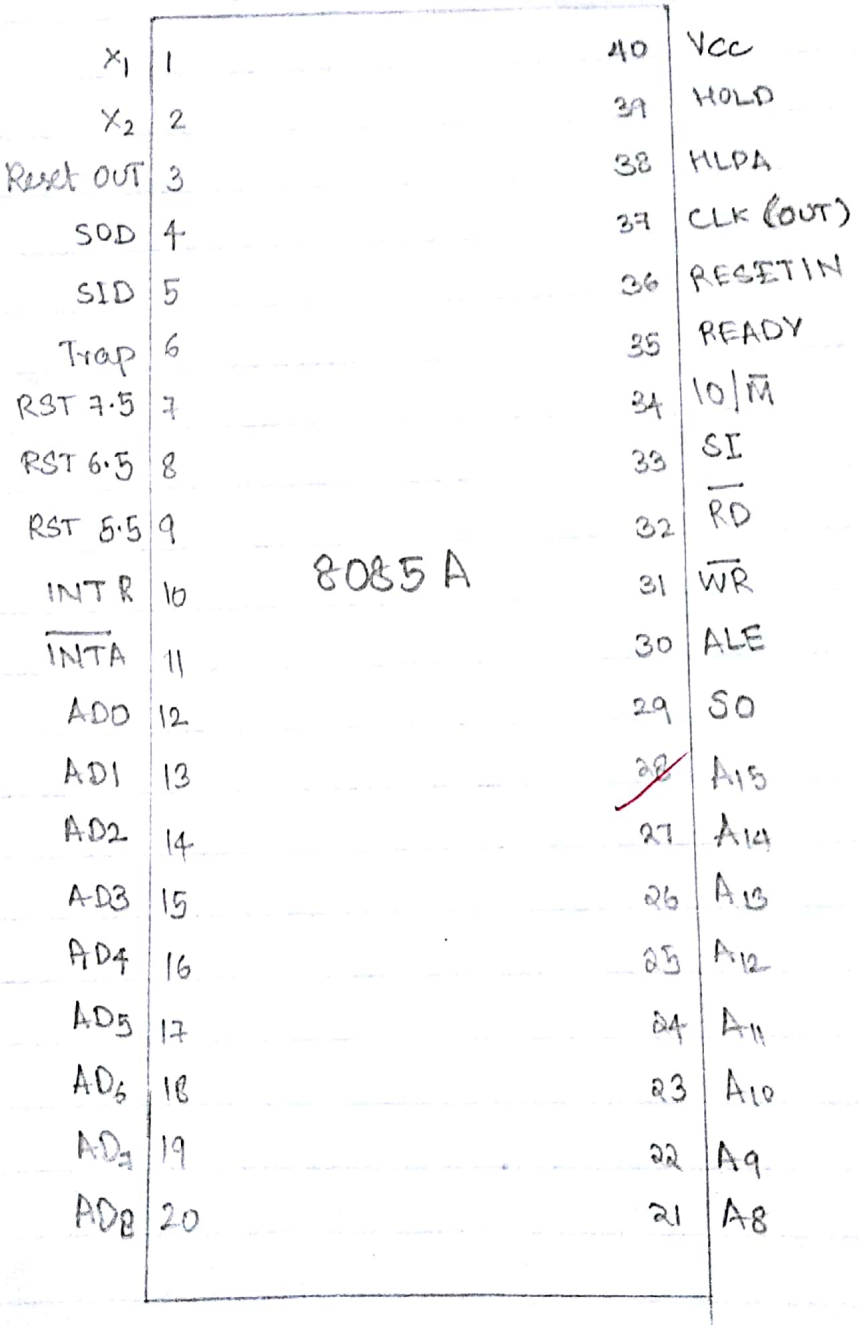
Serial No. Notes Date

EXPERIMENT - 1

AIM :- Introduction to 8085 Microprocessor



8085 Pinout

Signature

Serial No. 02 Notes Date 13/2/16

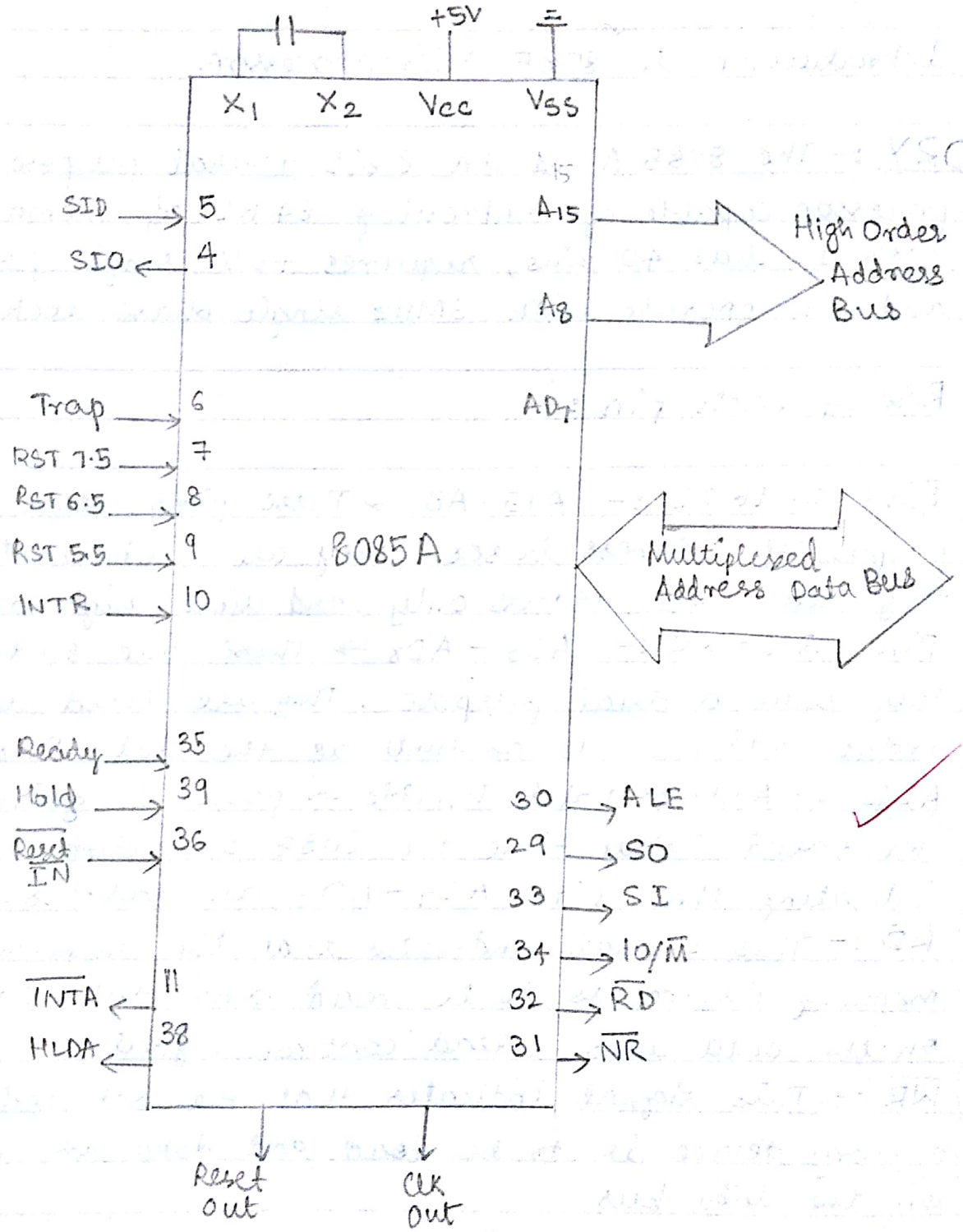
EXPERIMENT-1

AIM: Introduction to 8085 Microprocessor.

THEORY :- The 8085 A is an 8 bit general purpose microprocessor capable of addressing 64 KB of memory. The 8085 IC has 40 pins, requires +5 V single power supply and can operate with 3MHz single phase clock.

Role of each pin :-

1. Pins 21 to 28 :- A15 - A8 → These pins serve the purpose of address buses. They are unidirectional, as they deal with address only and store high order address
2. Pins 12 to 20 :- $AD_0 - AD_8$ → These are bidirectional, they serve a dual purpose. They are used as the low order address bus as well as the data Bus.
3. ALE :- Address Latch Enable → positive going pulse generated every time the 8085 operation begins indicating that bits $AD_7 - AD_0$ are address bits.
4. $\overline{RD}$:- This signal indicates that the selected I/O or memory device is to be read and data are availab on the data bus. (Read control signal).
5. $\overline{WR}$:- This signal indicates that the selected I/O or memory device is to be load and data are available on the data bus.
6. IO/$\overline{M}$:- When high, it indicates an I/O operation, when low it indicates memory operation
7. SI and SO :- Status signals
8. V_{cc} :- +5 Power supply



Signature

9. Vss :- Ground Refrence
10. X_1, X_2 :- A crystal is connected at these two points/pins. The frequency is initially divided by 2. se, crystal should have freqrency of 6MHz
11. CLK (OUT) :- Clock Output → This signal can be used as a system clock for other devices.
12. INTR :- Interrupt request → used as a general purpose interrupt.
13. $\overline{INTA}$:- Interrupt Acknowledge → This is used to acknowledge an interrupt.
14. RST 7.5 / RST 6.5 / RST 5.5 :- Restart Interrupts → These are the interrupts that transfer the program control to specific memory locations. They have higher priorities than INTR interrupt. The priority order among these three is 7.5 > 6.5 > 5.5.
15. HOLD :- This signal indicates that a peripheral such as DMA (Direct Memory access) controller is requesting the use of the address and data buses
16. TRAP :- Non-markable interrupt having highest priority
17. HLDA :- Hold Acknowledge. The signal acknowledges the hold request.
18. READY :- The signal is used to delay the Microprocessor Read or Write cycles until a slow responding peripheral is ready to send or accept data.
19. Reset IN :- When signal goes low, the program counter is set to zero.
20. Rest OUT :- Signals indicate the MPU is being reset. The signals can be used to reset other devices.
21. SIO / SOO :- Sometimes microprocessor works with serial data rather than parallel.

KEYPAD :-

Reset	KB INT	C VRFY	D PRRD	E TRPD	F TPWR	IN BYTE	OUT BYTE
F1	INSERT	8 H	9 L	A PROB	B BLNK	SINGLE STEP	BLK MOVE
COMP	DELETE	4 SPH	5 SPL	6 PCH	7 PCL	EXAM REG	EXAM MEM
NEXT	PREV	0 FILL	1	2	3 I	GO	EXE

Signature

Specifications of 8085 :-

1. CPU :- 8085 is operated at 3072 MHz
2. Memory :- Three 28-pin IEBEC sockets offer 64 KB of memory as follows :-
   16 KB of firmware is one 27128
   32 KB of static RAM using one 62256 with optional battery back up.
   16 KB of PROM/RAM (optional) three jumper selections allow the socket to accomodate 2764/27128/6264.
3. INTERRUPTS :- These are available to user.
4. POWER SUPPLY :- 5V (± 0.1V, 3.0A); ±12V (± 1.0V), 250mA; −12V (± 1.0V), 250mA; 30V (± 2.0V), 100mA

ROLE OF EACH KEY :-

1. Examine/Modify Memory :- Display/modifies the contents of memory location.
2. Examine/Modify Register :- Display/modifies 8085 registers content.
3. Single step :- Executes a single user program instruction
4. GO :- Transfer control from monitor to user program.
5. Block Move :- Moves a block of data from one position to another.
6. Insert :- Inserts one or more instructions in the user program.
7. Delete :- Delets one or more instructions in the user program.
8. Input Byte :- Input a byte from a specified post.
9. Ouput Byte :- Output a byte to the specified post.
10. Compare Memory :- Compare two blocks of memory.

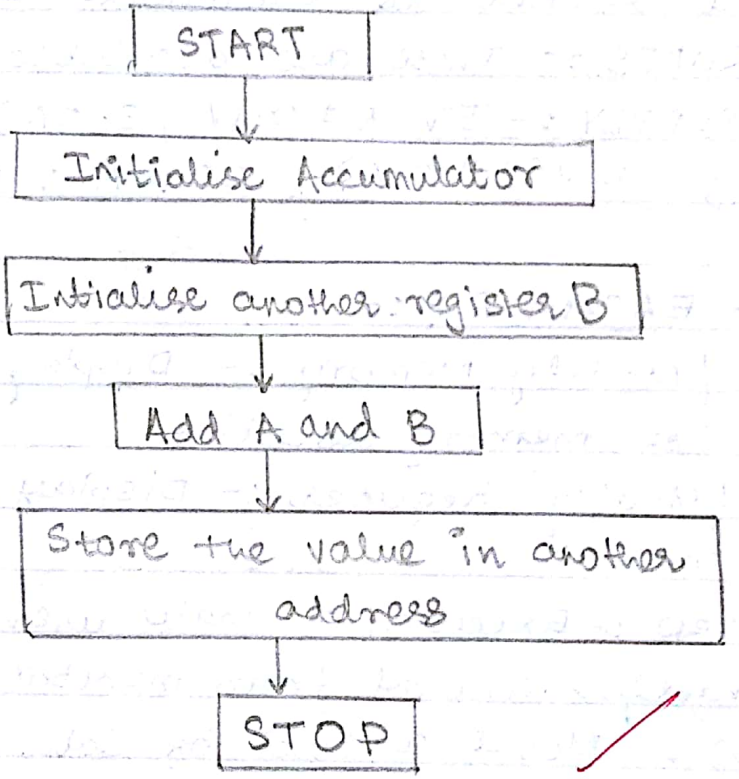
Returned 20/3/18

EXPERIMENT - 2

AIM :- To perform Addition of two 8-bit numbers using assembly language code using Microprocessor 8085 kit.

Apparatus :- 8085 Microprocessor kit, power supply.

Flow chart :-



Result :- Addition of two 8 bit numbers using assembly language code using microprocessor 8085 kit has been performed successfully.

Signature

Serial No. 05 | Notes | Date 13/8/18

EXPERIMENT-2

AIM:- To perform Addition of two 8-bit numbers using assembly language code using microprocessor 8085 kit.

Apparatus :- 8085 Microprocessor kit, power supply.

Program :-

ADDRESS	MNEMONICS	HEX CODE	COMMENTS
8000	MVI A 02	3E	Initialises accumulator
8001		02	stores 02 value in A
8002	MVI B 03	06	Initialises register B
8003		03	Stores 03 in B
8004	ADD B	80	Add the value B in A
8005	STA 8050	32	Stores the added
8006		50	Value at 8050
8007		80	address
8008	HLT	76	Stops program execution

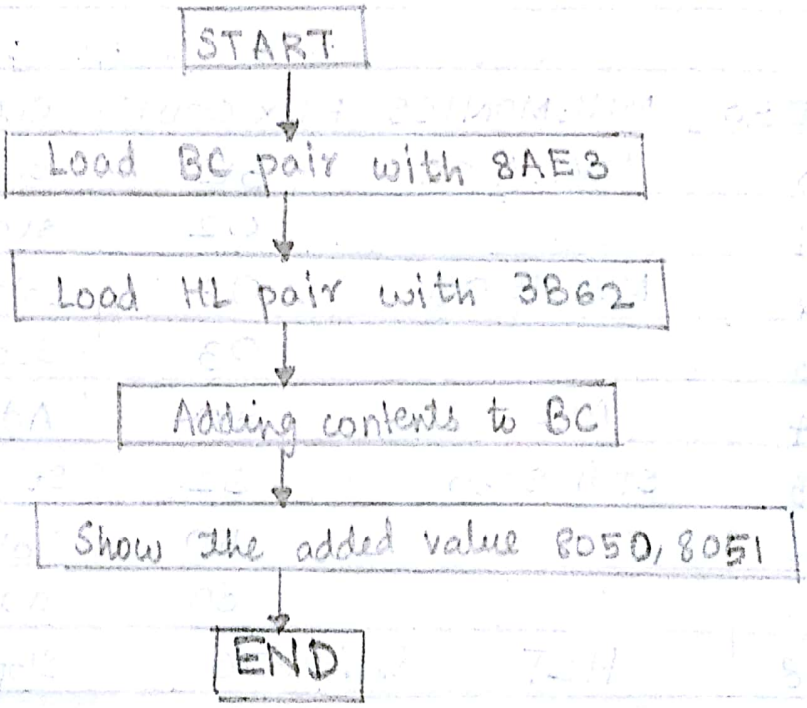
Result :- Addition of two 8 bit numbers using assembly language code using microprocessor kit has been performed successfully.

Akshira 20/8/18

AIM :- Addition of two 16 bit numbers using assembly language code with microprocessor 8085.

Apparatus :- 8085 Microprocessor kit, power supply.

FLOW CHART :-



Signature

EXPERIMENT-3

AIM: Addition of two 16 bit numbers using assembly language code with microprocessor 8085.

Apparatus :- 8085 Microprocessor kit, power supply.

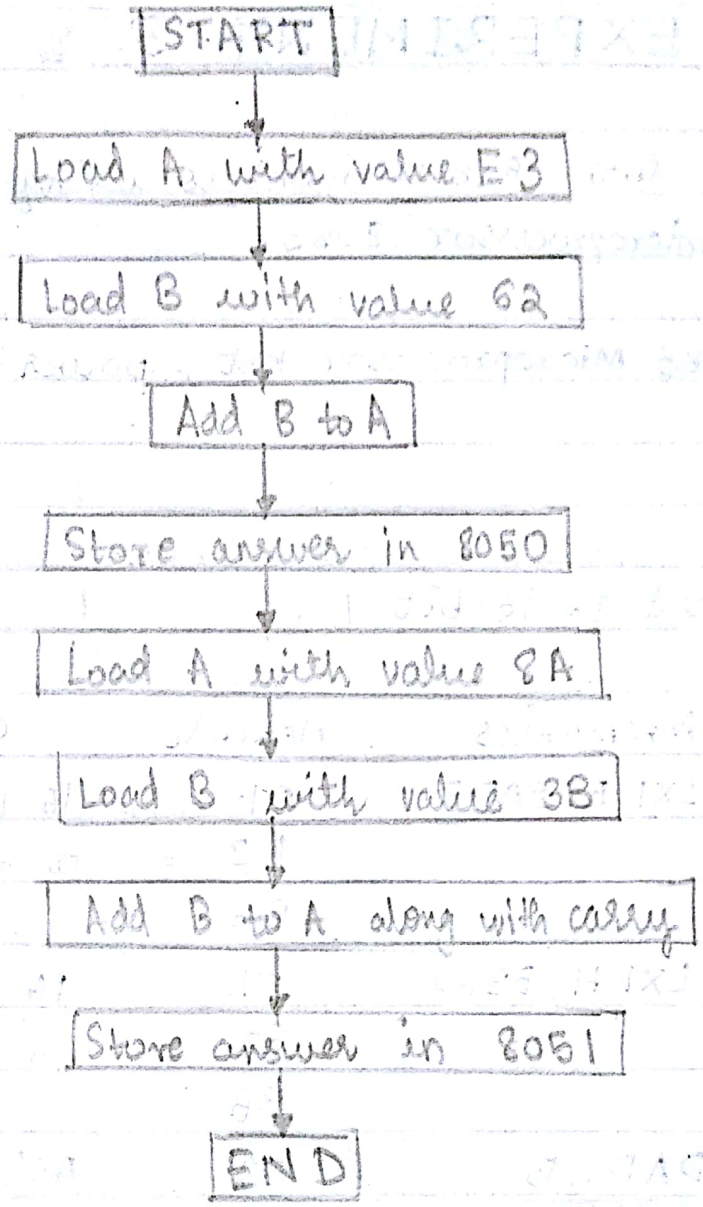
Program :-

(i) Register used as 16-bit

Address	Mnemonics	Hexcode	Comments
8000	LXI B, 8AE3	01	16 bit data added to register BC.
8001		E3	
8002		8A	
8003	LXI H, 3B62	21	16 bit data added to register HL.
8004		62	
8005		3B	
8006	DAD B	09	Adding BC to HL.
8007	SHLD 8050	22	Storing of data i.e, lower bits to 8050 and
8008		50	
8009		80	higher bits to 8051
800A	HLT	76	Stops program execution

(ii) Register used as 8-bit

Address	Mnemonics	Hexcode	Comments
8000	MVI A, E3	3E	Initialising accumulator
8001		E3	to E3 value



Result :- Addition of two 16 bit numbers using assembly language code using microprocessor 8085 has been performed successfully.

Signature

Serial No. 07 Notes Date

8002	MVI B, 62	06	Initialise B register
8003		62	with 62 value
8004	ADD B	80	Addition
8005	STA 8050	32	Result stored in
8006		50	Accumulator at
8007		80	location 8050
8008	MVI A, 8A	3E	Initialise A register
8009		8A	with 8A
800A	MVI B, 3B	06	Initialise B register
800B		3B	with 3B
800C	ADC B	88	Addition with carry
800D	STA 8051	32	Result stored in
800E		51	address 8051
800F		80	
8010	HLT	76	Stops the program
			execution

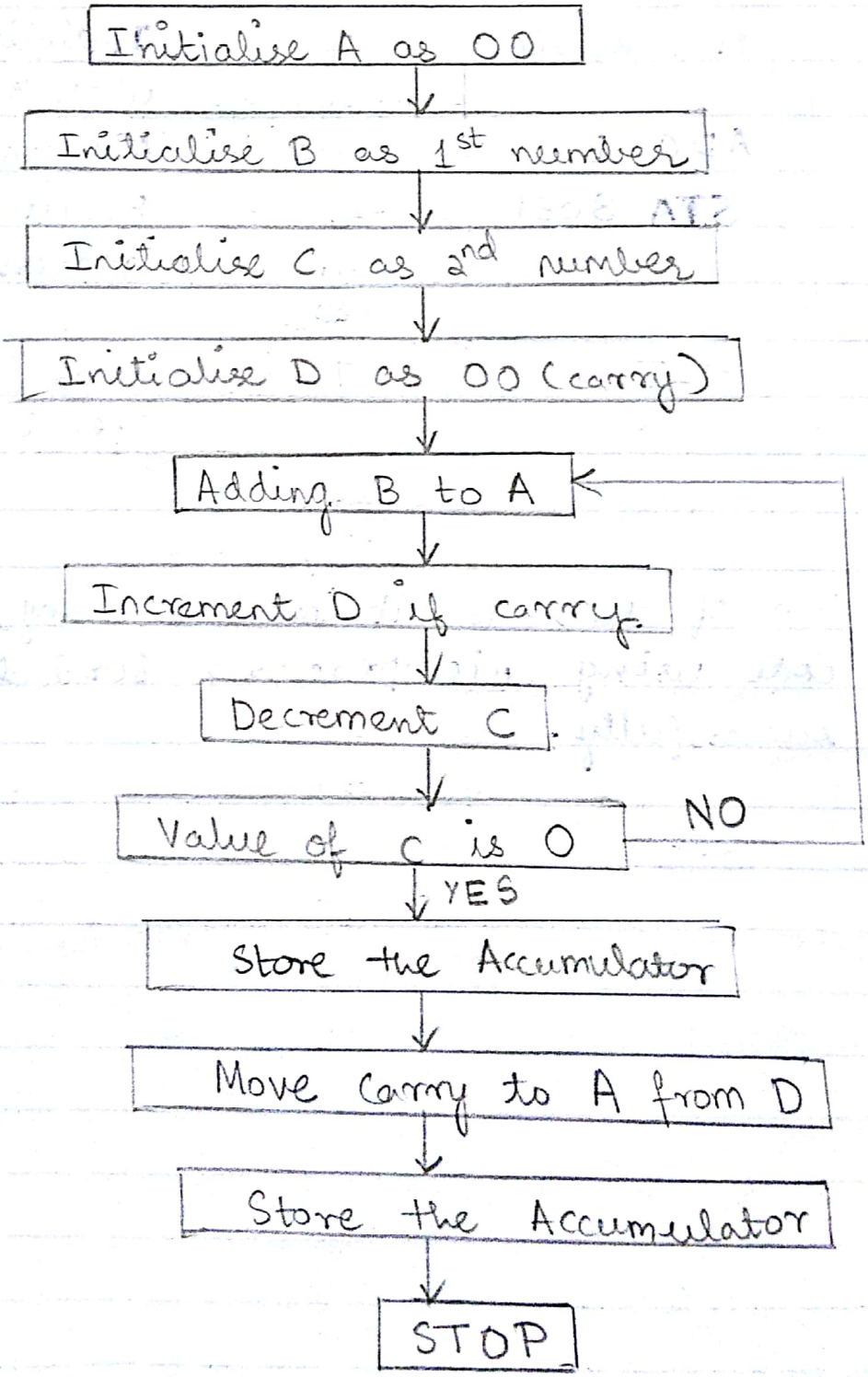
Result :- Addition of two 16-bit numbers using assembly language code using microprocessor 8085 has been performed successfully.

17/9/18

EXPERIMENT - 4

AIM :- Multiplication of two 8-bit numbers using 8085 microprocessor kit.

Apparatus :- 8085 microprocessor kit, power supply.



Signature

EXPERIMENT - 4

AIM: Multiplication of two 8-bit numbers by using 8085 microprocessor kit.

Apparatus :- 8085 microprocessor kit, power supply.

PROGRAM :-

Address	Mnemonics	Hexcode	Comments
8000	MVI A, 00	3E	Stores 00 in the
8001		00	accumulator A
8002	MVI B, 48	06	Stores 48 in the B
8003		48	register
8004	MVI C, 04	0E	Stores 04 in the C
8005		04	register
8006	MVI D, 00	16	Stores 00 in the D
8007		00	register
8008	ADD B	80	Adding B to A
8009	JNC 800D	D2	Jump to address 800D
800A		0D	if no carry is generated
800B		80	
800C	INR D	14	Increment D
800D	DCR C	0D	Decrement C
800E	JNZ 8008	C2	Jump to address 8008
800F		08	if no zero is generated
8010		80	
8011	STA 8051	32	Stores the value of
8012		51	accumulator into the
8013		80	address 8051

EXPERIMENT 4

Result :- Multiplication of two 8-bit numbers has been performed successfully.

Signature

Serial No. 09 Notes Date

8014	MOV A, D	7A	Moves contents from D to A
8015	STA 8050	32	Stores the value of
8016		50	carry in the accumulator
8017		80	to address 8050.
8018	HLT	76	Stops the execution
			of the program

Result :- Multiplication of two 8-bit numbers has been performed successfully.

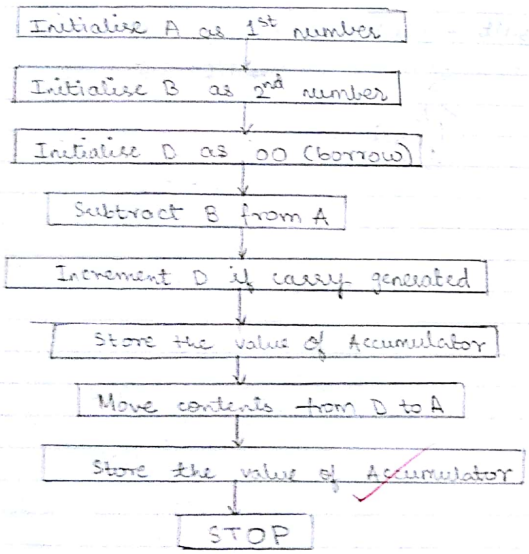
21/9/18

Serial No. _____ Notes _____ Date _____

EXPERIMENT-5

M :- Subtraction of two 8 bit numbers with or without borrow using 8085 microprocessor kit.

ratus :- 8085 microprocessor kit, power supply.



Signature

Serial No. 10 Notes _____ Date _____

EXPERIMENT-5

AIM: Subtraction of two 8 bit numbers with or without borrow using 8085 microprocessor kit

Apparatus :- 8085 microprocessor kit, power supply.

PROGRAM :-

Address	Mnemonics	Hexcode	Comments
8000	MVI A, 06	3E	Stores the value
8001		06	in accumulator A
8002	MVI B, 02	06	Stores the value
8003		02	in register
8004	MVI D, 00	16	Stores the value
8005		00	in register D
8006	SUB B	90	Subtracting B fro
8007	JNC 800B	D2	Jump to address
8008		0B	if no carry
8009		80	generated
800A	INR D	14	Increment D
800B	STA 8050	32	Store the value
800C		50	A to the addr
800D		80	8050
800E	MOV A, D	7A	Move contents from
800F	STA 8051	32	Store the value
8010		51	(Carry) to the
8011		80	address 8051
8012	HLT	76	Stops the execu
			of program

Serial No. Notes Date

ult :- Subtraction of two 8-bit numbers with or without borrow has been performed successfully.

Serial No. 11 Notes Date

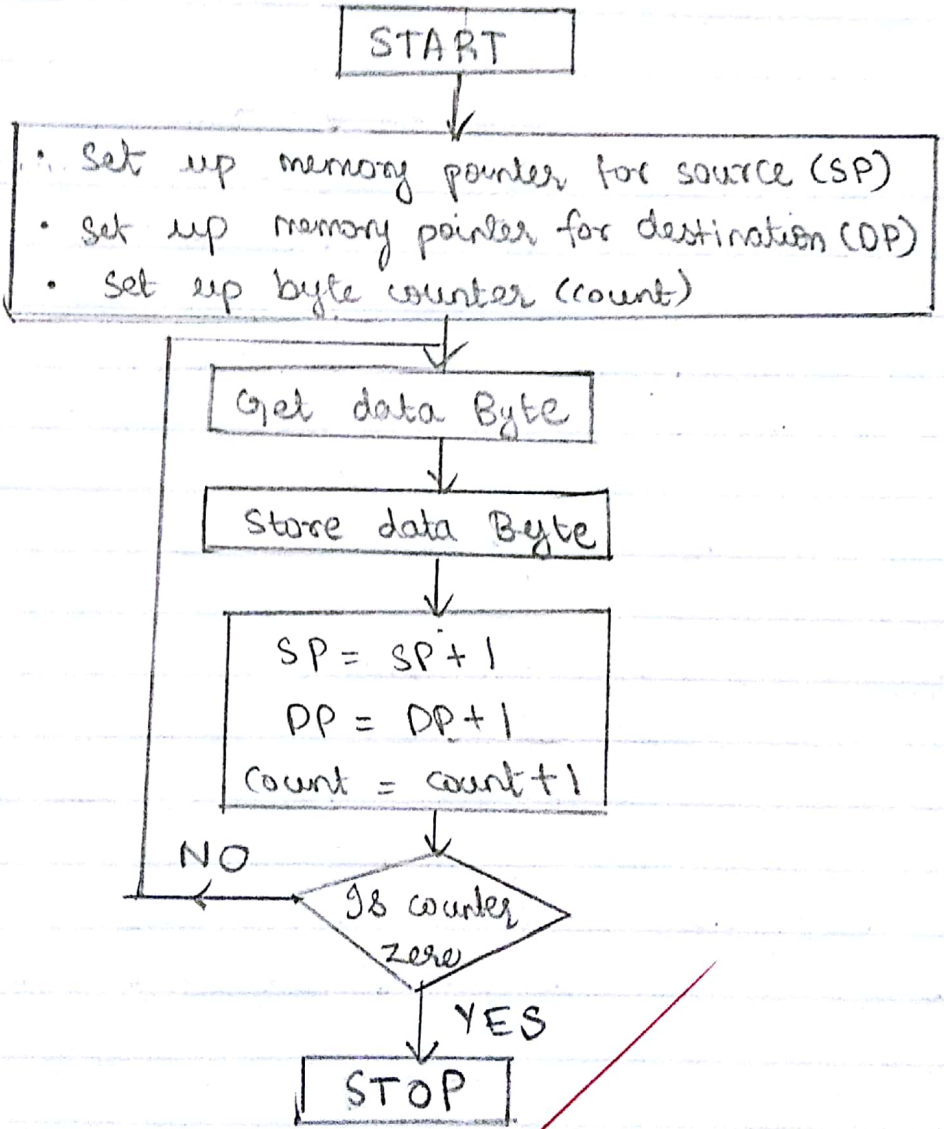
Result :- Subtraction of two 8-bit numbers with or without borrow has been performed successfully

3/9/18

EXPERIMENT-6

AIM:- Write a program to transfer 10 bytes of data from 2000 to 8050 in (i) SAME ORDER, (ii) REVERSE ORDER.

Apparatus :- 8085 Microprocessor kit, power supply.



Signature

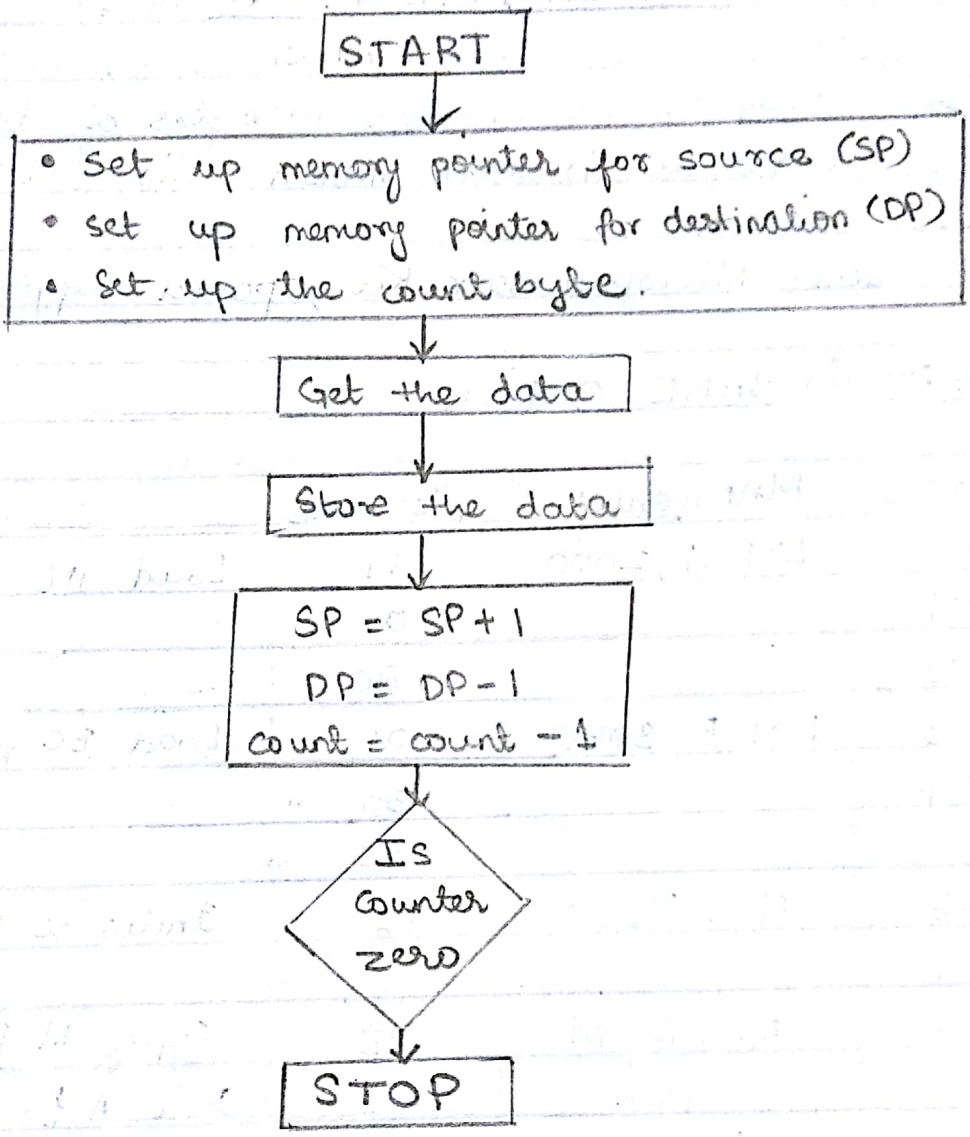
EXPERIMENT-6

AIM:- Write a program to transfer 10 bytes of data from 2000 to 8050 in (i) SAME ORDER ; (ii) REVERSE ORDER.

Apparatus:- 8085 Microprocessor kit, power supply.

PROGRAM:- (i) SAME ORDER

Address	Mnemonics	Hexcode	Comments
8000	LXI H, 2000	21	Load HL with 2000
8001		00	
8002		20	
8003	LXI B, 8050	01	Load BC pair with 8050
8004		50	
8005		80	
8006	MVI D, 10	16	Initialise D as 10
8007		10	
8008	MOV A, M	7E	Copy M to A
8009	STAX B	02	Store A to the memory in BC
800A	INX H	23	Increment HL pair
800B	INX B	03	Increment BC pair
800C	DCR D	15	Decrement D
800D	JNZ 8008	C2	Jump to 8008 if there is
800E		08	no zero
800F		80	
8010	HLT	76	Stop the execution



Result :- Transfer of 10 data bytes from one memory location to other in same & reverse order is studied.

Signature

Serial No. 13 Notes Date

(ii) REVERSE ORDER

Address	Mnemonics	Hexcode	Comments
8000	LXI H, 8000	21	Load HL with 2000
8001		00	
8002		20	
8003	LXI B, 8059	01	Load BC with 8059
8004		50	
8005		80	
8006	MVI D 0A	16	Initialise register D=0A
8007		0A	
8008	MOV A, M	7E	Move contents of M to A
8009	STAX B	02	Store the contents of A at BC
800A	INX H	23	INR HL pair
800B	DCX B	0B	Decrement BC pair
800C	DCR D	15	Decrement the conts of D by 1
800D	JNZ 8008	C2	Jump to 8008 if there
800E		08	is no zero
800F		80	
8010	HLT	76	Stop program execution

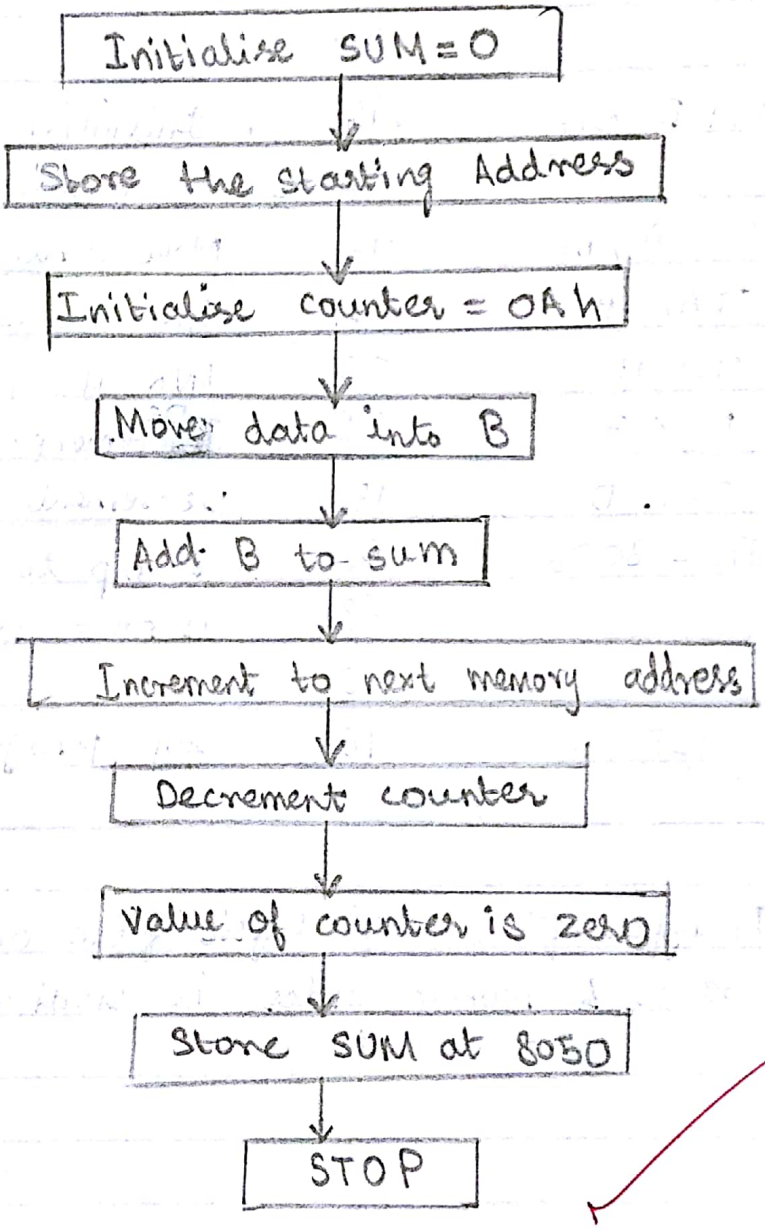
RESULT:- Transfer of 10 data bytes from one memory location to other in same & reverse order is studied.

Ashwini 7/10/18

EXPERIMENT-7

AIM :- Write a program to add 10 bytes of data starting from address 2000.

Apparatus :- 8085 Microprocessor kit, power supply.



Signature

EXPERIMENT-7

AIM:- Write a program to add 10 bytes of data starting from address 2000.

Apparatus:- 8085 Microprocessor kit, power supply.

PROGRAM:-

Address	Mnemonics	Hexcode	Comments
8000	MVI A, 00h	3E	Initialise SUM = 0 by
8001		00	storing A = 0
8002	LXI H, 2000h	21	Storing the starting
8003		00	address from which
8004		20	addition should take place
8005	MVI C, 0Ah	0E	For number of datas
8006		0A	here C = 0Ah (10)
8007	MOV B, M.	46	Moving data from M to B
8008	ADC B	88	Adding B to A with carry
8009	INX H	23	Incrementing memory address
800A	DCR C	0D	Decrementing counter
800B	JNZ 8007h	C2	If not zero, then
800C		07	move to address 8007
800D		80	
800E	STA 8050	32	Storing the sum at
800F		50	8050h address
8010		80	
8011	HLT	76	Stops the execution of
			program

Serial No. Notes Date

lt :- 10 data bits starting from address 2000 have been successfully added and stored at 8050.

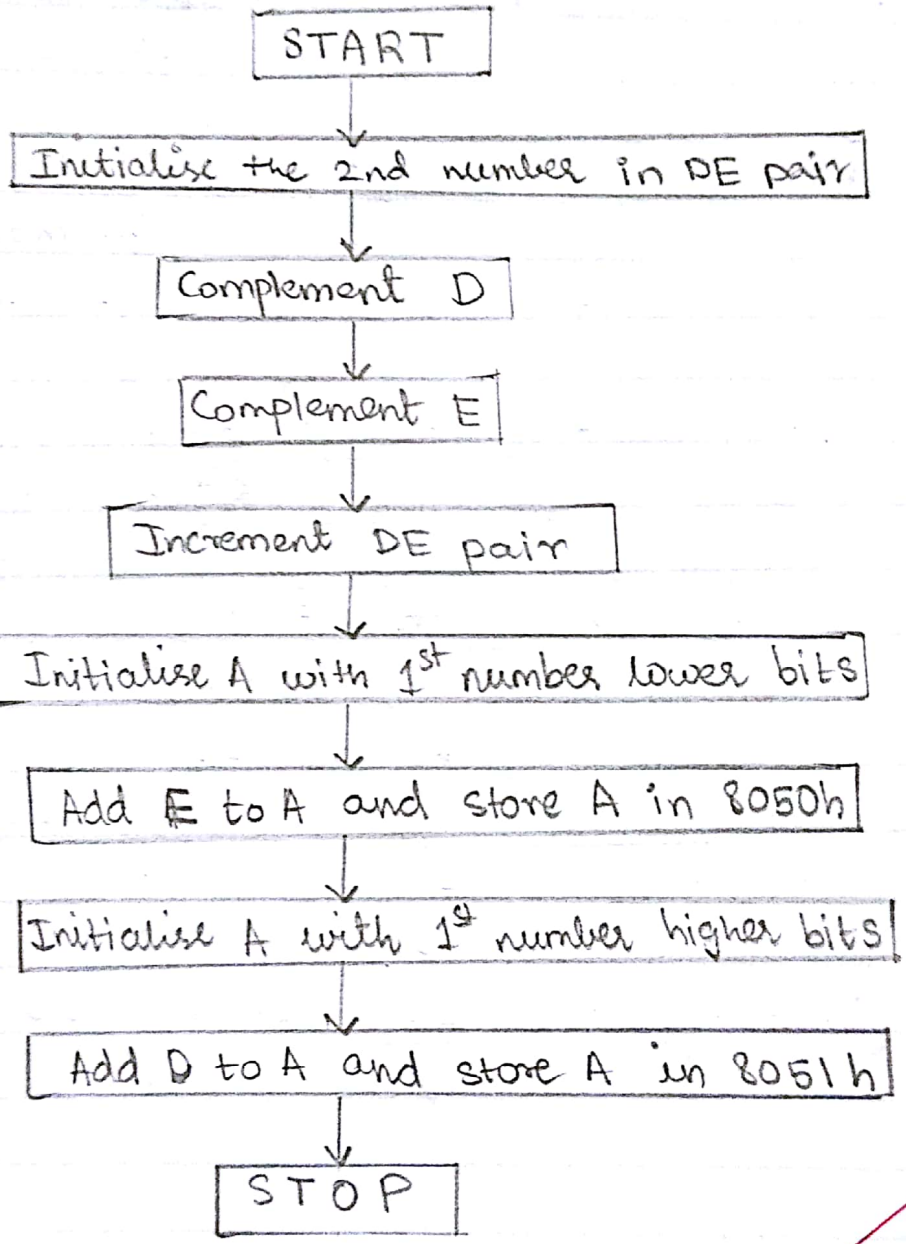
Serial No. 15 Notes Date

RESULT :- 10 data bits starting from address 2000 have been successfully added and stored at 8050.

EXPERIMENT-8

AIM :- To subtract two 16-bit numbers using the 2's complement method.

Apparatus :- 8085 Microprocessor kit, power supply.



EXPERIMENT-8

AIM:- To subtract two 16-bit numbers using the 2's complement method.

Apparatus:- 8085 Microprocessor kit, power supply.

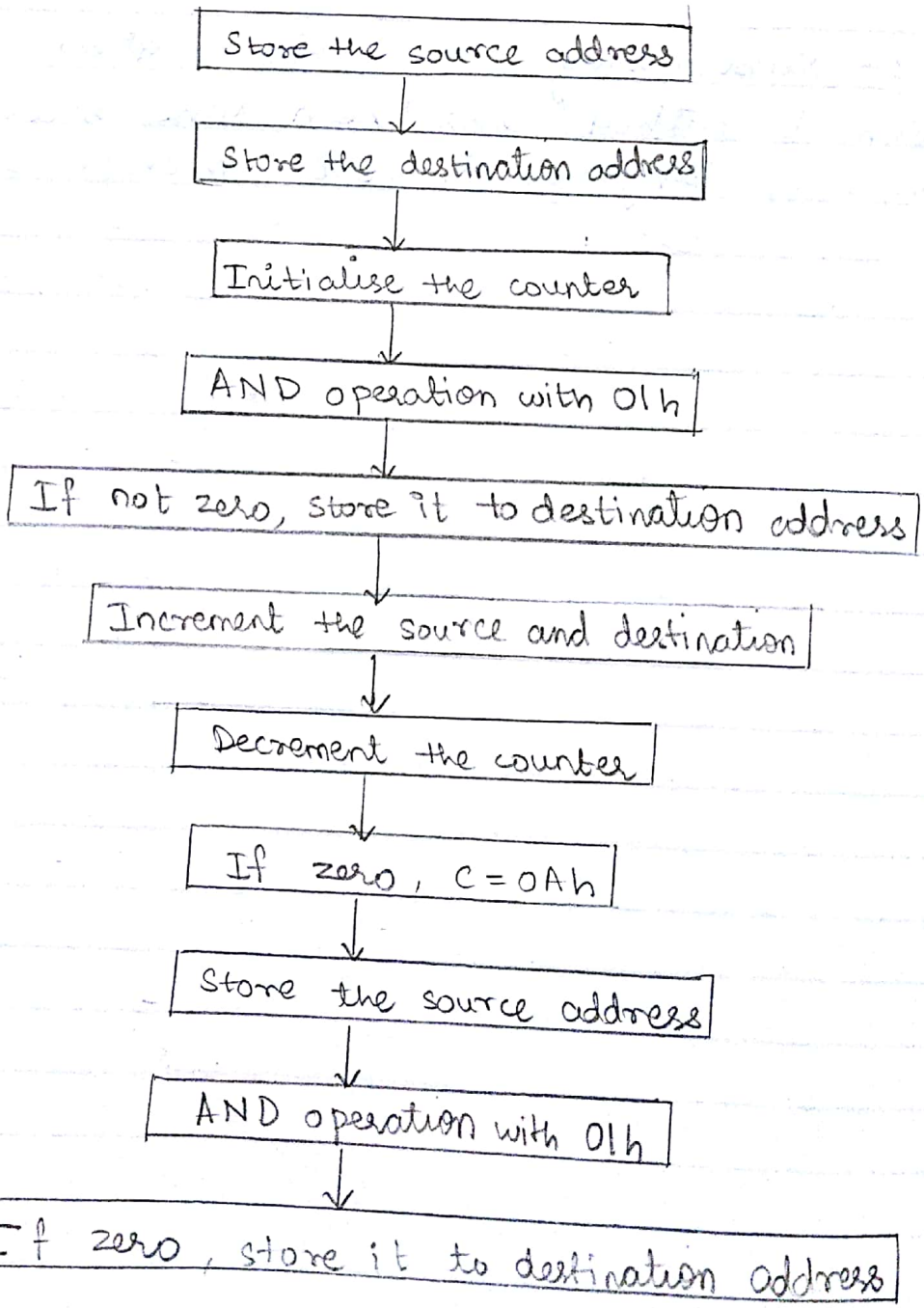
PROGRAM:-

Address	Mnemonics	Hexcode	Comments
8000	LXI D, 2612h	11	Store the number to
8001		12	be subtracted in DE
8002		26	register pair
8003	MOV A, D	7A	Moving D to A
8004	CMA	2F	Complementing A
8005	MOV D, A	57	Moving A to D
8006	MOV A, E	7B	Moving E to A
8007	CMA	2F	Complementing A
8008	MOV E, A	5F	Moving A to E
8009	INX D	13	Incrementing DE pair
800A	MVI A, 32h	3E	Store first number's lower
800B		32	bits in A
800C	ADD E	83	Adding E to A
800D	STA 8050h	32	Store the lower data
800E		50	bits at 8050h address
800F		80	
8010	MVI A, 78h	3E	First number's higher
8011		78	data bits are stored in A
8012	ADC D	8A	Adding D to A with carry

EXPERIMENT - 9

AIM :- To seperate and store even and odd data from contents of consecutive address

Apparatus :- 8085 Microprocessor kit, power supply.



Signature

Serial No. 18 Notes Date

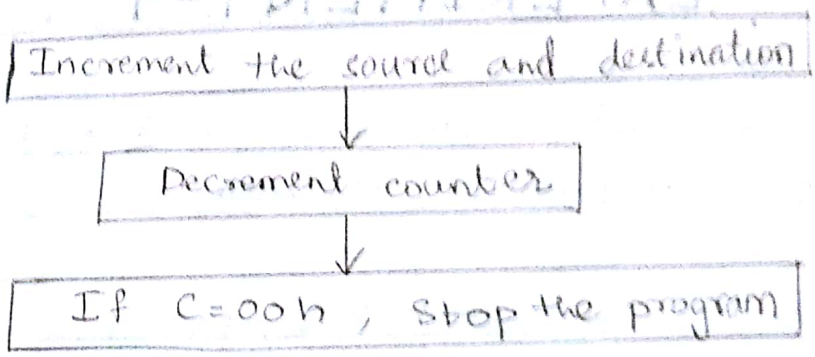
EXPERIMENT-9

AIM:- To seperate and store even and odd data from contents of consecutive addresses.

Apparatus :- 8085 Microprocessor kit, power supply.

PROGRAM :-

Address	Mnemonics	Hexcode	Comments
8000	LXI H, 2000h	21	storing the memory address
8001		00	from where the contents start
8002		20	
8003	LXI D, 8050h	11	The address where the contents
8004		50	seperated are stored in copied
8005		80	in DE pair
8006	MVI C, 0Ah	0E	Counter for 10 data addresses
8007		0A	initialised
8008	MOV A, M	7E	Move content from memory to A
8009	MVI B, 01h	06	Storing 01h in B register
800A		01	
800B	ANA B	A0	AND operation of B & A
800C	JZ 8012h	CA	if zero then move to 8012h
800D		12	address
800E		80	Move content into A from memory
800F	MOV A, M	7E	storing A at DE address
8010	STAX D	12	
8011	INX D	13	Incrementing memory address
8012	INX H	23	Incrementing memory address
8013	DCR C	0D	Decrementing the counter.



RESULT :- The even and odd contents are seperated and stored successfully.

Serial No. 19 | Notes | Date

8014	JNZ 8008h	C2	If not zero then jump
8015		08	to 8008 address
8016		80	
8017	MVI C, 0Ah	0E	Counter for 10 bits of
8018		0A	data initialized
8019	LXI H, 2000h	21	Storing the source address
801A		00	in memory
801B		80	
801C	MOV A, M	7E	Moving data into A from memory
801D	ANA B	A0	AND operation of A & B
801E	JNZ 8024h	C2	If not zero, move to
801F		24	8024h address
8020		80	
8021	MOV A, M	7E	Moving data from M to A.
8022	STAX D	12	Storing A at DE address
8023	INX D	13	Incrementing DE
8024	INX H	23	Incrementing HX
8025	DCR C	0D	Decrementing the counter
8026	JNZ 801Ch	C2	If not zero, move to
8027		1C	801C address.
8028		80	
8029	HLT	76	Stops the execution of program

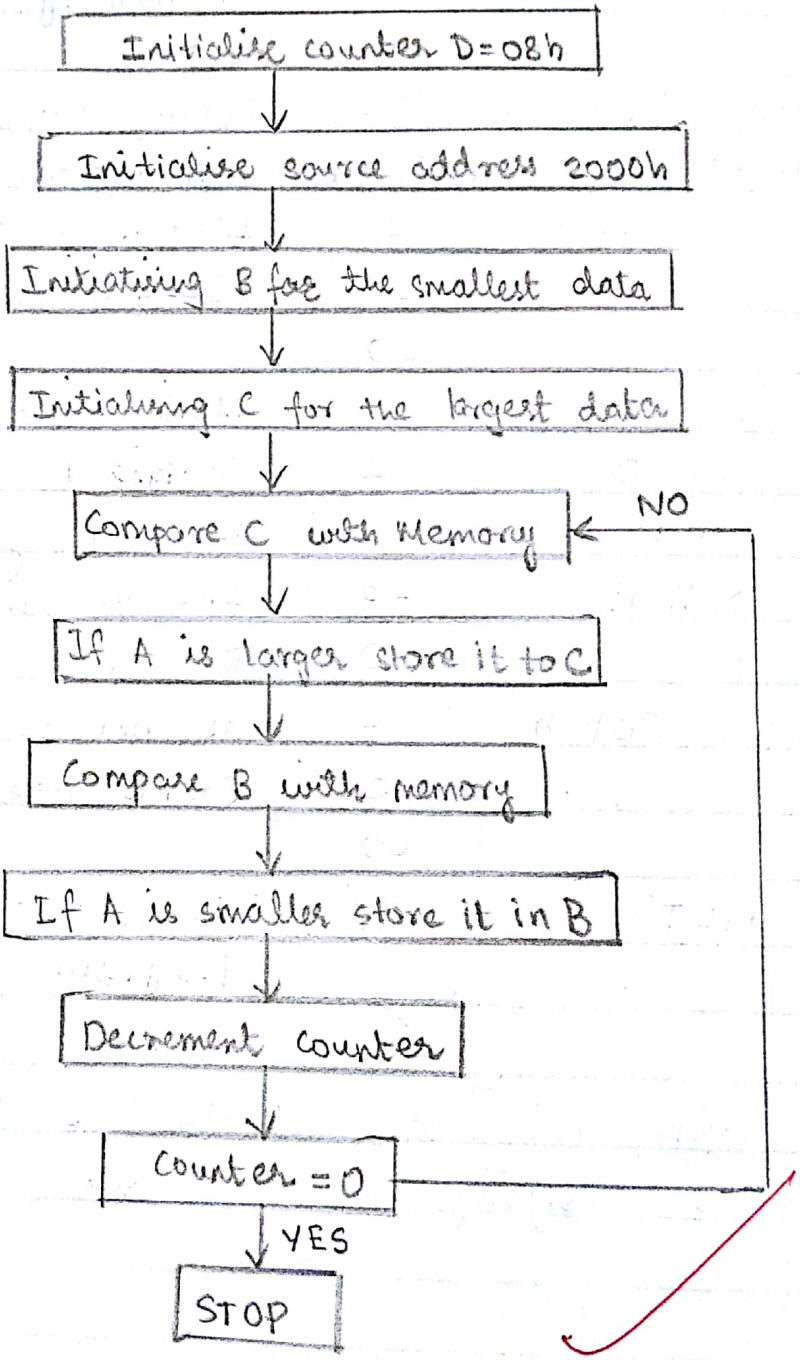
Result :- The even and odd contents are seperated and stored successfully.

15/10/18

EXPERIMENT - 10

AIM :- To find the maximum and minimum number from the given data.

Apparatus :- 8085 Microprocessor kit, power supply.



Signatu

Serial No. 20 | Notes | Date

EXPERIMENT-10

AIM: To find the maximum and minimum number from a given data.

Apparatus :- 8085 Microprocessor kit, power supply.

PROGRAM :-

Address	Mnemonics	Hexcode	Comments
8000	MVI D, 08h	16	Initialise D with no. of
8001		08	data given
8002	LXI H, 2000h	21	Initialise Memory with the
8003		00	source of address of data
8004		20	
8005	MVI B, FFh	06	B for storing smallest data
8006		FF	
8007	MVI C, 00h	0E	C for storing largest data
8008		00	
8009	MOV A, M	7E	Moving from M to A
800A	CMP C	B9	Comparing A with C
800B	JC 800Fh	DA	If carry, jump to 800Fh
800C		0F	address
800D		80	
800E	MOV C, M	4E	Move M to C
800F	MOV A, M	7E	Move M to A
8010	CMP B	B8	Comparing A with B
8011	JNC 8015h	D2	If no carry, move to 8015h
8012		15	address
8013		80	
8014	MOV B, M	46	Move M to B

Result :- The maximum and minimum number has been found from the given data

Sign

Serial No. 21

Notes

Date

8015	INX H	23	Increment Memory
8016	DCR D	15	Decrement Counter
8017	JNZ 8009h	C2	If no zero, move to
8018		09	8009 h address
8019		80	
801A	LXI H, 8050h	21	For storing the smallest
801B		50	data
801C		80	
801D	MOV M, B	70	Move B to M
801E	INX H	23	For storing the longest data
801F	MOV M, C	71	Move C to M
8020	HLT	76	Stops execution of program

Result :- The maximum and minimum number has been found from the given data.

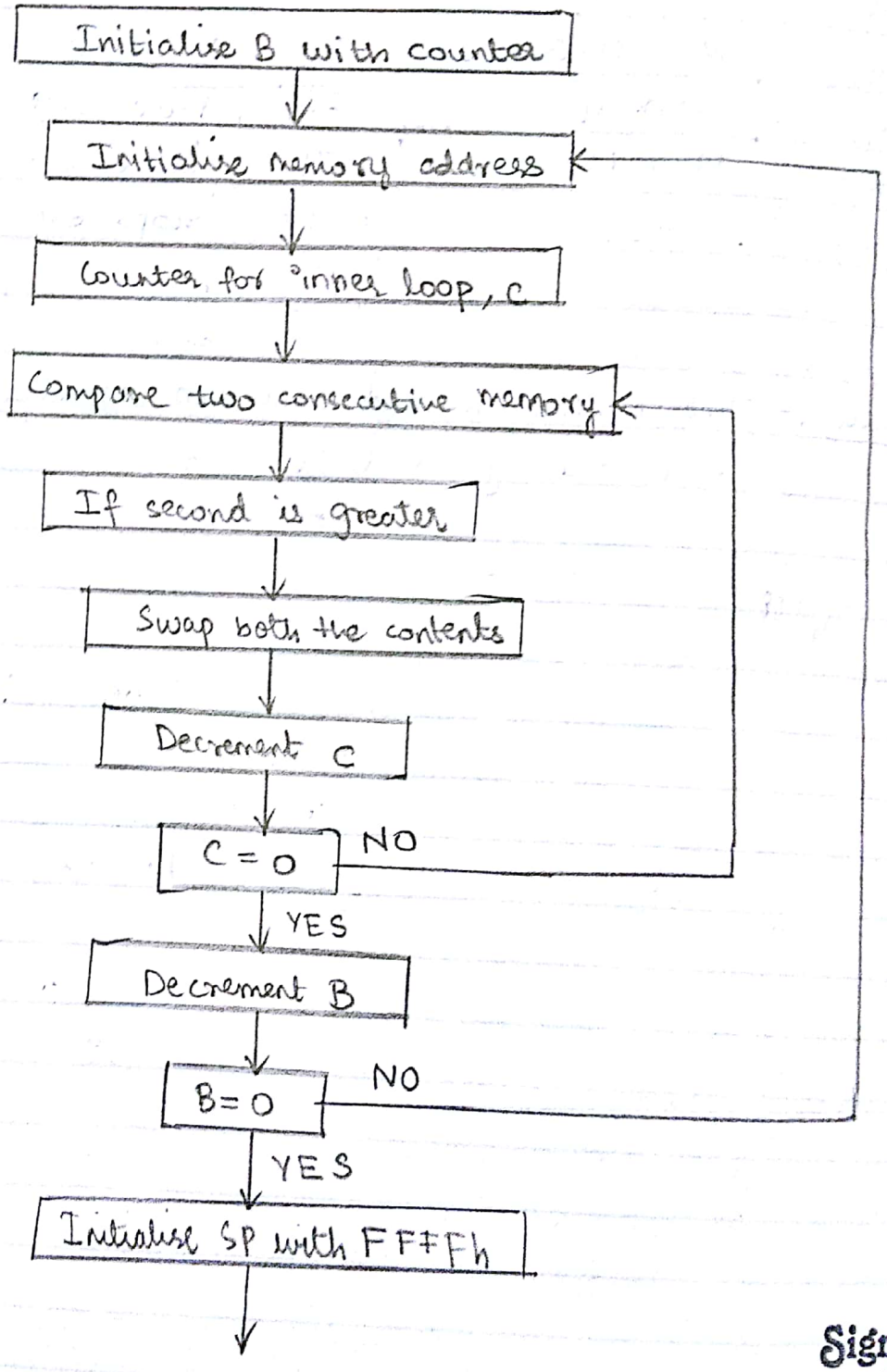
29/11/18

EXPERIMENT-11

AIM:- To arrange the given data in ascending and descending order.

Apparatus:- 8085 Microprocessor kit, power supply.

Flow Chart:-



Signature

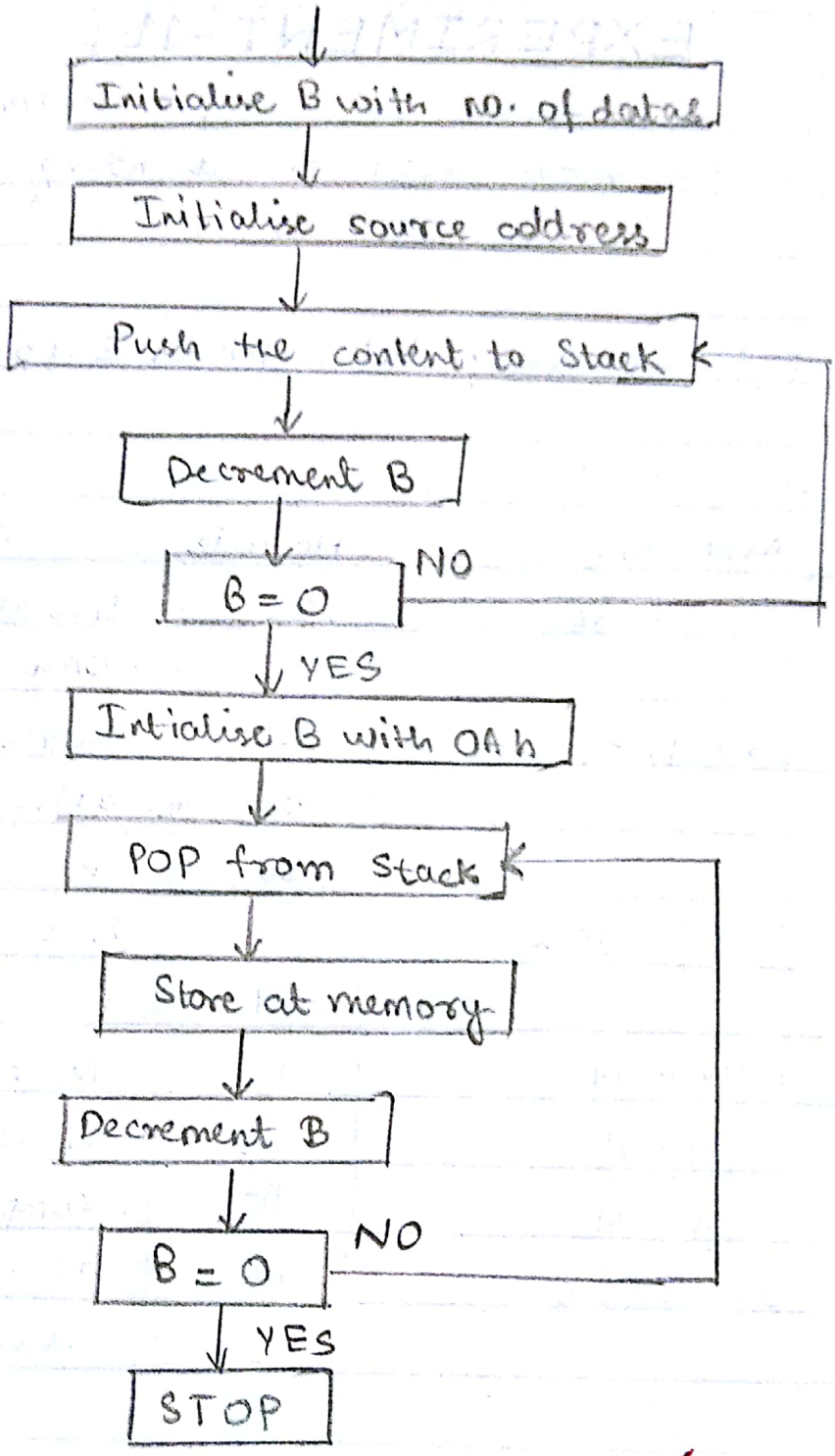
EXPERIMENT-11

AIM:- To arrange the given data in ascending and descending order.

Apparatus :- 8085 Microprocessor kit, power supply.

PROGRAM :-

Address	Mnemonics	Hexcode	Comments
8000	MVI B, 0Ah	06	Initialise counter with
8001		0A	0Ah
8002	LXI H, 2000h	21	Source data at 2000h
8003		00	address
8004		20	
8005	MVI C, 09h	0E	Initialise C with 09h
8006		09	,
8007	MOV A, M	7E	Move M to A
8008	INX H	23	Increment Memory
8009	CMP M	BE	Compare M with A
800A	JC 8012h	DA	If carry, move to 8012h
800B		12	address
800C		80	
800D	MOV D, M	56	MOVE M to D
800E	MOV M, A	77	Move A to M
800F	DCX H	2B	Decrement Memory
8010	MOV M, D	72	Move D to M
8011	INX H	23	Increment Memory
8012	DCR C	0D	Decrement C
8013	JNZ 8007	C2	If non zero, move to
8014		07	8007h address



Signature

8015		80	
8016	DCR B	05	Decrement B
8017	JNZ 8002h	C2	If non-zero, move to
8018		02	8002h address
8019		80	
801A	LXI SP, FFFh	31	Initialise stack pointer
801B		FF	with FFFh
801C		FF	
801D	MVI B, 0Ah	06	Initialise B with 0Ah
801E		0A	
801F	LXI H, 2000h	21	Initialise memory with
8020		00	2000h address
8021		20	
8022	MOV D, M	56	Move M to D
8023	PUSH D	D5	Push D into Stack
8024	INX H	2B	Increment memory
8025	DCR B	05	Decrement B
8026	JNZ 8022h	C2	If non zero, move to
8027		22	8022h address
8028		80	
8029	MVI B, 0Ah	06	Initialise B with 0Ah
802A		0A	
802B	POP D	D1	Pop D from Stack
802C	MOV M, D	56	Move D to M
802D	INX H	2B	Increment memory
802E	DCR B	05	Decrement B
802F	JNZ 802B	C2	If non-zero, move to
8030		28	802Bh address.
8031		80	
8032	HLT	76	Stops the execution of
			the program

Serial No. Notes Date

Result :- The data is arranged in ascending and descending order successfully.

Serial No. 24 Notes Date

Result :- The content or data is arranged in ascendin and descending order successfully.

29/10/20

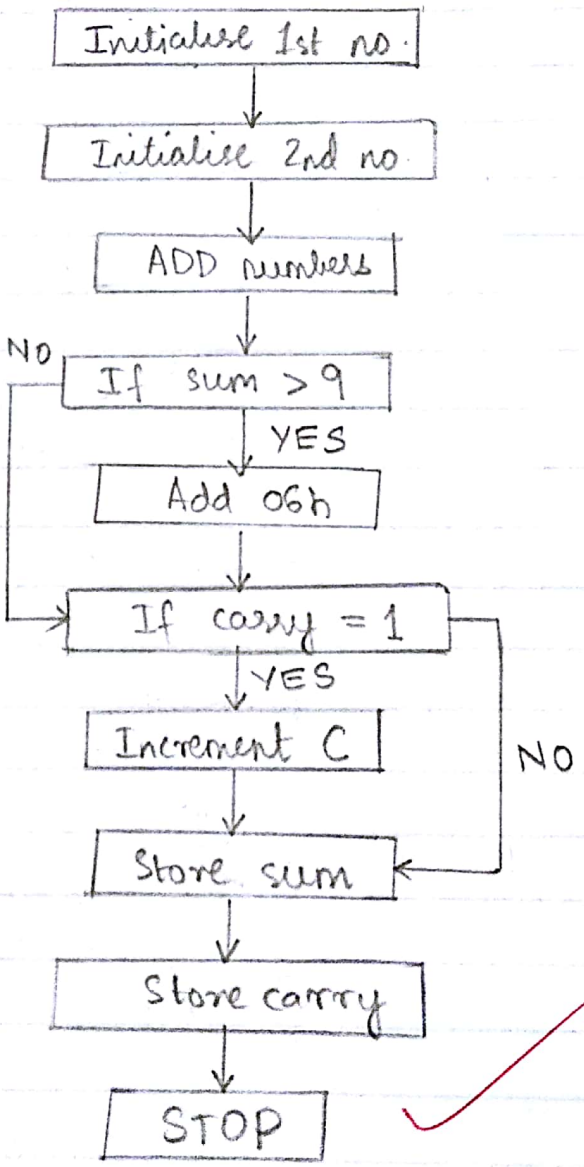
EXPERIMENT - 12

AIM :- To perform addition of two BCD numbers.

Apparatus :- 8085 Microprocessor kit, power supply.

Flow Chart :-

(a)



Signature

EXPERIMENT-12

AIM:- To perform addition of two BCD numbers

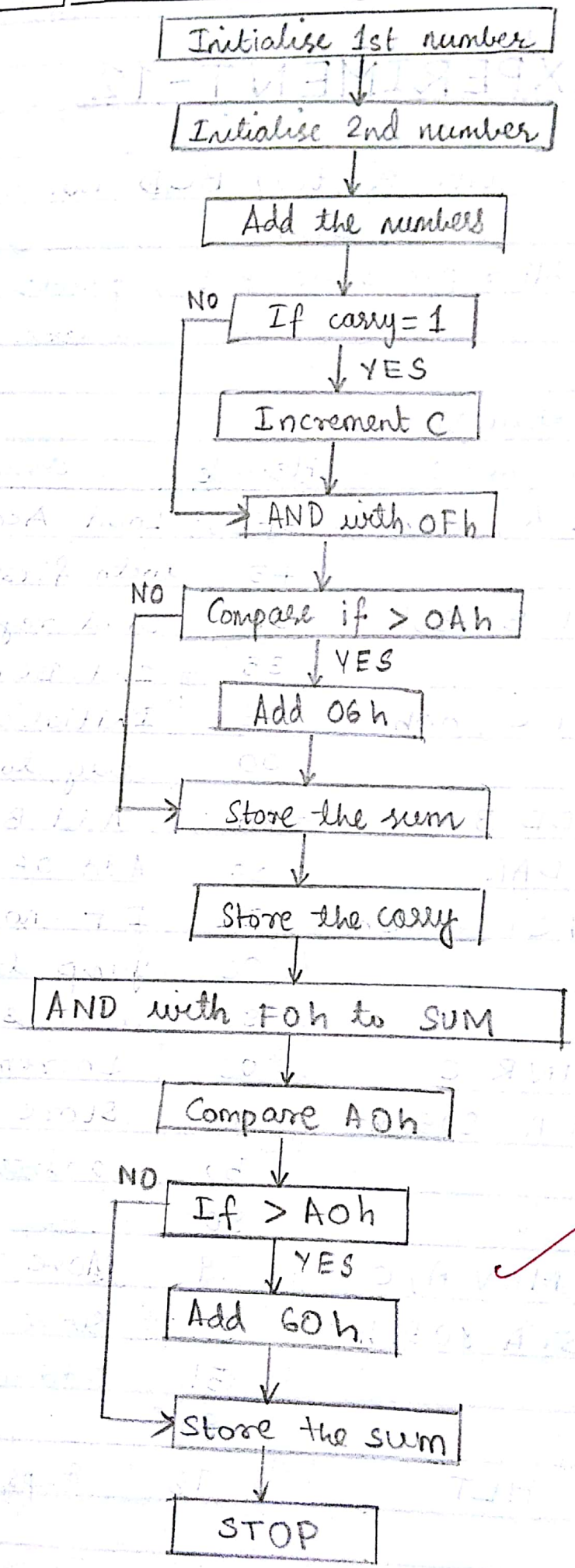
Apparatus :- 8085 Microprocessor kit, power supply.

PROGRAM :-

a) Using DAA Instruction :

Address	Mnemonics	Hexcode	Comments
8000	MVI A, 45h	3E	Load Accumulator
8001		45	with first BCD no.
8002	MVI B, 35h	06	Load register B with
8003		35	2nd BCD no.
8004	MVI C, 00h	0E	Initialise the value of
8005		00	carry to zero
8006	ADD B	80	Add B to A
8007	DAA	27	Add 06 if sum > 9
8008	JNC 800Ch	D2	If no carry then
8009		0C	jump to 800C h
800A		80	address
800B	INR C	0C	Increment carry
800C	ST A 8050h	32	Store sum at 8050h
800D		50	address
800E		80	
800F	MOV A, C	79	Move carry to A
8010	STA 8051h	32	Store carry at 8051h
8011		51	address
8012		80	
8013	HLT	76	Stops execution

(b)



Signature

b) Without using DAA instruction :

Address	Mnemonics	Hexcode	Comments
800D	MVI A, 45h	3E	Load accumulator A
8001		45	with 45h
8002	MVI C, 00h	0E	Load carry with
8003		00	00h
8004	ADI 35h	CE	Add 35h to the
8005		35	value in accumulator
8006	JNC 800Ah	D2	If no carry then
8007		0A	jump to 800Ah
8008		80	address
8009	INR C	0C	Increment carry
800A	MOV B, A	47	Move A to B
800B	ANI 0Fh	E6	AND with 0Fh
800C		0F	
800D	CPI 0Ah	FE	Compare with 0Ah
800E		0A	
800F	JC 8016h	DA	If carry then jump
8010		16	to 8016h address
8011		80	
8012	MOV A, B	78	Move B to A
8013	ADI 06h	C6	ADD 06 to sum
8014		06	
8015	MOV B, A	47	Move A to B
8016	MOV A, B	78	Move B to A
8017	STA 8055h	32	Store the sum at
8018		55	8055h address
8019		80	
801A	MOV A, C	79	Move C to A
801B	STA 8056h	32	Store carry at
801C		56	8056h address

Result :- Addition of two BCD numbers has been performed successfully.

Signatu

Serial No. 27 Date

801D		80	
801E	MOV A, B	78	Move B to A
801F	ANI F0h	E6	AND with F0h
8020		F0	
8021	CPI A0h	FE	Compare with A0h
8022		A0	
8023	JC 802Dh	DA	If carry then move
8024		2A	to address 802Ah
8025		80	
8026	MOV A, B	78	Move B to A
8027	ADI 60h	C6	ADD 60h to A
8028		60	
8029	MOV B, A	47	Move A to B
802A	MOV A, B	78	Move B to A
802B	STA 8058h	32	Store at 8058h
802C		58	address
802D		80	
802E	HLT	76	Stops the execution of program.

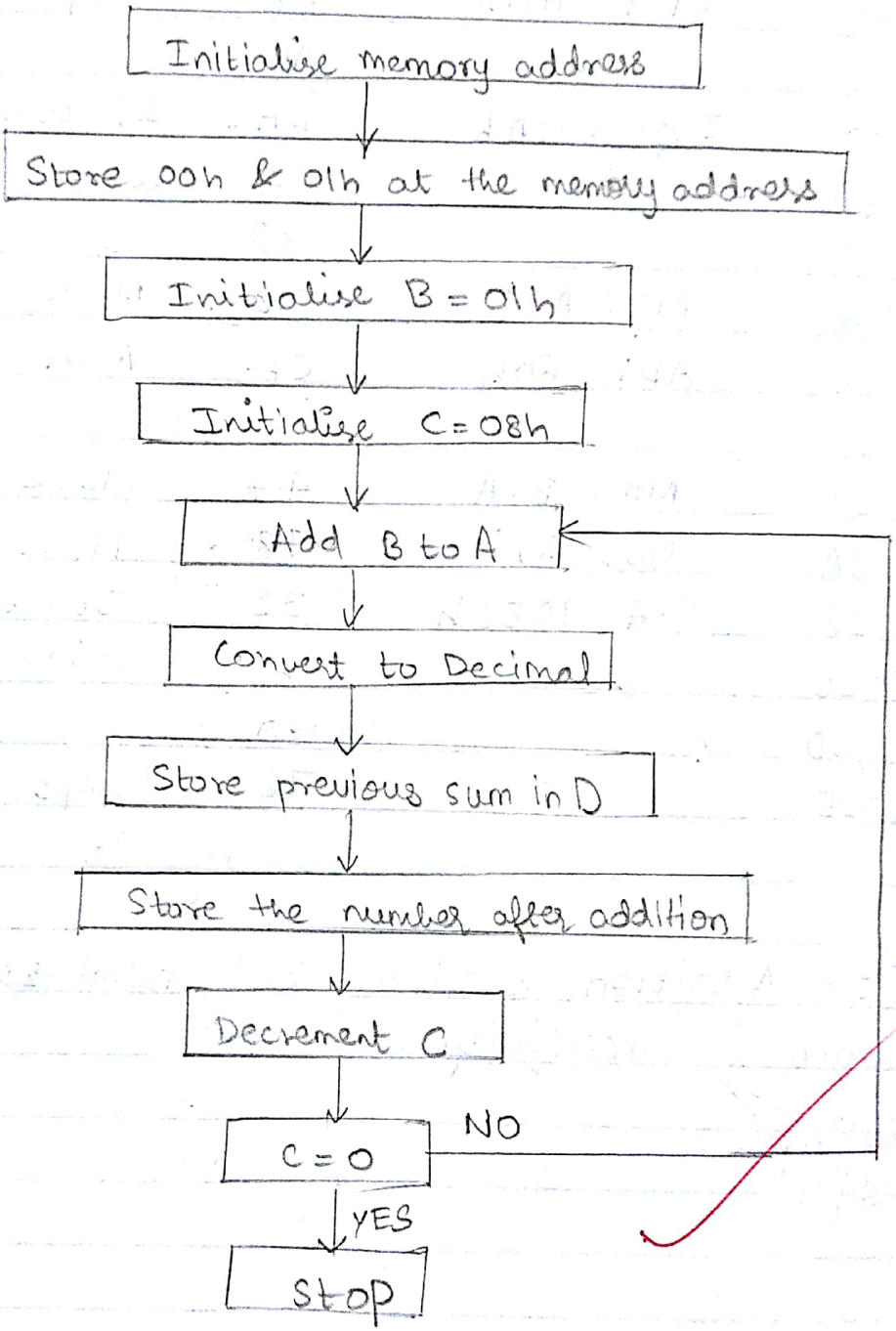
RESULT :- Addition of two BCD numbers has been performed sucessfully.

12/11/18

EXPERIMENT-13

AIM:- To implement Fibonacci series.

Apparatus:- 8085 Microprocessor kit, power supply.



Signature

EXPERIMENT-13

AIM:- To implement Fibonacci series.

Apparatus:- 8085 microprocessor kit, power supply.

PROGRAM :-

Address	Mnemonics	Hexcode	Comments
8000	LXI H, 2000h	21	Initialise memory at
8001		00	2000h address
8002		20	
8003	MVI A, 00h	3E	Initialise accumulator with
8004		00	00h value
8005	MOV M, A	77	Move A to M
8006	MVI A, 01h	3E	Initialise accumulator with
8007		01	01h value
8008	INX H	23	Move A to M
8009	MOV M, A	77	Increment the memory
800A	INX H	23	Increment the memory
800B	MOV M, A	77	move A to M
800C	MVI B, 01h	06	Initialise B = 01 h
800D	MVI C, 08h	01	Initialise the register
800E		0E	C with 08 h as a
800F		08	counter
8010	MOV D, A	57	Move A to D
8011	ADD B	80	Add B to A
8012	DAA	27	Convert to Decimal
8013	MOV B, D	42	Move D to B
8014	INX H	23	Increment memory address
8015	MOV M, A	77	Move A to M

Result :- Fibonacci Series has been implemented successfully.

Sign

Serial No. 29 Notes Date

	8016	DCR C	0D	Decrement counter C
	8017	JNC 8010h	C2	If not zero then jump
	8018		10	to 8010h address
	8019		80	
	801A	HLT	76	Stops the execution of
				program.

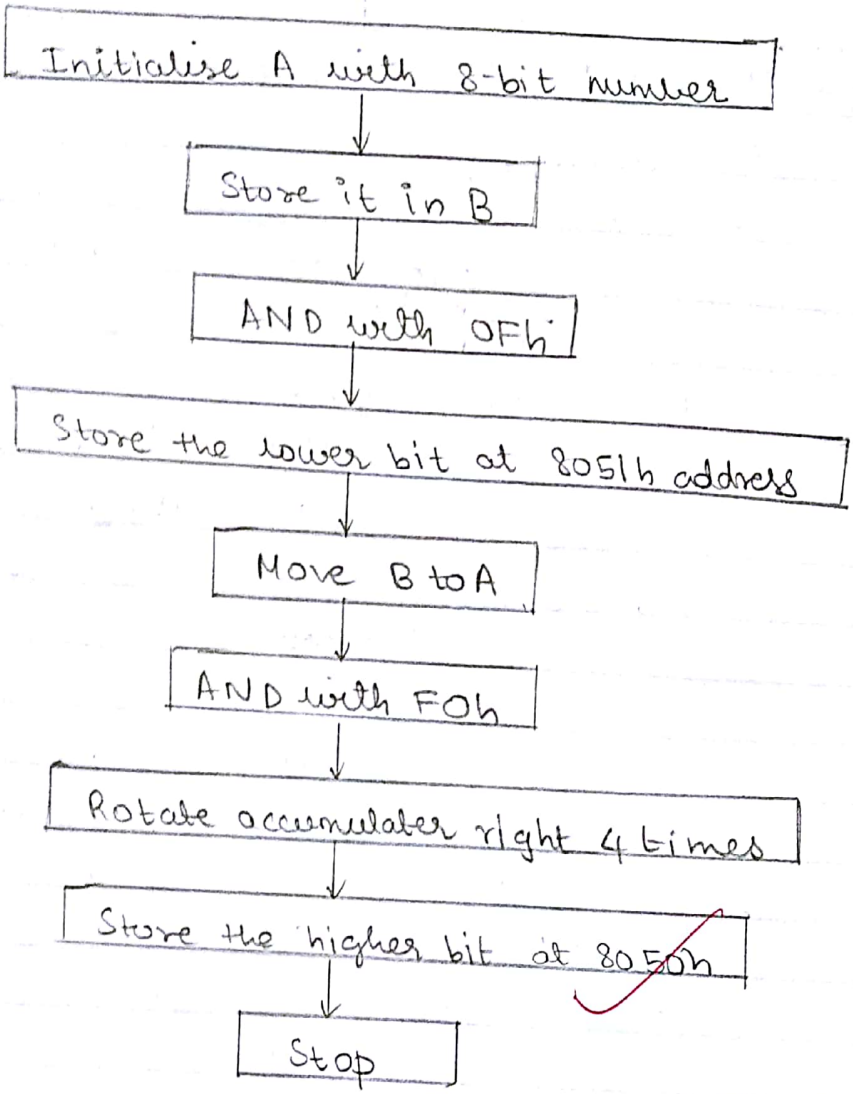
Result :- Fibonacci Series has been implemented successfully.

19/11/18

EXPERIMENT - 14

AIM:- To unpack at 8 bit BCD number.

Apparatus :- 8085 Microprocessor kit, power supply.



Result :- 8-bit BCD number has been unpacked succesfully

Signature

EXPERIMENT-14

AIM:- To unpack at 8 bit BCD number.

Apparatus:- 8085 Microprocessor kit, power supply.

PROGRAM:-

Address	Mnemonics	Hexcode	Comments
8000	MVI A, 28h	3E	Initialise accumulator with
8001		23	the number to be unpacked
8002	MOV B, A	47	Move A to B
8003	ANI 0Fh	E6	AND with 0Fh
8004		0F	
8005	STA 8051h	32	Store to 8051h address
8006		51	
8007		80	
8008	MOV A, B	78	Move B to A
8009	ANI F0h	E6	AND with F0h
800A		F0	
800B	MVI C, 04h	0E	Initialise C = 04h
800C		04	
800D	RRC	0F	Rotate right without carry
800E	DCR C	0D	Decrement C
800F	JNZ 800Dh	C2	If not zero then jump to
8010		0D	step rotate or 800Dh address
8011		80	
8012	STA 8050h	32	Store at 8050h address
8013		50	
8014		80	
8015	HLT	76	Stop execution of program.

19/11/18

Result:- 8-bit BCD number has been unpacked successfully.

PARTICULARS OF THE EXPERIMENTS PERFORMED

S. No.	Name of The Experiment	Date	Page	Remarks
1.	Introduction to 8085 Microprocessor.	13/8/18	02-04	
2.	To perform addition of two 8 bit numbers using assembly language code using microprocessor 8085 kit.	13/8/18	05	
3.	Addition of two 16 bit numbers using assembly language using microprocessor 8085	20/8/18	06-07	
4.	Multiplication of two 8 bit numbers using 8085 microprocessor kit	17/9/18	08-09	
5.	Subtraction of two 8 bit numbers with or without borrow using 8085 microprocessor kit	17/9/18	10-11	
6.	Write a program to transfer 10 bytes of data from 2000 to 8050 in (i) SAME ORDER and (ii) REVERSE ORDER	24/9/18 01-10-18	12-13	
7.	Write a program to add 10 bytes of data starting from 2000.	01/10/18 03-10-18	14-15	

PARTICULARS OF THE EXPERIMENTS PERFORMED

S. No.	Name of The Experiment	Date	Page	Remarks
8.	To subtract two 16-bit numbers using 2's complement method.	01/10/18	16-17	
9.	To seperate and store even and odd data from contents of consecutive addresses	08/10/18	18-19	
10.	To find the maximum and the minimum number from a given data	15/10/18	20-21	
11.	To arrange the given data in ascending and descending order.	15/10/18	22-24	
12.	To perform addition of two BCD numbers.	29/10/18	25-27	
13.	To implement Fibonacci series	12/11/18	28-29	
14.	To unpack a 8-bit BCD number	12/11/18	30	