Microprocessor Lab Report

Abhiroop Mukherjee
Enrl. No: 510519109

Department of Computer Science and Technology
Indian Institute of Engineering Science and Technology, Shibpur

Contents

1 Assignment 1

1.1 Objective

Find out the sum of the first 30 natural numbers.

1.2 Tool/Experimental setup considered

- Jubin's 8085 Simulator

1.3 Procedure

We know that

$$1 + 2 + 3 + ... + 29 + 30 = \frac{30 \times 29}{2} = 435 = 01D1H$$

This result is not possible to store in a single register, so we need to use register pair to store the result.

1.4 Program

```
;end result is 465, more than 255 hence we need to do extended additions
       MVI D,1E; setup D, the counter as 30
       MVI C,01; setup BC as 1

L1:    DAD B; Double add BC to HL
       INX B; extended increment BC
       DCR D; decrement D
       JNZ L1; if D becomes 0, Z flag becomes 0 and we break
       HLT
; Ans will be in HL
```

Listing 1: assembly program to find sum of the first 30 natural numbers

1.5 Experimentation

Register	Value	7	6	5	4	3	2	1	0
Accumulator	00	0	0	0	0	0	0	0	0
Register B	00	0	0	0	0	0	0	0	0
Register C	1F	0	0	0	1	1	1	1	1
Register D	00	0	0	0	0	0	0	0	0
Register E	00	0	0	0	0	0	0	0	0
Register H	01	0	0	0	0	0	0	0	1
Register L	D1	1	1	0	1	0	0	0	1
Memory(M)	00	0	0	0	0	0	0	0	0

1.6 Conclusion

We see that after the execution of program, the data stored in HL register pair is 01D1H, which is the hexadecimal value of 435.
Hence the Program is working as expected

2 Assignment 2

2.1 Objective

From an array of 10-byte size integers (unsigned) find out the maximum and minimum.

2.2 Tool/Experimental setup considered

- Jubin's 8085 Simulator

2.3 Procedure

The idea is to linearly iterate through all the values of the arr, and update the register for minimum(C) and maximum(B) values.

2.4 Program

```
;Actual Program
# ORG 5000H
# ARR: DB 5,2,3,4,F,C,7,A,B,1
# ORG 0000
       LXI H,ARR
       MOV B,M  ;B is maximum val
       MOV C,M  ;C is minimum val
       MVI D,0A
;CMP R does (A - R) in background
;If A - R > 0 then Cy = 0, Z = 0
;If A - R = 0 then Cy = 0, Z = 1
;If A - R < 0 then Cy = 1, Z = 1

LP:    MOV A,M
       CMP B
       JC MIN   ;will Jump when Cy = 1, A - B < 0, A < B
       MOV B,A  ;will only happen if A > B

MIN:   CMP C
       JNC SKIP ;will Jump when Cy = 0, A - C > 0, A > C
       MOV C,A  ;will only happen if A < C

SKIP:  INX H
       DCR D
       JNZ LP
       HLT
```

Listing 2: assembly program to find minimum and maximum number in 10-byte unsigned array

2.5 Experimentation

Register	Value	7	6	5	4	3	2	1	0
Accumulator	01	0	0	0	0	0	0	0	1
Register B	0F	0	0	0	0	1	1	1	1
Register C	01	0	0	0	0	0	0	0	1
Register D	00	0	0	0	0	0	0	0	0
Register E	00	0	0	0	0	0	0	0	0
Register H	50	0	1	0	1	0	0	0	0
Register L	0A	0	0	0	0	1	0	1	0
Memory(M)	00	0	0	0	0	0	0	0	0

2.6 Conclusion

We see that that after program execution, B has the maximum value of array(F), and C has the minimum value of the array(C).
Hence the Program is working as expected

3 Assignment 3

3.1 Objective

Write a routine that produces a delay. The delay value must be passed to register pair DE.

3.2 Tool/Experimental setup considered

- Jubin's 8085 Simulator

3.3 Procedure

Idea is to define DE as a very big value (say FFFF), and decrement it in a loop till DE becomes 0 to produce delay in execution.

3.4 Program

```
1        LXI D,07D0
2        CALL DELAY
3        HLT
4  ;delay: this subroutine produces delay
5  ;in: value in DE pair
6  ;out: none
7  ;destroys: A
8
9  DELAY: DCX D    ;doesn't affect any flags, that's why doing OR
10       MOV A,E
11       ORA D    ;will give 0 only when both D and E 00
12       JNZ DELAY
13       RET
```

Listing 3: assembly program to produce delay

3.5 Conclusion

We see that the code runs for sometime, then it completes it's execution, signifying that the delay function worked and delayed execution of CPU for some time.
Hence the Program is working as expected