



2.5 位操作类指令



2.5.1 逻辑运算指令

逻辑运算指令对字节或字数据进行按位的操作。
包括AND、OR、NOT、XOR、TEST等。

2.5.1 逻辑运算指令

1. AND (逻辑与)

指令格式:

AND DST, SRC ; (DST) ← (DST) ∧ (SRC)

逻辑与指令可用于屏蔽操作数的数据位;

例: **AND AH, 0FH** ; 屏蔽寄存器AH的高四位, 保留其低四位的数据。

01100111
AND 11011100

01000100

2.5.1 逻辑运算指令

2. OR (逻辑或)

指令格式:

OR DST, SRC ; $(DST) \leftarrow (DST) \vee (SRC)$

逻辑或指令可对操作数的某些位置1;

例: OR BX, 0F00H ; 将寄存器BH的低四位置1, 其他位数据不变。

01000110
OR 01011010
<hr/>
01011110

2.5.1 逻辑运算指令

3. NOT (逻辑非)

指令格式:

NOT OPR ; (OPR) \leftarrow (OPR)

逻辑非指令用于对操作数按位求反

NOT 01000110
10111001

2.5.1 逻辑运算指令

4. XOR (异或)

指令格式:

XOR DST, SRC ; (DST) \leftarrow (DST) \oplus (SRC)

异或指令用于测试两个数据是否相等或将某操作数按位求反;

例: 指令 **XOR CX, 00FFH** 的操作结果是寄存器CH数据保持不变, 对寄存器CL数据求反。

01000110
XOR 01011010
<hr/>
00011100

2.5.1 逻辑运算指令

5. TEST (测试)

指令格式:

TEST OPR1, OPR2 ; $(\text{OPR1}) \wedge (\text{OPR2})$

测试指令的操作和逻辑与指令相同，只是不保存测试结果。

例: TEST AL, 00000001B

如果寄存器AL最低位是0，则使ZF=1。

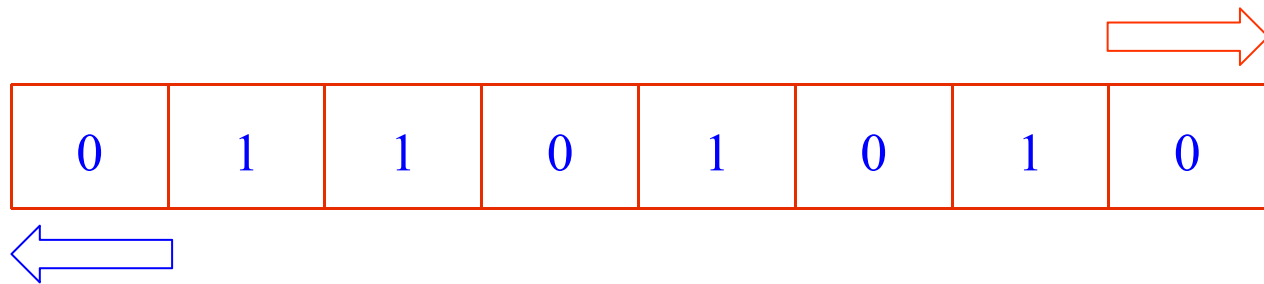
2.5.1 逻辑运算指令

注意：

- ①逻辑非指令为单操作数指令，不允许使用立即数，也不影响标志位。
- ②其他4条指令为双操作数指令，当源操作数不是立即数时，两个操作数中的一个要采用寄存器寻址方式，另一个操作数可以采用任何寻址方式。
- ③运算结果将影响标志位ZF、SF和PF，使CF、OF清0，对AF无影响。

2.5.2 移位指令

- ◇ 移位指令根据移位方向分为左移或右移；
- ◇ 按移位功能可以分为算术移位、逻辑移位和循环移位；
- ◇ 从移位次数上可以分为一次移位或多次移位。



1 算术逻辑移位指令

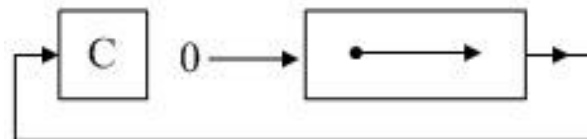
算术移位和逻辑移位

逻辑左移 SHL OPR, CNT

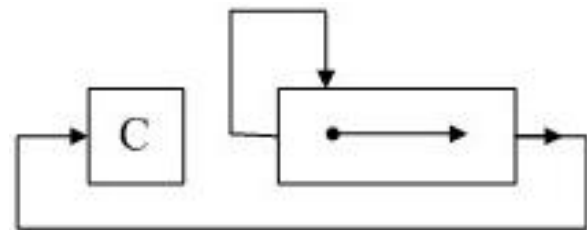
算数左移 SAL OPR, CNT



逻辑右移 SHR OPR, CNT



算数右移 SAR OPR, CNT



2.5.2 移位指令

由图可见：

- ①算术左移指令与逻辑左移指令相同，左移一位最高位移入**CF**，最低位补零，可用于无符号数乘2操作；
- ②逻辑右移指令右移一位移出位进入**CF**，最高位补零，可用于无符号数除2操作；
- ③算术右移指令右移一位，移出位进入**CF**，最高位保持不变，可以用于有符号数除2操作。

算术和逻辑移位指令是双操作数指令，操作数**OPR**可以使用除立即数寻址之外的任何寻址方式；**CNT**决定移位次数，**CNT=1**时只移位一次，**CNT**为寄存器**CL**时，移位次数由寄存器**CL**的内容决定。

移位指令举例

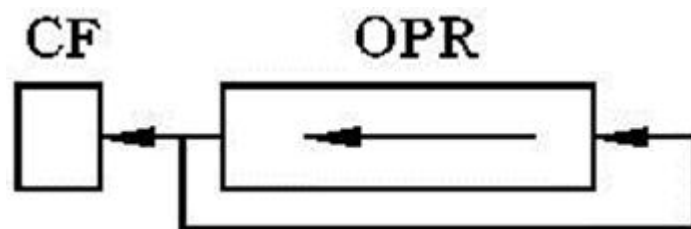


[例] 当 $(BX) = FFFCH$ 时，执行指令 “SAR BX, 1” 后， $(BX) = FFFE H$ ，寄存器 BX 的内容由 -4 变为 -2。

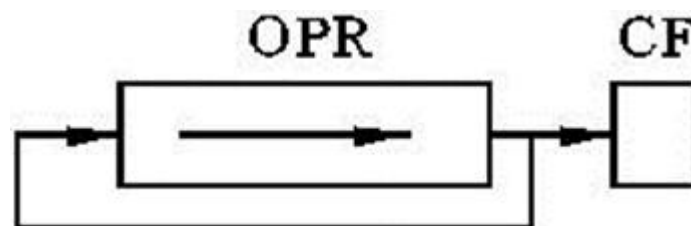


2 循环移位指令

循环左移 ROL OPR, CNT

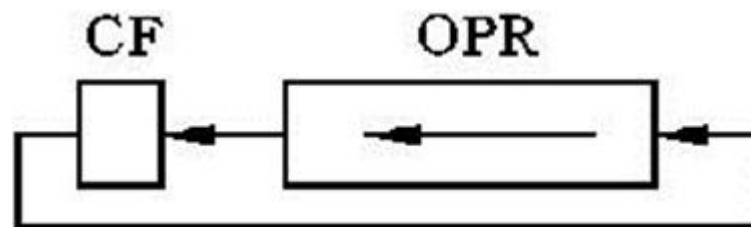


循环右移 ROR OPR, CNT



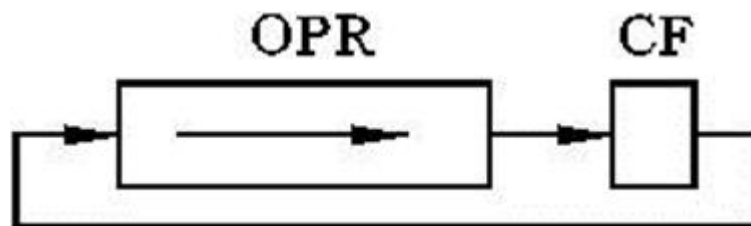
带进位循环左移

RCL OPR, CNT




带进位循环右移

RCR OPR, CNT



3 移位指令应用举例



例 已知：(DS) = 7B00H, (SI) = 1B79H,
(7CB79H) = 07H

指令：MOV CL,3
SAL BYTE PTR [SI],CL

指令执行结果：(7CB79H) = 38H, CF=0

程序设计中，连续测试循环移位指令执行后CF的值，可确定操作数中含有“1”或“0”的个数。另外，将算术逻辑移位指令和循环移位指令结合使用，可以实现多字节（多精度）数据的移位操作。

3 移位指令应用举例

例 用移位指令计算 $Y=5 \times X$ (假设 X 为8位数据且已在 AL 中)
，程序段如下：

```
MOV BL,AL      ;  $BL \leftarrow X$   
SHL AL,1       ;  $2 \times X$   
SHL AL,1       ;  $4 \times X$   
ADD AL,BL      ;  $5 \times X$ 
```

4 移位指令对状态标志的影响

所有移位指令都可以对字或字节操作数进行移位。

移位结果对状态标志的影响是：

- ①算术、逻辑移位指令将影响标志位**SF**、**ZF**、**CF**和**PF**；
- ②循环移位指令只影响标志位**CF**， **OF**， 对其他标志位无影响。
- ③标志位**OF**只是在**CNT=1**时有效， 移位后最高有效位（**MSB**） 的值发生变化时**OF=1**， 否则**OF=0**。

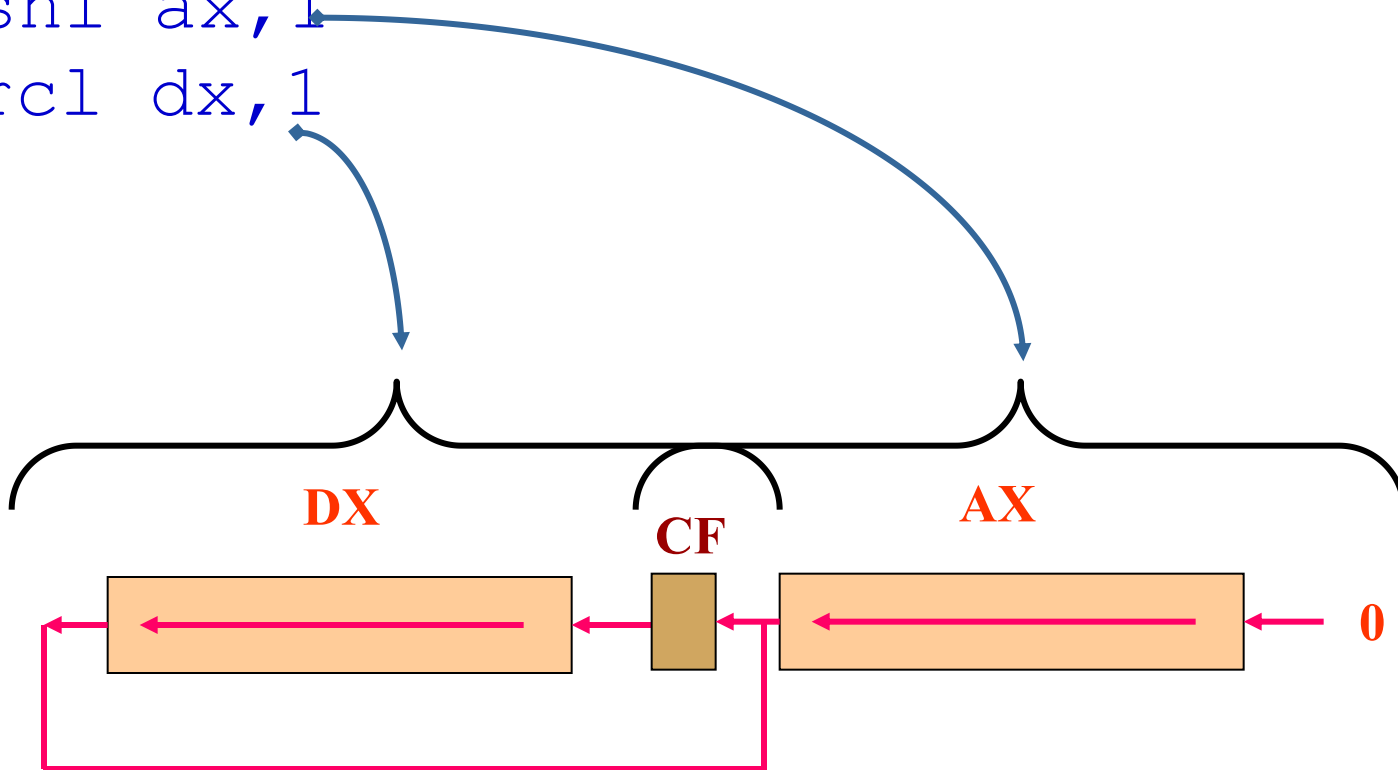
移位指令应用举例

例：32位数据移位

；将DX.AX中32位数值左移一位

```
shl ax, 1
```

```
rcl dx, 1
```



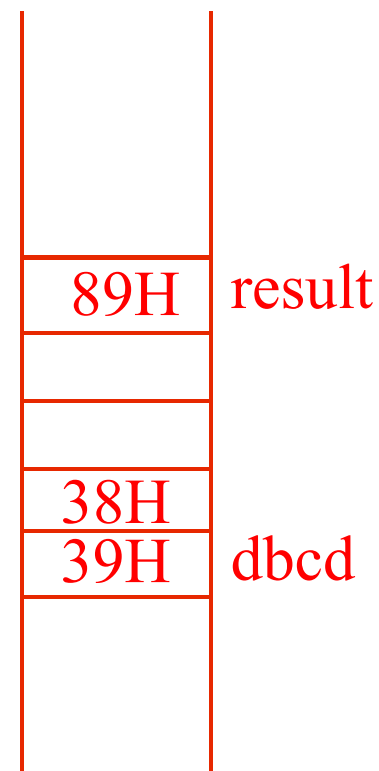
移位指令应用举例

例：将变量**dbcd**中存储的两位非压缩BCD码合并为压缩BCD，存入**result**单元。

；假设变量**dbcd**中有2位非压缩BCD码

```

mov dl,dbcd          ; 取低字节
and dl,0fh           ; 只要低4位
mov dh,dbcd+1        ; 取高字节
mov cl,4
shl dh,cl             ; 低4位移到高4位
or dl,dh              ; 合并到DL
mov result dl
  
```





本小节到此结束
谢谢！



SHL和SAL指令的功能

SHL reg/mem,1 (SAL reg/mem, 1)



SHR指令的功能

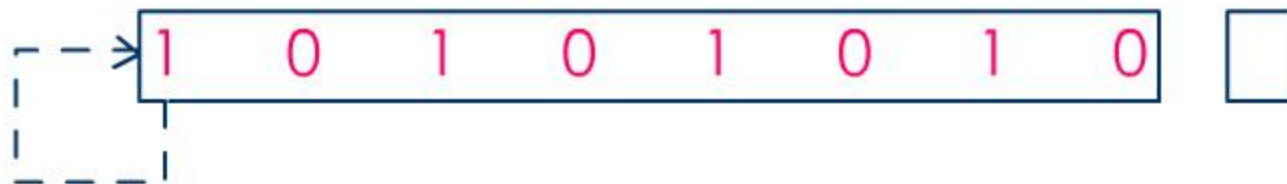
SHR reg/mem, 1



SAR指令的功能

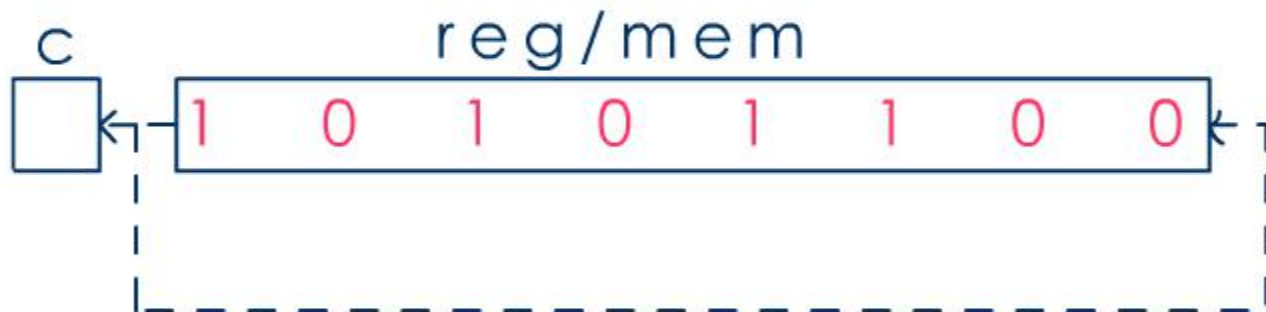
SAR reg/mem, 1

reg/mem



ROL指令的功能

ROL reg/mem, 1



播放

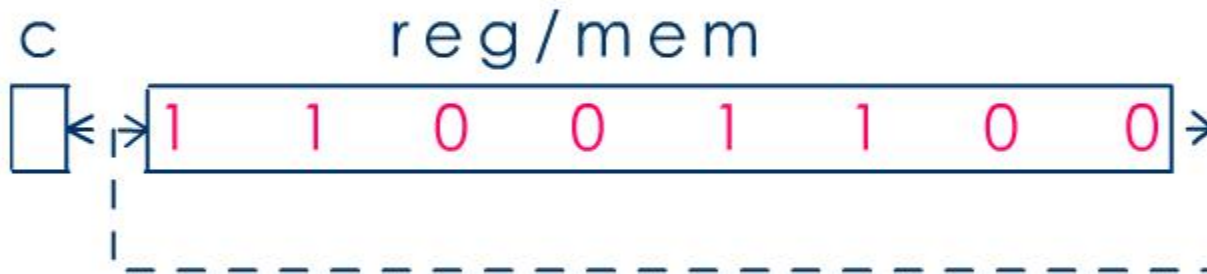


返回



ROR指令的功能

ROR reg/mem, 1



RCL指令的功能

RCL reg/mem, 1



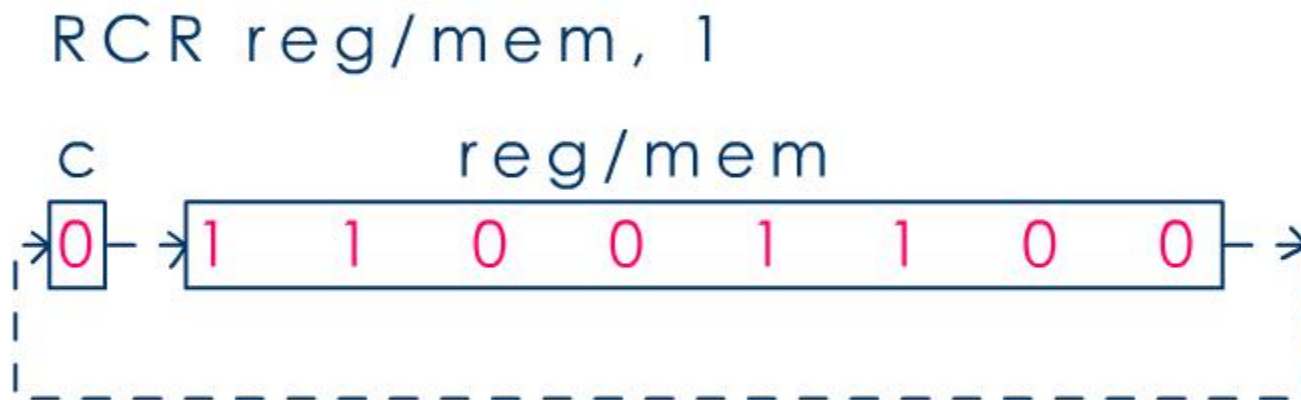
播放




返回



RCR指令的功能





SAL: Shift Arithmetic Left
SHL: SHift Logic Left
SAR: Shift Arithmetic Right
SHR: SHift Logic Right
ROL: ROtate Left
ROR: ROtate Right
RCL: Rotate Left through CF
RCR: Rotate Right through CF