1. Write a program for the addition of two 8 bit numbers outh cony.

# PROGRAM STATEMENT

To curite an assembly language program for the addition of two 8 bit numbers exith confy.

# DROGRAM ANALY 615

#### ALGORI THM!

STEP 1! Load the data into accumulator from memory location 2000H.

STEP 2: Mae the data into B register

STEP 3: Load tre data into accumulater from memory location 2001 H.

STEP 1! Add the registers.

STEP 6: Check for com

STEP 6: Store the value of sum and comy in momony location 20024 and 20034 respectiely

STEP 7! Terminate the program.

# ASSEMBLY LANGUAGE CODE

MVI C,00H "Intralize C register to 004 accumulates. 20004 // Load the volve of memory lucotion 20001 to Brokens LOA Ma B,A 11 More the content of accumulator to register B LDA 20014 // Load the value of memory location 20034 to accumulator. B . // Add the content of accumulator outn register B : ada JNC // Jump on no carry to X. X INR Increament (Content of register C by 1. Store the value of accumulator to memory location X'. STA 20024 20024 (Sum)

MOVA, C More the content of register C to accumulator.

11 Store the statue of accumulater to memory location 2008 (Corry) STA 2003 H // Terminate the program. HLT

Japut:

Memory Location:

2000H - 9AH

20001 H - 9FH

Register:

B - 1 9AH

Ac 9FH

Output:

2003H — 39H

2004H -> 01 H

2. Curite a program for the addition of two 16-bit numbers come corry.

## PROGRAM STATEMENT

To curite a assembly longuage program for the addition of two 16-bit numbers cuth conf.

#### PROGRAM ANALYSIS

#### ST ALGODRITHM;

immediale

STEP 1: Load 16 bit data on register poir HL and DE respectively

STEP 2: Add the register poir DE and HL directly.

STEP3: Check for com

STEP 4: State the Sum and corry in memory location 2055H and 2060H respectively.

STEP 6: Terminole the program

#### ASSEMBLY LANGUAGE CODE

MUIC, OOH // Intelize C outn data OOH.

LXI H, 9A9FH // Load 16 bit data to HL poir immediately

LXI D, 95 A 2 H // Load 16 bit data to DE pair immediately.

DAD D // Double odd recision pair HI and DE directly.

DAD D

11 Double add register pair HL and DE directly.

JNC X

11 Jump to X if no corry

INR C 11 Increament content of register C by 1.

X! SHLD 2065H // Store the control of HL por to mem. location 2065H.

MON A, c 11 Mar content of register to accumulater

STA 2060H // Stare the content of accumulator to regizable

HLT // Terminate the Drogram

#### Japat

HL - 949FH

DEL 9FAZH

Output

2055H → 41H 2056H → 30H

2060H -> 01H

3 CUAP to add the content of 9101H and 9102H and place the Sum at 9103H and corry at 9104H.

#### PROGRAM STATEMENT

To axite on assembly longuage program to odd content of memory location 91014 and 91024 and place the sum of 91034 and conf at 91094.

#### PROGRAM ANALYSIS

#### ALGORITHM:

STEP 1: Intialize to store the comy bit.

STEP 2: Load data to accumulate from memory lucation 9101H.

STEP 3! More the content of accumulator to register B.

STEP 4! Load data to accumulate from memory location 9102H.

STEP S! Add the content of accumulator and register B.

STEP 6: Check for corry or not?

STEPAL 0) If NO,

- i) Store the result in a manay location 91034 and corry of 91044.
- ii) Terminate the program
- b) If Yes,
  - i) Increase the content of register C by one.
- ii) Store the addition result in memory location 91034.
- ii) Stere the Cony in Amonog location 91044.

STEP 7: Terminate the progress.

```
// Introlize C register to OUH.
MUIC, OOH
HIDE BIOTH // Load the content of memory location GIGIH
                  accumulator.
               11 Mue the content of accumulator to register B
MOU BID
               11 Load the content of memory location 9102H tu
HOOLB ACO
                  accumulates.
               11 Add the content of register B outh accumulator.
DDD
X! STA 9503H
JNC X / Jump to X 4 not corry.
             11 Increment content of register C by 1.
INR C
X: STA 9103H // Store the content of accumulator to memory
                  location 910317.
           11 More content of register c by 0111.
MOU GA, C
              11 Store the content of accomulated to mamony location
STA 910911
                9104 H.
               11 Terminate the program
 HLT
```

# Jnput 9101 H ← 7FH 9102 H ← 5BH 9103 H → DAH B ← 7FH A ← 6BH

4. CUMP for the Bubtraction of two 8-bit number cum borrow.

# PROGRAM STATEMENT

To write on assembly longuage program for the subtraction of two 8-bit number with borrae.

#### PROGRAM. ANALYSIS

#### ALGORITHM;

STEP 1: Initialize register C to store borrow bit.

STEP 2: Load the data in accumulater from memory location 2000H.

STEP 3 : More the content of accumulator to register B

STEP 4! load the data to accumulate from memory location 2001H.

STEP 5! Subtract content of accumulater and regulater
B.

STEP 6: If complete sent toke 2's complement of accumulator.

STEP 7. Store the value of borrow in memory location

STEP8: Store the differe value in many location 20034.

STEP9: Terminate the program

// Initialize OOH to register C MUI (, 00H // Load the content of memory lucation 2000H to LOA 2000H accumulater A 11 Move the content of accumulater to register B. MON BIA // Load the content of memory location 2001H to LDA 2001H occumulater A. 11 Subtract content of occumulator and majester B. SUB B 11 Jump to x if no corry (x) JNC X 11 Complement accumulator contents. CMA 11 Increment value in Accumulator INR A Increment value in Register C. INIR ( 11 Store the value of accomplater to memory address X! STA 2002H 2002H. MOU AC More control of 11 Store the value of accumulator to mamony location STA 2003H 2003H. // Terminot the Program. HLT

#### JNPUT

200H ← 1FH

2001HE- GAH

BE JFH

AC SAH

2002H -> 7BH 2003H -> 00H 6. CUMP to subtract the content of register c and register D and store the register H and barrow at register L.

## PROGRAM STATEMENT

To crite on assembly language paragram to subtract the control of register c and register D and stee the differe of H and bornow at negister L.

### PROGRAM ANALYBIS

#### ALGORITHM:

STEP 1: Intialize register B to stone borrow bit.

STEP 3! More the content of register C immediately.

to regislar D immediately. STEP 4: Load the data

STEP 5: Subtract content of register C and register D.

STEP 6: If comy is present take 2's complement of accumulator.

STEP 7: Store the diffrence value in arranging register н,

STEP 8: Stare the value of borrow in register L.

STEP 9: Terminale the program.

MVI B, OOH // Load OUH date on regists 18 to store
borrow bit
MUI C, 35H // Load 35H date on register C immediately.

MOU A, C // Move the content of regists C to occumulate.

MUI D, 78H // Load 76H date on register D immediately.

SUB D // Subtract the content of register BA cuth register D.

JNC X // Jump to X if not corry (borrow).

CMA // Complement content & A

INRA // Josephent content of A by OIH.

INRB // Josephant content of B by OIH.

X: MWHA // More content of accumulated to register H.

MONDIB. Il More content of register B to accomplater.

MOV LIA 11 More content of accumulator to register L.

HLT // Terminate the program.

## JNPUT

( - 35H

DE-75H

Output

H-> 43H

L-> 01 H

7. CUAP to Create a continuous loop using unconditional Jump

## PROGRAM STATEMENT

To curite on assembly language program to create a continuous loop using unconditional Jump location.

## PROGRAM ANALYSIS

### ALGORITHM:

STEP 1: load Accumulater with data 10H immediately.

STEP 2: Load register & cuth data OIH immediately.

STEP 3! Subtract content of accumulator cuitn register B.

STEP 4: Jump \$ to > 7 if the result on accumulate is zoro.

STEP 5: Jump to X.

STEP 6: Moe content of occumulator to register C.

STEP7: Terminake the program

# ASSEMBLY LANGUAGE COPE

MVI, A, 10H // Load lot value to accumulater immediately.

X: MUJB, OLH // Load OIH data value to register B immediately.

SUB B 11 Subtract the content of accumulate with register B.

JZ Y / Jump to Y if content of accumulator is zero.

JMP X // Unconditional Jump to X.

Y: MON C, A // More the content of accumulator to register C.

HLT // Terminote The Program.

INPUT

AL 10H

B 6 01 H

OUTPUT

C -> 00H

Coste a program to test the Jump on zero Condition on an metic operation.

# PROGRAM STATEMENT

assembly language program to test the Jump on Zara Condition on anithmetic operation.

## PROGRAM ANALYGIS

#### ALGORI THM:

STEPS: Load immediate Value into occumulator.

STEP 2: Load immediate value into register B

STEP3: Load immediate value into regular C.

STEP4: Add contents of register B to occumulator.

STEP 5! Compare contents of accumulator and register (.

STEP 6: If accumulater is equal to register c, jump to

STEP 7: Load immediale value 11 H Into register E7. No STEP 8: More content of register E to accumulator. Case

STEP 9: Terminate the Drigram

STEP10: Load immediale Value 22H into register D 1 401
STEP11: More content of register D to accumulator. Decision
Case
STEP 12: Terminale the Drogram

# · For Satisfied Condition

MVI A, OSH // Load immediak value OSH into Occumulator.

MUI B, OZH // Lood immediale Value OZH into accum register B.

MUJ C, 07H // Lood immediak Value 07H into register C.

ADD B // Add content of accumulater cuth register B.

CMP C // Compose contents of occumulates and register C.

JZ X // If accumulated is equal to register C, Gump to

MUI E, 11H // Load immediale value 11H into register &.

MON A, E // More content of register & to accumulates

HLT // Terminale The Drogram

X: MUJ D, 22H // Load immediak value 22H into register D

MON A, D // Mue content of orgister D to occumulate

HLT // Terminole the program

# Japut

AL OSH

Bc- 02 H

C --- 07H

## Output

A-> 22H

# · For Unsatisfied Condition

MIVI A, OSH

MVJ B, 02 H

MVJ C, O1H

ADD B

CMP C

J2 X

MVI E, IIH

MOU DIE

HLT

X! MUJ D, 22H

MOIDIO

HLT

Jnput

AL OSH

B 6- 02 H

C 6- 0J. H.

Cutput

A > 11 H

9. Curite a program to test the Jump on Even Parity condition on logical operation.

# PROGRAM STATEMENT

To arrive on assembly language program to test the Jump on Franchist Condition on logical apparation

#### PROGRAM ANALYSIS

#### ALGORITHM:

STEP 1! Lood immediate value into accumulator

STEP 2: Load immediate value into register B.

STEP 3: Load immediak value into register C.

STEP 4: AND content of register B auth accumulation.

STEPS: Compor contents of accumulated and register (

STEP 6: If the point is even jump to label X.

STEP 7: Load immediale value 11H into register E.

STEP 8: Mode content of register E to accumulator.

STEP 5: Terminate the program.

STEP 10: Load immediale value 22H-into agristar 1).

STEP 11: Mae content of rigister D to occumulate.

STEP 12: Terminal the program

# · For Satisfied Condition

MUIA, OSH // Lood immedial value OSH into occumulator.

MUIB, OZH // Lood immedial Value OZH into register B.

MUIC, OTH // Lood immedial Value OTH into register C.

AND B // AND content of accumulator cum register B.

CMP C // Compase content of accumulator and register C.

JPE X // Jf the Porty is even, jump to label X.

MUIE, IIH // Lood immedial value IIH into register E.

MUNDE, IIH // Lood immedial value IIH into register E.

MUNDE // Max content of register E to accumulator.

HLT // Terminal tre Program,

X'MUID, 27H // Lood immedial Value 22H Into register ID.

MOND, D // Max content of register D to accumulator.

HLT // Terminal tre Program,

Japut		Output
AL OSH		A -> 22 H
BE 02H		
C <- 07H		

# · Far Unstatisfied Condition

MVI A, 02H
MVI B, 02H
MVI C, 07H
ANA B
CMP C

JPE X

MVI E, 11H MW A, E HLT X: MVI D, 22H

MW 0,0

HLT

Japut

A - 02 H

B - 02 H

( - 07 H

Output

111 (- A

10. Write a program to test the Jump on Negative condition on an orithmetic operation.

## PROGRAM STATEMENT

To civile an assembly language program to test the Jump on Nagotive condition on an onlynmetic approxime.

# PROGRAM ANALYSIS

# Algorithm:

Step 1: Load immediate value into accumulator.

Step 2! Load immediate value into register B.

Step 3: Load immediate value into register C.

Step 4: Subtract the content of accumulator and register B.

Stps: Jump to label x if the content of accumulator is negative:

Step 6: More the content of register ( to accumulater.

Stp7: Terminot the program.

Step 8! Load immedial Value into accumulator.

Step 9: More the content of register D to accumulator.

Stplo: Terminate the Program.

# ASSENTALY LANGUAGE CODE

PTVI A, OZH // Load immediate Value OZH 19th Occumulator.

MUJ B, OSH // Load immediat value OSH into are register 13.

SUB B // Subtract the content of register B cuth accumu

JM X // Jump to Have lobel X if the content of

MUI C, UCH // Load immediate value Och into accumulator.

Ma Ac // Move content of registral to accumulate

HLT // Terminote the program.

PX: MUID, UM // Load immediate value OIH into accumulater.

May A,D 11 More content of register D To accumulater.

HLT // Terminate the program.

Japut Output

A→ 02H B→ 05H

· For Unsotisfied Condition

MUI A, OSH

MVI B, OZH

SUB B

X MT

MUIC, OUH

MW ALL

HLT

MO VD 11

HLT

INPUT

A -> 05H

B → 02H

DUTPUT

A> 0 0H