


# 移位操作指令

# 移位操作指令

- 控制二进制位向左或向右移动的指令

 非循环移位指令  
循环移位指令

# 移位操作指令说明

- 指令格式在形式上为双操作数，本质上为单操作数；
- 指令的目标操作数为被移动对象，源操作数为移动次数
  - 当目标为存储器操作数时，需要说明其字长
- 移动1位时由指令直接给出；移动两位及以上时，移位次数必须由CL指定。

**指令源操作数  
只能是1或CL**

# 1. 非循环移位指令

- 逻辑左移
- 算术左移
- 逻辑右移
- 算术右移

# 算术左移和逻辑左移

## 算术左移指令：

SAL OPRD, 1  
SAL OPRD, CL } 有符号数



## 逻辑左移指令：

SHL OPRD, 1  
SHL OPRD, CL } 无符号数

# 逻辑右移

格式:

SHR OPRD, 1

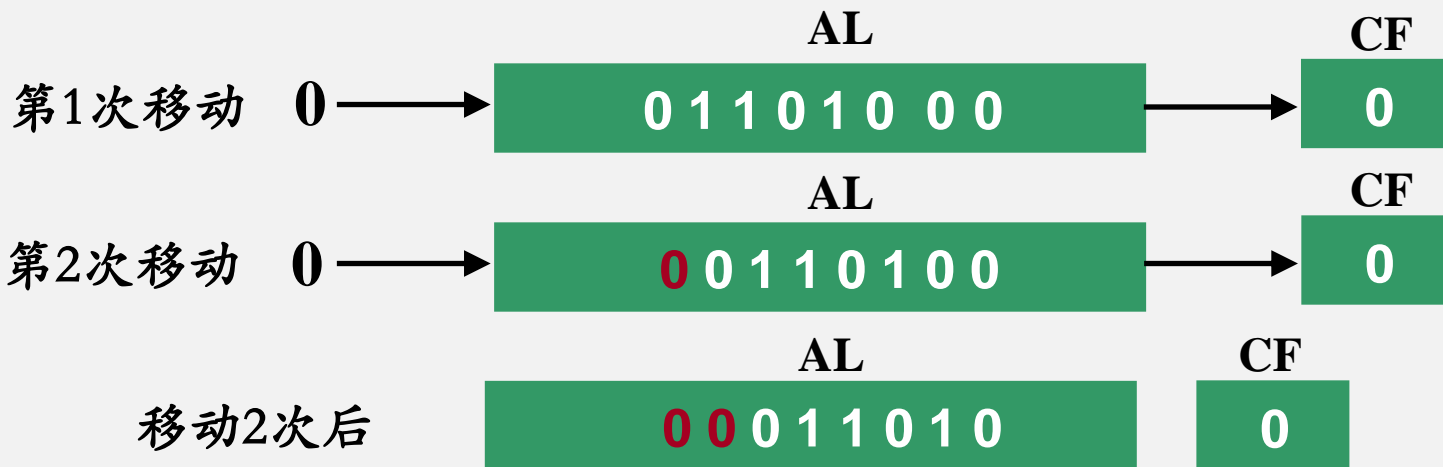
SHR OPRD, CL

无符号数  
的右移



# 逻辑右移例：

- MOV AL, 68H
- MOV CL, 2
- SHR AL, CL



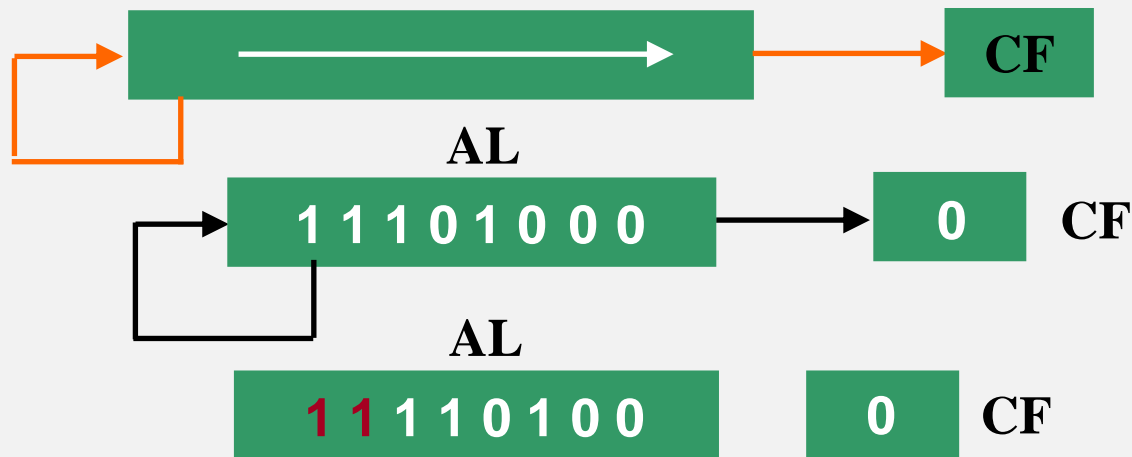
# 算术右移

格式:

SAR OPRD, 1

SAR OPRD, CL

有符号数  
的右移





# 非循环移位指令的应用

- 左移可实现乘法运算
- 右移可实现除法运算

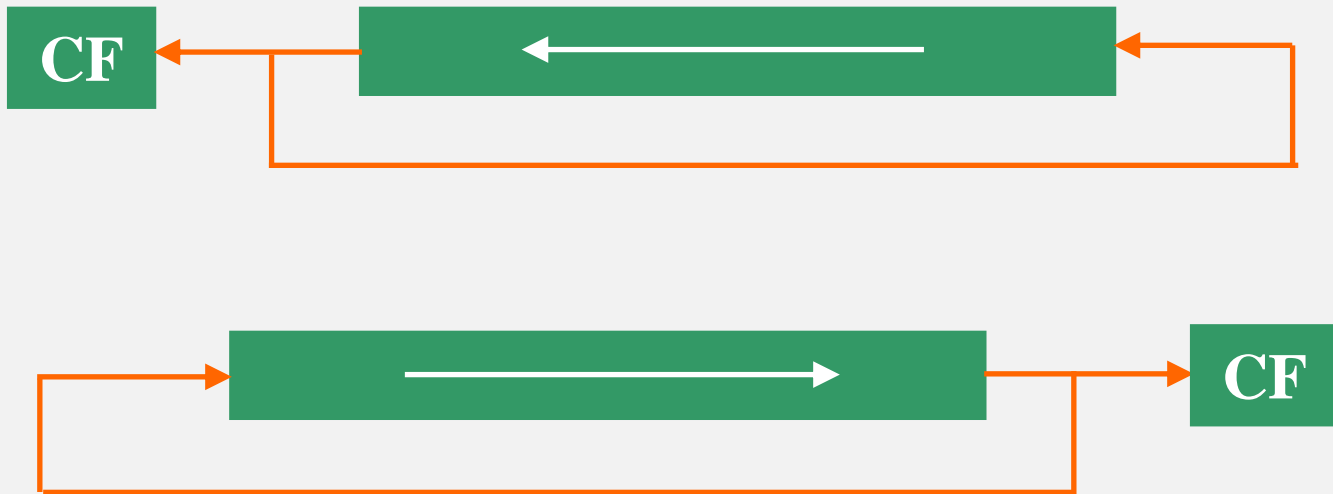
## 2. 循环移位指令

不带进位位的循环移位 { 左移 ROL  
右移 ROR

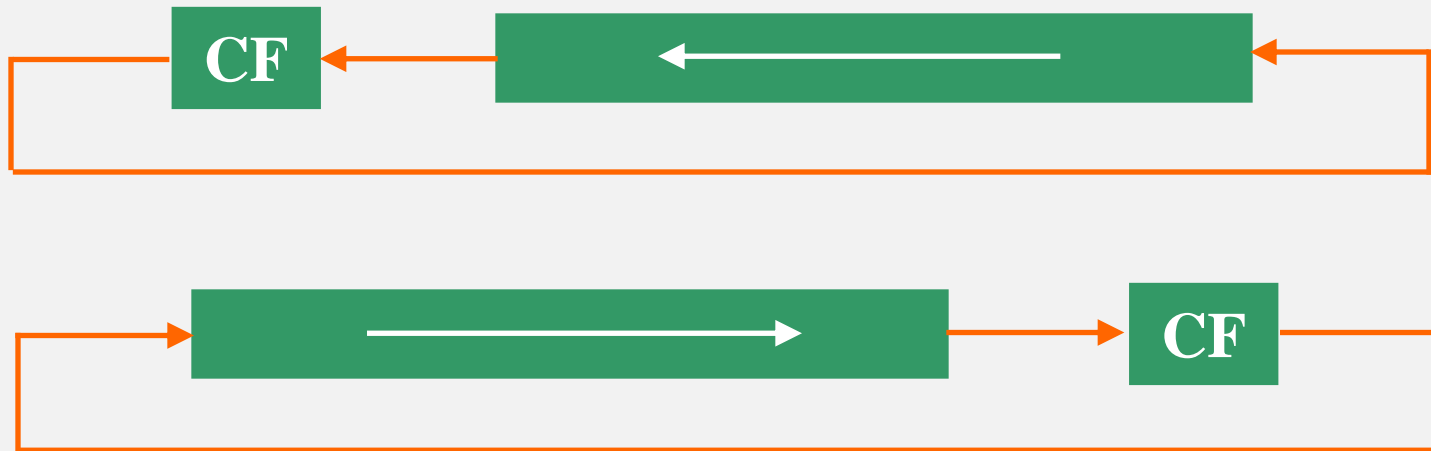
带进位位的循环移位 { 左移 RCL  
右移 RCR

指令格式、对操作数的要求与非循环移位指令相同

# 不带进位的循环移位



# 带进位的循环移位



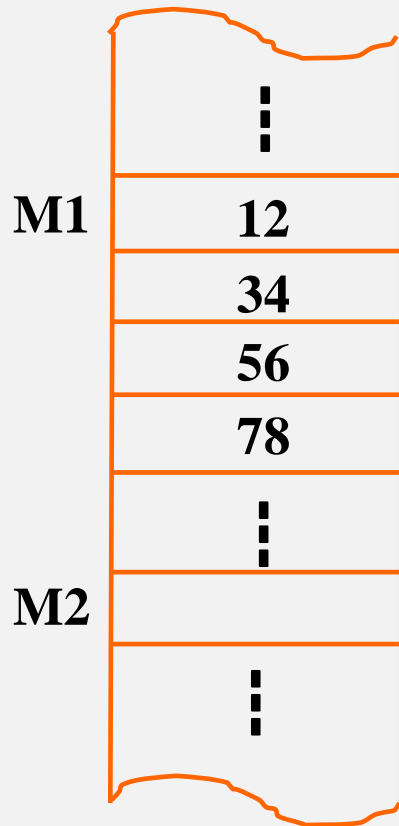
# 循环移位指令的应用

- 用于对某些位状态的测试;
- 高位部分和低位部分的交换;
- 与非循环移位指令一起组成32位或更长字长数的移位。

# 例：

在内存数据段M1为首地址的4个单元中存放了4个压缩BCD码。

要求：将这4个压缩BCD码分别转换为ASCII码，并将转换结果存放在同一逻辑段、M2为首的单元中。



例：

## ■ 题目分析：

- 压缩BCD码是用4位二进制码表示1位十进制数
- 转换ASCII码时需要分别转换高4位（十位数）和低4位（个位数）
- 0~9的ASCII码的高4位均为0011（03H）
- 转换低4位时应先使高4位清零，转换高4位时须先将高4位移动到低4位的位置。

1字节表示2

位压缩BCD



# 程序例

LEA SI,M1

LEA DI,M2

MOV CH,4

Next: MOV AL,[SI]

MOV BL,AL

AND AL,0FH

OR AL,30H

MOV [DI],AL

INC DI

MOV AL,BL

MOV CL,4

SHR AL,CL

OR AL,30H

MOV [DI],AL

INC DI

INC SI

DEC CH

JNZ Next

HLT

