Republic of Iraq
Ministry of Higher Education and Scientific Research
Al-Furat Al-Awsat Technical University
Engineering Technical College/Najaf
Al Najaf Al Ashraf, 31001. Iraq.

8085 Microprocessor

Lecture 9

المدرس ضرغام الخفاف الاسدي

Third Year lecture notes

Avionics Engineering Dept.

Engineering Technical College/ NAJAF 2020-2021

Lecturer: Dhurgham Al-Khaffaf Alasady

Logic Transfer Group

The logic group of instructions operates on registers, memory and conditional flags. This group of instructions contains ANDing, ORing, XORing, complementing and comparing of data.

1- ANA reg/M* (AND register/memory) & *ANI data where *reg = A, B, C, D, E, H or L.* In this instruction each bit of the given register contents are ANDed with each bit of the accumulator contents (bit by bit). The result is saved in the accumulator. It does not affect the contents of the given register.

$$[A] \leftarrow [A].AND.[M_{H-L}]$$

$$[A] \leftarrow [A].AND.data$$

$$[A] \leftarrow [A].AND.[reg]$$

The possible combinations of this instruction are:

ANA A
ANA B
ANA C
ANA D
ANA E
ANA H
ANA L

The *ANA reg* instruction clears (resets) the CY flag and all other flags are modified according to the data conditions of the result. This instruction is one byte instruction.

Let A = 73 H C = C3 H CY = 1

before the execution of the instruction *ANA C*. Then after the instruction is executed we get:

A = 0 1 1 1 0 0 1 1 CY = 1

C = 1 1 0 0 0 0 1 1

A = 0 1 0 0 0 0 1 1 CY = 0

2- ORA reg/M* (OR register) & *ORI data where *reg = A, B, C, D, E, H or L.* In this instruction each bit of the given register contents are ORed with each bit of the accumulator contents (bit by bit). The result is saved in the accumulator. It does not affect the contents of the given register.

$$[A] \leftarrow [A] \text{ OR } data$$
$$[A] \leftarrow [A] \text{ OR } [M_{H-L}]$$
$$[A] \leftarrow [A] \text{ OR } [reg]$$

The possible combinations of this instruction are:

ORA A
ORA B
ORA C
ORA D
ORA E
ORA H
ORA L

The *ORA reg* instruction clears (resets) the CY flag and all other flags are modified according to the data conditions of the result. This instruction is one byte instruction.

Let A = 73 H C = C3 H CY = 1

before the execution of the instruction *ORA B*. Then after the instruction is executed we get:

A = 0 1 1 1 0 0 1 1 CY = 1

B = 1 1 0 0 0 0 1 1

A = 1 1 1 1 0 0 1 1 CY = 0

***3- XRI data* (Exclusive OR Immediate)** In this instruction each bit of the given data is immediately XORed with each bit of the accumulator contents (bit by bit). The result is stored in the accumulator.

$$[A] \leftarrow [A] \; XOR \; data$$

The carry flag will be reset after the execution of this instruction and other flags will be affected as per the result. For example, if A = AB H and CY = 1 before the execution of ANI 12 H, then after the execution of this instruction we have:

A =	1	0	1	0	1	0	1	1	CY = 1
data =	0	0	0	1	0	0	1	0	
A =	1	0	1	1	1	0	0	1	CY = 0

***4- CMA* (Complement Accumulator)** This is one byte implied addressing instruction as no operand is required with the instruction. The execution of this instruction inverts each bit of the accumulator contents and the result is saved in the accumulator. Basically it produces 1's complement of the accumulator contents. No flag is affected with this instruction.

$$[A] \leftarrow [\overline{A}]$$

For example, if A = 0B H before the execution of *CMA* instruction, then after the execution of this instruction we have:

A =	0	0	0	0	1	0	1	1
=	1	1	1	1	0	1	0	0
A =	1	1	1	1	0	1	0	0

(F4)

Compare Instructions

***5- CMP reg* (Compare Register/memory)** The contents of the given register are compared with the accumulator contents. In fact the contents of the register are subtracted from the contents of accumulator and the accumulator contents remain unchanged. However, as a result of the subtraction the flags are modified as per the result.

The zero flag Z is set if [*A*] = [*reg/M*] otherwise reset.

Similarly, carry flag CY is set if [*A*]< [*reg/M*] otherwise reset.

***6- CPI data* (Compare immediate with data)** It is similar to CMP instruction with the difference that the data is directly given with the instruction. In this instruction the given data

***7- CMC* (Complement the carry)** This instruction complements the carry flag.

$$CY \leftarrow \overline{CY}$$

If CY = 1 before the execution of *CMC* instruction, the carry flag will be reset (CY = 0) after the execution of this instruction. Similarly, If CY = 0 before the execution of *CMC* instruction, the carry flag will be set (CY = 1) after the execution of this instruction. In this only carry flag will be affected and all other flags will not be affected.

8- *STC* (Set the carry) It sets the carry flag.

$$CY \leftarrow 1$$

The carry flag will be set (CY = 1) irrespective of the carry flag is set or reset before the execution of this instruction *STC*. Only carry flag gets affected with this instruction.

Thank you