ARITHMETIC GROUP

INTRODUCTION:

This group of instructions perform arithmetic operations such as addition, subtraction, increment and decrement.

ADDITION:- Any 8-bit number, or the contents of a register or the contents of a memory location can be added to the contents of the accumulator and the sum is stored in the accumulator. No two other 8 bit registers can be added directly.

SUBTRACTION:- Any 8 bit number, or the contents of a register or the contents of a memory location can be subtracted from the contents of the accumulator and the results stored in the accumulator. The subtraction is performed in 2's complement, and the results if negative, are expressed in 2's complement. No two other registers can be subtracted directly.

INCREMENT/DECREMENT:- The 8 bit contents of a register or a memory location can be incremented or decremented by 1. Similarly, the 16 bit contents of a register pair (such as BC) can be incremented or decremented by 1. These increment and decrement operations differ from addition and subtraction in an important way i.e they can be performed in any one of the registers or in a memory location.

The arithmetic group of instructions include following instructions:

1. ADD R
2. ADD M
3. ADC R
4. ADC M
5. ADI data
6. ACI data
7. DAD Rp
8. SUB R
9. SUB M
10. SBB R
11. SBB M
12. SUI data
13. SBI data
14. DAA
15. INR R
16. INR M
17. DCR R
18. DCR M
19. INX Rp
20. DCX Rp

ADDITION

1. ADD R

Mnemonic	ADD R
Operation	A ← A + R
No. of Bytes	1 byte
Machine cycle	1 (OF)

Algorithm	A ← A + R
Flags	All the flags are modified
Addr. mode	Register addressing mode
T-states	4

DESCRIPTION :- Add register R contents to accumulator

- This instruction adds the contents of register R and accumulator. The result is stored in the accumulator.
- The register R can be any general purpose register like A, B, C, D, E, H or L.
- In addition to the result in accumulator, all the flags are modified to reflect the result of operation.

EXAMPLE: ADD C A ← A + C

- Let A = 47H, C = 51H and instruction ADD C is executed

$$\begin{array}{lrrr} & A & 0100\ 0111 & = 47 \\ + & C & 0101\ 0001 & = 51 \\ \hline & & 1001\ 1000 & = 98 \end{array}$$

The addition is performed in hexadecimal no system. The status of the flags will be as follows :

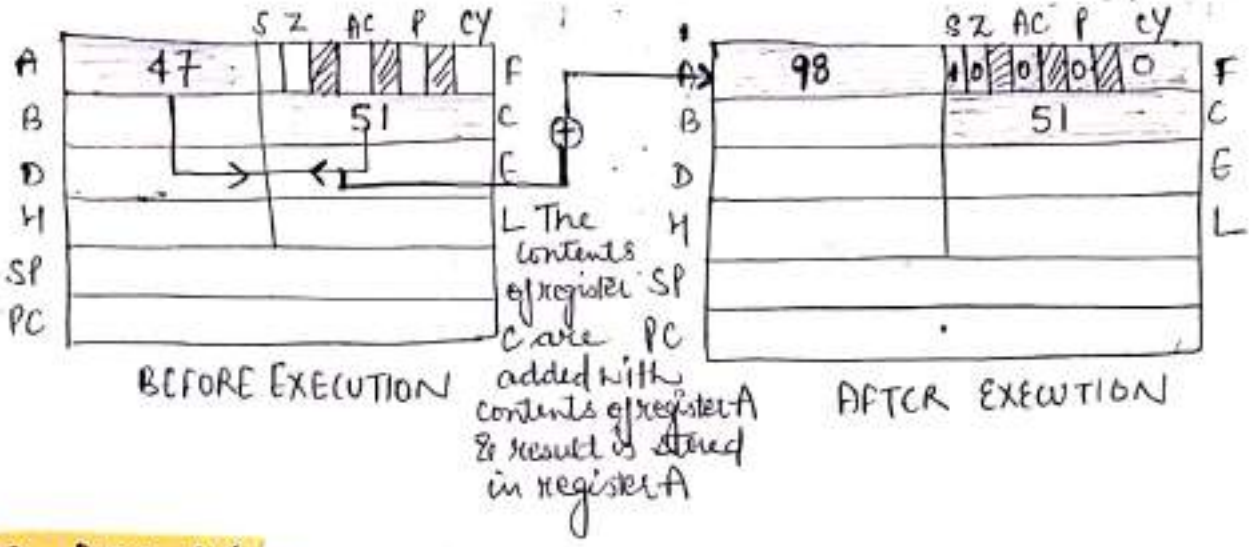
i) Zero : The zero flag is reset as the result is not equal to 0.

ii) Parity : The parity flag is reset as the result contains odd numbers of 1's.

iii) Sign : Sign flag is set as the MSB bit of the result is 1.

iv) Carry : The carry flag is reset as there is no carry from B_7 bit of result to next stage.

v) Auxiliary Carry : The AC flag is reset as there is no carry from B_3 bit to B_4 bit of result.



2. ADD M

Mnemonic	ADD M
Operation	A = A+M or A = A+(HL)
No. of Bytes	1 byte
Machine cycles	2 (OF + MR)

Algo	A ← A+M or A ← A+(HL)
Flags	All flags are modified
Addr. Mode	Indirect addressing mode
T-states	7 (4+3)

DESCRIPTION :- Add memory location contents to accumulator

- This instruction uses the HL register pair as a memory pointer. The contents of memory location addressed by HL pair are added to the contents of the accumulator. The result is stored in the accumulator.
- To reflect the status of result all the flags are modified.
- The contents of memory location remain unchanged.

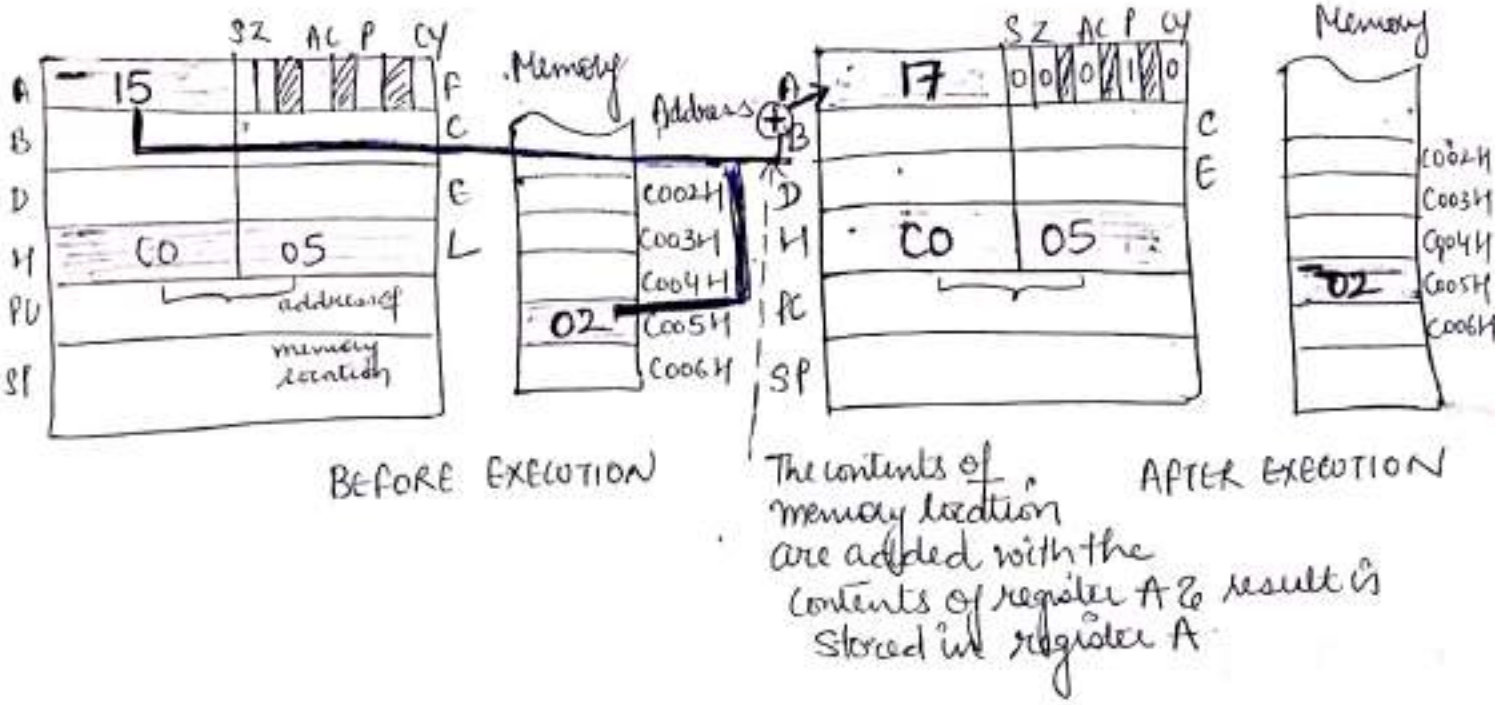
EXAMPLE : ADD M

- Let A = 15H H = C0H, L = 05H, at memory location C005 : 02H is stored and the instruction ADD M is executed.

$$A + (HL) \rightarrow A$$

$$15 + 02 \rightarrow 17$$

$$A = 17H$$



3. ADC R

Mnemonic	ADC R
Operation	A = A + R + CY
No of Bytes	1 bytes
Machine cycles	1 (OF)

Algorithm	A ← A + R + CY
flags	All flags are modified
Addr. Mode	Register addressing mode
T-states	4

DESCRIPTION :- Add register R and carry flag contents to accumulator.

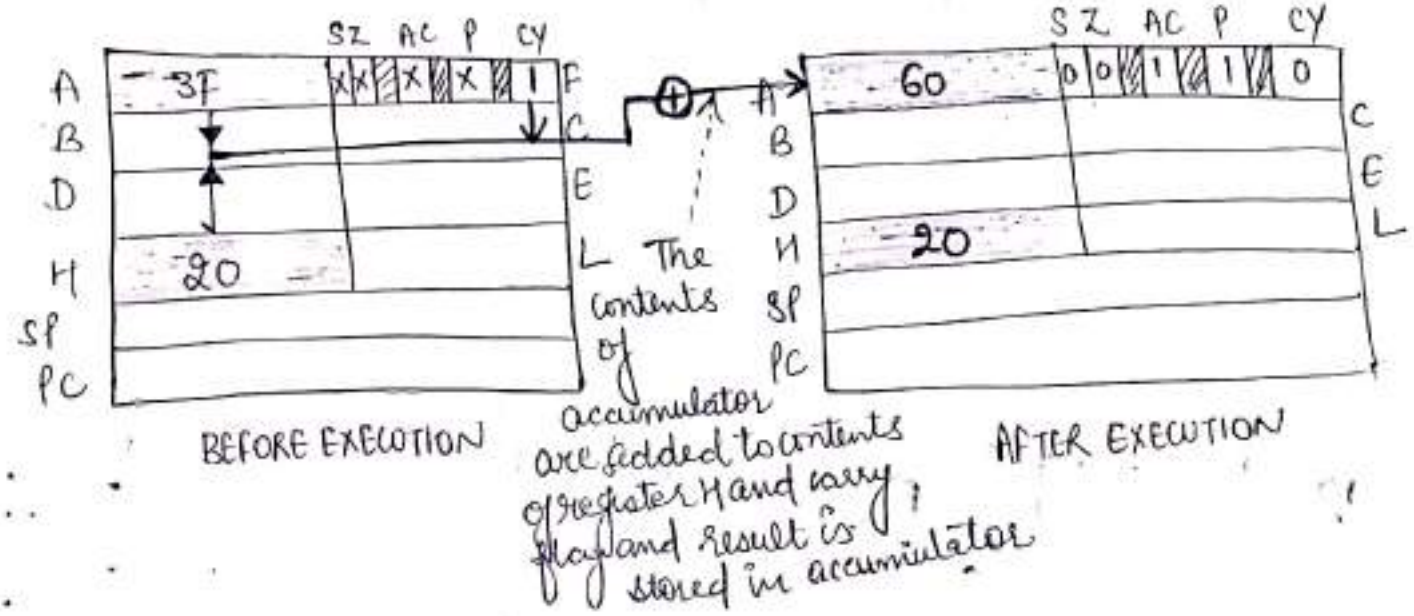
- This instruction adds the contents of the specified register R to the contents of accumulator with carry. The result is stored in accumulator.
- The register R is any general purpose register like A, B, C, D, E, H or L.

EXAMPLE : ADC H A = A + H + CY

- Let A = 3FH, H = 20H, CY = 1 and the instruction ADC H is executed

```
  A    0011 1111
+ H    0010 0000
+ CY            1
-------------------
  01100000  (Result)
```

- The flag status is as follows:
  CY=0, Z=0, P=1, AC=1, S=0
- The carry flag will be reset becz the addition has not produced a carry
- The parity flag is set becz the parity of addition is even parity. The auxiliary carry flag is set as there is carry from B_3 bit to B_4 bit of result.



4. ADC M

Mnemonic	ADC M
Operation	A = A+M+CY or A = A+(HL)+CY
No. of Bytes	1byte
Machine cycles	2(OF+MR)

Algorithm	A ← A+M+CY A ← A+(HL)+CY
Flags	All flags are modified
Addr. Mode	Indirect addressing Mode
T-states	7(4+3)

DESCRIPTION:- Add data in memory to accumulator with carry.

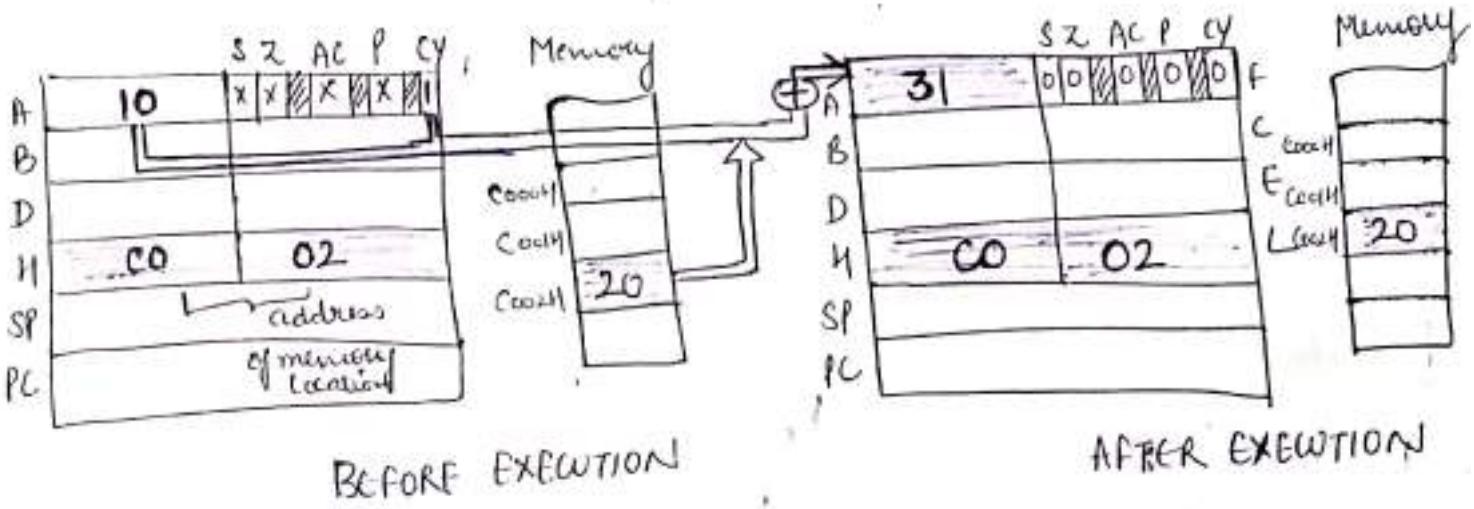
- This instruction uses the HL pair as memory pointer. The contents of the memory location addressed by the HL register pair and carry flag are added with the accumulator contents. The result is stored in the accumulator

EXAMPLE ADC M A = A+(HL)+CY

- Let A = 10H, H = C0H, L = 02H, CY = 1 and at memory location C002 : 20H is stored and the instruction ADC M. is executed.

$$\begin{array}{rll} & A & 0001\ 0000 \\ +\ (HL) & & 0010\ 0000 \\ +\ (CY) & & 1 \\ \hline & A & 0011\ 0001 \quad \text{(Result)} \end{array}$$

- The flag status is as follow :
  - CY = 0 as no carry is generated
  - Z = 0 Result is not zero
  - P = 0 as odd number of 1's are present in result
  - S = 0 as B_7 bit is zero.
  - AC = 0 as no carry from B_3 to B_4 bit.



5. ADI data

Mnemonic	ADI Data
Operation	A = A + data
No. of Bytes	2 bytes FIRST BYTE : opcode, SECOND BYTE : 8 bit data
Machine cycles	2 (OF + MR)

Algorithm	A ← A + data
Flags	All the flags are modified
Addr. Mode	Immediate Addressing Mode
T-State	7 (4+3)

DESCRIPTION :- Add immediate 8 bit data to accumulator.

- This instruction adds the 8-bit data given within the

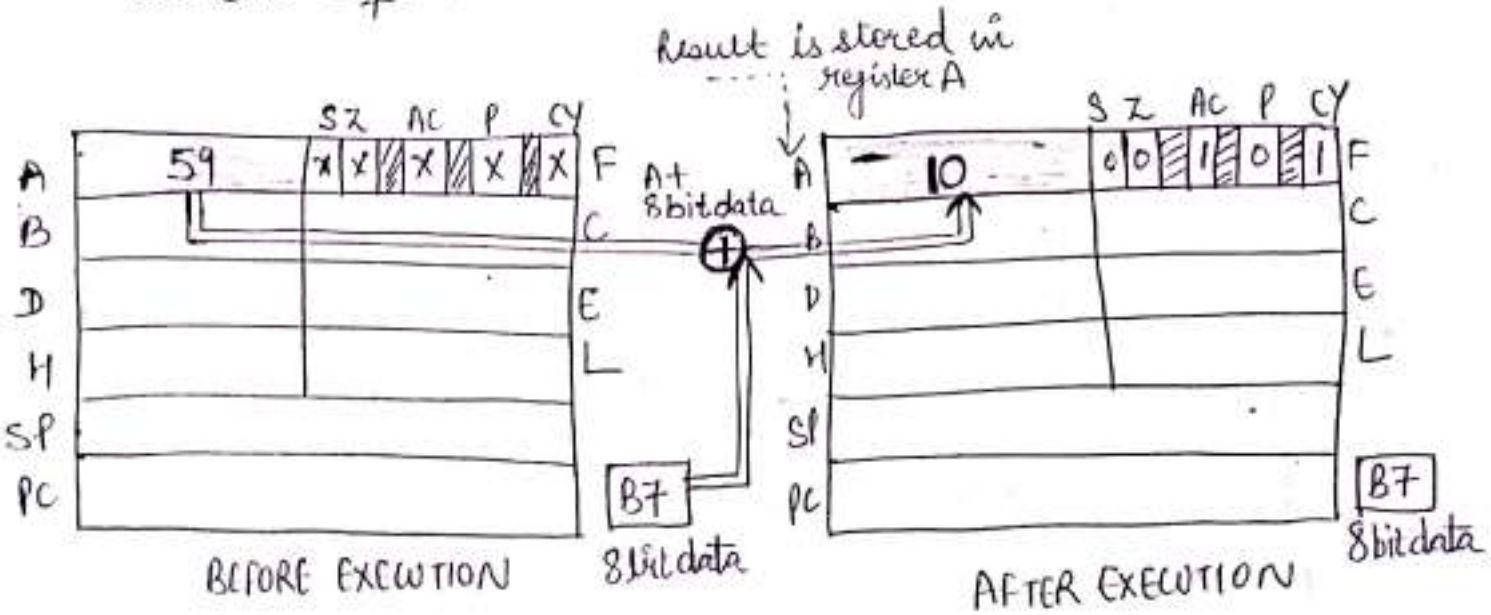
instruction to the data in accumulator. The result is stored in the accumulator.

EXAMPLE :- ADI B7H

- This instruction will add B7H to the accumulator & store result in the accumulator.
- Let A = 59H and the instruction ADI B7H is executed.

A	0101	1001
+8bit data	1011	0111
A [1]	0001	0000
CY		

- The result is A = 10H.
- The flag status is as follows:
  CY = 1 as carry is generated from B_7 bit
  Z = 0 as result is not zero
  AC = 1 as carry is generated from B_3 bit to B_4 bit
  P = 0 as odd number of 1's are present in result
  S = 0 as B_7 bit is zero.



6. ACI data

Mnemonic	ACI data
Operation	A = A + 8 bit data + CY
No of Bytes	2 bytes (first byte: opcode, second byte: 8 bit data)
Machine cycles	2 (OF + MR)

Algorithm	A ← A + data + CY
Flags	All flags are modified
Addr. Mode	Immediate addressing mode.
T-states	7 (4+3)

DESCRIPTION :- Add immediate data and carry to accumulator

- This instruction adds the immediate data, carry flag with the contents of accumulator and stores the result in the accumulator.

EXAMPLE :- ACI 20H

- This instruction will add 8 bit data, 20H to the contents of accumulator and result is stored in the accumulator.

Let A = C0H CY = 0

A	1100	0000
+ 8 bit data	0010	0000
+ CY		0
A	1110	0000 (Result)

- The flag status is as follows:

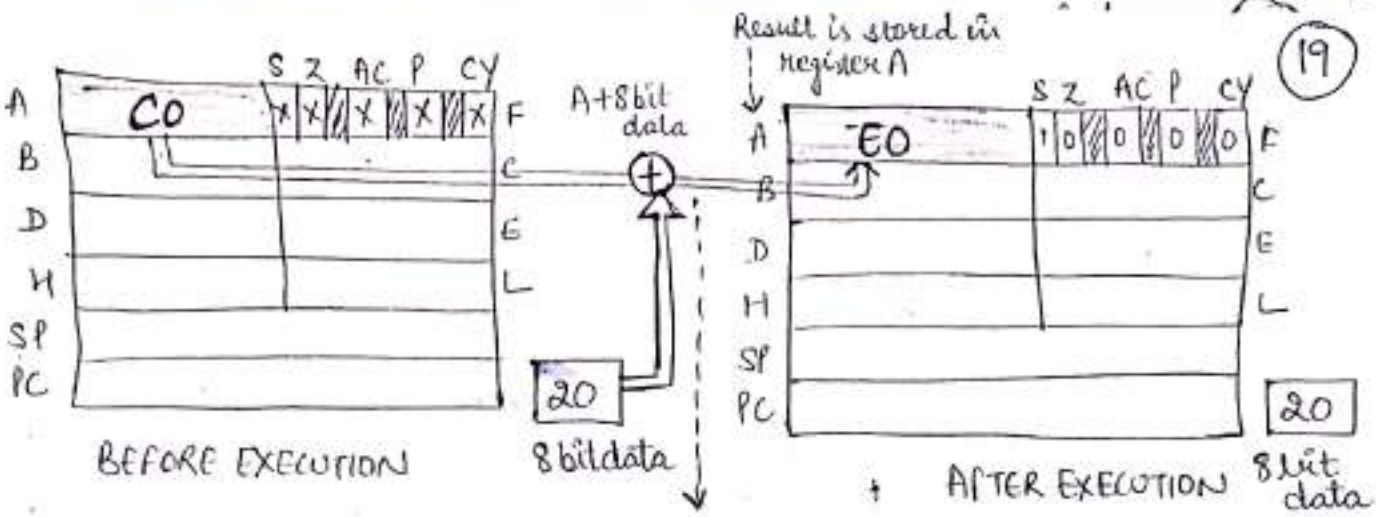
CY = 0 as no carry is generated from B_7 bit

P = 0 as odd number of 1's are present in the result

AC = 0 as no carry is generated from B_3 to B_4 bit

Z = 0 as result is not zero

S = 1 as B_7 bit is set.



The data 20H is added to the contents of register A along with carry and the result is stored in register A.

7. DAD Rp

Mnemonic	DAD Rp
Operation	HL = HL + Rp
No of Bytes	1 byte
Machine cycles	3 (OF + BI + BI)

Algorithm	HL ← HL + Rp
Flags	
Addr. Mode	Register Addressing Mode
T-States	10 (4+3+3)

DESCRIPTION:- Add the specified register pair to HL pair.

- This instruction adds the contents of the specified register pair to HL pair and stores the result in HL pair.
- The register pair Rp is a 16 bit register pair like BC, DE, HL or stack pointer.
- Only higher order register is to be specified for register pair within the instruction.
- Only carry flag is modified to reflect the status of result.

EXAMPLE:- DAD D

- Let D = 20H, E = 35H, H = 80H, L = 45H, F = 10X1X0X1

and instruction DAD D is executed.

DE 2035 H
HL + 8045 H

HL A07A (Result)

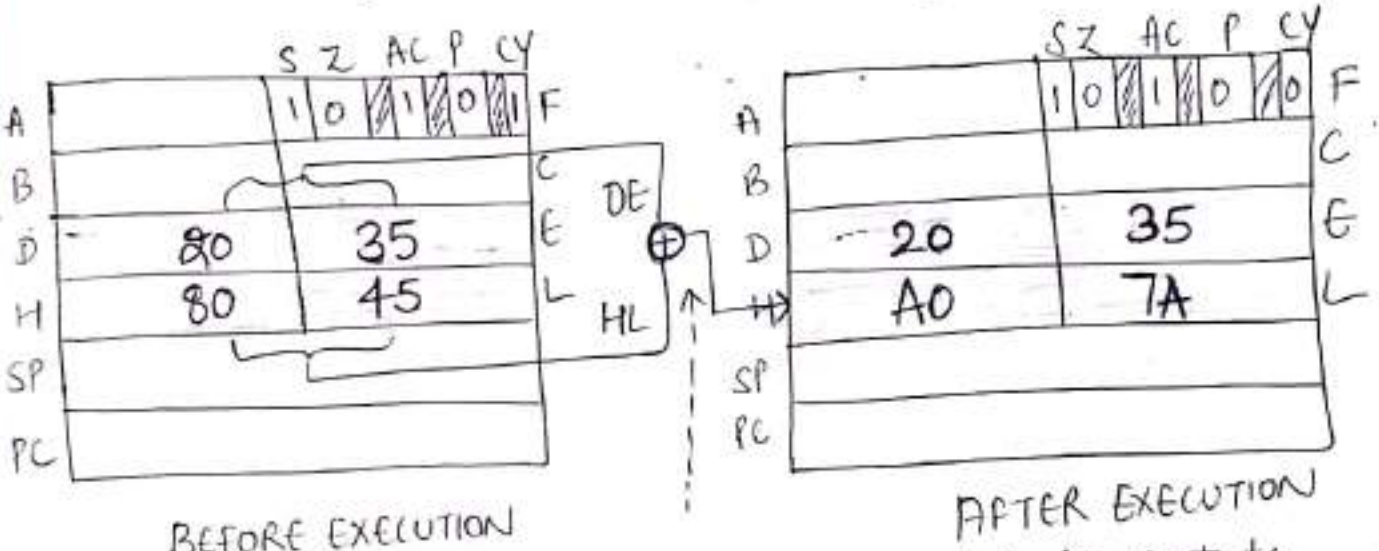
- The carry flag will be reset as there is no carry from By bit
- HL will contain A07A as result.

- The operation is a 16 bit addition. But as 8085 is a slow processor, it requires additional time to complete the instruction execution. In that time 8085 is busy in execution so it enters bus idle machine cycle. During the bus idle machine cycle no operation on the bus is performed. The fetching of next instruction is stopped by not giving control signal RD and ALE. PC is not incremented by 1.

In all total 3 machine cycles required:

(1) OPCODE fetch (2) Bus idle machine cycle

(3) Bus idle machine cycle.



BEFORE EXECUTION AFTER EXECUTION

The contents of register pair DE are added with contents of register pair HL and the result is stored in register pair HL

SUBTRACTION

1. SUB R

Mnemonic	SUB R
operation	A = A - R
No of Bytes	1 byte
Machine cycles	1 (OF)

Algorithm	A ← A - R
Flags	All flags are modified.
Addr. Mode	Register addressing mode.
T-states	4

DESCRIPTION:- Subtract register from accumulator

- This instruction subtracts the contents of the specified register from the contents of the accumulator and the result is stored in accumulator.
- The contents of register R are not altered.
- The Register R can be any general purpose register like A, B, C, D, E, H or L.

EXAMPLE: SUB B

Let A = 37H, B = 40H and the instruction SUB B is executed.

B = 0100 0000

2's complement of B = 1100 0000

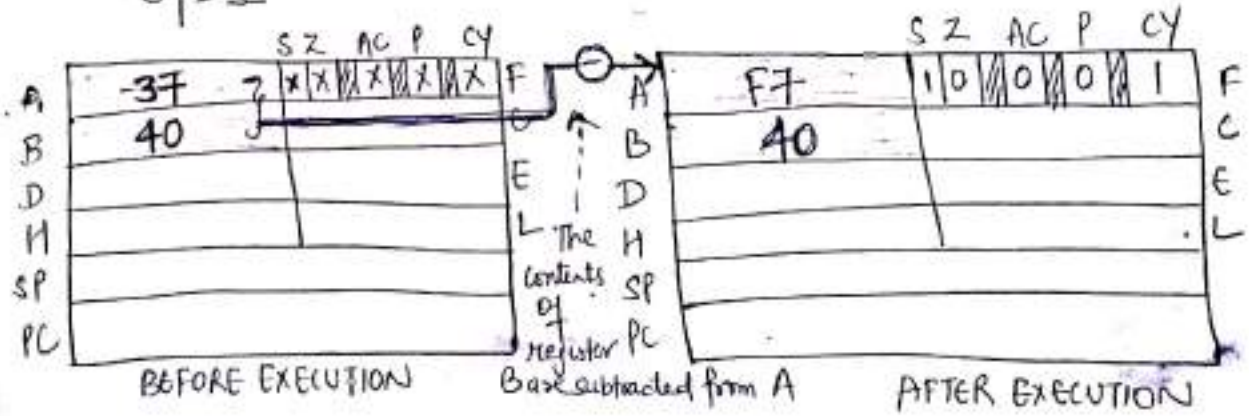
A = 0011 0111

2's complement of B = +1100 0000

A = 1111 0111

- No carry is generated, so CY = 0. The microprocessor complements the carry. So CY = 1. This indicates that result is negative and in 2's complement form.

- The flag status will be as follow:
  - Z = 0 as result is not zero.
  - P = 0 as odd no of 1's are present in the result.
  - AC = 0 as there is no carry from B_3 bit to B_4 bit.
  - S = 1 as B_7 bit is set.
  - CY = 1



Now Let A = 40H, B = 37H and the instruction SUB B is executed

B	0011	0111
2's complement of B	1100	1001
A	0100	0000
2's complement of B +	1100	1001
A [1] CY	0000	1001

Carry is generated. Hence, the microprocessor complements the carry, so CY = 0. CY = 0 represents that the result is positive and in normal form. The result in accumulator will be 09H.

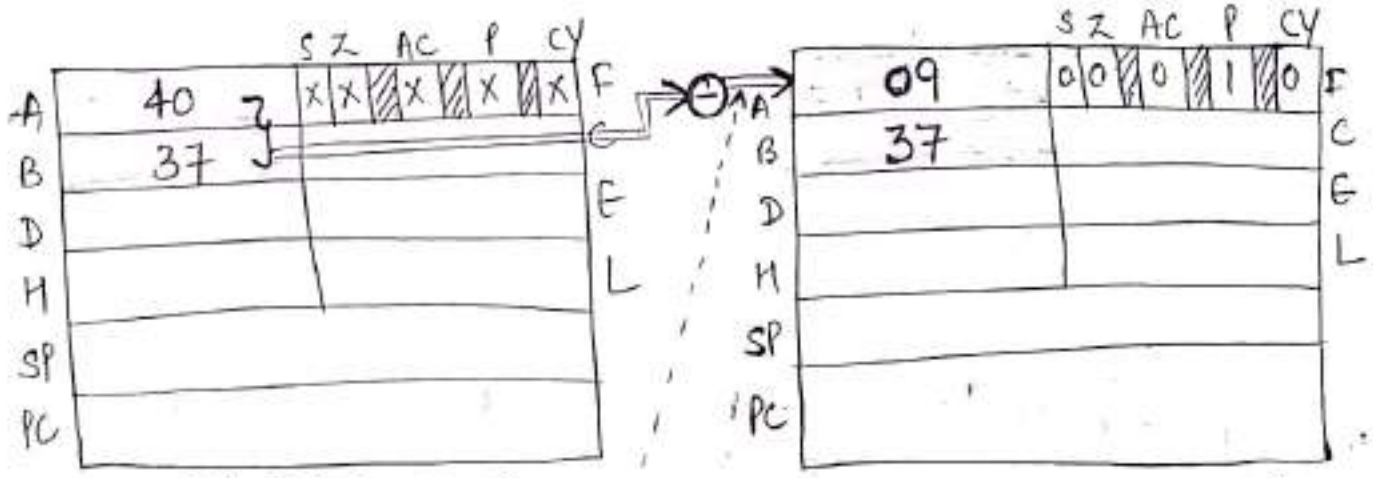
The status of flags will be as follows:

S = 0 as B_7 bit is zero

P = 1 as even number of 1's are present in result

AC = 0 as there is no carry from B_3 bit to B_4 bit.

CY = 0 Z = 0 as result is not zero



The contents of register B are subtracted from contents of register A and result is stored in register A.

2: SUB M

Mnemonic	SUB M
Operation	A = A - M or A = A - (HL)
No. of Bytes	1 byte
Machine Cycles	2 (OF + MR)

Algorithm	A ← A - M
Flags	All flags are modified
Addr. Mode	Indirect addressing Mode
T-states	7 (4+3)

DESCRIPTION:- Subtract data in memory from accumulator

- This instruction will subtract the contents of memory location pointed by HL register pair from the contents of the accumulator. The result is stored in the accumulator. The HL register pair is used as memory pointer.
- The subtraction is done in the same way as SUB R instruction

EXAMPLE:- SUB M A = A - (HL)

Let A = 50H, H = C2H, L = 00H at the memory location: C200H : 20H is stored and instruction SUB M is executed.

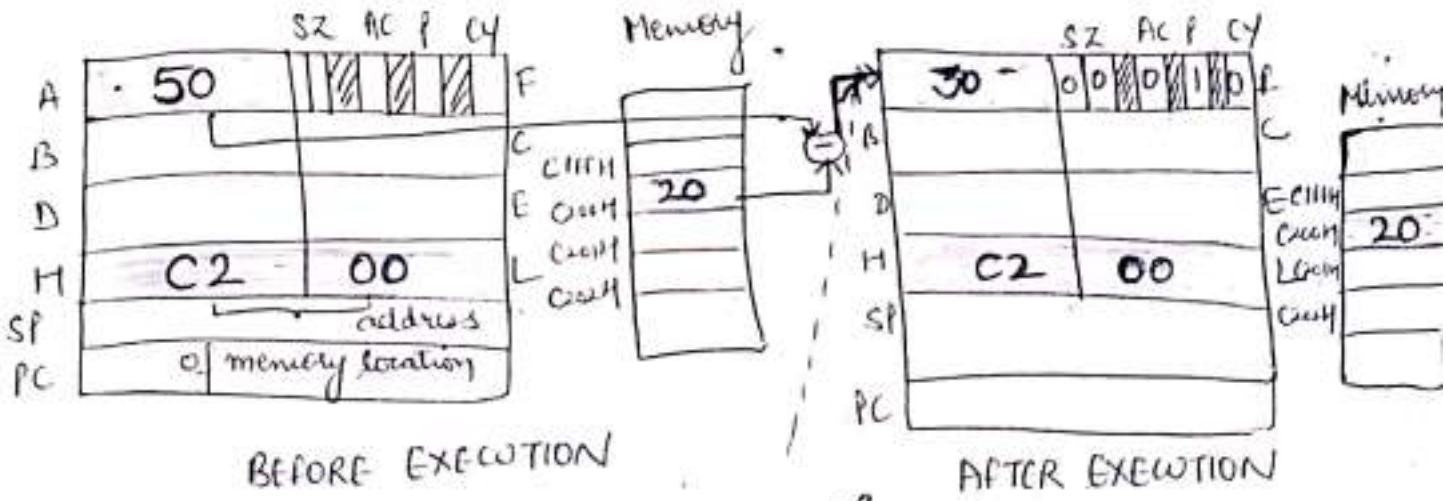
(HL)	0010 0000
2's complement of data at memory location	1110 0000
A	0101 0000
2's complement of data at memory location +	1110 0000
	1 0011 0000 (Result)
	CY

- Carry is generated. Hence the microprocessor complements the carry so CY = 0. This represents that the result is positive & in normal form.
- The status of the flag will be as follow:

Z = 0 as result is not zero S = 0 as B_7 bit is not set

CY = 0 AC = 0 as no carry is generated from B_3 to B_4 bit.

P = 1 as result contains even no of 1's.



The contents of memory location C200H are subtracted from the register A and result is stored in register A

3. SBB R

Mnemonic	SBB R
Operation	A = A - R - CY
No of Bytes	1 byte
Machine cycles	1 (OF)

Algorithm	A ← A - R - CY
Flags	All flags are modified
Addr Mode	Register addressing mode
T-states	4

DESCRIPTION:- Subtract register and borrow flag from accumulator.

- This instruction subtracts register R and borrow flag contents from the contents of the accumulator. The result is placed in the accumulator
- The carry flag is called as borrow flag for subtraction related instructions.
- The register R may be any general purpose register like A, B, C, D, E, H or L.

EXAMPLE :- SBB C A = A - C - CY

- The subtraction is performed using 2's complement method. First the borrow flag is added to register C and then this number is subtracted from the accumulator.

Let A = 37H, C = 3FH and carry i.e borrow flag is set and SBB instruction is executed:

```
C = 0011 1111
Borrow = +        1
         ---------
         0100 0000
         ---------
2's complement 1100 0000
          +
A 0011 0111
  ---------
A 1111 0111
  ---------
```

- No carry is generated Hence the microprocessor complements the carry, so carry flag is set, Hence borrow flag is set. This represents that the result is negative and in 2's complement form.

- The status of the flags will be as follows:

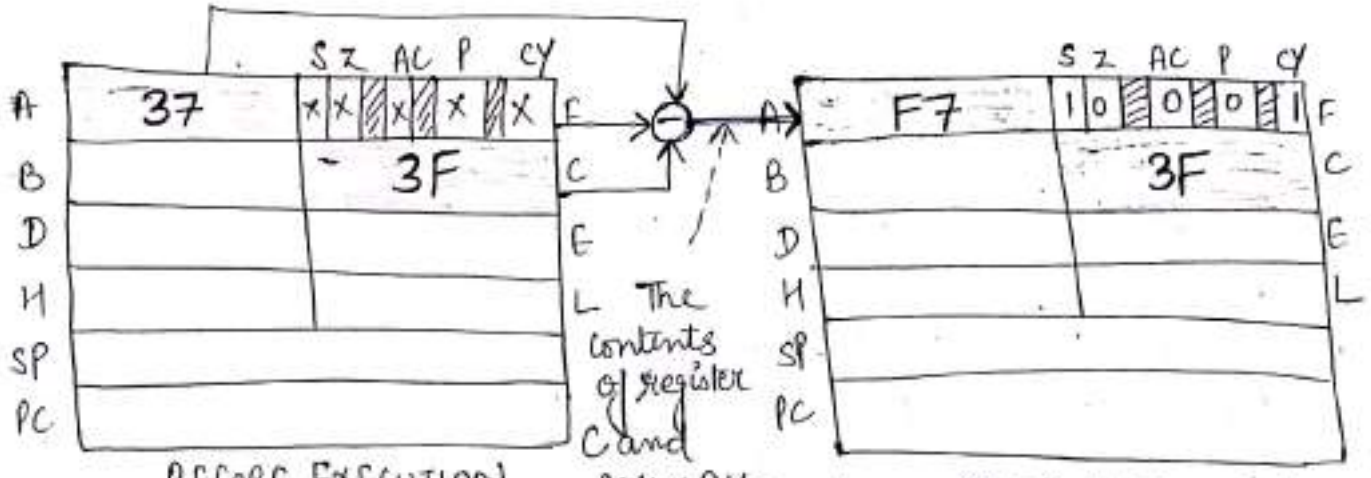
S = 1 as B_7 bit is 1

CY = 1

Z = 0 as the result is not zero

P = 0 as there are odd no. of 1's in the result.

AC = 0 as there is no carry from Bit B_3 to bit B_4



Subtracted from contents of register A and result is stored in register A

4. SBB M

Mnemonic	SBB M
operation	A = A-M-CY or A = A-(HL)-CY
No of Bytes	1byte
Machine Cycles	2(OF+MR)

Algorithm	A ← A-M-CY
Flags	All flags are modified
Addr. Mode	Indirect addressing mode
T-states	7(4+3)

DESCRIPTION :- Subtract data in memory and borrow flag from accumulator.

- This instruction subtracts the contents of memory location pointed by HL register pair and borrow flag from the contents of accumulator. The result is stored in the accumulator.
- The HL register pair acts as memory pointer.

EXAMPLE :- SBB M

Let A = 20H, H = 20H, L = 00H, CY = 1 at memory location 2000H : 4FH is stored and the instruction SBB M is executed

Data at memory location 2000H : 4FH

Borrow flag : + 1

50H (0101 0000)

2's complement of 50H : 1011 0000

A : + 0010 0000

A 1101 0000 (Result = D0H).

- Carry is not generated. The microprocessor complements the carry flag. Hence carry flag is set. The result is negative and in 2's complement form.
- The status of the flag will be as follow:

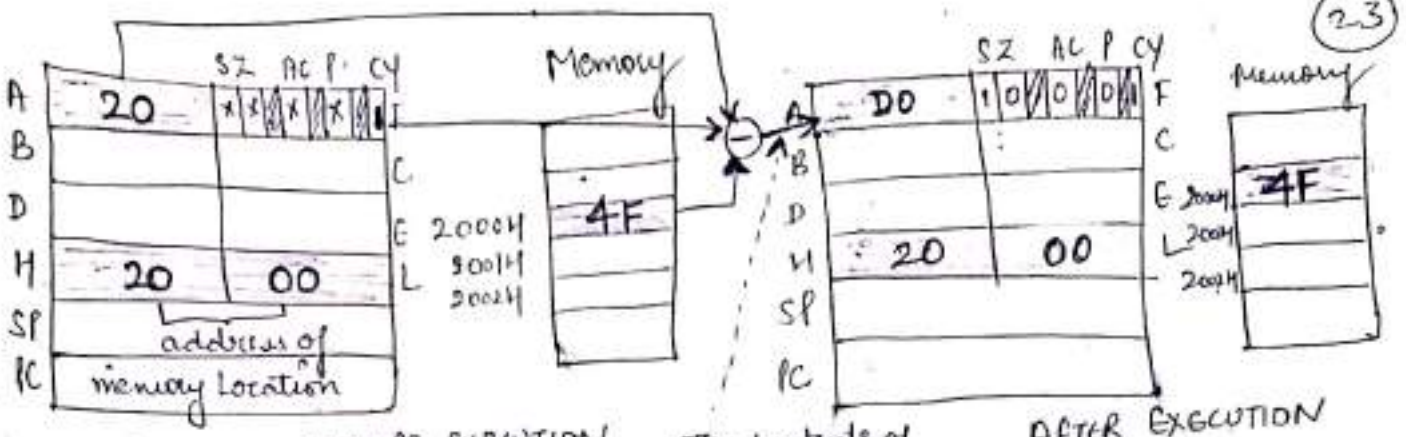
CY = 1

S = 1 as B_7 bit is 1.

AC = 0 as there is no carry from B_3 bit to B_4 bit.

P = 0 as there are odd number of 1's in result.

Z = 0 as the result is not zero.



memory location and borrow are subtracted from the contents of accumulator and result is stored in accumulator

5. SUI data

Mnemonic	SUI data
Operation	A = A - 8 bit data
No of Byte	2 bytes ; first byte : opcode, 2nd byte : 8 bit data
Machine cycles	2 (OF + MR)

Algorithm	A ← A - 8 bit data
Flags	All flags are modified
Addr. Mode	Immediate addressing mode.
T-States	7 (4+3)

DESCRIPTION :- Subtract immediate 8 bit data from accumulator

- This instruction subtracts the 8 bit data given within the instruction and borrow flag from the contents of the accumulator. The result is stored in accumulator.
- The subtraction is performed by using 2's complement method.

EXAMPLE :- SUI 50H A = A - 50H

Let A = 20H

Data : 0101 0000

2's complement : 1011 0000

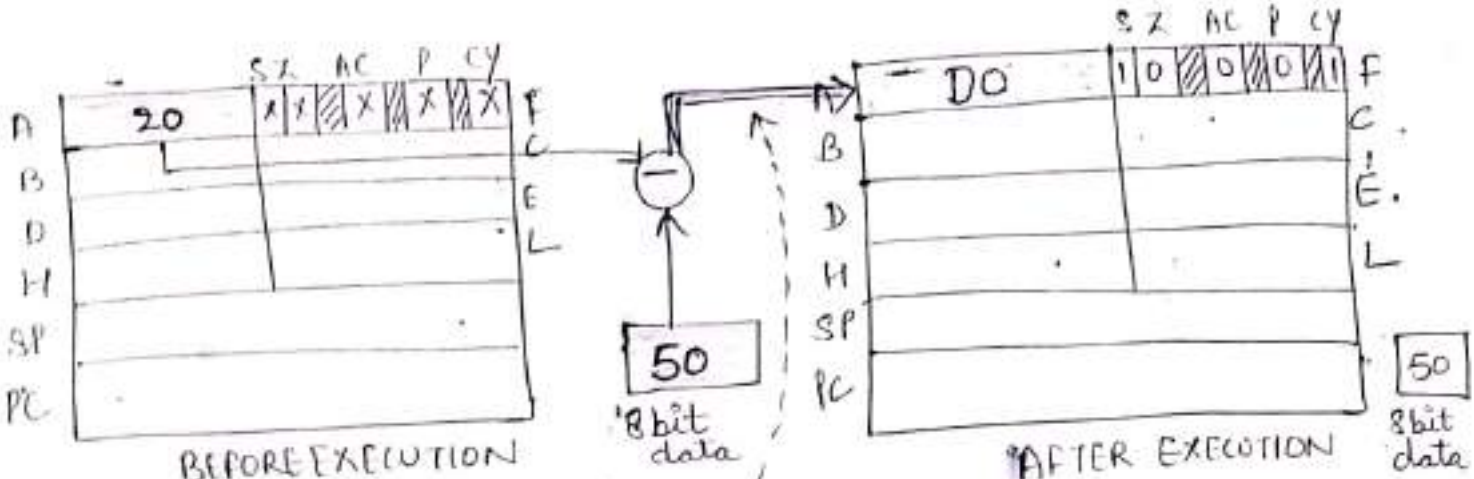
A : + 0010 0000

A 1101 0000 (Result D0H)

No carry is generated. The microprocessor complements the carry flag. Hence, the carry flag is set. This indicates that the result is negative and in the 2's complement form.

- The states of the flags will be as follows:

CY = 1
S = 1 as B_7 bit is 1
AC = 0 as there is no carry from bit B_3 bit to B_4 bit
P = 0 as there are odd No of 1's in the result.
Z = 0 as the result is not zero



8 bit data is subtracted from the accumulator and result is stored in accumulator.

6. SBI data

Mnemonic	SBI- data
Operation	A = A - 8 bit data - CY
No of Bytes	2 bytes; First byte: opcode; 2nd byte: 8bit data
Machine cycles	2 (OF + MR)

Algorithm	A ← A - 8bit data - CY
Flags	All flags are modified
Addr. Mode	Immediate addressing mode
T-States	7 (4+3)

DESCRIPTION :- Subtract immediate 8 bit data and borrow flag from Accumulator.

- This instruction subtracts the 8 bit data given within the instruction and the borrow flag from the contents of the accumulator. The result is stored in the accumulator.

EXAMPLE :- SBI 4FH

- This instruction will subtract 4FH from the contents of the accumulator and the result is stored in the accumulator.
- Let A=20H, CY=1 and the instruction SBI 4FH is executed.

Data 4FH
CY +1
50H (0101 0000)

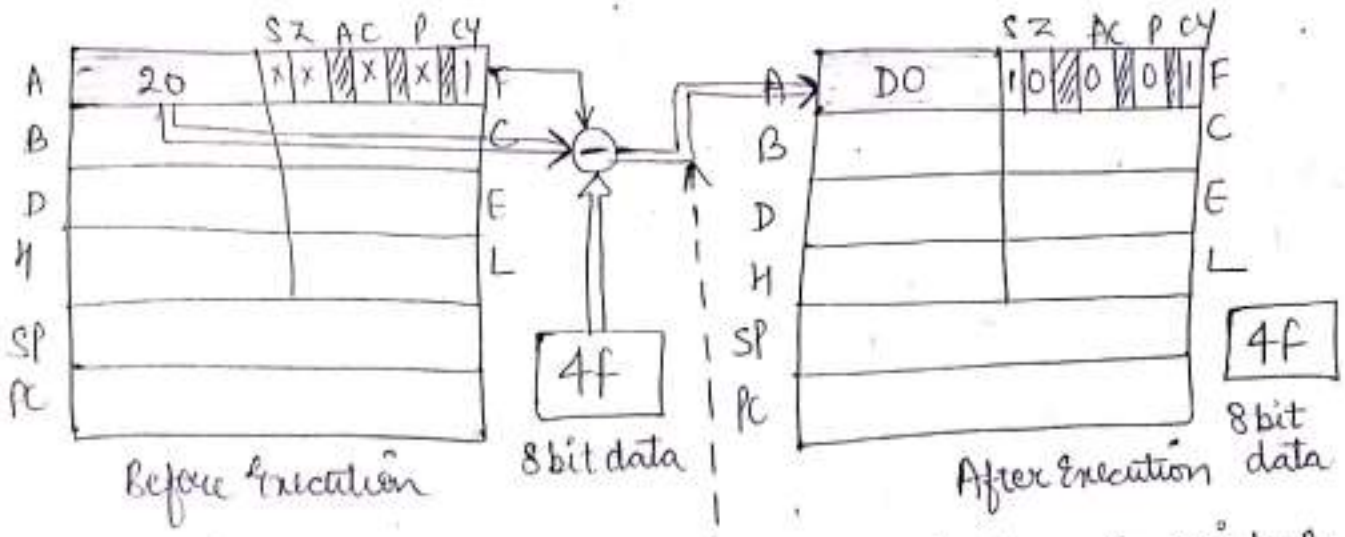
2's complement of 50H: 1011 0000

A: + 0010 0000

A 1101 0000 (Result = D0H)

- No carry is generated. Hence carry flag is set. This indicates that the result is negative and in the 2's complement form.
- The status of the flag will be as follows:

CY = 1
S = 1 as B_7 bit is 1
Z = 0 as the result is not zero.
AC = 0 as there is no carry from bit B_3 to bit B_4.
P = 0 as there are odd number of 1's in the result.



The 8 bit data and borrow are subtracted from the register A and result is stored in register A.

7. DAA

Mnemonic	DAA
Operation	If $A_{3-0} > 9$ or AC=1 then $A_{3-0} = A_{3-0} + 06H$ or If $A_{7-4} > 9$ or CY=1 then $A_{7-4} = A_{7-4} + 06H$
No. of Bytes	1 byte
Machine cycles	1(OF)

Algorithm	• If lower nibble of A > 9 or AC=1 then lower nibble of A = lower nibble of A + 06 H • If higher nibble of A > 9H or CY=1 then higher nibble of A = higher of A + 06H
Flags	All flags are modified
Addr. Mode	Implied addressing mode
T-State	4

DESCRIPTION: Decimal adjust accumulator

A ← sum in A is adjusted to packed BCD format.

- This instruction adjusts accumulator to packed BCD after adding two BCD numbers.
- This is the only instruction that uses the auxiliary carry flag.

The instruction works as follows:

1. If the value of low order 4 bits $B_3 - B_0$ in the accumulator is greater than 9 or if the AC flag is set, then the instruction adds 6 to the lower order 4 bits of the accumulator.
2. If the value of the higher order 4 bits $B_7 - B_4$ in accumulator is greater than 9 or if the carry flag is set, then the instruction adds 6 to the higher order 4 bits of the accumulator.

- The DAA instruction is used with the add instructions eg ADD, ADI etc to perform addition of a number in BCD. The add instructions adds the two BCD numbers in hexadecimal form and the DAA instruction converts this hexadecimal result to BCD format.

EXAMPLE :-

```
MVI A, 12H
ADI 39H
DAA
```

To add two BCD numbers 12 and 39 and get the result in BCD form.

A	0001 0010	12
Data +	0011 1001	39
	0100 1011	4B

The value of lower order 4 bits is greater than 9. Hence, 6 is added to low order 4 bits.

DAA	0100 1011	4B
+	0110	
	0101 0001	51

Thus, result in accumulator = 51 in BCD form.

INCREMENT

1. INR R

Mnemonic	INR R
Operation	R=R+1
No of Bytes	1byte
Machine cycles	1(OF)

Algorithm	R ← R+1
flags	Except carry all other flags are modified.
Addr mode	Register addr. Mode
T-States	4

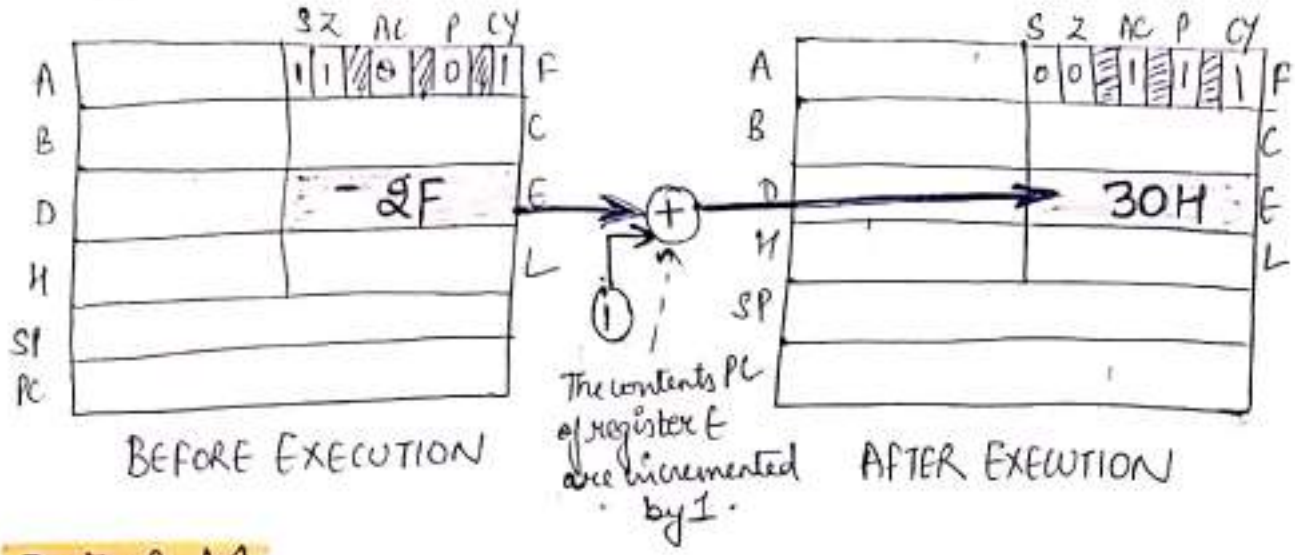
DESCRIPTION :- Increment Specified register

- This instruction increments the contents of specified register by 1 and result is stored in the same Register
- The Register R can be any general purpose register like A, B, C, D, E, H or L.
- All the flags except the carry flag are modified.

EXAMPLE :- INR E E → E+1

Let E=2FH, flag register = 11X0X0X1 and instruction INR E is executed

E	2FH
+	1H
E	30H



2. INR M

Mnemonic	INR M
Operation	(HL) = (HL) + 1 or (M) = (M) + 1
No of Bytes	1 byte
Machine cycles	3 (OF + MR + MW)

Algorithm	M ← M + 1 or (HL) ← (HL) + 1
Flags	All flags except carry flag is modified
Addr. Mode	Indirect addr. mode
T-States	10 (4 + 3 + 3)

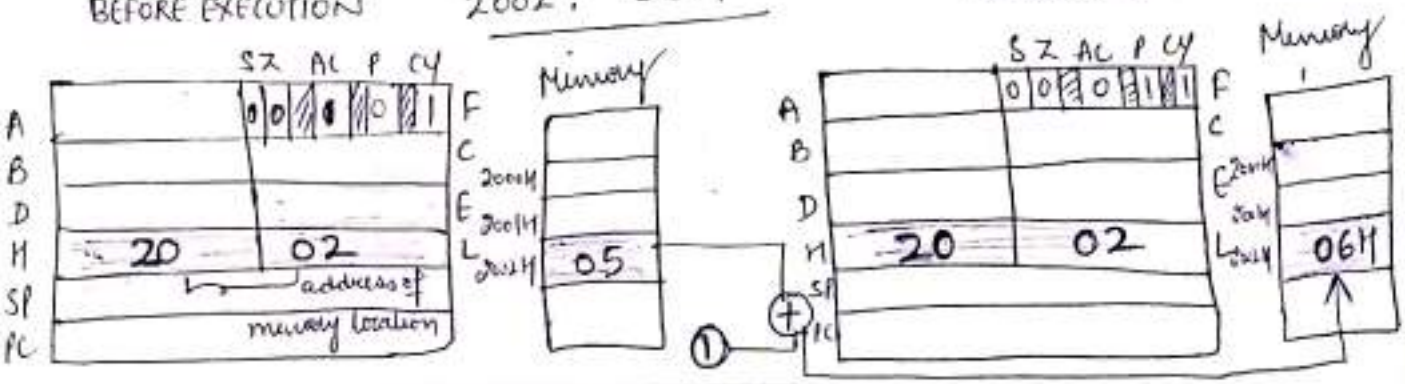
DESCRIPTION: Increment contents of memory location contents by one

- This instruction increments the contents of memory location addressed by HL register pair by 1 and result is stored back at the same memory location.

EXAMPLE : INR M

Let H = 20H, L = 02H, at memory location 2002 : 05 is stored flag register = 00X 1X0X1 and the instruction INR M is executed.

2002 : 05H
+ 1H
2002 : 06H



3. INX Rp

Mnemonic	INX Rp
Operation	Rp = Rp + 1
No of Byte	1 byte
Machine cycles	1(OF)

Algorithm	Rp ← Rp + 1
Flags	No flags are modified
Addr. Mode	Register addressing mode
T-States	6

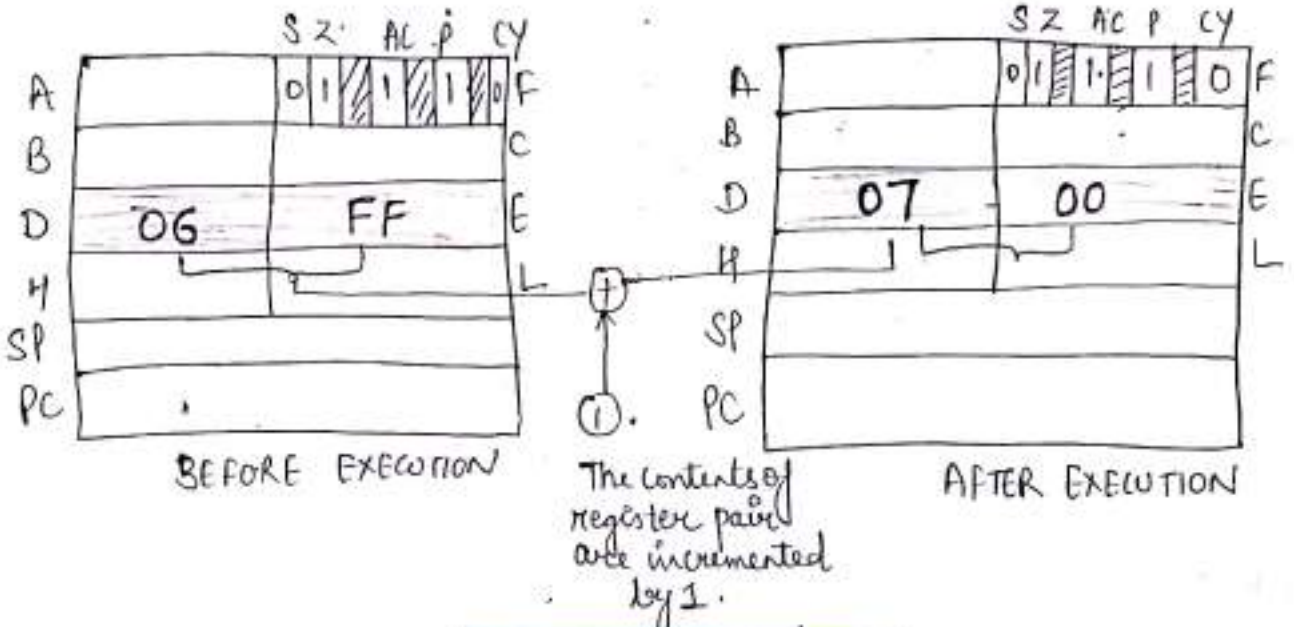
DESCRIPTION:- Increment specified register pair by 1.

- This instruction increments the contents of specified register pair by one. The result is stored in the same register pair.
- Rp is any valid register pair like BC, DE, HL or SP.

EXAMPLE:- INX D DE + 1 ⟶ DE.

Let D = 06, E = FFH, flag register = 0 1 X 1 X 1 X 0 and the instruction INX D is executed

```
DE   06FF H
   +    1 H
-----------
DE   0700 H
-----------
```



DECREMENT

1. DCR R

Mnemonic	DCR R
Operation	R = R-1
No of Bytes	1 byte
Machine cycles	1 (OF)

Algorithm	R ← R-1
Flags	Except carry, all other flags are modified
Addr. Mode	Register addressing mode
T-States	4

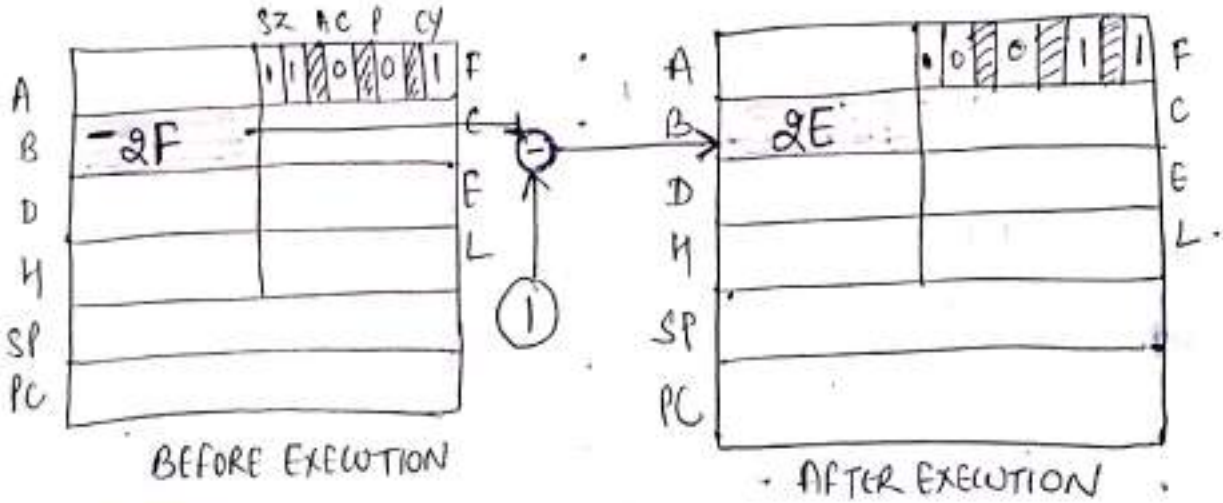
DESCRIPTION:- Decrement specified register contents by one.

- This instruction decrements register contents by 1 and the result is stored in the same register.
- Register R can be any general purpose register like A, B, C, D, E, H.

EXAMPLE: DCR B B = B-1

Let B = 2F, flag register = 01X0X0X1 and the instruction DCR B is executed.

```
  B   2F H
- ____1 H
  B   2E H
```



2. DCR M

Mnemonic	DCR M
operation	M = M-1 or (HL) = (HL) - 1
No. of Bytes	1 byte
Machine cycles	3 (OFt + MR + MW)

Algorithm	M ← M-1 or (HL) ← (HL) - 1
Flags	All the flags except carry flag are affected
Addr. Mode	Indirect addressing mode
T-States	10 (4+3+3)

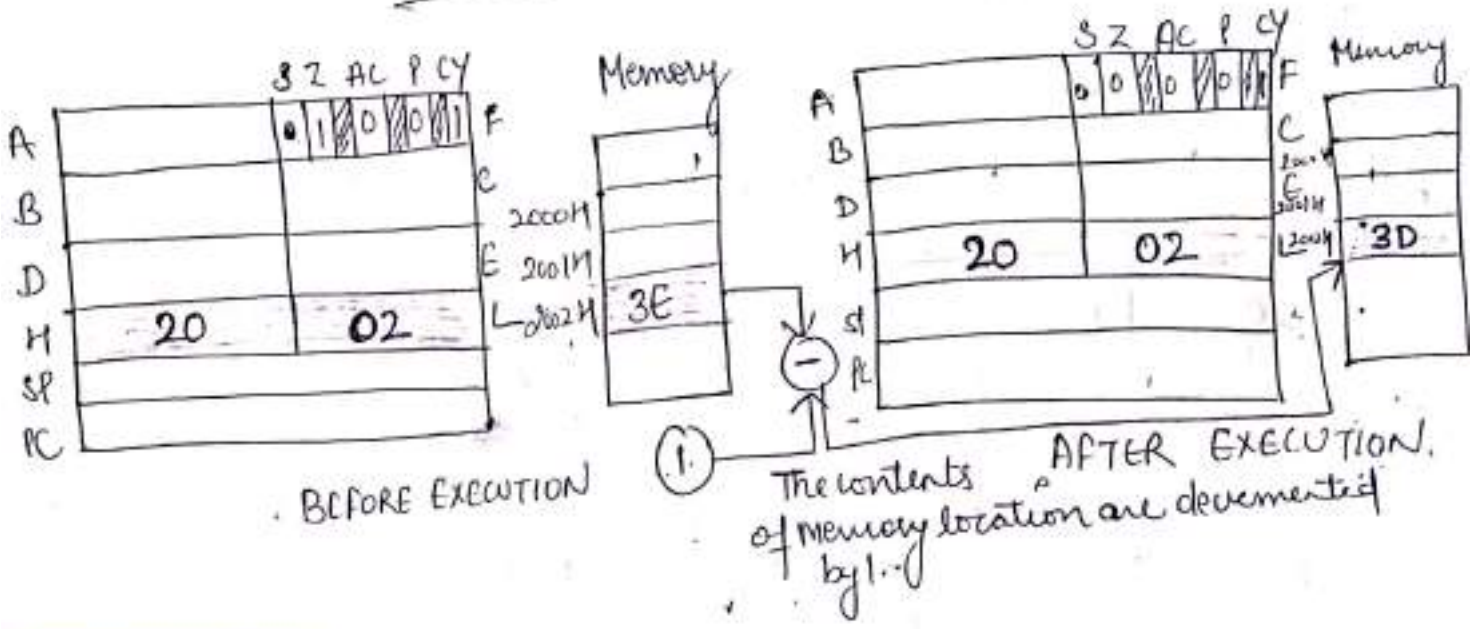
DESCRIPTION :- Decrement data in memory

- This instruction decrements the contents of memory location addressed by HL register pair by 1. The result is stored back at same memory location. The HL register pair acts as the memory pointer.

EXAMPLE : DCR M

Let H = 20H, L = 02H, at memory location 2002 : 3E data is stored, flag register = 01X0X0X1 and instruction DCR M is executed.

2002 : 3E H
− 1H
2002 : 3D H



3. DCX Rp

Mnemonic	DCX Rp
operation	Rp = Rp − 1
No of Bytes	1 byte
Machine cycles	1(OF)

Algorithm	Rp ← Rp − 1
Flags	NO flags are modified
Addr. Mode	Register addr. mode
T-states	6

DESCRIPTION: Decrement Register pair by one

- This instruction decrements the contents of register pair by 1. The result is stored in the same register pair.
- The register pair Rp is a valid register pair like BC, DE, HL or stack pointer (SP)

EXAMPLE: DCX D

Let D = 20H, E = FFH, flag register = 0X1X1X0 and instruction DCX D is executed.

$$\begin{array}{rr} DE & 20FF\,H \\ - & 1\,H \\ \hline DE & 20FE\,H \end{array}$$