位CF、ZF、SF和OF的使用频率较高，而标志位PF和AF的使用频率较低。

### 4、零标志ZF(Zero Flag)

零标志ZF用来反映运算结果是否为0。如果运算结果为0，则其值为1，否则其值为0。

在判断运算结果是否为0时，可使用此标志位。

### 5、符号标志SF(Sign Flag)

符号标志SF用来反映运算结果的符号位，它与运算结果的最高位相同。在微机系统中，有符号数采用补码表示法，所以，SF也就反映运算结果的正负号。运算结果为正数时，SF的值为0，否则其值为1。

### 6、溢出标志OF(Overflow Flag)

溢出标志OF用于反映有符号数加减运算所得结果是否溢出。如果运算结果超过当前运算位数所能表示的范围，则称为溢出，OF的值被置为1，否则，OF的值被清为0。

“溢出”和“进位”是两个不同含义的概念，不要混淆。如果不太清楚的话，请查阅《计算机组成原理》课程中的有关章节。

## 二、状态控制标志位

状态控制标志位是用来控制CPU操作的，它们要通过专门的指令才能使之发生改变。

### 1、追踪标志TF(Trap Flag)

当追踪标志TF被置为1时，CPU进入单步执行方式，即每执行一条指令，产生一个单步中断请求。这种方式主要用于程序的调试。

指令系统中没有专门的指令来改变标志位TF的值，但程序员可用其它办法来改变其值。

### 2、中断允许标志IF(Interrupt-enable Flag)

中断允许标志IF是用来决定CPU是否响应CPU外部的可屏蔽中断发出的中断请求。但不管该标志为何值，CPU都必须响应CPU外部的不可屏蔽中断所发出的中断请求，以及CPU内部产生的中断请求。具体规定如下：

- (1)、当IF=1时，CPU可以响应CPU外部的可屏蔽中断发出的中断请求；
- (2)、当IF=0时，CPU不响应CPU外部的可屏蔽中断发出的中断请求。

CPU的指令系统中也有专门的指令来改变标志位IF的值。

### 3、方向标志DF(Direction Flag)

方向标志DF用来决定在串操作指令执行时有关指针寄存器发生调整的方向。具体规定在第5.2.11节——字符串操作指令——中给出。在微机的指令系统中，还提供了专门的指令来改变标志位DF的值。

## 三、32位标志寄存器增加的标志位

### 1、I/O特权标志IOPL(I/O Privilege Level)

I/O特权标志用两位二进制位来表示，也称为I/O特权级字段。该字段指定了要求执行I/O指令的特权级。如果当前的特权级别在数值上小于等于IOPL的值，那么，该I/O指令可执行，否则将发生一个保护异常。

### 2、嵌套任务标志NT(Nested Task)

嵌套任务标志NT用来控制中断返回指令IRET的执行。具体规定如下：

- (1)、当NT=0，用堆栈中保存的值恢复EFLAGS、CS和EIP，执行常规的中断返回操作；
- (2)、当NT=1，通过任务转换实现中断返回。

### 3、重启动标志RF(Restart Flag)

重启动标志RF用来控制是否接受调试故障。规定：RF=0时，表示“接受”调试故障，否则拒绝之。在成功执行完一条指令后，处理机把RF置为0，当接受到一个非调试故障时，处理机就把它置为1。

### 4、虚拟8086方式标志VM(Virtual 8086 Mode)

如果该标志的值为1，则表示处理机处于虚拟的8086方式下的工作状态，否则，处理机处于一般保护方式下的工作状态。

#### 标志位操作指令

标志位操作指令是一组对标志位置位、复位、保存和恢复等操作的指令。

##### 1、进位CF操作指令

☐、清进位指令CLC(Clear Carry Flag)：CF←0 ☐、置进位指令STC(Set Carry Flag)：CF←1 ☐、进位取反指令CMC(Complement Carry Flag)：CF←not CF

##### 2、方向位DF操作指令

☐、清方向位指令CLD(Clear Direction Flag)：DF←0 ☐、置方向位指令STD(Set Direction Flag)：DF←1

##### 3、中断允许位IF操作指令

☐、清中断允许位指令CLI(Clear Interrupt Flag)：IF←0其功能是不允许可屏蔽的外部中断来中断其后程序段的执行。

☐、置中断允许位指令STI(Set Interrupt Flag)：IF←1其功能是恢复可屏蔽的外部中断的中断响应功能，通常是与CLI成对使用的。

##### 4、取标志位操作指令

☐、LAHF(Load AH from Flags)：AH←Flags的低8位 ☐、SAHF(Store AH in Flags)：Flags的低8位←AH

##### 5、标志位堆栈操作指令

☐、PUSHF/PUSHFD(Push Flags onto Stack)：把16位/32位标志寄存器进栈； ☐、POPF/POPDF(Pop Flags off Stack)：把16位/32位标志寄存器出栈；

CF：	进位标志符号比	排在第0位
PF：	奇偶标志	排在第2位
AF：	辅助进位标志	排在第4位
ZF：	零标志	排在第6位
SF：	符号标志	排在第7位
TF：	追踪标志	排在第8位
IF：	中断允许标志	排在第9位
DF：	方向标志	排在第10位
OF：	溢出标志	排在第11位

#### 在debug 中标志位（标志寄存器）的表示方法

标 志 名	标 志 为1	标 志 为0
OF      溢出	OV      (是)	NV      (否)
DF      方向	DN      (减 量)	UP      (增量)
IF      中断	EI      (允许)	DI      (关闭)
SF      符号	NG      (为 负)	PL      (为正)
ZF      零	ZR      (是)	NZ      (否)
AF      辅助进位	AC      (是)	NA      (否)
PF      奇偶	PE      (偶)	PO      (奇)
CF      进位	CY      (是)	NC      (否)

#### 标志位介绍

标签：[http](#) [io](#) [ar](#) 使用 [sp](#) [strong](#) [on](#) [数据](#) [div](#)

原文地址：<http://www.cnblogs.com/qintangtao/p/4161912.html>