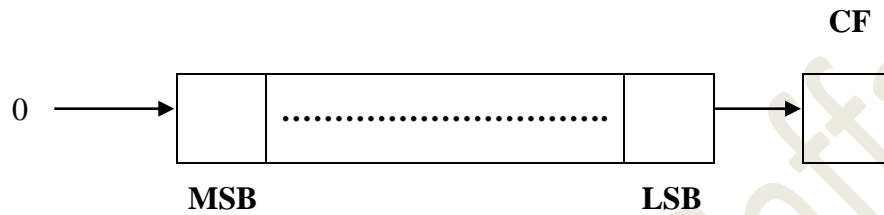


3.2.4 Shifting and Rotating instructions

3.2.4.1 Shifting instructions

i. Logical Shifting

1. **SHR** op1,op2 . shift right op1(logical) by op2. The shift to the right can be done by enter 0 from the left and every bit jump to the other bit to the right & the (**LSB**) from right will be transferred to the carry flag.

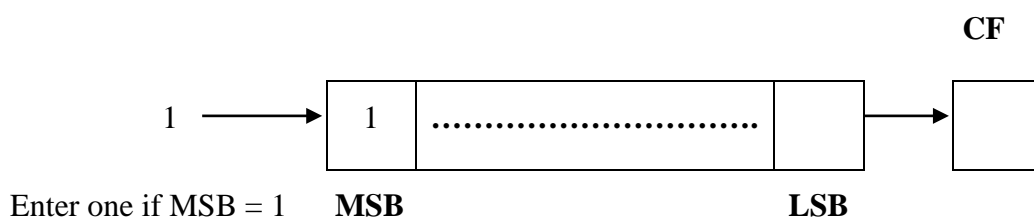
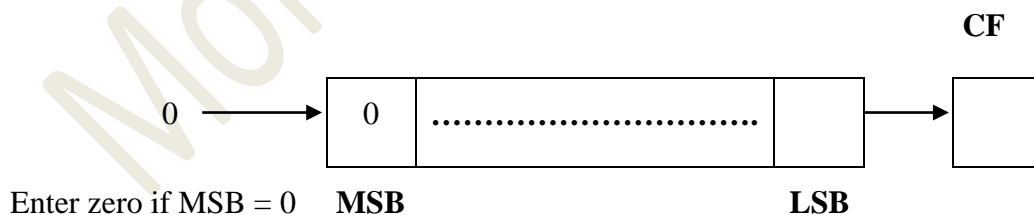


2. **SHL** op1,op2 . shift left op1(logical) by op2. The shift to the left can be done by enter 0 from the right and every bit jump to the other bit to the left & the (**MSB**) will be transferred to the carry flag.

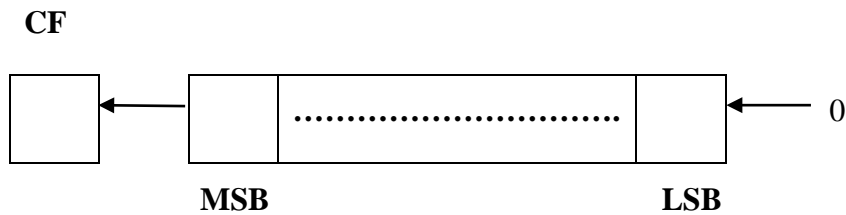


ii. Arithmetic Shifting

1. **SAR** op1,op2 . shift right (arithmetic). This instruction check the sign bit (**MSB**) & see if 0 enter zero from left if 1 enter one from left & keep the sign bit. i.e. copy the sign bit as many times as shift & the (**LSB**) transferred to the carry flag.

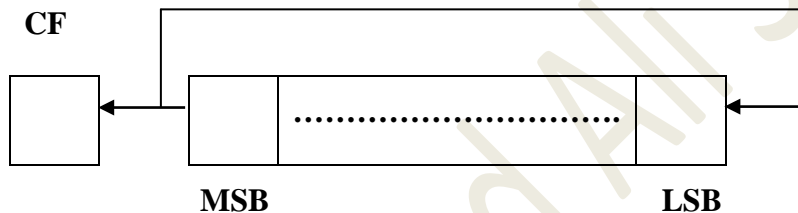


2. **SAL** op1, op2 . shift left op1(arithmetic) by op2. The shift to the left can be done by enter 0 from the right and every bit jump to the other bit to the left & the (**MSB**) will be transferred to the carry flag.

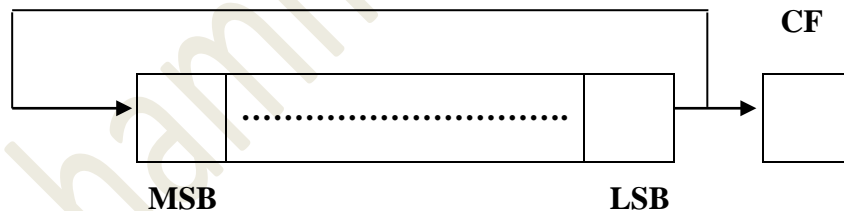


3.2.4.2 Rotating instructions

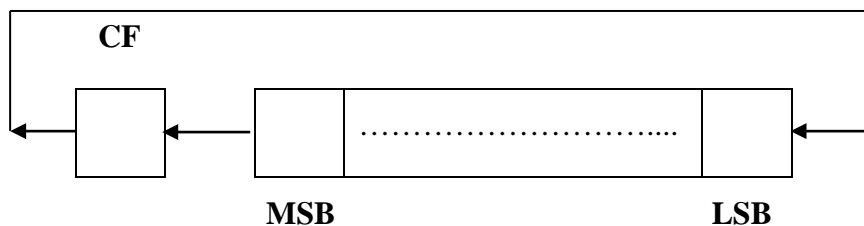
1. **ROL** op1, op2. Rotate data from right to left & the last bit (**MSB**) rotated from left transferred to carry flag.



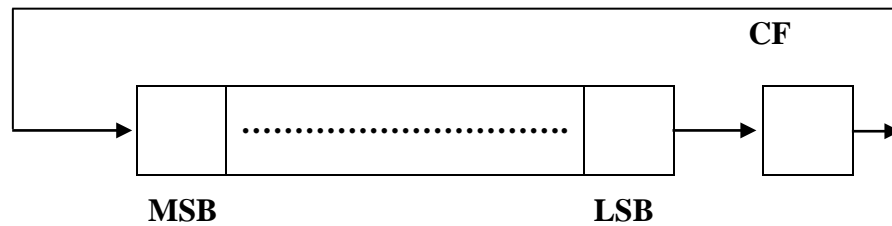
2. **ROR** op1, op2. Rotate data from left to right & the last bit (**LSB**) rotated from right transferred to carry flag.



3. **RCL** op1, op2. Rotate data from right to left through carry & the old value of the carry transferred to the (**LSB**).



4. **RCR** op1, op2. Rotate data from left to right through carry & the old value of the carry transferred to the (**MSB**).



Example 12: Let AH = 1000 0000 & CF = 0, what is the result of AH & CF after executing the following instructions?

INSTRUCTION	RESULT	CF
SHL AH,1		
SHR AH,1		
SAL AH,1		
SAR AH,1		
ROL AH,1		
ROR AH,1		
RCL AH,1		
RCR AH,1		

Important Notes:

1. *All* rotation & shifting operations change the carry flag.
2. If we want to rotate or shift data for more than one time we must use **CL** register for this purpose.

Example 13: Rotate reg. **AX** 6 times to the left without carry:

MOV CL,06H

ROL AX,CL

3. every **n** times shift to the right represent divide by 2^n .

Example 14: Let AL = 0000 1000 = 08H

SHR AL,1 AL = 0000 0100 = 04H

SHR AL,1 AL = 0000 0010 = 02H

SHR AL,1 AL = 0000 0001 = 01H

4. Every **n** times shift to the left represent multiply by 2^n .

Example 15: Let AL = 0001 0000 = 10H

SHL AL,1 AL = 0010 0000 = 20H

SHL AL,1 AL = 0100 0000 = 40H

SHL AL,1 AL = 1000 0000 = 80H

5. The **SHL**, **SAL** & **SHR** cause of losing data.

Example 16: Let BX = FFFFH

So after executing the following instructions:

MOV CL,10H

SHR BX,CL

BX = 0000H

6. The **SAR** cause either losing data or make all bits of data = 1

Example 17: Let AL = 0100 0100 = 44H

MOV CL,07

SAR AL,CL

Since **MSB** = 0 the result will be AL = 0000 0000 = 00H.

Example 18: Let AL = 1001 0110 = 96H

MOV CL,07

SAR AL,CL

Since **MSB** = 1 the result will be AL = 1111 1111 = FFH.

H.W. :Write a set of instructions to perform the following operations:

- ❖ 7AL without using MUL instruction. (AL = 05H).
- ❖ Copy the value of 3rd bit in M.L. [2000] to all bits of reg. BH.

Mohammed Ali Saffah