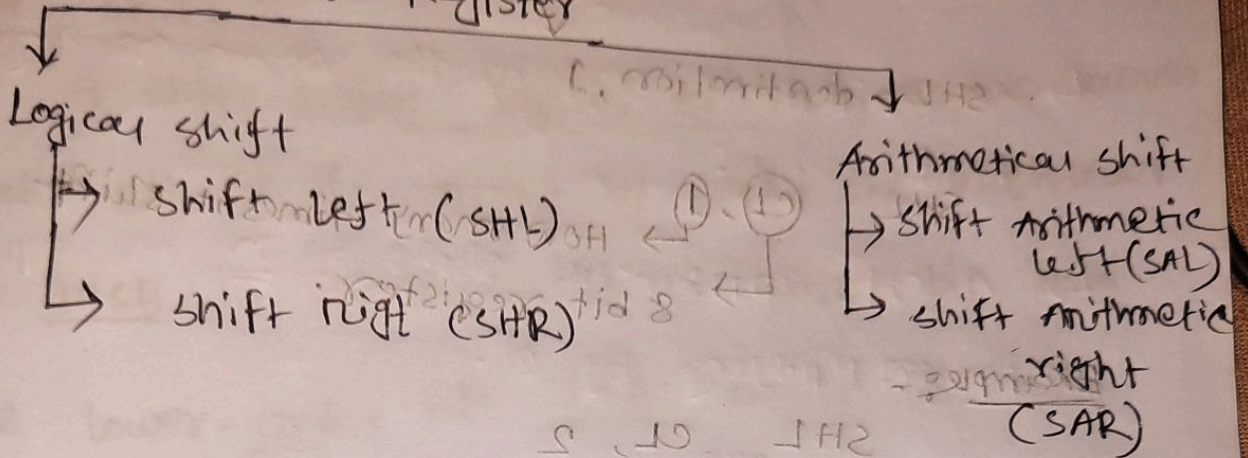


CH-07

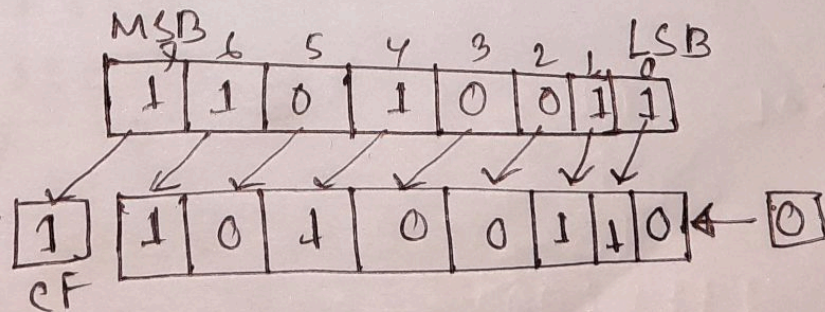
Shift Register



Left shift

A shift left logical of one position moves each bit to the left by one.

The low order bit LSB is replaced by a zero (0) bit and the high-order bit MSB moves to CF (Carry Flag)



shift left one position

* The format for a single shift is,

SHL destination, 1

① - ① → How many times it shift
8 bit register.

Example -

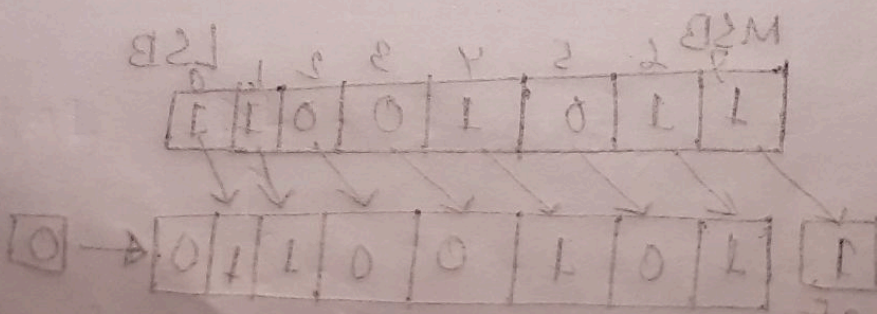
SHL CL, 2

cf ↑
 2 3

0 0010 0011 ← 0 1st shift
0 0100 0110 ← 0 2nd shift
0 1000 1100

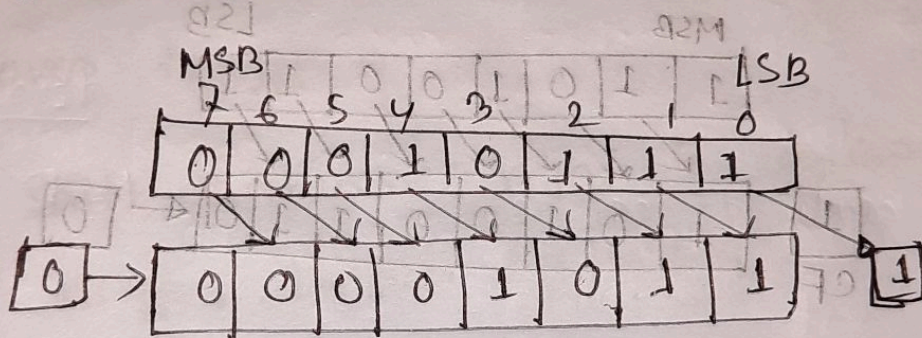
8 c ← CL

8c Load CL Register

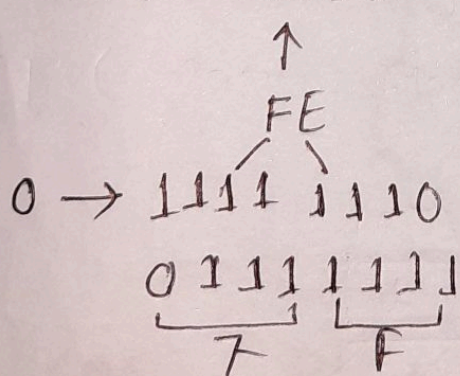


Right shift

A shift right logical of one position moves each bit to the right by one. The high-order bit MSB is replaced by a zero bit and the lower-order bit LSB moves to CF.

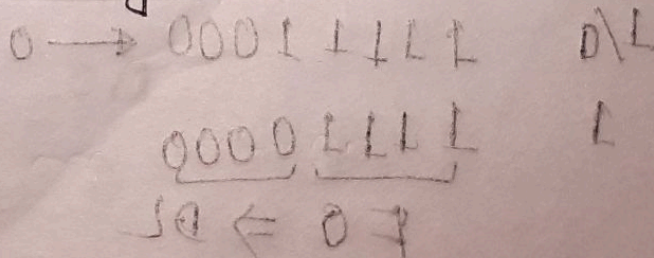


** SHR AH, 1



CF = 0/1

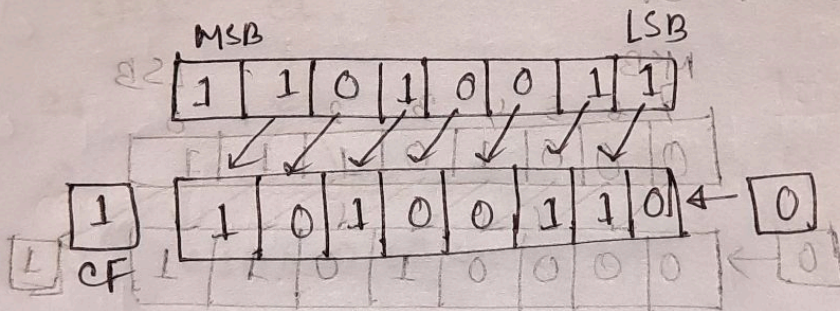
7F load AH Register

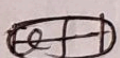


① Left Arithmetic shift

↳ An arithmetic left is the same as logical left shift.

↳ A left arithmetic shift by $\frac{n}{2}$ is equivalent to multiplying by 2^n .



*  SAL one position means $2^1 = 2$

$$11010011 = 211 \times 2$$

$$40100110 = 4220$$

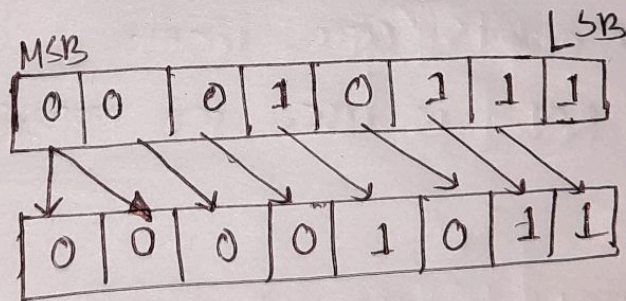
* SAL DL, 1

↑
 OF F8 11111000 ← 0
 1 11110000
 F8 ⇒ DL

⑤ Shift Arithmetic Right

↳ A shift arithmetic right of one position moves each bit to the right by one. The higher-order bit MSB is replaced by sign bit and the low-order bit LSB moves to CF.

↳ A shift arithmetic right is equivalent to integer division by 2^n .



** SAR DL, 02

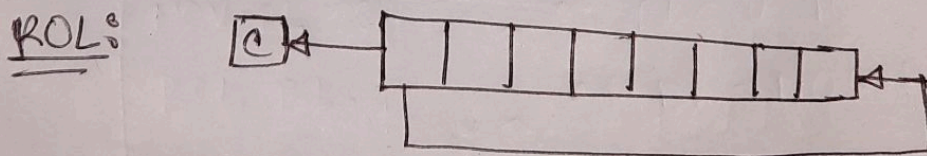
↑	
F2	CF
1	0/1
11110010	
11111001	0
11111100	1
FC ⇒ DL	

Rotate instruction (6)

- Rotate left (ROL)
- Rotate Right (ROR)
- Rotate carry left (RCL)
- Rotate carry right (RCR)

ROL Instruction

- ↳ ROL shifts each bit to the left.
- ↳ The highest bit is copied into both the carry flag and into the lowest bit
- ↳ No bits are lost.
- ↳ ROL is used for unsigned data.



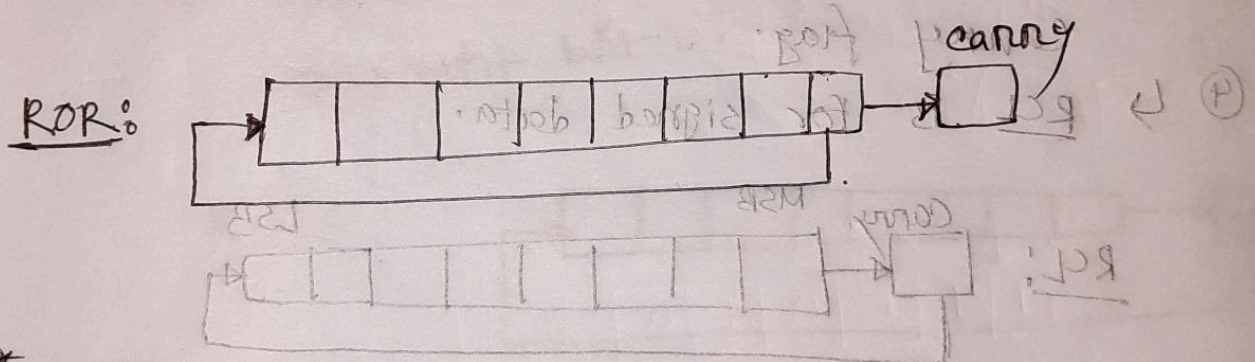
** ROL CH, 2

carry ↑
 28
 0 ← 00101000
 0 ← 01010000 First rotation
 0 10100000
 A0 → CH

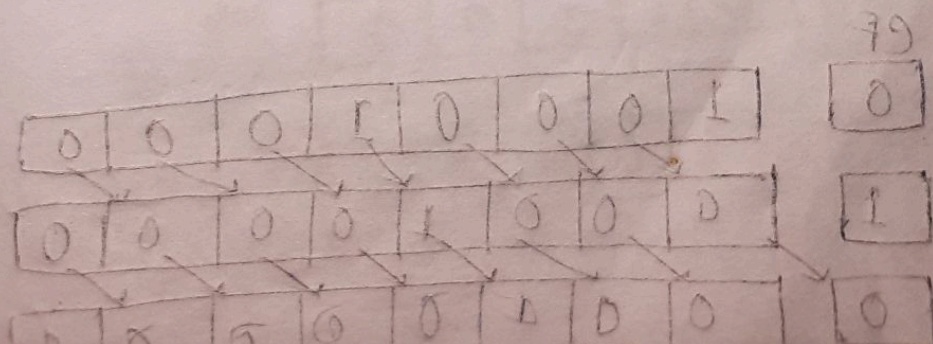
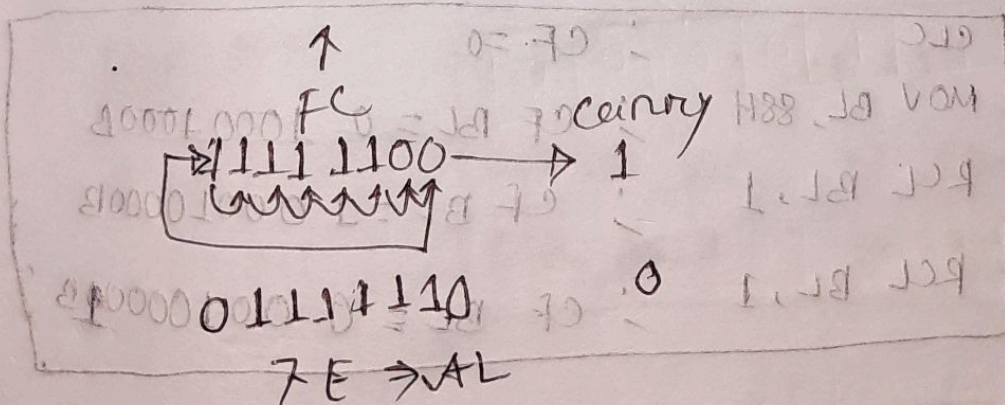
⑦ ⑧

Rotate Right instruction

- ↳ ROR (rotate right) shifts each bit to right
- ↳ The lowest bit is copied into both the carry flag and into the higher bit.
- ↳ No bit are loss.
- ↳ ROR is for unsigned data.



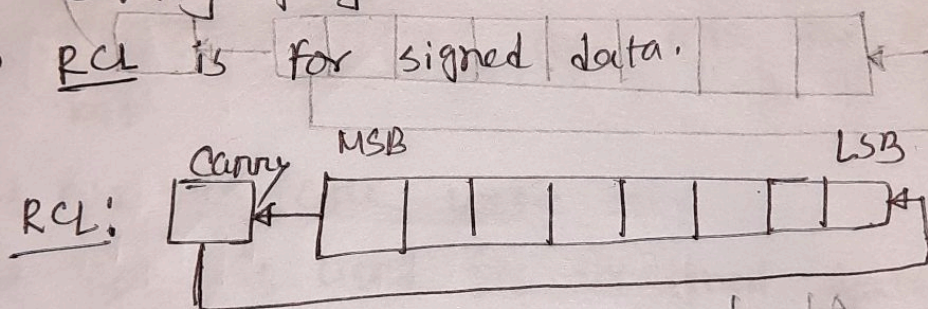
ROR AL, 1



⑧ Rotate carry ^{left} instruction

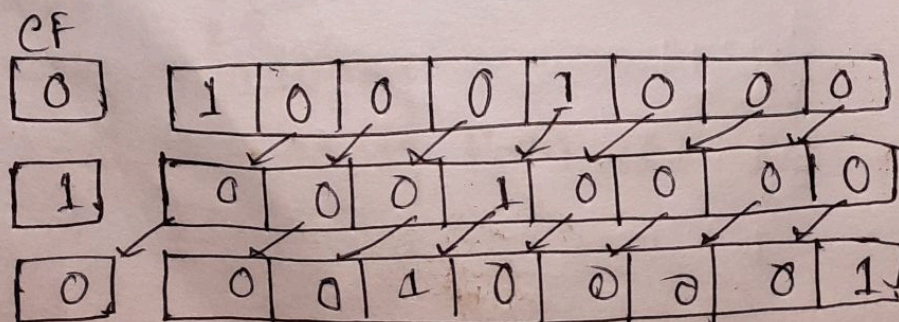
RCL:

- ① → RCL (rotate carry left) shifts each bit to the left.
- ② → copies the most significant bit to the carry flag.
- ③ → copies the carry flag to the least significant bit.
- ④ → RCL is for signed data.



Example

CLE	: CF = 0	↑
MOV BL, 88H	: CF BL = 0	10001000B
RCL BL, 1	: CF BL = 1	00010000B
RCL BL, 1	: CF BL = 0	01010000B



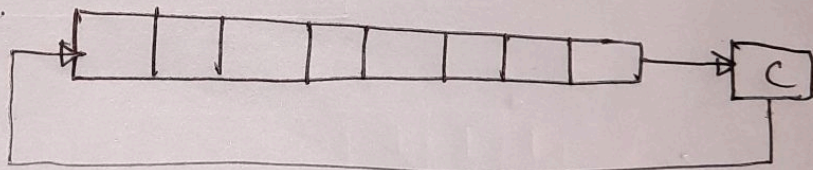
⑨

Rotate carry right instruction

RCR:

- ↳ RCR (rotate carry right) shifts each bit to the right.
- ↳ copies the least significant bit to the carry flag.
- ↳ copies the carry flag to the most significant bit.

RCR:



Example:

```

STC      ; CF = 1
MOV AH, 10H ; CF, AH = 00010000 1
RCR AH, 1 ; CF, AH = 10001000 0
  
```

