

* Name :- Diksheet Agarwal

Scholar ID : 1912059

Class :- Computer Science and Engineering

Section :- A

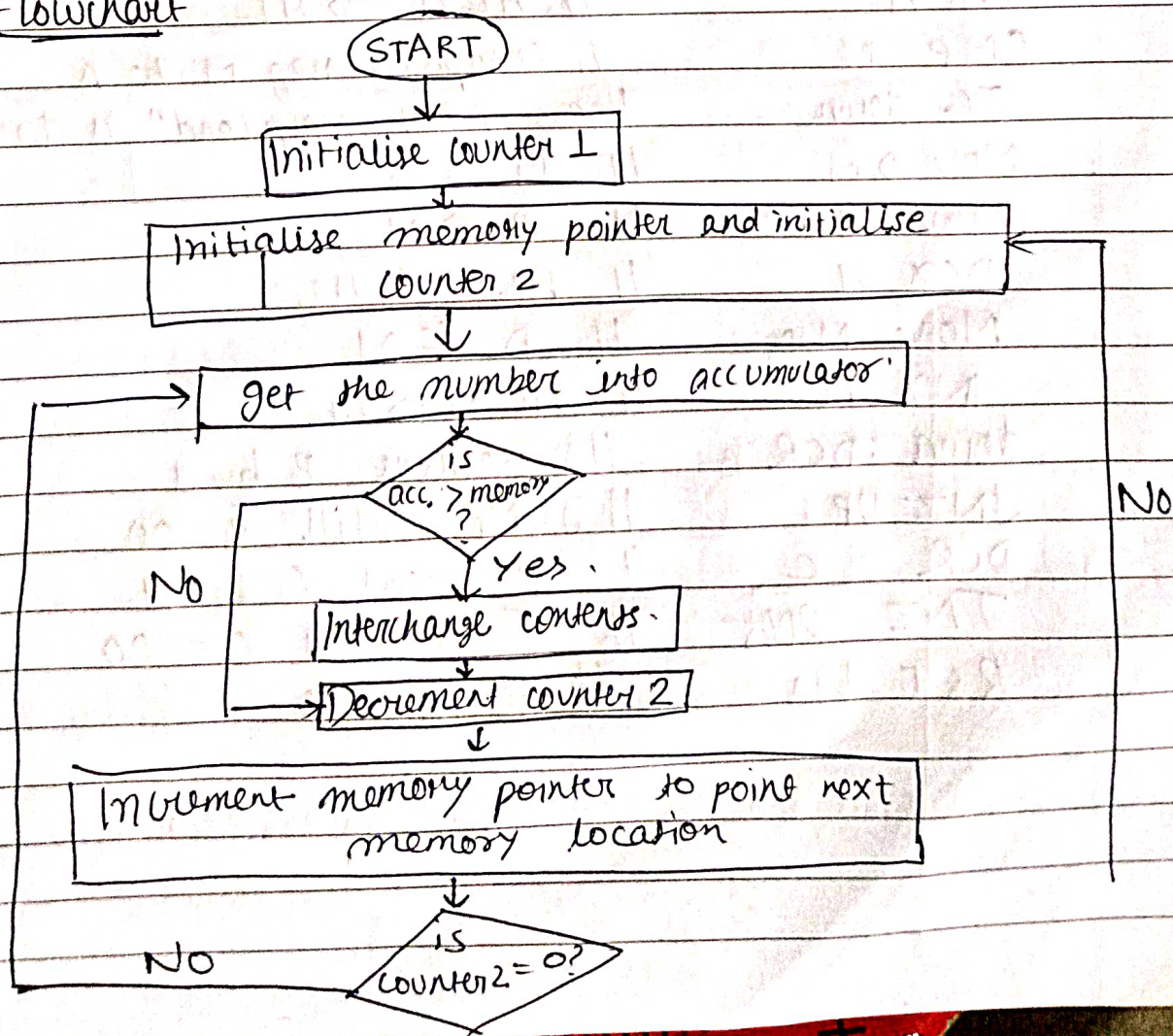
Experiment - 6

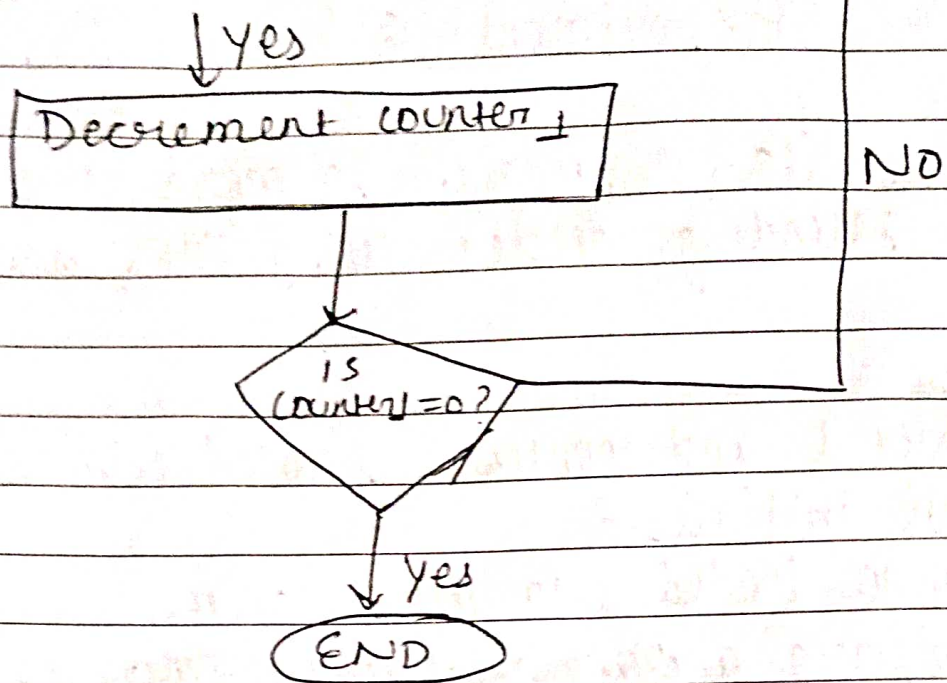
Aim:- Write a program to arrange a number in ascending order using 8085 and verify.

Theory

- ① Counter 1 and counter 2 loaded with counters and memory points initialised
- ② The no. loaded into accumulator.
- ③ Compare accumulator and memory pointer.
- ④ if carry is set, then go for step 7.
- ⑤ The contents are interchanged.
- ⑥ memory pointer incremented to point next memory location.
- ⑦ counter 2 decremented
- ⑧ if counter 2 is not zero, go to step 2.
- ⑨ counter 1 decremented.
- ⑩ if counter 1 is not zero, go to step 1.
- ⑪ Terminate.

Flowchart





Code

ORG 2000H.

LDA F100.

//load count from F100 to A

DCR A

// Decrement A By 1

MOV C,A

// A \Rightarrow C.

MOV B,C

// B \Rightarrow C.

LXI H,F200

// HL \Leftarrow F200.

UP: MOV A,M

// [HL] \Rightarrow A

INX H

// HL \Rightarrow HL.

CMP M

// Compare reg M to A.

JC down

// e.g. A < M jump condⁿ is true

MOV D,M

// M \Rightarrow D

MOV M,A

// A \Rightarrow M

DCX H

// HL-1 \Rightarrow HL

MOV M,D

// D \Leftarrow M

INX H

// HL+1 \Rightarrow HL

down: DCR B

// Decrement B by 1

JNZ UP

// JUMP untill B=00

DCR C

// Decrement C by 1

TJNZ 2005

// Jump untill C=00

RST 1

// terminate.

#ORG F100H // store counter of the address.
 #DB 04 // store counter.
 #ORG F200H // store nos of address.
 #DB, DD, LC, BB, AA // store nos.

Input : F100-04H, F200-DDH, F201-0CH, F202-BEH, F203-AAH
 Output : F200-AAH, F201-BBH, F202-0CH, F203-DDH

Assembler						
* Address	Label	Mnemonics	Hexcode	Bytes	M-Cycles	T-States
✓ 2000		MVI A,09	3E	2	2	7
2001			09			
✓ 2002		MVI B,07	06	2	2	7
2003			07			
✓ 2004		CMP B	B8	1	1	4
✓ 2005		JZ DOWN	CA	3	3	10
2006			17			
2007			20			
✓ 2008		JNC SHIFT	D2	3	3	10
2009			0E			
200A			20			
✓ 200B		MOV C,A	4F	1	1	4
✓ 200C		MOV A,B	78	1	1	4
✓ 200D		MOV B,C	41	1	1	4
✓ 200E	SHIFT	SUB B	90	1	1	4
✓ 200F		CMP B	B8	1	1	4
✓ 2010		JZ MOVE	CA	3	3	10
2011			16			
2012			20			

Terminate

Simulate

Start From → 2000

Run all At a Time Step By Step

Registers :									
Register	Value	7	6	5	4	3	2	1	0
Accumulator	01	0	0	0	0	0	0	0	1
Register B	01	0	0	0	0	0	0	0	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	00	0	0	0	0	0	0	0	0
Register L	00	0	0	0	0	0	0	0	0
Memory(H)	00	0	0	0	0	0	0	0	0

Register	Value	S	Z	*	AC	*	P	*	CY
Flag Register	54	0	1	0	1	0	1	0	0

Type	Value
Stack Pointer(SP)	FFFE
Memory Pointer (HL)	0000
Program Status Word(PSW)	0154
Program Counter(PC)	0009
Clock Cycle Counter	244
Instruction Counter	38

SOD	SID	INTR	TRAP	R7.5	R6.5	R5.5
0	0	0	0	0	0	0

For SIM instruction

SOD	SDE	*	R7.5	MSE	M7.5	M6.5	M5.5
0	0	0	0	0	0	0	0

For RIM instruction

SID	I7.5	I6.5	I5.5	IE	M7.5	M6.5	M5.5
0	0	0	0	0	0	0	0

No. Converter Tool :

Hexadecimal	Decimal	Binary
0		0

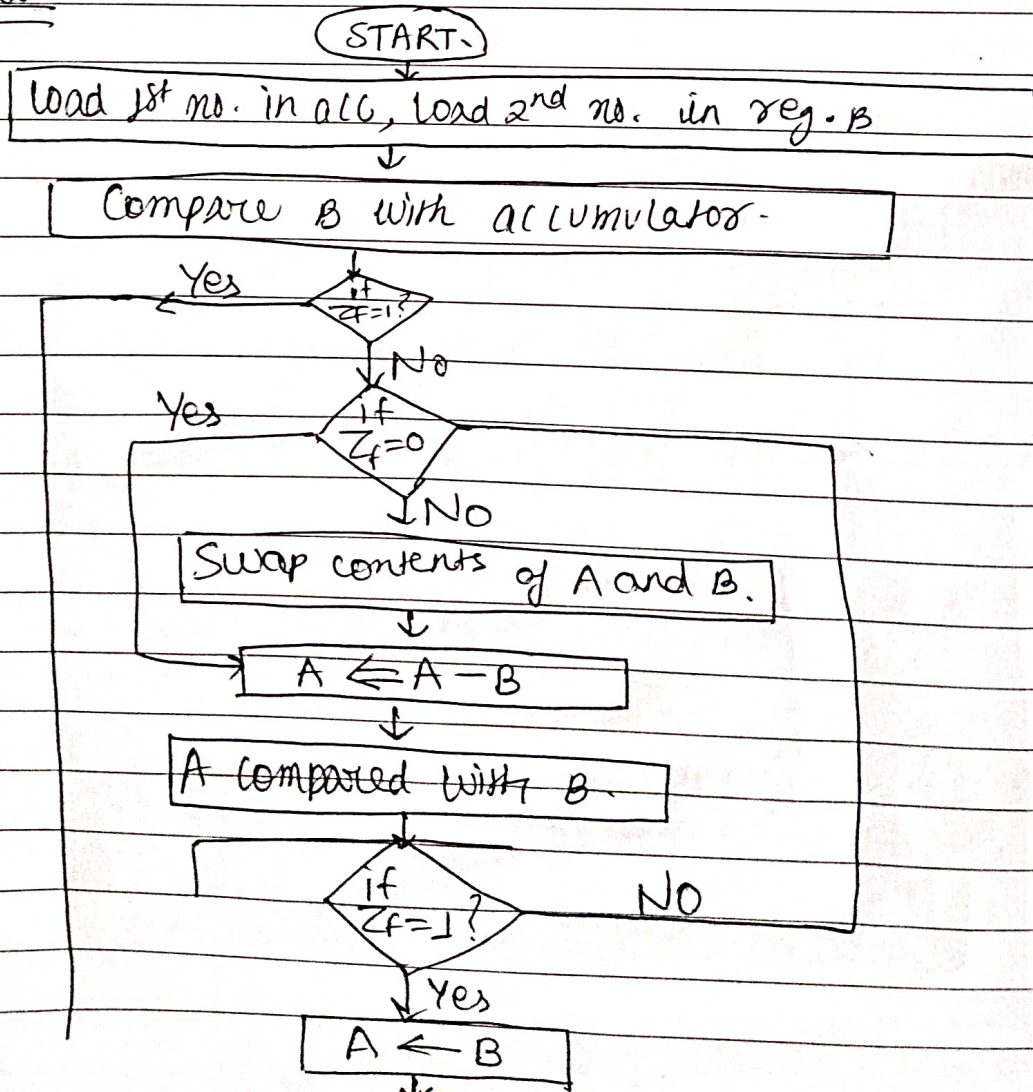
Experiment 8

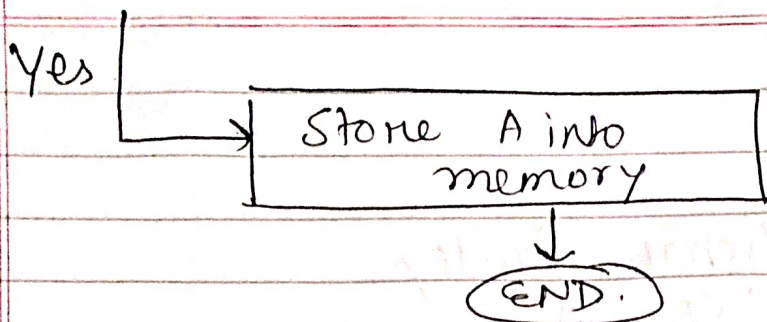
Aim:- Write a program to find GCD of two numbers using 8085 and verify.

Algorithm/Theory

- ① The numbers loaded in accumulator and reg B.
- ② Reg B compared with A.
- ③ if zero flag is set go to step 2.
- ④ If carry flag is reset, go to step 6.
- ⑤ Swap the contents of A and B.
- ⑥ Subtract A and B.
- ⑦ compared B with A.
- ⑧ if zero flag is set go to step 10
- ⑨ Jump to step 4
- ⑩ move contents of B into A.
- ⑪ store contents of A into an address.
- ⑫ Stop.

Flowchart





Code :

#ORG 2000H.

MVI A, 09	// load first no. in reg A.
MVI B, 07	// load second no. in reg B.
CMP B	// compare B to A.
JZ down	// True if A = B.
JNC shift	// True if A > B.
MOV C, A	// $A \Rightarrow C$.
MOV A, B	// $A \Leftarrow B$.
MOV B, C	// $C \Leftarrow B$.
shift: SUB B	// $A - B \Rightarrow A$.
CMP B	// compare B to A.
JZ move	// True if A = B.
JMP 2008	// Jump until A = B.
move: MOV A, B	// $B \Rightarrow A$.
down: STA F200	// $A \Rightarrow [Address]$.
RST 1	// Terminate.

Input: A - 09H, B - 07H.

Output: A - 01H, F200 - 01H.

☐ Assembler

* Address	Label	Mnemonics	Hexcode	Bytes	M-Cycles	T-States
✓ 2000		LDA F100	3A	3	4	13
2001			00			
2002			F1			
✓ 2003		DCR A	3D	1	1	4
✓ 2004		MOV C,A	4F	1	1	4
✓ 2005		MOV B,C	41	1	1	4
✓ 2006		LXI H,F200	21	3	3	10
2007			00			
2008			F2			
✓ 2009	UP	MOV A,H	7E	1	2	7
✓ 200A		INX H	23	1	1	6
✓ 200B		CHP H	8E	1	2	7
✓ 200C		JC DOWN	DA	3	3	10
200D			14			
200E			20			
✓ 200F		MOV D,H	56	1	2	7
✓ 2010		MOV H,A	77	1	2	7
✓ 2011		DCX H	2B	1	1	6
✓ 2012		MOV M,D	72	1	2	7

☐ Simulate

Start From →

0000

Run all At a Time

Step By Step

☐ Memory Editor

Memory Range: 0000 — FFFF

Memory Address	Value
200F	56
2010	77
2011	2B
2012	72
2013	23
2014	05
2015	C2
2016	09
2017	20
2018	0D
2019	C2
201A	05
201B	20
201C	CF
F100	04
F200	DD
F201	CC
F202	BB
F203	AA

- ☐ Show entire memory content
- ☒ Show only loaded memory location
- ☐ Store directly to specified memory location