2-arithmetic instructions

The instruction set of the 8086 microprocessor contains an extensive set of arithmetic instructions such as addition subtraction multiplication and division operations

***execution of the arithmetic instructions affect the status of the flag register bits the flags that are affected by the arithmetic instructions are:- CF ,AF ,SF ,ZF ,PF OF.**

Addition instructions (ADD,ADC)

the addition instruction has two forms the first ADD that adds a source operand with the destination operand ;and the second form ADC .that adds the source and destination operands with the carry flags ,as shown below .

mnemonics	meaning	format	operation	Flags affected
ADD	Addition	ADD D,S	(S)+(D) (D) Carry (CF)	CF,AF,ZF
ADC	Addition with carry	ADC D,S	(S)+(D)+(CF) Carry (CF)	SF,PF,OF

The source/ destination operands type are;-

Reg ↔ reg

Reg ↔ memory

Reg ← immediate

Memory ← immediate

Ex:-

ADD AX,BX (AX)=(AX)+(BX)

ADC DI,[SI] (DI)=(DI)+((DS)0+(SI))+CF

ADD CL,77H (CL)=(CL)+77H

ADC [BX+2],245FH ((DS)0+(BX)+2)=((DS)0+(BX)+2)=245F+CF

Ex:-write an assembly language program that will adds two consecutive bytes of data stored in memory data segment with offset location NUMB and NUMB+1 with the contents of al register

Sol/ LEA DI,NUMB ;load effective address

MOV AL,0 ;clear AL

ADD AL,[DI] ;add NUMB

ADD AL ,[DI+1] ;add NUMB+1

Ex:-write an assembly language program that adds a 32bit number stored in Bx and AX with a 32 bit number stored at Dx and CX

sol/ ADD AX,CX

ADC BX,DX

H.W write an assembly language program that adds two 16-bit numbers stored ad memory data segment with offset specified by DI reg .then store the result of addition at memory location followed the two consecutive numbers?

Subtraction instruction (SUB,SBB)

The subtraction instruction are similar to the addition instruction and its basic format shown below:

Mnemonic	meaning	format	operation	Flags affected
SUB SBB	Subtract Subtract with borrow	SUB D,S SBB D,S	(D)-(S)→(D) (D)-(S)-(CF)→(D)	CF,AF,ZF,OF,SF,PF

The source/ destination operands type are

reg↔reg SUB CL , BL CL=CL_BL

reg ↔ memory SUB DX , [DI]

reg ← immediate SUB AX , $23A5_H$

memory ← immediate SUB [SI] , 3514_H

increment &decrement instructions

the increment INC or decrement DEC instruction are used to add or subtract one to from register or memory location

mnemonic	meaning	format	operation	Flags affected
INC DEC	Increment by 1 Decrement by 1	INC D DEC D	(D)+1→(D) (D)-1→(D)	CF,AF,ZF,OF,SF,PF

the destination operand can be one of the following type:-

reg 16 INC AX DEC SI

reg 8 INC CL DEC AH

Memory INC [DI] DEC [BP]

((OS)0+(DI))=((DS)0+(DI))+1 ((SS)0+(BP))=((SS)0+(BP))-1

Ex/assume the contents of registers AX,BX, and CX are f014h, 2007H ,1623H respectively ;and the carry flag is 0 .what will be its values after executing the following set of instructions

ADD AX , CX

INC CX

SBB BX , CX

Sol/ executing the 1'st instruction result:-

(AX)=(AX)+(CX)→ (AX)=F014+1623=0632H

(CF)=1

Executing the 2'nd instruction give:-

(CX)=(CX)+1→ (CX)=1623+1=1624H

executing the 1'st instruction result:-

(BX)=(BX)-(CX)-(CF)→ (BX)=2007-1624-1=09E2H

Therefore finally

(AX)=0637H,(BX)=09E2H,(CX)=1624H,CF=1

H.W if the initial contents of DS is 1000H .DI is 2000H AL is 34H and memory location 12000H is 9AH.what is its contents after executing the following code assume CF=1initially.

ADC [DI] , AL

DEC AL

SUB AL,19H

Negate instruction (NEG)

Execution of this instruction causes the value of the specified operand to be replaced by its negative. This is actually done by subtracting from zero using 2's complement the specified operand .

Mnemonic	*Meaning*	*Format*	*Operation*	*Flag affected*
NEG	*Negate*	*NEG D*	*0 –(D) ⟶(D)* *1 ⟶(cf)*	*Cf , zf , Af , pf , sf , of*

the operand type can be a register or a memory location

EX/ NEG AX

NEG [si]

Example / assuming that register BX contains 3AH , what is the result of executing the instruction NEG BX

Sol /

000016 – (BX) = 000016 + 2's complement of (003A)16

=0000 + FFC6 =FFC6H

(BX) = FFC6

In the other words , we can say that the negate instruction takes the 2's complement of the operand .