

# LAB MANUAL

MICROPROCESSOR & MICROCONTROLLER



DEPARTMENT OF ELECTRONICS  
AND TELECOMMUNICATION  
ENGINEERING

Year & Semester: 2ND Year, 4<sup>TH</sup> Semester

## **INDEX**

SL NO	NAME OF THE EXPERIMENT	PAGE NO
01	To Study About the 8085 Microprocessor with the help of NVI5585 Trainer kits.	03
02	To study about the addition of two 8-bit number.	09
03	Write a Programme to Study About Subtraction of Two 8-bit Number with the Help of NVIS 5585 Trainer Kit.	11
04	Write a programme to Study about Multiplication of two 8-bit Numbers with the Help of NVIS5585 Trainer kit.	13
05	Write a Programme to Study about Division of Two 8-bit Number with the Help of NVIS 5585 Trainer kit.	16
06	To Program for 1's complement of an 8-bit numbers using 8085 instructions.	19
07	To write a program for 2's complement of an 8-bit numbers using 8085 instructions.	21
08	To write a Program for addition of two 16-bit number without carry using 8086 instructions.	23
09	To write a Program for subtraction of two 16-bit number without borrow using 8086 instructions	25
10	To write Programs for Multiplication of two 16-bit numbers without borrow using 8086 instructions.	27
11	To write a Program for Division of Two 16-bit number using 8086 instructions.	30
12	Write a program to find Smallest Number in data array using 8086 microprocessor.	32
13	Write a program to find Largest Number in data array using 8086 microprocessor.	35

## EXPERIMENT - 1

**AIM OF THE EXPERIMENT:** - To Study About the 8085 Microprocessor with the help of NVI5585 Trainer kits.

**APPARATUS REQUIRED:** -

- NVIS 5585 Trainer Kit.
- +5-VoltSupply Voltage.

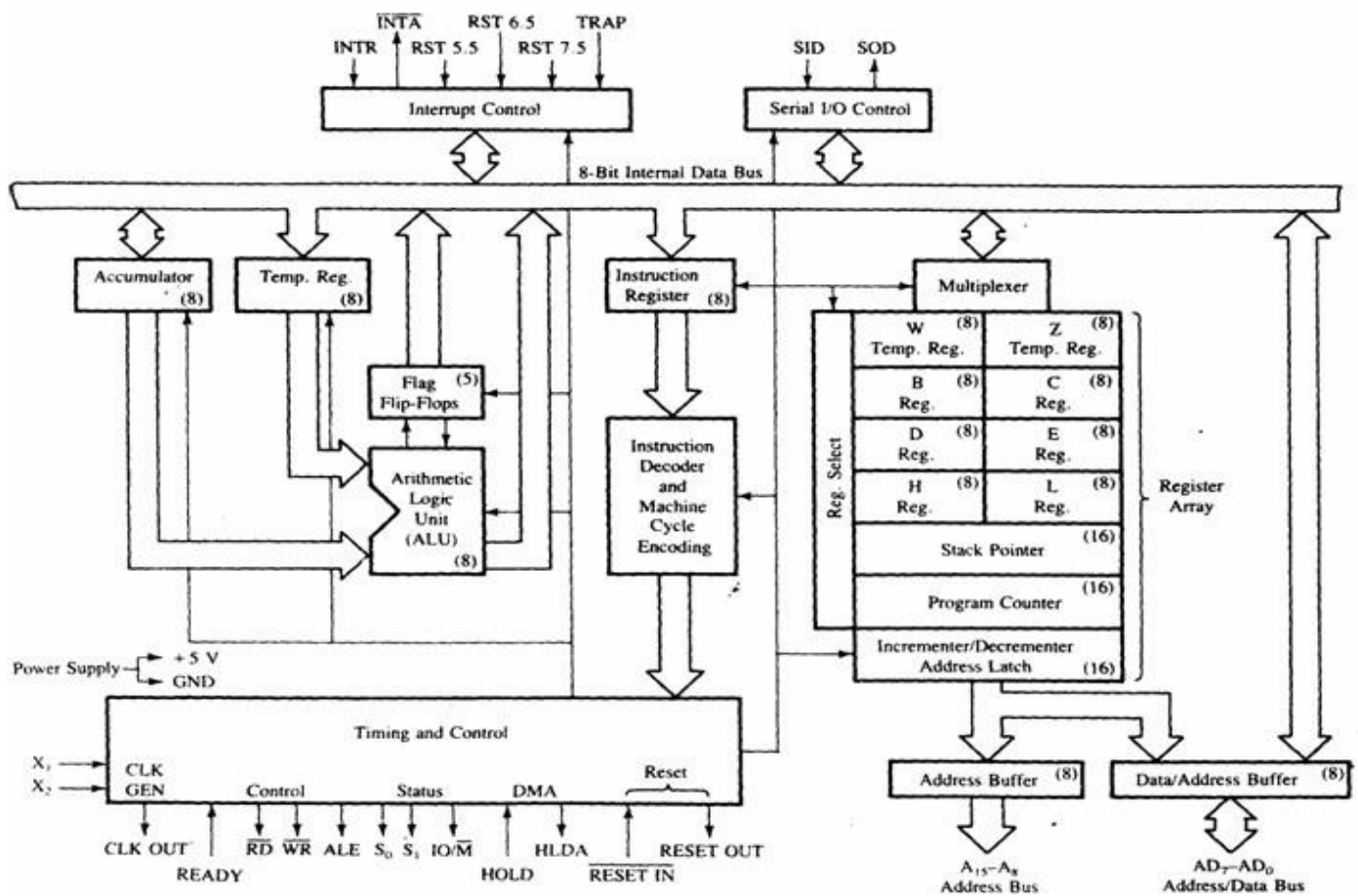
**THEORY:** -

- CPU:- 8 bit up the 8085.
- RAM:- 8k bytes user RAM and expandable up to 64 kb.
- TIMER :- 16-bit programmable timer using 8253,
- KEYBOARD :- Consist of 10 keys for command 16 keys for hexadecimal data entry.
  - 1 key for RESET.
  - 1 key for RST 7.5.
- DISPLAY :- 6-digit seven segment displays and digit for address field and 2 for Data field.
- POWER SUPPLY :- +5v connected to Vcc.
- OPREATING TEMPERATURE :- 0-to-50-degree C.

**HARDWARE DESCRIPTION:** -

- i. CPU: The system has got 8085 up as CPU. The clock frequency for the system is 3.04 MHz Generated from a crystal oscillator of 6.14 MHz, 8085 of Faiz 16 address lines or 8-bit Tata lines.
- ii. MEMORY: NU5585 provide 8 KB of RAM and 8 KB powerful monitor EPROM. The total on board memory can be expandable off to 64 KB.
- iii. I/O DEVICE: The device Ayo chips used in NU 5585 R8 to 8279, 8255, 8253.

- iv. it is a8279 (keyboard and display controller) 8279 is general purpose programmable keyboard and display. I/O interface device design for use with the 8085 Microprocessor.
- v. 8155 (programmable IO port and Timur interface): The 8155 is a programmable IO part and timer interface design to use with 8085. Up the 8155 includes 256 bytes RW memory, 3 IO part and a timer. 3 basically acts as a general-purpose IO device to interface with peripheral equipment to the system bus.
- vi. 8053 (programmable internal timer): This is used for the generation of accurate time delay under software control.
- vii. DISPLAY: - NUI5585 provide. 6 digits of 7 segment display for digit or for displaying the address where s the rest of 2 digits are meant for displaying the content of memory location or register. All the 6 digits of the display are in Hexadecimal number.



(INTERNAL ARCHITECTURE OF 8085 MP)

# Keyboard Description

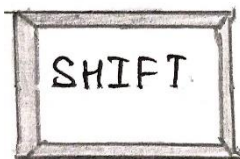
RESET	VEN INT	SHIFT	C	D	E	F
EX-REQ	INS DATA	DEL DATA	8 H	9L	A	B
GO	B.M	REL EX-MEM	4 PCH	5 PCL	6 SPH	7 SPL
STRING PRE	MEMO NEXT	FILL	0	1	2 SER	3

## KEYWORD DESCRIPTION: -

**RESET:** -Reset the system, this key indicates the initializes of the " NV5585 trainer kit and displays NV 5585 on the display the left indicates that the system is expecting a valid command.



Hardware interrupts via keyboard RST 7.5



Provide a second level command to all keys.



To execute the program on pressing this key the PC connect are display in the address view with data field A to F in the address can be changed if it needed.



To execute the program in single step mode

EX  
REG

Examine register it allows user to examine and modify the contents of different Resistor.

REL  
EXMEM

Examine memory, it allows user to examine any memory location and modify and RAM location.

PRE

Previous is used on intermediate terminals in case of examine memory. It decrements the content of PC.

NEXT

Increment (next) is used as a intermediate term for in case of n examine memory and examine resistor. It increments the content of PC.

.

Terminator is used to terminate the command and white the data in data field.

DEL

Delete the part of program or data.

INS

Insert the part of Program or Data

B.M

Allow user to move a block of memory to RAM location

FILL

Allow user to fill RAM with constant

REL

Reallocates a Program Same memory are and to be transferred to Other memory area.

INS  
DATA

Insert one or more data bytes in the user program/ data area

DEL  
DATA

Delete one or more data bytes from the user Program / data area.

STRING

Not used

MEMO

Memory compares two blocks of memory for equality.

0.E

Hexadecimal key's

**CONCLUSION:-**

From the Above Experiment We Studied and Verified the Working of 8085 Microprocessor with the help of NVIS 5585 Trainer kit.



## EXPERIMENT - 2

### AIM OF THE EXPERIMENT: -

To study about the addition of two 8-bit number.

### APPARATUS REQUIRED: -

- NVIS 5585 trainer kit.

### PROGRAM: -

Let us take a number you in the memory location 2501 and another number 56 will be in the memory location 2502 and the result will be stored in the memory location 2503.

ADDRESS	MACHINE CODE	OPCODE	OPRAND	COMMENT
2000	21,01,25	LXI	H, 2501	Get the First No in Memory Location 2501.
2003	7E	MOV	A, M	Name the Memory Location to the Accumulator
2004	23	INX	H	Increase the memory
2005	86	ADD	M	Add the Memory
2006	32,03,25	STA	2503	Store the Result in Memory Location 2503
2009	76	HLT	-	Stop

### PROCEDURE: -

- RESET
- RELEX memory
- Starting address 2000
- Next
- Fill
- RELEX memory
- Data address

- viii. NEXT
- ix. First data
- x. Next
- xi. Second data
- xii. Fill
- xiii. GO
- xiv. Starting address
- xv. Fill
- xvi. RESET
- xvii. RELEX memory
- xviii. Restore location or answer address
- xix. Next

**DATA: -**

2501-49H (Hexa)

2502-56 H

**RESULT:-**

2503- 9FH

**CONCLUSION: -**

From the above experiment we studied the addition of two. 8-bit numbers by the help of NUIS 5585 Trainer kit.

## EXPERIMENT - 3

### AIM OF THE EXPERIMENT: -

Write a programme to Study about Subtraction of two 8-bit number with the help of NVIS 5585 trainer kit.

### APPARATUS REQUIRED: -

- NVIS 5585 Trainer kit

### PROGRAMME: -

Let us take a number 39 in the memory location 2501 and the other number 29 in the memory location 2502. Now result will be stored in the memory location 2503.

ADDRESS	MACHINE CODE	MNEMONICS CODE	OPERAND	COMMENT
2000	21,01,25	LXI	H, 2501	Get the First No in Memory Location 2501
2003	7E	MOV	A.M	Move Memory Location to Accumulator
2004	23	INX	H	Increment The Memory
2005	96	SUB	M	Subtract Memory
2006	23	INX	H	Increment The Memory
2007	77	MOV	A.M	Move Memory Location to Accumulator
2008	76	HLT	-	STOP

### Procedure:-

- RESET
- RELEX memory
- Starting address 2000
- Next
- Fill
- RELEX memory
- Data address

- viii. NEXT
- ix. First data
- x. Next
- xi. Second data
- xii. Fill
- xiii. GO
- xiv. Starting address
- xv. Fill
- xvi. RESET
- xvii. RELEX memory
- xviii. Restore location or answer address
- xix. Next

**DATA:-**

2501- 39

2502- 29

**RESULT:-**

2503-10

**CONCLUSION: -**

From the above Experiment we studied the Subtraction of Two 8 bitNumbers by the Help of NUIS 5585 Trainer kit.

## EXPERIMENT - 4

### **AIM OF THE EXPERIMENT: -**

Write a programme to Study about Multiplication of two 8-bit number with the help of NVIS 5585 Trainer kit.

### **APPARATUS REQUIRED: -**

- NVIS 5585 Trainer kit

### **THEORY: -**

The 8085 has no multiplication operation. to get the result of multiplication, we should use the repetitive addition method. After multiplying two 8-bit numbers it may generate 1-byte or 2-byte numbers, so we are using two registers to hold the result. We are serving the data at location memory 8000H and 8001H. The result is to be stored in the memory location 8050H and 8051H.

### **PROCEDURE: -**

#### STEP 1- STORAGE OF PROGRAM MACHINE CODES

- Machine codes of the program to be executed, should be stored in their memory available on 8085 trainer kit as
- RESET→REL EXMEM→starting address of program (2000) →NEXT→Now enter all the machine codes, one after the other followed by the key 'NEXT'.

#### STEP 2-DECLARATION OF INPUTS

- Input values to the program should be declared as RESET→REL EXMEM→input (data) address→NEXT→Now enter all the inputs values followed by the 'NEXT'.

#### STEP 3-EXECUTION OF PROGRAM

- The program can be executed as RESET→GO→Starting address of the program (2000) →FILL

#### STEP 4-VERIFICATION OF OUTPUT

- Output of the executed program can be verified as RESET→REL EXMEM→output (data) address→NEXT

**PROGRAM:-**

ADDRESS	MACHINE CODES	LABLES	MNEMONICS CODES	OPERAND	COMMENT
F000	21,00,80		LXI	H,8000H	Address of 1st number in H-L pai
F003	46		MOV	B, M	1st number in register B
F004	23		INX	H	Content of H-L pair increases from 8000H to 8001
F005	AF		XRA	A	Initialize 00 data in accumulator/clear accumulator
F006	4F		MOV	C, A	Copy data 00 in register C
F007	86	LOOP	ADD	M	Add accumulator data with memory data
F008	D2,0C,F0		JNC AHEAD		Is carry? No, go to AHEAD
F00B	0C		INR	C	Yes, increment C
F00C	05	AHEAD	DCR	B	Decrement B
F00D	C2,07,F0		JNZ LOOP		Is count (register B) =0? No, jump to loop
F010	21,50,80		LXI	H,8050H	Load destination address
F013	71		MOV	M, C	Store register C content into memory
F014	23		INX	H	Increase H-L pair
F015	77		MOV	M, A	Store accumulator content to memory
F016	76		HLT		Terminate the program

**INPUT: -**

8000→DC

8001→AC

**OUTPUT:-**

8050→93

8051→D0

**CONCLUSION:-**

From the above Experiment we studied the Multiplication of Two 8 bit Numbers by the Help of NUIS 5585 Trainer kit.

## EXPERIMENT - 5

### AIM OF THE EXPERIMENT:-

Write a programme to Study about Division of two 8-bit number with the help of NVIS 5585 Trainer kit.

### APPARATUS REQUIRED: -

- NVIS 5585 Trainer kit

### THEORY:-

The 8085 has no division operation to get the result of division, we should use the repetitive subtraction method. By using this program, we will get the quotient and the remainder. 2504H will hold the quotient, and 2505H will hold the remainder. We save the data at memory location 2501H (LSBs of dividend), 2502H (MSBs of dividend) and 2503H (divisor). The result is to be stored in the memory location 2504H (quotient) and 2505H (remainder).

### PROCEDURE:-

#### STEP 1- STORAGE OF PROGRAM MACHINE CODES

- Machine codes of the program to be executed, should be stored in their memory available on 8085 trainer kit as
- RESET→REL EXMEM→starting address of program (2000) →NEXT→Now enter all the machine codes, one after the other followed by the key 'NEXT'.

#### STEP 2-DECLARATION OF INPUTS

- Input values to the program should be declared as RESET→REL EXMEM→input (data) address→NEXT→Now enter all the inputs values followed by the 'NEXT'.

#### STEP 3-EXECUTION OF PROGRAM

- The program can be executed as RESET→GO→Starting address of the program (2000) →FILL



#### STEP 4-VERIFICATION OF OUTPUT

Output of the executed program can be verified as RESET→REL EXMEM→output (data) address→NEXT

#### **PROGRAM:-**

MEMORY ADDRESS	MACHINE CODES	LABLES	MNEMONICS	OPERANDS	COMMENT
2400	2A,01,25		LHLD	2501H	Get dividend in H-L pair
2403	3A,03,25		LDA	2503H	Get divisor from 2503 H
2406	47		MOV	B,A	Divisor in register B
2407	0E,08		MVI	C,08	Count=08 in register C
2409	29	LOOP	DAD	H	Shift dividend and quotient left by one bit.
240A	7C		MOV	A,H	Most significant bits of dividend in accumulator.
240B	90		SUB B	B	Subtract divisor from most significant bits of dividend.
240C	DA,11,24		JC	AHEAD	Is most significant part of dividend > divisor? No, go to AHEAD
240F	67		MOV	H,A	Most significant bits of dividend in register H
2410	2C		INR	L	Yes, add 1 to quotient
2411	0D	AHEAD	DCR	C	Decrement count
2412	C2,09,24		JNZ	LOOP	Is count=0? No, jump to LOOP
2415	22,04,25		SHLD	2504H	Store quotient in 2504 H

2418	76		HLT		Stop the program
------	----	--	-----	--	------------------

**INPUT:-**

2501→98 H, LSBs of dividend

2502→48 H, MSBs of dividend

2503→1A H, Divisor

**OUTPUT:-**

2504→F2, Quotient

2505→07, Remainder

**CONCLUSION:-**

From the above Experiment we studied the Division of Two 8-bit Numbers by the Help of NUIS 5585 Trainer kit.

## EXPERIMENT - 6

### AIM OF THE EXPERIMENT:-

To Program for 1's complement of an 8-bit numbers using 8085 instructions.

### APPARATUS REQUIRED:-

- NVIS 5585Trainer kit

### THEORY: -

To obtain 1's complement of a number its '0' bits are replaced by '1' and '1' by '0'. The number is placed in the memory location 2501H. The result is stored in the memory location 2502H.

### PROCEDURE:-

#### STEP 1- STORAGE OF PROGRAM MACHINE CODES

- Machine codes of the program to be executed, should be stored in their memory available on 8085 trainer kit as
- RESET→REL EXMEM→starting address of program (2000) →NEXT→Now enter all the machine codes, one after the other followed by the key 'NEXT'.

#### STEP 2-DECLARATION OF INPUTS

- Input values to the program should be declared as RESET→REL EXMEM→input (data) address→NEXT→Now enter all the inputs values followed by the 'NEXT'.

#### STEP 3-EXECUTION OF PROGRAM

- The program can be executed as RESET→GO→Starting address of the program (2000) →FILL

#### STEP 4-VERIFICATION OF OUTPUT

- Output of the executed program can be verified as RESET→ REL EXMEM→output (data) address→NEXT

**PROGRAM:-**

MEMORY ADDRESS	MACHINE CODES	MNEMONICS	OPERANDS	COMMENTS
2000	3A,01,25	LDA	2501 H	Get data in accumulator.
2003	2F	CMA		Take its 1's complement.
2007	32,02,25	STA	2502 H	Store result in 2502 H.
2008	76	HLT		Stop.

**DATA:**

2501→96 H .

**RESULT:**

2502→69 H

**CONCLUSION:-**

From the above Experiment we studied the 1's complement of 8-bit Numbers by the Help of NUIS 5585 Trainer kit.

## EXPERIMENT - 7

### AIM OF THE EXPERIMENT:-

To write a program for 2's complement of an 8-bit numbers using 8085 instructions.

### APPARATUS REQUIRED:-

- NVIS 8085 Trainer kit

### THEORY:-

- 2's complement of a number is obtained by adding 1 to the 1's complement of the number.
- The data is placed in the memory location 2501H.
- The result is to be stored in the memory location 2502H.

### PROCEDURE:-

#### STEP 1- STORAGE OF PROGRAM MACHINE CODES:

- Machine codes of the program to be executed, should be stored in their memory available on 8085 trainer kit as
- RESET→REL EXMEM→starting address of program (2000) →NEXT→Now enter all the machine codes, one after the other followed by the key 'NEXT'.

#### STEP 2-DECLARATION OF INPUTS

- Input values to the program should be declared as RESET→REL EXMEM→input (data) address→NEXT→Now enter all the inputs values followed by the 'NEXT'.

#### STEP 3 - EXECUTION OF PROGRAM

- The program can be executed as RESET→GO→Starting address of the program (2000) →FILL

#### STEP 4-VERIFICATION OF OUTPUT

- Output of the executed program can be verified as RESET→REL EXMEM→output (data) address→NEXT

**PROGRAM:-**

MEMORY ADDRESS	MACHINE CODES	MNEMONICS	OPERANDS	COMMENT
2000	3A,01,25	LDA	2501, H	Get data in accumulator.
2003	2F	CMA		Take its 1's complement
2004	3C	INR	A	Take its 2's complement.
2005	32,02,26	STA	2502, H	Store result in 2502 H
2008	76	HLT		Stop

DATA:-

2501→96 H

RESULT: -

2502→ 6A H

**CONCLUSION:-**

From the above Experiment we studied the 2's complement of 8-bit Numbers by the Help of NVIS 5585 Trainer kit.

## EXPERIMENT - 8

### AIM OF THE EXPERIMENT: -

To write a program for addition of two 16-bit number without carry using 8086 instruction.

### APPARATUS REQUIRED:-

- NVIS 8086 Trainer kit

### THEORY:-

- We can simply take the numbers from memory to AX and BX register respectively, then add them using ADD instruction.
- When the carry is present store carry into the memory, otherwise result moved to the memory location.

### PROCEDURE:-

Assembly language programs for microprocessor 8086 can be executed on its trainer kit as per the following steps

#### STEP 1- STORAGE OF PROGRAM MACHINECODES

- RESET→EB/AX (Examine byte)→ Starting address of program (2000)→NEXT→Now enter all the machine codes, one after the other followed by the key 'NEXT'.

#### STEP 2-DECLARATION OF INPUTS

- Input values to the program should be declared as →TYY

#### STEP 3-EXECUTION OF PROGRAM

- The program can be executed as GO→ Starting address of the program (2000) → TYY

#### STEP 4-RESULT VERIFICATION OF REGISTERS

- Output of the executed program can be verified as ER/BX (Register B) → TYY→ER/BX→EX/AX (To check the content of register A

**PROGRAM:-**

EFFECTIVE ADDRESS	MNEMONIC CODES	LABLE	MNEMONICS	OPERAND	COMMENTS
2000	8B,06,0D,10		MOV	AX, [100D]	Move the contents of 100D to AX
2004	03,06,0F,10		ADD	AX, [100F]	Add the contents of 100F with AX
2008	89,06,11,10		MOV	[1011], AX	Move the content of AX to 1011
200C	CC		INT 3		Interrupt program

**Input:**

100D- 23

100E- 34

100F- 12

1010- 22

**Output:**

1011- 35

1012- 56

**CONCLUSION:-**

From the above Experiment we studied the Addition of two 16-bit number without carry using 8086 instructions.



## EXPERIMENT - 9

### AIM OF THE EXPERIMENT:-

To write a program for subtraction of two 16-bit number without borrow using 8086 instructions.

### APPARATUS REQUIRED:-

- NVIS 8086 trainer kit

### THEORY:-

- We can simply take the numbers from memory to AX and BX register respectively, then subtract them using SUB instruction.
- When the borrow is present, the CY flag will be 1, so we can store borrow into the memory, otherwise result moved to the memory location.

### PROCEDURE:-

Assembly language programs for microprocessor 8086 can be executed on its trainer kit as per the following steps: -

#### STEP 1- STORAGE OF PROGRAM MACHINE CODES

- RESET→EB/AX (Examine byte) →starting address of program (2000) → NEXT→Now enters all the machine codes, one after the other followed by the key 'NEXT'.

#### STEP 2-DECLARATION OF INPUTS

- Input values to the program should be declared as →.TYY

#### STEP 3-EXECUTION OF PROGRAM

- The program can be executed as GO→starting address of the program (2000)→.TYY

#### STEP 4-RESULT VERIFICATION OF REGISTERS

- Output of the executed program can be verified as ER/BX (Register B)→ TYY→ER/BX→EX/AX (To check the content of register A

### **PROGRAM:-**

EFFECTIVE ADDRESS	MNEMONIC CODES	LABLE	MNEMONICS	OPERAND	COMMENTS
2000	8B,06,0D,10		MOV	AX,[100D]	Move the contents of 100D to AX
2004	03,06,0F,10		SUB	AX, [100F]	Add the contents of 100F with AX
2008	89,06,11,10		MOV	[1011], AX	Move the content of AX to 1011
200C	CC		INT 3		Interrupt program

### **Input:**

100D- 34

100E- 56

100F- 12

1010- 34

### **Output:**

1011- 22

1012- 22

### **CONCLUSION:-**

From the above Experiment we studied the Subtraction of two 16-bit number without carry using 8086 instructions.

## EXPERIMENT - 10

### AIM OF THE EXPERIMENT:-

To write a program for Multiplication of two 16-bit number without borrow using 8086 instructions.

### APPARATUS REQUIRED:-

- NVIS 8086 trainer kit.

### THEORY:-

- Initialize the pointer to the memory for data and result.
- Load the multiplier value into AX register.
- Load multiplicand value in BX register.
- Multiply of these two data.
- Store the result into Memory address 1520.

### PROCEDURE:-

Assembly language programs for microprocessor 8086 can be executed on its trainer kit as per the following steps

#### STEP 1- STORAGE OF PROGRAM MACHINE CODES

- RESET→EB/AX (Examine byte) →starting address of program (2000)  
→NEXT→Now enter all the machine codes, one after the other followed by the key 'NEXT'.

#### STEP 2-DECLARATION OF INPUTS

- Input values to the program should be declared as →TYY

#### STEP 3-EXECUTION OF PROGRAM

- The program can be executed as GO→starting address of the program (2000)  
→. TYY

#### STEP 4-RESULT VERIFICATION OF REGISTERS

- Output of the executed program can be verified as ER/BX (Register B)→  
.TYY→ER/BX→EX/AX (To check the content of register A)

**PROGRAM:-**

EFFECTIVE ADDRESS	OPCODES	MNEMONICS	OPERANDS	COMMENTS
1100	BE 00 15	MOV	SI,1500	Load 1500 into SI
1103	AD	LOD	SW	Load the multiplicand value
1104	89 C3	MOV	BX, AX	Load AX value into BX
1106	AD	LOD	SW	Load the multiplier value
1107	F7 E3	MUL	BX	Multiply two data
1109	BF 0 5 15	MOV	DI, 1520	Load 1520 address into DI
110C	89 05	MOV	[DI], AX	Store AX value into DI
110E	47	INC DI		
110F	47	INC	DI	Increment the DI
1110	89 15	MOV	[DI], BX	Store BX value into DI
1112	CC	INT 3		Break point

**INPUT:**

1500→02

1501→00

1502→03

1503→00

**OUTPUT:**

1520→06

1521→00

**CONCLUSION:-**

From the above Experiment we studied the Multiplication of two 16-bit number without carry using 8086 instructions.

## EXPERIMENT - 11

### AIM OF THE EXPERIMENT:

To write a Program for Division of Two 16-bit number using 8086 instructions.

### APPARATUS REQUIRED:-

- NVIS 8086 trainer kit.

### THEORY:-

- Initialize the pointer to the memory for result.
- Load the dividend value into AX register.
- Load the divisor value into BX register.
- Divide these two data's.
- Store the result into Memory address 1520.

### PROCEDURE:-

Assembly language programs for microprocessor 8086 can be executed on its trainer kit as per the following steps

#### STEP 1- STORAGE OF PROGRAM MACHINE CODES

- RESET→EB/AX (Examine byte)→starting address of program (2000)→NEXT→Now enter all the machine codes, one after the other followed by the key 'NEXT'.

#### STEP 2-DECLARATION OF INPUTS

- Input values to the program should be declared as →.TYY

#### STEP 3-EXECUTION OF PROGRAM

- The program can be executed as GO→Starting address of the program (2000) →.TYY

#### STEP 4-RESULT VERIFICATION OF REGISTERS

- Output of the executed program can be verified as ER/BX (Register B) → .TYY→ER/BX→EX/AX (To check the content of register A)

**PROGRAM:-**

EFFECTIVE ADDRESS	OPCODES	MNEMONICS	OPERANDS	COMMENTS
1100	BA 00 00	MOV	DX, 0000	Clear DX registers
1103	B8 FD FF	MOV	AX, FFFD	Load the dividend in AX
1106	B9 0F 00	MOV	BX, 0F	Load the divisor value in BX
1109	F7 F1	DIV	BX	Divide the two data's
110B	BF 00 15	MOV	DI, 1520	Load 1520 address into DI
110E	88 05	MOV	[DI], AL	Load AL value into DI
1110	47	INC	DI	Increment DI
1111	88 25	MOV	[DI], AH	Load AH value into DI
1113	47	INC	DI	Increment DI
1114	89 15	MOV	[DI], DX	Load DX value into DI
1116	CC	INT3		Break point

**Input:**

AX (dividend) =0083

BX (divisor) =02

**Output:**

1522→41 (quotient)

1523→01 (remainder)

**CONCLUSION:-**

From the above Experiment we studied the Division of two 16-bit number without carry using 8086 instructions.

## EXPERIMENT - 12

### AIM OF THE EXPERIMENT:-

Write a program to find smallest number in data array using 8086 Microprocessors.

### APPARATUS REQUIRED:-

- NVIS 8086 Trainer kit.

### THEORY: -

- The given data array is 8341, 7258, 4639, 8453, 9630. These are hexadecimal numbers.
- As there are five 16-bit numbers in the given data array, the count is 0005. Two consecutive memory locations store a 16-bit numbers i.e. two bytes of a 16-bit number.

### PROCEDURE:-

Assembly language programs for microprocessor 8086 can be executed on its trainer kit as per the following steps

#### STEP 1- STORAGE OF PROGRAM MACHINE CODES

- RESET→EB/AX (Examine byte)→starting address of program (2000)→NEXT→Now enter all the machine codes, one after the other followed by the key 'NEXT'.

#### STEP 2-DECLARATION OF INPUTS

- Input values to the program should be declared as →.TYY

#### STEP 3-EXECUTION OF PROGRAM:

- The program can be executed as GO→Starting address of the program (2000)→.TYY

#### STEP 4-RESULT VERIFICATION OF REGISTERS

- Output of the executed program can be verified as ER/BX (Register B)→.TYY→ER/BX→EX/AX (To check the content of register A



**PROGRAM:-**

EFFECTIVE ADDRESS	MNEMONICS CODES	LABEL	MNEMONICS	OPERANDS	COMMENTS
0101	B8,FF,FF		MOV	AX,FFFF	Initial value for comparison.
0104	BE,00,02		MOV	SI,0200	Memory address in SI.
0107	8B,0C		MOV	CX,[SI]	Count in CX
0109	46	BACK	INC	SI	Increment SI
010A	46		INC	SI	Increment SI
010B	3B,04		CMP	AX,[SI]	Compare previous smallest with next number
010D	72,02		JB	GO	Jump if number in AX is smaller i.e. CF=1
010F	8B,04		MOV	AX,[SI]	Save next smaller
0111	E2,F6	GO	LOOP	BACK	Jump to back until CX becomes zero
0113	A3,51,02		MOV	[0251],AX	Store smallest number in memory
0116	CC		INT 3		Interrupt program

**DATA :-**

0201-00 H

0202-41 H

0203-83 H

0204-58 H

0205-72 H

0206-39 H

0207-46 H

0208-53 H

0209-84 H

020A-30 H

020B-96 H

**RESULT:-**

0251-39 H

0252-46 H

**CONCLUSION:-**

From the above Experiment we studied the find smallest number in data array using 8086 microprocessor.

## EXPERIMENT - 13

### AIM OF THE EXPERIMENT:-

Write a program to largest number in data array using 8086 Microprocessors.

### APPARATUS REQUIRED:-

- NVIS 8086 Trainer kit.

### THEORY:-

- The given data array is: 8341, 7258, 4639, 8453, and 9630. These are hexadecimal numbers.
- Since there are five 16-bit number in a given data array, the count is 0005. Two consecutive memory locations store a 16-bit number i.e., two bytes of a 16-bit binary number.

### PROCEDURE:-

Assembly language programs for microprocessor 8086 can be executed on its trainer kit as per the following steps

#### STEP 1- STORAGE OF PROGRAM MACHINE CODES

- RESET→EB/AX (Examine byte)→starting address of program (2000)→NEXT→Now enter all the machine codes, one after the other followed by the key 'NEXT'.

#### STEP 2-DECLARATION OF INPUTS

- Input values to the program should be declared as →.TYY

#### STEP 3-EXECUTION OF PROGRAM

- The program can be executed as GO→starting address of the program (2000)→.TYY

#### STEP 4-RESULT VERIFICATION OF REGISTERS

- Output of the executed program can be verified as ER/BX (Register B)→.TYY→ER/BX→EX/AX (To check the content of register A)

**PROGRAMME:-**

ADDRESS	MNEMONIC CODES	LABLE	MNEMONICS	OPERANDS	COMMENTS
0101	B8, 00, 00		MOV	AX, 0000	Initial value for comparison
0104	BE, 00, 02		MOV	SI, 0200	memory address in SI
0107	8B, 0C		MOV	CX, [SI]	count in CX
0109	46	BACK	INC	SI	increment SI
010A	46		INC	SI	increment SI
010B	3B, 04		CMP	AX, [SI]	compare previous largest number with next number
010D	73, 02		JAE	GO	Jump if number in AX is greater i.e. CF = 0
010F	8B, 04		MOV	AX, [SI]	save next larger number in AX
0111	E2, F6	GO	LOOP	BACK	jump to BACK until CX become zero
0113	A3, 51, 02		MOV	[0251], AX	store largest number in memory
0116	CC		INT3		interrupt program

DATA:-

0200-05H

0201-00H

0202-41H

0203-83H

0204-58H

0205-72H

RESULT:-

2052-96H

0251-30H

0206-39H

0207-46H

0208-53H

0209-84H

020A-30H

020B -96H

**CONCLUSION:-**

From the above Experiment we studied the find Largest Number in Data Array using 8086 microprocessor.