NAME-ARPAN MANDAL DEPT-CSE YEAR-2nd

ROLL-001910501061 GROUP-A3

SUBJECT-MICROPROCESSOR AND ASSEMBLY LANGUAGE LAB

ASSIGNMENT-1

-------------------------------------------------------------------------------------------------------------------------------------

1. Load the contents of the memory locations 2200H and 2201H into registers. Add these registers and store the result in memory locations 2202H and 2203H.

2. Find the sum of N numbers stored in consecutive locations starting from 2500H. The value of N is stored in 2200H. Store the result in locations 2300H and 2301H.

3. Find the sum of the least significant 4 bits and most significant 4 bits of a byte stored in memory location 2500H. Store the result in 2550H.

4. Write a program to count the ‘1’s and ‘0’s of a byte stored in 2500H. Store the result in 2610H and 2511H, respectively.

5. Write a program to sum two 16-bits binary numbers.

1.

MVI C,00H //[C]🡨00H

LDA 2200H //[A]🡨[2200H]

MOV B,A //[B]🡨[A]

LDA 2201H //[A]🡨[2201H]

ADD B //[A]🡨[A]+[B]

JNC SKIP //JUMP IF CS=0

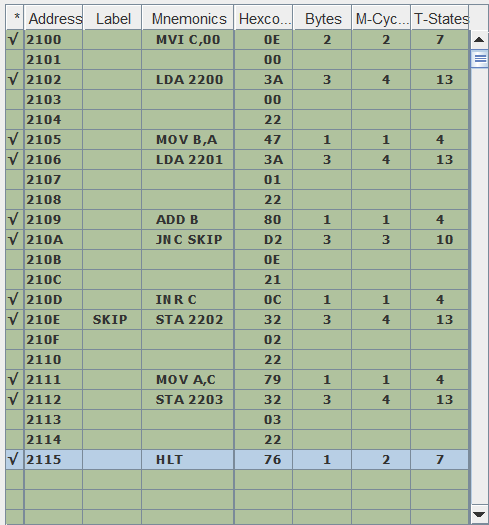
INR C //[C]🡨[C]+1

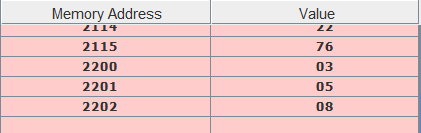
SKIP:STA 2202H //[2202H]🡨[A]

MOV A,C //[A]🡨[C]

STA 2203H //[2203H]🡨[A]

HLT //HALT





2.

LXI H,2500 //[H-L]🡨2500H

LDA 2200H //[A]🡨[2200H]

MOV C,A //[C]🡨[A]

MVI A,00H //[A]🡨00H

MVI B,00H //[B]🡨00H

LOOP: ADD M //[A]🡨[A]+[[H-L]]

JNC SKIP //JUMP IF CS=0

INR B //[B]🡨[B]+1

SKIP:INX H //[H-L]🡨[H-L]+1

DCR C //[C]🡨[C]-1

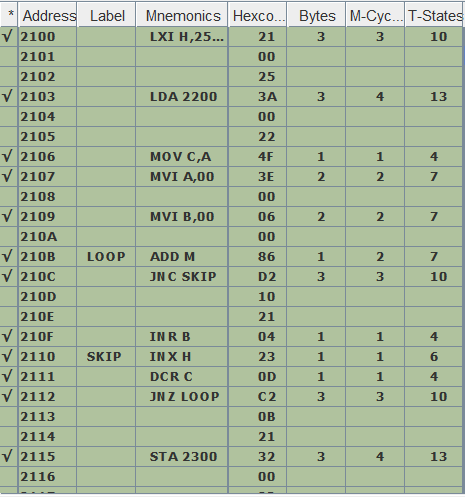
JNZ LOOP //JUMP IF [C]!=0

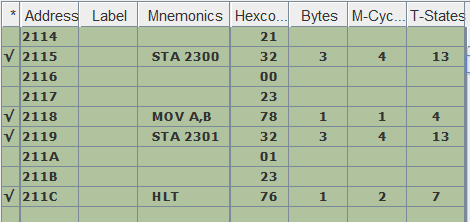
STA 2300H //[2300H]🡨[A]

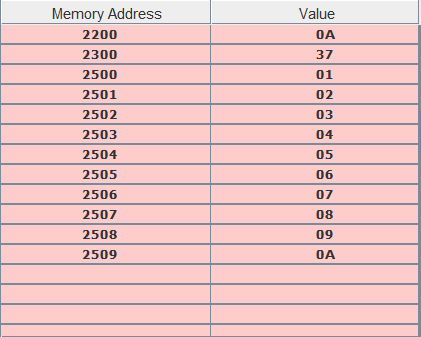
MOV A,B //[A]🡨[B]

STA 2301H //[2301H]🡨[A]

HLT //HALT







\*ALL VALUES ARE IN HEXADECIMAL

3.

LDA 2500H //[A]🡨[2500H]

ANI F0H //[A]🡨[A]^[F0H]

RRC //ROTATE ACCUMULATOR RIGHT

RRC // ‘’

RRC // ‘’

RRC // ‘’

MOV B,A //[B]🡨[A]

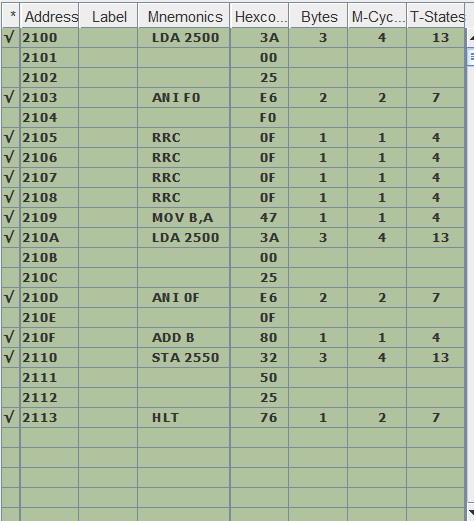
LDA 2500H //[A]🡨[2500H]

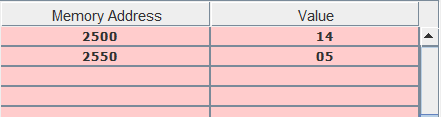
ANI 0FH //[A]🡨[A]^[0FH]

ADD B //[A]🡨[A]+[B]

STA 2550H //[2550H]🡨[A]

HLT //HALT





4.

LDA 2500H //[A]🡨[2500H]

MOV E,A //[E]🡨[A]

MVI B,00H //[B]🡨00H

MVI C,00H //[C]🡨00H

MVI D,08H //[D]🡨08H

LOOP:MOV A,E //[A]🡨[E]

RRC //ROTATE ACCUMULATOR RIGHT

MOV E,A //[E]🡨[A]

RLC //ROTATE ACCUMULATOR LEFT

ANI 01H //[A]🡨[A]^[01H]

JZ SKIP //JUMP IF [A]=0

INR B //[B]🡨[B]+1

SKIP:DCR D //[D]🡨[D]-1

JNZ LOOP //JUMP IF [D]!=0

MOV A,B //[A]🡨[B]

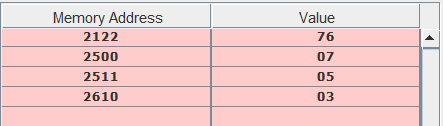
STA 2610H //[2610H]🡨[A]

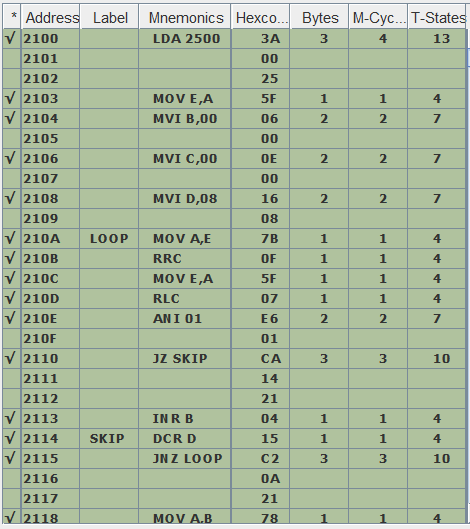
MVI A,08H //[A]🡨08H

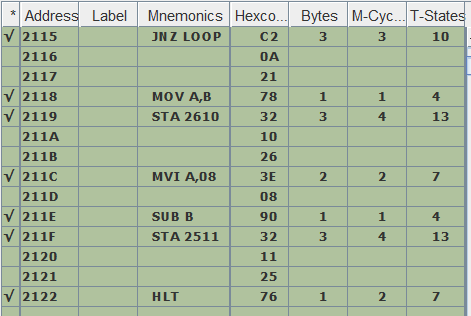
SUB B //[A]🡨[A]-[B]

STA 2511H //[2511H]🡨[A]

HLT //HALT







5.

MVI H,00H //[H]🡨00H

LXI B,ABCDH //[B-C]🡨ABCDH

LXI D,5141H //[D-E]🡨5141H

MOV A,C //[A]🡨[C]

ADD E //[A]🡨[A]+[E]

STA 2500H //[2500H]🡨[A]

MOV A,B //[A]🡨[B]

ADC D //[A]🡨[A]+[D]+CY(CARRY)

JNC SKIP //JUMP IF CY=0

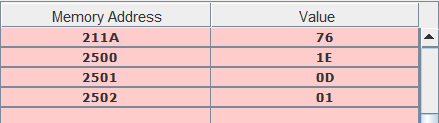
INR H //[H]🡨[H]+1

SKIP:STA 2501H //[2501H]🡨[A]

MOV A,H //[A]🡨[H]

STA 2502H //[2502H]🡨[A]

HLT //HALT



ABCDH+5141H=0D1E CY=1

