汇编快速入门(4)

By：凌晨五点

知识点

- 分段管理
- 标志寄存器

分段管理

分段管理

- 一个存储单元有一个物理地址，还有多个逻辑地址

物理地址

- 物理地址：
  - 就是一个存储单元的编号；
  - 每个物理存储单元都有一个20位编号；
  - 8086CPU物理地址范围：00000H~FFFFFH

逻辑地址

- 逻辑地址：
  - 用户编程时，采用逻辑地址，形式为：

    段基地址：段内偏移地址
  - 物理地址：将逻辑地址左移4位，加上偏移地址就得到20位物理地址
    - 逻辑地址：1230:100　　1030:2100　　1100:1400
    - 物理地址：12400　　12400　　12400

      1230*10H + 100

段寄存器与逻辑段

- 8086CPU有4个段寄存器，每个段寄存器用来确定一个逻辑段的起始位置，每种逻辑段均有各自的用途：
  - CS（代码段）：指明代码的起始地址
    - 利用CS：IP取得下一条要执行的指令
  - SS（堆栈段）：指明堆栈段的起始地址
    - 利用SS：SP操作堆栈顶的数据
  - DS（数据段）：指明数据的起始地址
    - 利用DS：EA存取数据段中的数据
  - ES（附加段）：指明附加段的起始地址
    - 利用ES：EA存取附加段中的数据

段寄存器与逻辑段

- 没有指明段前缀时，一般的数据访问在DS（数据）段
- 例如：
  - MOV AX,[1000H]; = MOV AX,DS:[1000H];
    从默认的DS段中取出数据
  - MOV AX,CS:[1000H]
    从指定的CS段取出数据

标志寄存器

标志寄存器

8086CPU 的 flag 寄存器的结构如图 11.1 所示。

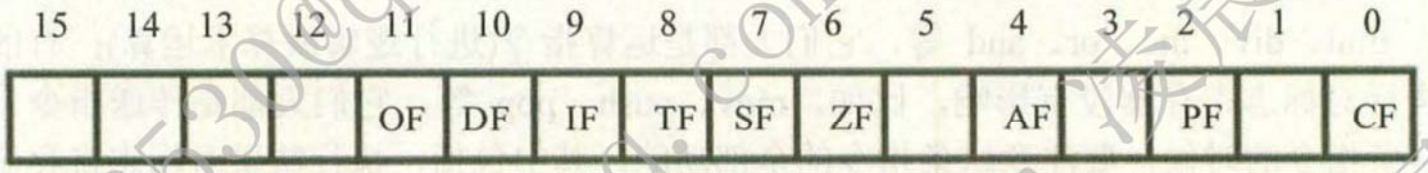

图 11.1　flag 寄存器各位示意图

flag 的 1、3、5、12、13、14、15 位在 8086CPU 中没有使用，不具有任何含义。而 0、2、4、6、7、8、9、10、11 位都具有特殊的含义。

标志寄存器

标志位	标志位名称及外语全称	=1	=0
CF	进位标志/Carry Flag	进位	无进位
PF	奇偶标志/Parity Flag	偶	奇
AF	辅助进位标志/Auxiliary Carry Flag	进位	无进位
ZF	零标志/Zero Flag	等于零	不等于零
SF	符号标志/Sign Flag	负	非负
TF	跟踪标志/Trace Flag		
IF	中断标志/Interrupt Flag	允许	禁止
DF	方向标志/Direction Flag	减少	增加
OF	溢出标志/Overflow Flag	溢出	未溢出

标志位

- 状态标志：

  用于记录程序运行结果的状态信息。

  CF　ZF　SF　PF　OF　AF

- 控制标志：

  用于控制处理器执行指令。

  DF　IF　TF

进位标志CF

- 进位标志CF（Carry Flag），当运算结果的最高有效位有进位（加法）或借位（减法）时候，进位标志置1，即CF=1；否则CF=0
- 示例：

2C + 7C = A8， 没有进位：CF = 0

C2 + C7 = (1)89， 有进位：CF = 1

零标志ZF

- 零标志ZF（Zero Flag），若运算结果为0，则ZF=1；否则ZF=0
- 示例：

2C + 7C = A8，结果不为0：ZF = 0

39 + C7 = (1)00，结果为0：ZF = 1

符号标志SF

- 符号标志SF（Sign Flag），若运算结果最高位为1，则SF=1；否则SF=0
- 示例：

2C + 7C = A8，二进制： 10101000B

最高位为1：SF = 1

39 + C7 = (1)00，二进制： （1）00000000B

最高位为0：SF = 0

奇偶标志PF

- 奇偶标志PF（Parity Flag），若运算结果最低字节中“1”的个数为零或偶数时，则PF=1；否则PF=0
- 示例：

2C + 7C = A8， 二进制：10101000

结果中有3个1，是奇数：PF = 0

39 + C7 = (1)00，二进制：（1）00000000

结果中有0个1，是偶数：PF = 1

溢出标志OF

- 溢出标志OF（Overflow Flag），若运算结果有溢出，则OF=1；否则OF=0
- 溢出：如果运算结果超出了范围，就产生了溢出，有溢出，说明有符号数的运算结果不正确

- 说明：

我们通常认为溢出（上溢）就是因为进位时当前存储格式（1B、2B、4B等）的位数（8bit、16bit、32bit）不够而引起的。比如8位寄存器：11111111B +1B=100000000B超过了八位的1被认为是溢出寄存器（放不下），当然也是进位进上去的1

溢出与进位（1）

- **溢出标志（OF）**：表示有符号数运算结果是否超出范围，运算结果已经不正确
- **进位标志（CF）**：表示有符号数运算结果是否超出范围，运算结果仍然正确

- 有符号无符号指的是最高位是否是符号位，即是以补码的形式看待还是以原码的形式看待。

CF范围：0~255/0X00~0XFF（8位）、
0~65535/0X0000~0XFFFF（16位）
OF范围：-128~127/0X80~0XEF（8位）、
-32768~32767/0X8000~0XEFFF（16位）

溢出与进位（2）

（1）、8H+8H：

对于signed：(8)+(8)=16,没超过[-128,127]的范围，OF为0
对于unsigned：(8)+(8)=16,没超过[0,255]的范围，CF为0

（2）、80H+81H：

对于signed：(-128)+(-127)=-255,超过[-128,127]的范围，OF为1
对于unsigned：(128)+(129)=257,超过[0,255]的范围，CF为1

（3）、FCH+05H：

对于signed：(-4)+(5)=1,没有超过[-128,127]的范围，OF为0
对于unsigned：(252)+(5)=257,超过[0,255]的范围，CF为1

（4）、7FH+2H：

对于signed：(127)+(2)=129,超过[-128,127]的范围，OF为1
对于unsigned：(127)+(2)=129,没超过[0,255]的范围，CF为0

辅助进位标志

- 辅助进位标志AF（Auxiliary Carry Flag），若运算时D_3（低半字节）有进位或借位时， AF=1；否则AF=0

- 示例：

  33H + 78H = ABH

  低四位 3，加上，低四位8，进位为零，即：AF=0。

  39H + 78H = B1H

  低四位 9，加上，低四位8，进位为 1 ，即：AF=1。

方向标志DF

- 方向标志DF（Direction Flag），用于串操作指令中，控制地址的变化方向：
  - 设置DF为0，存储器地址自动增加
  - 设置DF为1，存储器地址自动减少

- 示例：

  CLD指令用于复位方向标志，执行后DF=0；

  STD指令用于置位方向标志，执行后DF=1。

中断允许标志IF

- 中断允许标志IF（Interrupt-enable Flag），用于控制外部可屏蔽中断是否可以被处理器响应：
  - 设置IF为0，则禁止中断
  - 设置IF为1，则允许中断

- 示例：

  CLI指令用于复位中断标志，执行后IF=0；

  STI指令用于置位中断标志，执行后IF=1。

陷阱标志TF

- 陷阱标志TF（Trap Flag），用于控制处理器进入单步操作方式：
  - 设置TF为0，处理器正常工作
  - 设置TF为1，处理器单步执行指令

- 单步执行指令：处理器在每条指令执行结束时，便产生一个编号为1的内部中断
- 这种内部中断称为单步中断，所以TF也称单步标志；
  利用单步中断可对程序进行逐条指令的调试；

回顾

标志位	标志位名称及外语全称	=1	=0
CF	进位标志/Carry Flag	进位	无进位
PF	奇偶标志/Parity Flag	偶	奇
AF	辅助进位标志/Auxiliary Carry Flag	进位	无进位
ZF	零标志/Zero Flag	等于零	不等于零
SF	符号标志/Sign Flag	负	非负
TF	跟踪标志/Trace Flag		
IF	中断标志/Interrupt Flag	允许	禁止
DF	方向标志/Direction Flag	减少	增加
OF	溢出标志/Overflow Flag	溢出	未溢出

谢谢

看雪公众号：ikanxue