Microprocessor - 8085 Instruction Sets

Let us take a look at the programming of 8085 Microprocessor.

Instruction sets are instruction codes to perform some task. It is classified into five categories.

S.No.	Instruction & Description
1	Control Instructions Following is the table showing the list of Control instructions with their meanings.
2	Logical Instructions Following is the table showing the list of Logical instructions with their meanings.
3	Branching Instructions Following is the table showing the list of Branching instructions with their meanings.
4	Arithmetic Instructions Following is the table showing the list of Arithmetic instructions with their meanings.
5	Data Transfer Instructions Following is the table showing the list of Data-transfer instructions with their meanings.

8085 – Demo Programs

Now, let us take a look at some program demonstrations using the above instructions −

Adding Two 8-bit Numbers

Write a program to add data at 3005H & 3006H memory location and store the result at 3007H memory location.

Problem demo −

```
(3005H) = 14H
   (3006H) = 89H
```

Result −

14H + 89H = 9DH

The program code can be written like this −

```
LXI H 3005H   : "HL points 3005H"
MOV A, M      : "Getting first operand"
INX H         : "HL points 3006H"
ADD M         : "Add second operand"
INX H         : "HL points 3007H"
MOV M, A      : "Store result at 3007H"
HLT           : "Exit program"
```

Exchanging the Memory Locations

Write a program to exchange the data at 5000M& 6000M memory location.

```
LDA 5000M   : "Getting the contents at5000M location into accumulator"
MOV B, A    : "Save the contents into B register"
LDA 6000M   : "Getting the contents at 6000M location into accumulator"
STA 5000M   : "Store the contents of accumulator at address 5000M"
MOV A, B    : "Get the saved contents back into A register"
STA 6000M   : "Store the contents of accumulator at address 6000M"
```

Arrange Numbers in an Ascending Order

Write a program to arrange first 10 numbers from memory address 3000H in an ascending order.

```
MVI B, 09          :"Initialize counter"
START              :"LXI H, 3000H: Initialize memory pointer"
MVI C, 09H         :"Initialize counter 2"
BACK: MOV A, M     :"Get the number"
INX H              :"Increment memory pointer"
CMP M              :"Compare number with next number"
JC SKIP            :"If less, don’t interchange"
JZ SKIP            :"If equal, don’t interchange"
MOV D, M
MOV M, A
DCX H
MOV M, D
INX H              :"Interchange two numbers"
SKIP:DCR C         :"Decrement counter 2"
JNZ BACK           :"If not zero, repeat"
DCR B              :"Decrement counter 1"
JNZ START
HLT                :"Terminate program execution"
```