A Laboratory Manual
for

Microprocessor

(22415)

Semester – IV

(CO, CM, CW)

Maharashtra State
Board of Technical Education, Mumbai
(Autonomous) (ISO 9001 :2015) (ISO/IEC 27001:2013)

Sr.no	Practical Outcome	Date of performance	Date of submission	Marks	Remark
1	Identify various pins of the given microprocessor.	**06/01/2024**	**13/01/2024**		
2	Use Assembly Language Programming Tools and functions	**13/01/2024**	**20/01/2024**		
3	Use different addressing mode instruction in program Write an Assembly Language Program (ALP) to add two given 8 and 16 bit numbers. (b) Write an Assembly Language Program (ALP) to subtract two given 8 and 16 bit numbers.	**20/01/2024**	**27/01/2024**		
4	(a) Write an ALP to multiply two given 8 and 16 bit unsigned numbers. (b) Write an ALP to multiply two given 8 and 16 bit signed numbers.	**27/01/2024**	**03/02/2024**		
5	(a) Write an ALP to divide two unsigned numbers 02 (b) Write an ALP to divide two signed numbers	**03/02/2024**	**10/02/2024**		
6	Write an ALP to add, subtract, multiply, divide two BCD numbers.	**10/02/2024**	**17/02/2024**		
7	(a) Write an ALP to perform block transfer data using string instructions 02 (b) Write an ALP to perform block transfer data without using string instructions.	**17/02/2024**	**24/02/2024**		
8	Implement loop in assembly language program (a) Write an ALP to find sum of series of Hexadecimal Numbers. (b) Write an ALP to find sum of series of BCD numbers.	**24/02/2024**	**02/03/2024**		

9	(a) Write an ALP to find smallest number from array of n numbers. (a) Write an ALP to find largest number from array of n numbers.	**02/03/2024**	**09/03/2024**		
10	(a) Write an ALP to arrange numbers in array in ascending order. (b) Write an ALP to arrange numbers in array in descending order.	**09/03/2024**	**16/03/2024**		
11	Write an ALP for Z = (A + B) * (C + D) using Procedure	**16/03/2024**	**16/03/2024**		
12	Write an ALP for Z = (A + B) * (C + D) using MACRO.	**16/03/2024**	**16/03/2024**		

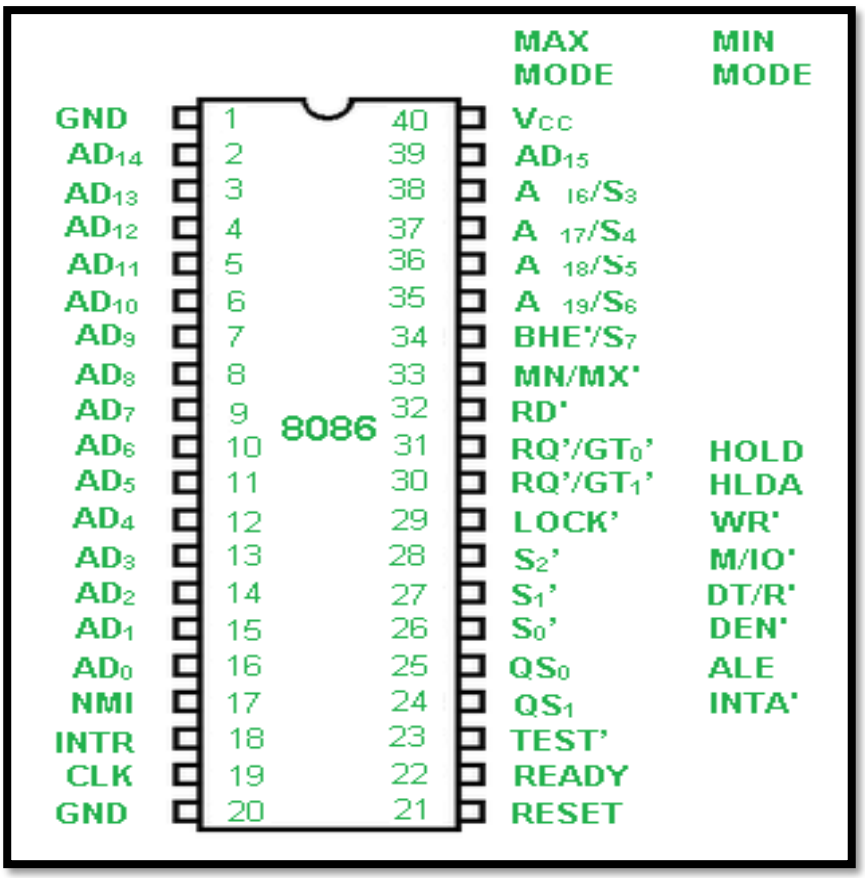
Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: MICROPROCESSOR	**Subject Code**:22415
Semester: 4th Semester	**Course:** Computer Engineering
Laboratory No: L004B	**Name of Subject Teacher**: MISS.PRAGATI MALI
Name of Student: Aditya G. Makwana	**Roll Id:** 22203A0042

Experiment No:	1
Title of Experiment	Identify the various pins of the 8086 Microprocessor

- **AIM:** Identify the various pins of the 8086 Microprocessor.

- **DIAGRAM:**



~ADITYA MAKWANA (2203A0042)

- **PIN DESCRIPTION:**

Pin number	*Pin name*	*Function*
16-2, 39	AD0 – AD15	Address/Data bus. These are low order address bus. They are multiplexed with data.
38-35	A16-A19	High order address bus. These are multiplexed with status signals.
28	M/IO’	M/IO’ signal is used to distinguish between memory and I/O operations. When it is high, it indicates I/O operation and when it is low indicates the memory operation
27	DT/R’	DT/R’ stands for Data Transmit/Receive signal. It decides the direction of data flow through the trans receiver. When it is high, data is transmitted out and vice-versa.

26	DEN’	DEN’ stands for Data Enable and is available at pin 26. It is used to enable Trans receiver 8286. The trans receiver is a device used to separate data from the address/data bus.
25	ALE	ALE stands for address enable latch and is available at pin 25. A positive pulse is generated each time the processor begins any operation. This signal indicates the availability of a valid address on the address/data lines.
29	WR’ OR LOCK	WR’ stands for write signal and is available at pin 29. It is used to write the data into the memory or the output device depending on the status of M/IO signal.
18	INTR	It is an interrupt request signal, which is sampled during the last clock cycle of each instruction to determine if the processor considered this as an interrupt or not

21	RESET	It causes the processor to immediately terminate its present activity. This signal is active high for the first 4 clock cycles to RESET the microprocessor.
22	READY	It is an acknowledgement signal from I/O devices that data is transferred. It is an active high signal. When it is high, it indicates that the device is ready to transfer data. When it is low, it indicates wait state.
23	TEST’	When this signal is high, then the processor has to wait for IDLE state, else the execution continues
24	INTA’	When the microprocessor receives this signal, it acknowledges the interrupt.
30	HLDA	This signal acknowledges the HOLD signal.
31	HOLD	HOLD signal indicates to the processor that external devices are requesting to access the address/data buses.

32	RD’	RD’ is available at pin 32 and is used to read signal for Read operation
33	MN/MX’	MN/MX’ stands for Minimum/Maximum and is available at pin 33. It indicates what mode the processor is to operate in; when it is high, it works in the minimum mode and vice-versa.
34	BHE’	BHE stands for Bus High Enable. It is available at pin 34 and used to indicate the transfer of data using data bus AD8-AD15. This signal is low during the first clock cycle, thereafter it is active.
17	NMI	NMI stands for non-maskable interrupt and is available at pin 17. It is an edge triggered input, which causes an interrupt request to the microprocessor.
1,20	GND	It provides ground for the microprocessor
40	VCC	It has 5V DC supply

- **CONCLUSION**:

8086 was the first 16-bit microprocessor available in 40-pin DIP chip. By this practical we can understand the pin number with their respective functions.

~***ADITYA MAKWANA (2203A0042)***

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: MICROPROCESSORS	**Subject Code:22415**
Semester: 4th Semester	**Course: Computer Engineering**
Laboratory No: L004C	**Name of Subject Teacher: PRAGATI MALI**
Name of Student: ADITYA G. MAKWANA	**Roll Id: 22203A0042**

Experiment No:	**2**
AIM of Experiment	**Use Assembly Language Programming Tools and Functions**

- **ASSEMBLY LANGUAGE TOOLS**

Tools	Function	Software Used
Assembler	An assembler is a program that converts source code program written in assembly language into object files in machine language.	TASM
Linker	A linker is a program that combines object file created by the assembler with other object files and link libraries and produces a single executable program.	TLINK for TASM and LINK.EXE
Debugger	A debugger is a program that allows you to trace the execution of a program and examine the content of registers and memory.	Turbo Debugger for TASM
Editor	An editor is used to create assembly language source files.	Notepad

- **CONCLUSION**

Learned about essential programming tools, including text editors for code creation assemblers for converting code to machine code , linker for combining multiple code modules, and debuggers for identifying and resolving programming errors.

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject:- Microprocessors	Subject Code: 22415
Semester: 4th Semester	Course: Computer Engineering
Laboratory No: L004B	Name of Subject Teacher: Pragati Mali
Name of Student: Aditya Makwana	Roll Id: 22203A0042

Experiment No:	3
Title of Experiment	Write an Assembly Language program to perform Addition and subtraction of two 8-bit and 16-bit numbers

- <u>Software Used:-</u>

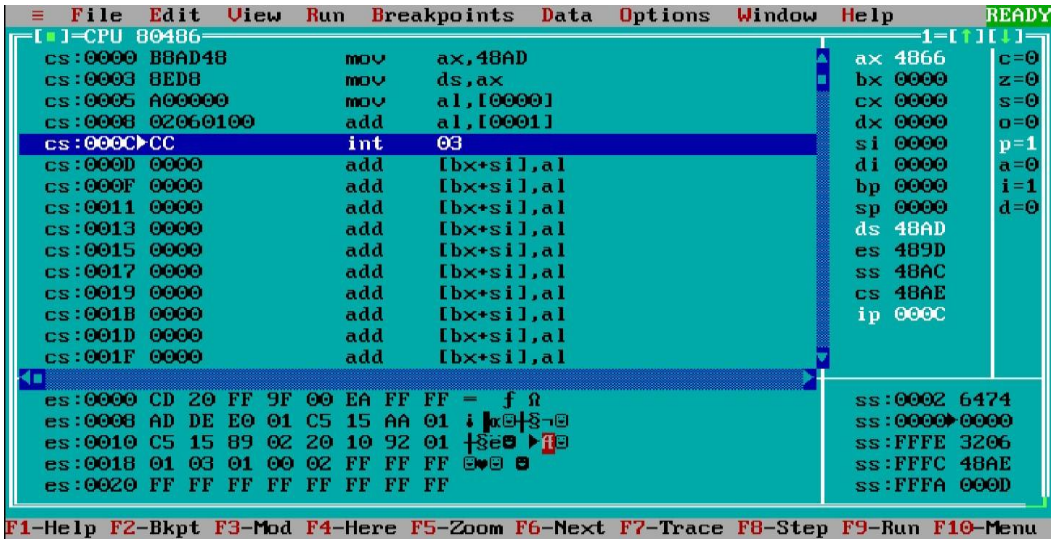
TASM 1.4
Text Editor (Notepad)

1. Addition of 8-bit numbers:-

- <u>Program Code</u>

```
DATA SEGMENT

      NUM1 DB 55H

      NUM2 DB 11H

DATA ENDS

CODE SEGMENT

      ASSUME CS: CODE, DS: DATA

START:

      MOV AX, DATA

      MOV DS, AX

      MOV AL, NUM1

      ADD AL, NUM2

      INT 3

CODE ENDS

END START
```

- <u>Output</u>

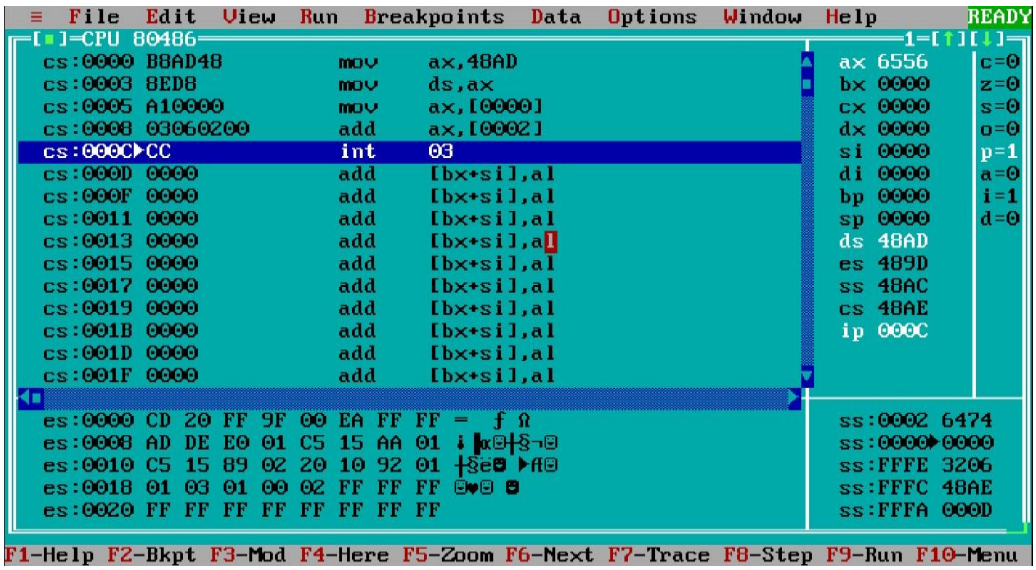


2. Addition of 16-bit numbers:-

- <u>Program Code</u>

```
DATA SEGMENT

      NUM1 DB 5555H

      NUM2 DB 1001H

DATA ENDS

CODE SEGMENT

      ASSUME CS: CODE, DS: DATA

START:

      MOV AX, DATA

      MOV DS, AX

      MOV AX, NUM1

      ADD AX, NUM2

      INT 3

CODE ENDS

END START
```

- <u>Output</u>

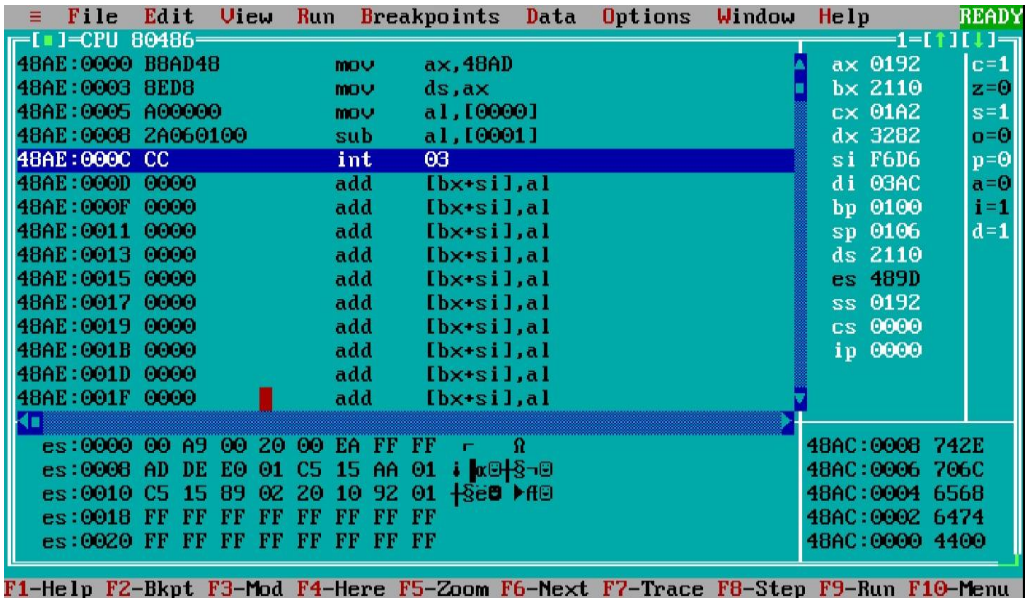


3. Subtraction of 8-bit numbers:-

- <u>Program Code</u>

```
DATA SEGMENT

      NUM1 DB 55H

      NUM2 DB 11H

DATA ENDS

CODE SEGMENT

      ASSUME CS: CODE, DS: DATA

START:

      MOV AX, DATA

      MOV DS, AX

      MOV AL, NUM1

      SUB AL, NUM2

      INT 3
CODE ENDS
END START
```

- <u>Output</u>

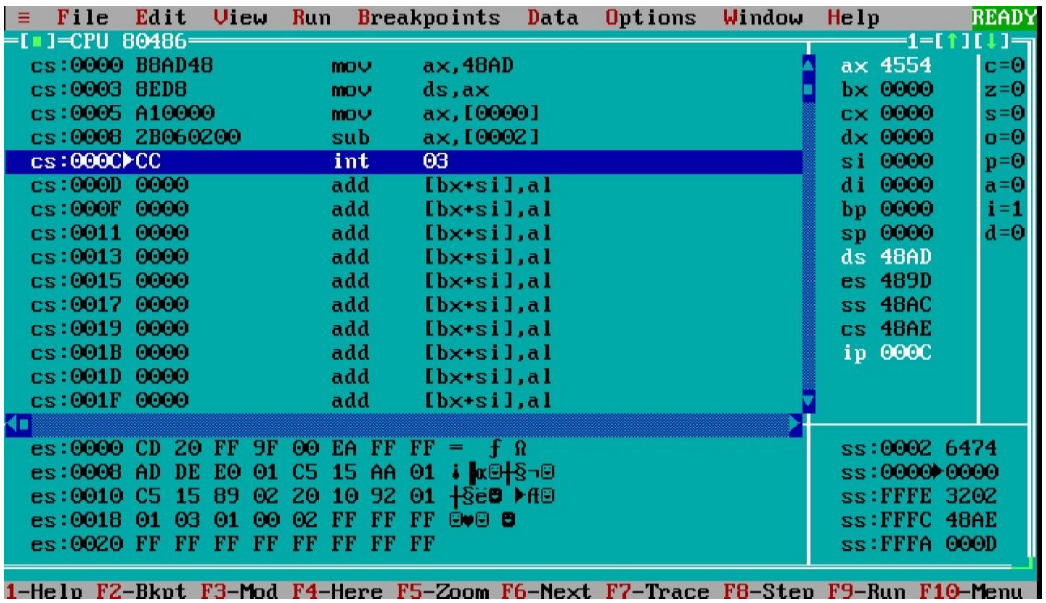


4. Subtraction of 16-bit numbers:-

- <u>Program Code</u>

```
DATA SEGMENT

      NUM1 DW 5555H

      NUM2 DW 1001H

DATA ENDS
CODE SEGMENT

      ASSUME CS: CODE, DS: DATA

START:

      MOV AX, DATA

      MOV DS, AX

      MOV AX, NUM1

      SUB AX, NUM2

      INT 3

CODE ENDS
END START
```

- <u>Output</u>



- <u>CONCLUSION :-</u>

In this practical we learned how to perform addition and subtraction operations on 8-bit and 16-bit numbers in assembly language programming.

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: MIC	Subject Code:22414
Semester:4th Semester	Course: Computer Engineering
Laboratory No: L004B	Name of Subject Teacher: Pragati Mali
Name of Student: Aditya Makwana	Roll Id: 22203A0042

Experiment No:	4
Title of Experiment	Write an assembly language program for multiplication of 8 bit and 16 bit signed and unsigned numbers

- **Software used :**

  Emu8086

- **PROGRAM CODE :**

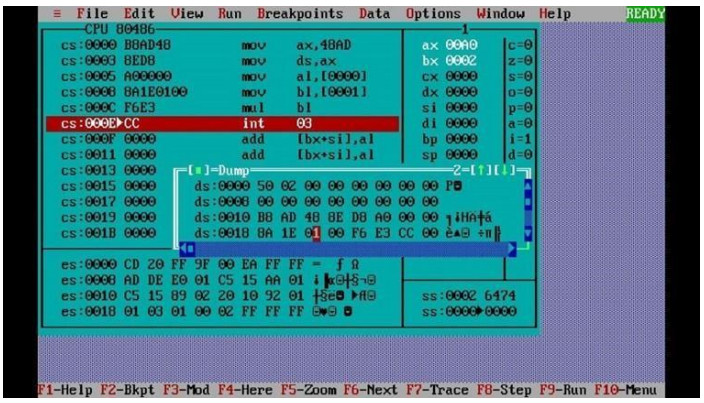
1. Write an ALP to multiply two given 8-bits numbers.

ANS: DATA SEGMENT

```
NUM1 DB 10H
NUM2 DB 05H
DATA ENDS
CODE SEGMENT ASSUME
CS:CODE,DS:DATA
START:

MOV AX,DATA
```

```
        MOV DS,AX
        MOV AL,NUM1
        MUL AL,NUM2
        INT 3
        CODE ENDS
END START
```

Output:



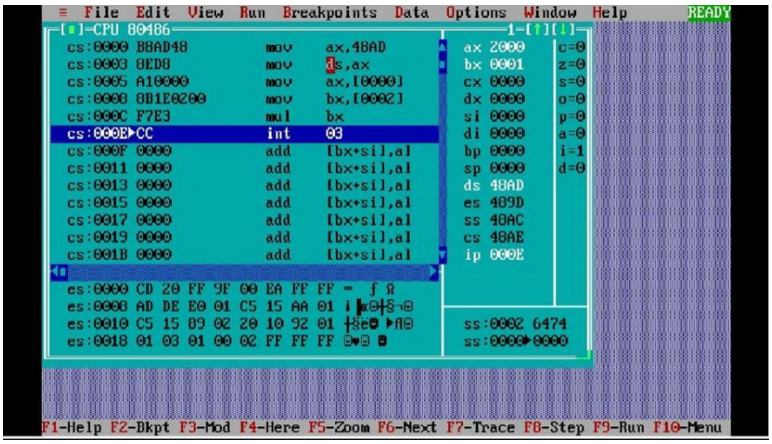
2. Write an ALP to multiply two given 16-bits numbers.

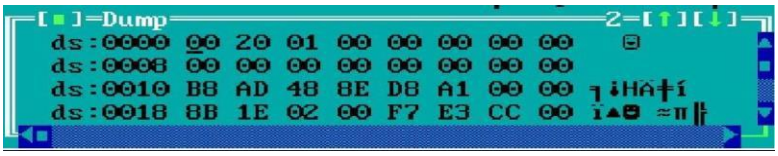
ANS: DATA SEGMENT

```
        NUM1 DW 2000H
        NUM2 DW 0001H
        DATA ENDS
        CODE SEGMENT ASSUME
        CS:CODE,DS:DATA
        START:

        MOV AX,DATA
        MOV DS,AX
        MOV AX,NUM1
        MOV BX,NUM2
```

```
MUL BX
INT 3

CODE ENDS
END START
```

Output:





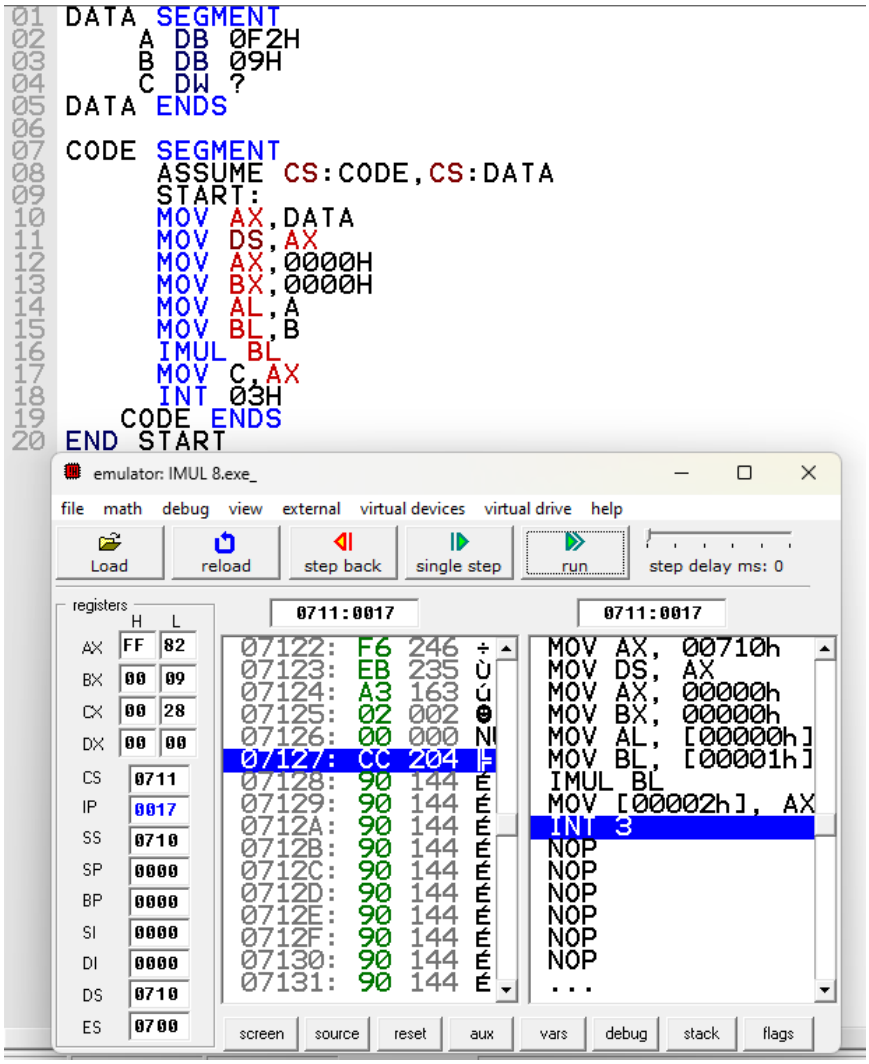
3. Write an ALP to multiply two given 8-bits signed numbers.

ANS:

```
DATA SEGMENT
    A DB 0F2H
    B DB 09H
    C DW ?
DATA ENDS
CODE SEGMENT
    ASSUME CS:CODE,CS:DATA
    START:
    MOV AX,DATA
    MOV DS,AX
```

```
MOV AX,0000H

MOV BX,0000H

MOV  AL,A

MOV BL,B

IMUL BL

MOV C,AX

INT 03H

CODE ENDS

END START
```

Output:



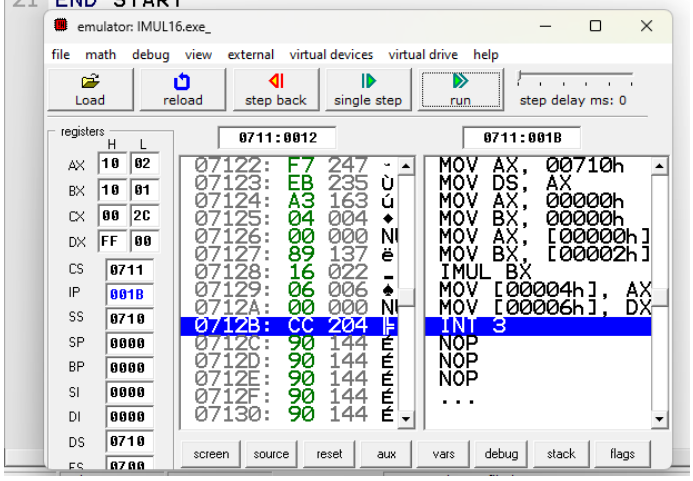
4. Write an ALP to multiply two given 16-bits signed numbers.

ANS:

```
DATA SEGMENT
    A DW 0F002H
    B DW 1001H
    C DD ?
DATA ENDS

CODE SEGMENT
    ASSUME DS:CODE,CS:DATA
    START:
    MOV AX,DATA
    MOV DS,AX
    MOV AX,0000H
    MOV BX,0000H
    MOV AX,A
    MOV BX,B
    IMUL BX
    MOV WORD PTR C,AX
    MOV WORD PTR C+2, DX
    INT 3
  CODE ENDS
END START
```

Output:

```
DATA SEGMENT
     A DW 0F002H
     B DW 1001H
     C DD ?
DATA ENDS

CODE SEGMENT
     ASSUME DS:CODE,CS:DATA
     START:
     MOV AX,DATA
     MOV DS,AX
     MOV AX,0000H
     MOV BX,0000H
     MOV AX,A
     MOV BX,B
     IMUL BX
     MOV WORD PTR C,AX
     MOV WORD PTR C+2, DX
     INT 3
   CODE ENDS
END START
```



- **Conclusion :**

In above practical we learn how to do the multiplication of signed and Unsigned 8 bit and 16-bit numbers

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: MIC	Subject Code:22414
Semester:4th Semester	Course: Computer Engineering
Laboratory No: L004B	Name of Subject Teacher: Pragati Mali
Name of Student: Aditya Makwana	Roll Id: 22203A0042

Experiment No:	5
Title of Experiment	Write an assembly language program for division of 8 bit and 16 bit signed and unsigned numbers

- **Software used :**

  Emu8086

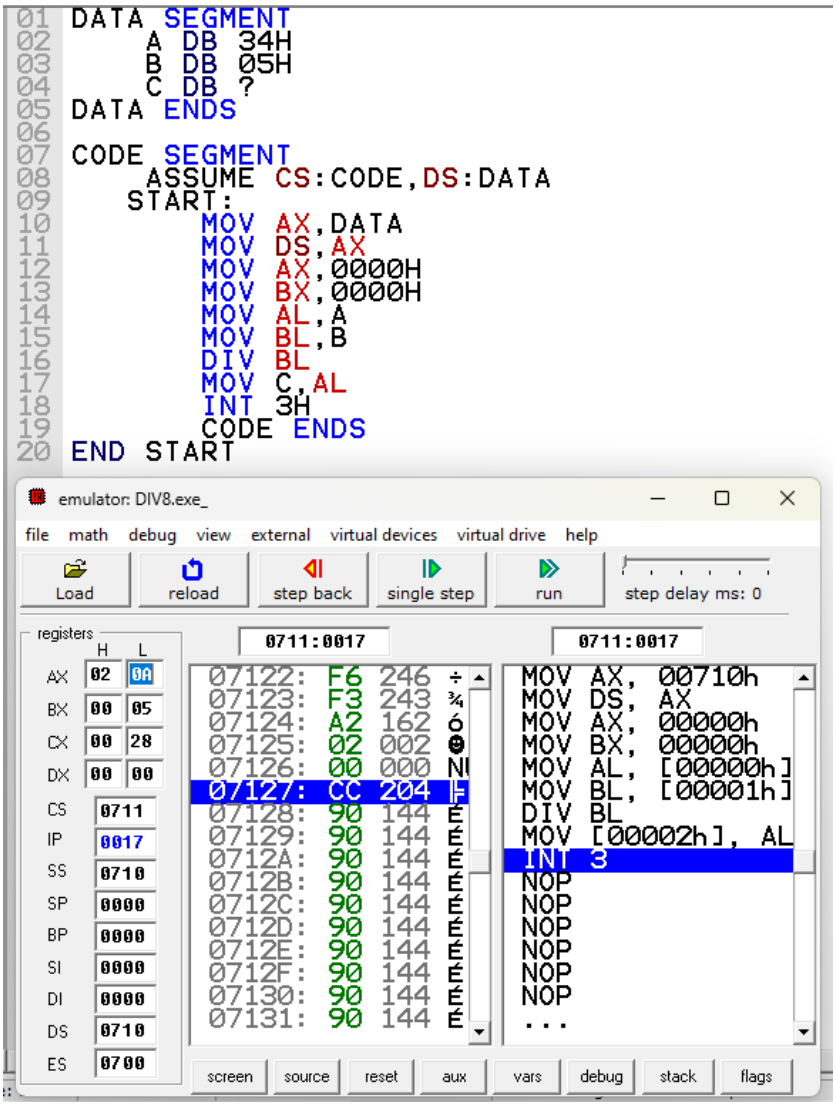
- **PROGRAM CODE :**

1. Write an ALP to divide two given 8-bits numbers.

ANS:
```
DATA SEGMENT
    A DB 34H
    B DB 05H
    C DB ?
DATA      ENDS
CODE SEGMENT
    ASSUME CS:CODE,DS:DATA
   START:
      MOV AX,DATA
```

```
        MOV DS,AX
        MOV AX,0000H
        MOV BX,0000H
        MOV  AL,A
        MOV BL,B
        DIV BL
        MOV C,AL
        INT 3H
        CODE ENDS
    END START
```

Output:



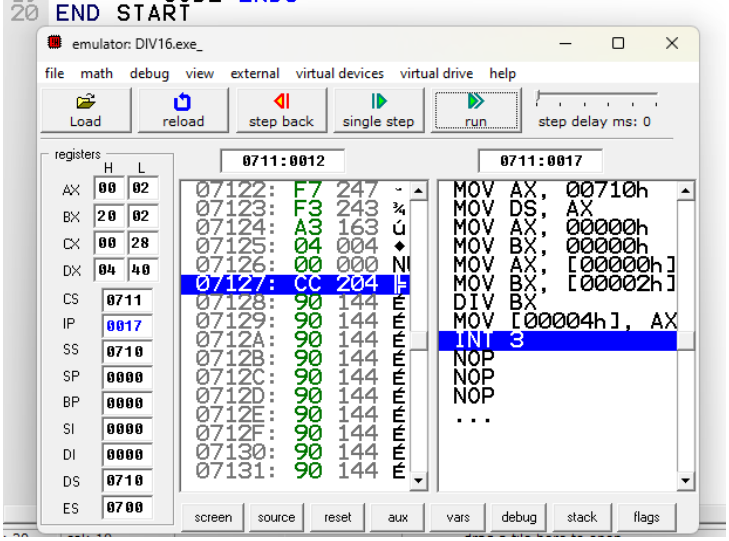
2. Write an ALP to divide two given 16-bits numbers.

ANS:

```
DATA SEGMENT
    A  DW  4444H
    B  DW  2002H
    C DW ?
DATA ENDS

CODE SEGMENT
    ASSUME CS:CODE,DS:DATA
   START:
      MOV AX,DATA
      MOV DS,AX
      MOV AX,0000H
      MOV BX,0000H
      MOV AX,A
      MOV BX,B
      DIV BX
      MOV C,AX
      INT 3H
      CODE ENDS
END START
```

Output:

```
DATA SEGMENT
     A DW  4444H
     B DW  2002H
     C DW ?
DATA ENDS

CODE SEGMENT
    ASSUME CS:CODE,DS:DATA
   START:
        MOV AX,DATA
        MOV DS,AX
        MOV AX,0000H
        MOV BX,0000H
        MOV AX,A
        MOV BX,B
        DIV BX
        MOV C,AX
        INT 3H
        CODE ENDS
END START
```



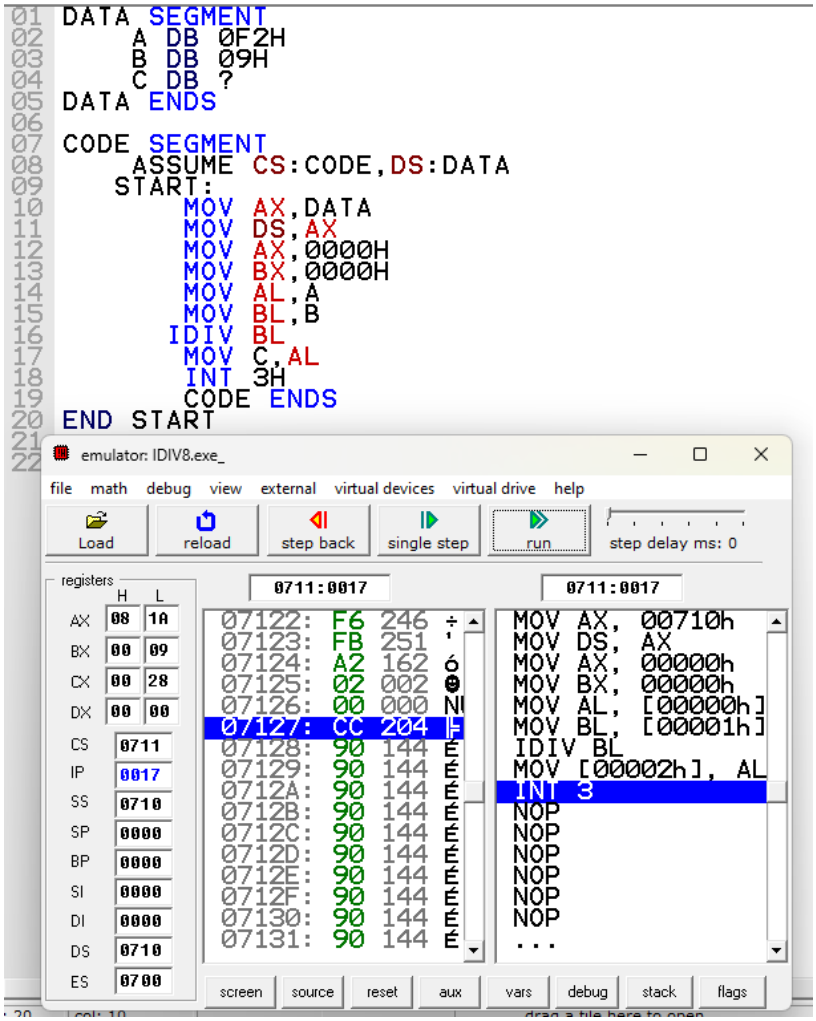
3. Write an ALP to divide two given 8-bits signed numbers.

ANS: DATA SEGMENT

```
        A DB 0F2H

        B DB 09H

        C DB ?

    DATA ENDS


    CODE SEGMENT

        ASSUME CS:CODE,DS:DATA

       START:

          MOV AX,DATA
```

```
        MOV DS,AX
        MOV AX,0000H
        MOV BX,0000H
        MOV  AL,A
        MOV BL,B
       IDIV BL
        MOV C,AL
        INT 3H
        CODE ENDS
    END START
```

Output:



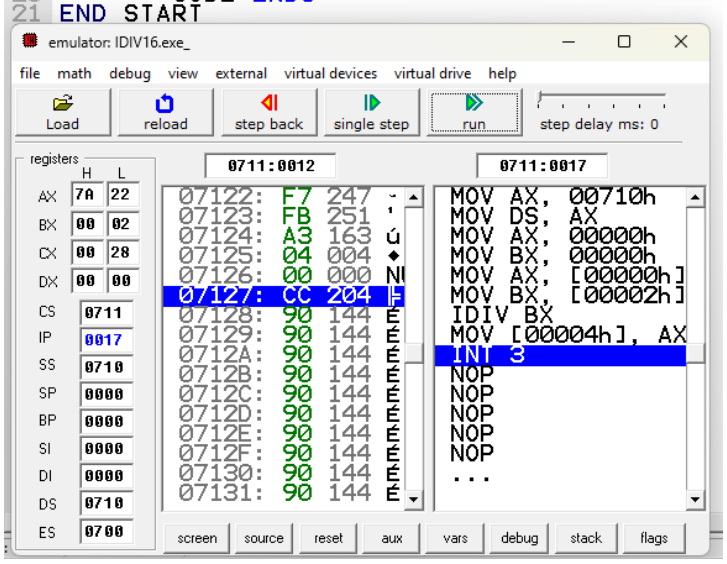
4. Write an ALP to divide two given 16-bits signed numbers.

ANS:

```
DATA SEGMENT
    A DW 0F444H
    B DW  0002H
    C DW ?
DATA ENDS

CODE SEGMENT
    ASSUME CS:CODE,DS:DATA
   START:
      MOV AX,DATA
      MOV DS,AX
      MOV AX,0000H
      MOV BX,0000H
      MOV AX,A
      MOV BX,B
      IDIV BX
      MOV C,AX
      INT 3H
      CODE ENDS
END START
```

Output:

```
DATA SEGMENT
     A DW  0F444H
     B DW  0002H
     C DW ?
DATA ENDS

CODE SEGMENT
    ASSUME CS:CODE,DS:DATA
   START:
        MOV AX,DATA
        MOV DS,AX
        MOV AX,0000H
        MOV BX,0000H
        MOV AX,A
        MOV BX,B
        IDIV BX
        MOV C,AX
        INT 3H
        CODE ENDS
END START
```



- **Conclusion :**

  In above practical we learn how to do the division of signed and Unsigned 8 bit and 16-bit numbers

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: MIC	Subject Code:22414
Semester:4th Semester	Course: Computer Engineering
Laboratory No: L004B	Name of Subject Teacher: Pragati Mali
Name of Student: Aditya Makwana	Roll Id: 22203A0042

Experiment No:	6
Title of Experiment	Write an assembly language program for BCD addition, subtraction, multiplication and division of 8 bit numbers

- **Software used :**

  Emu8086

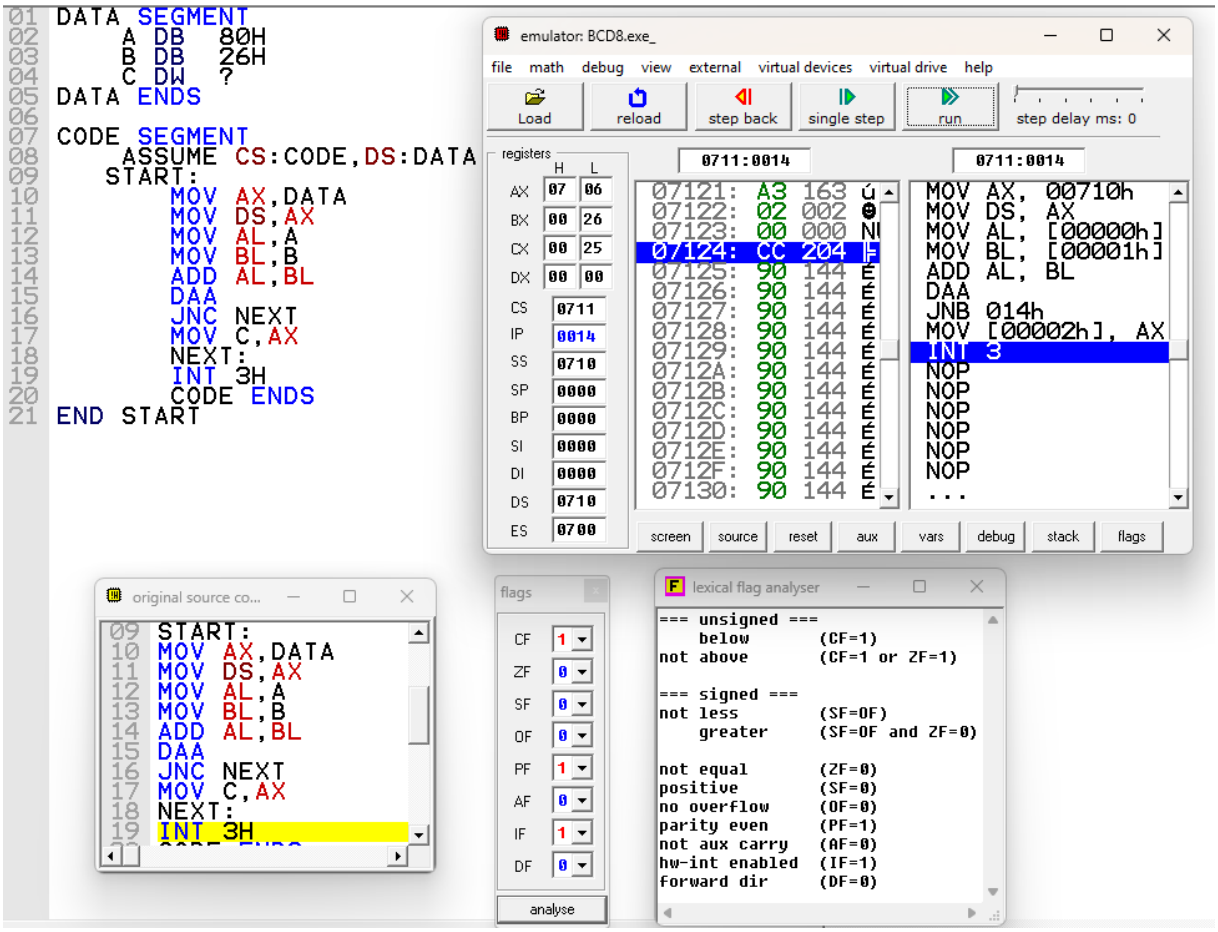
- **PROGRAM CODE :**

1. Write an ALP to BCD add two given 8-bits numbers.

ANS:
```
DATA SEGMENT
    A DB  80H
    B DB 26H
    C DW  ?
DATA ENDS

CODE SEGMENT
    ASSUME CS:CODE,DS:DATA
    START:
```

```
        MOV AX,DATA
        MOV DS,AX
        MOV AL,A
        MOV BL,B
        ADD AL,BL
        DAA
        JNC NEXT
        MOV C,AX
        NEXT:
        INT 3H
        CODE ENDS
    END START
```

Output:

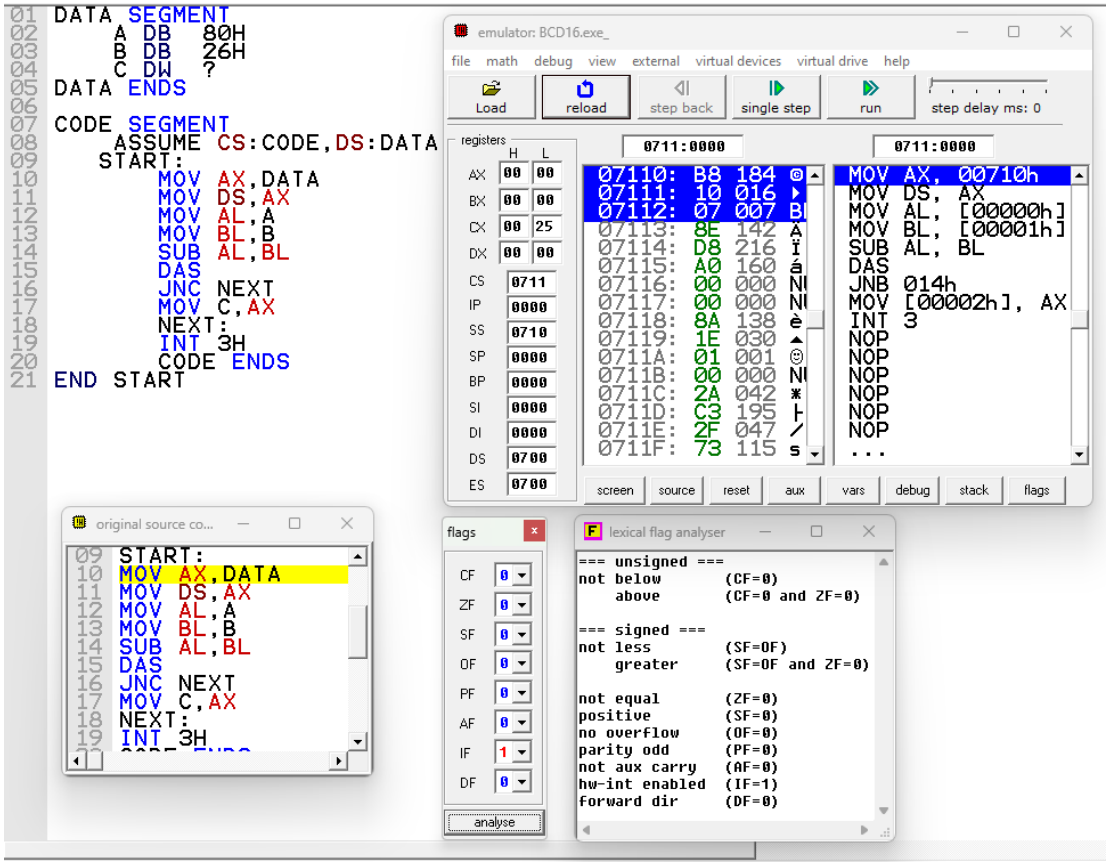


2. Write an ALP to BCD subtract two given 16-bits numbers.

ANS:

```
DATA SEGMENT
    A  DW  4444H
    B  DW  2002H
    C DW ?
DATA ENDS

CODE SEGMENT
    ASSUME CS:CODE,DS:DATA
   START:
      MOV AX,DATA
      MOV DS,AX
      MOV AX,0000H
      MOV BX,0000H
      MOV AX,A
      MOV BX,B
      DIV BX
      MOV C,AX
      INT 3H
      CODE ENDS
END START
```

Output:



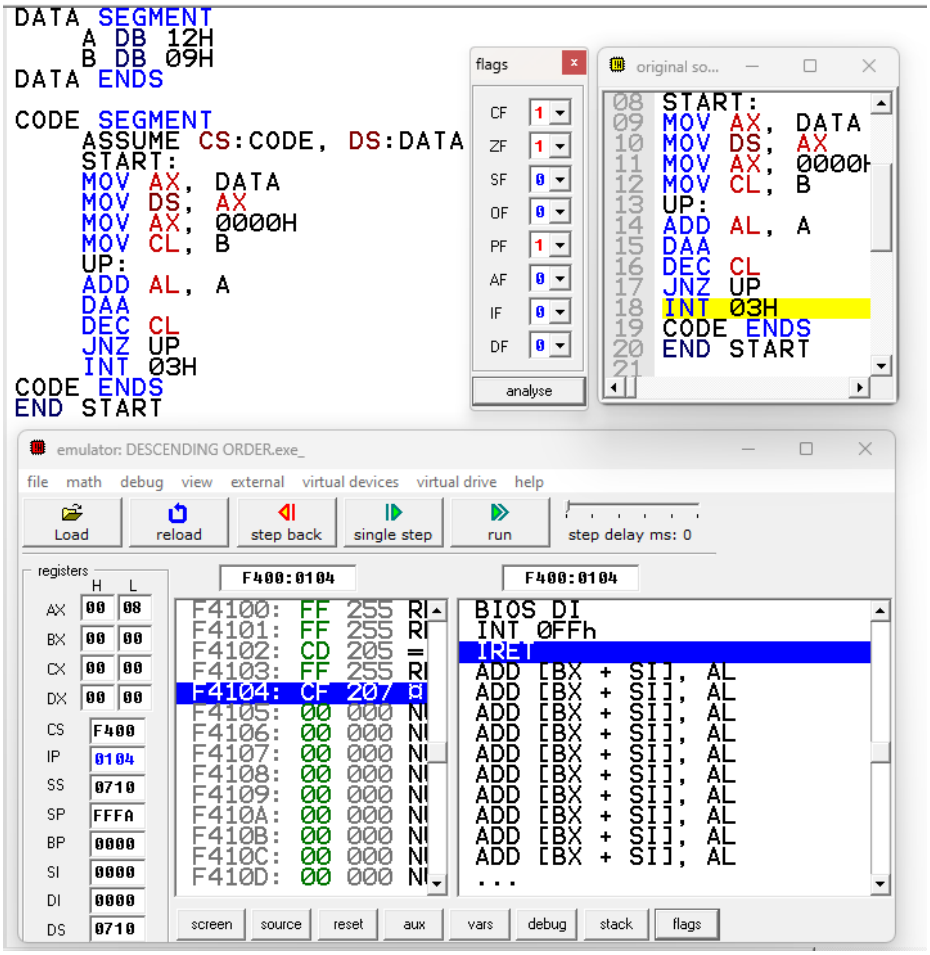
3. Write an ALP to BCD multiply two given 8-bits numbers.

ANS:

```
DATA SEGMENT
    A DB 12H
    B DB 09H
DATA ENDS

CODE SEGMENT
    ASSUME CS:CODE, DS:DATA
    START:
    MOV AX, DATA
    MOV DS, AX
```

```
    MOV AX, 0000H
    MOV CL, B
    UP:
    ADD AL, A
    DAA
    DEC   CL
    JNZ   UP
    INT 03H
CODE ENDS
END START
```

Output:

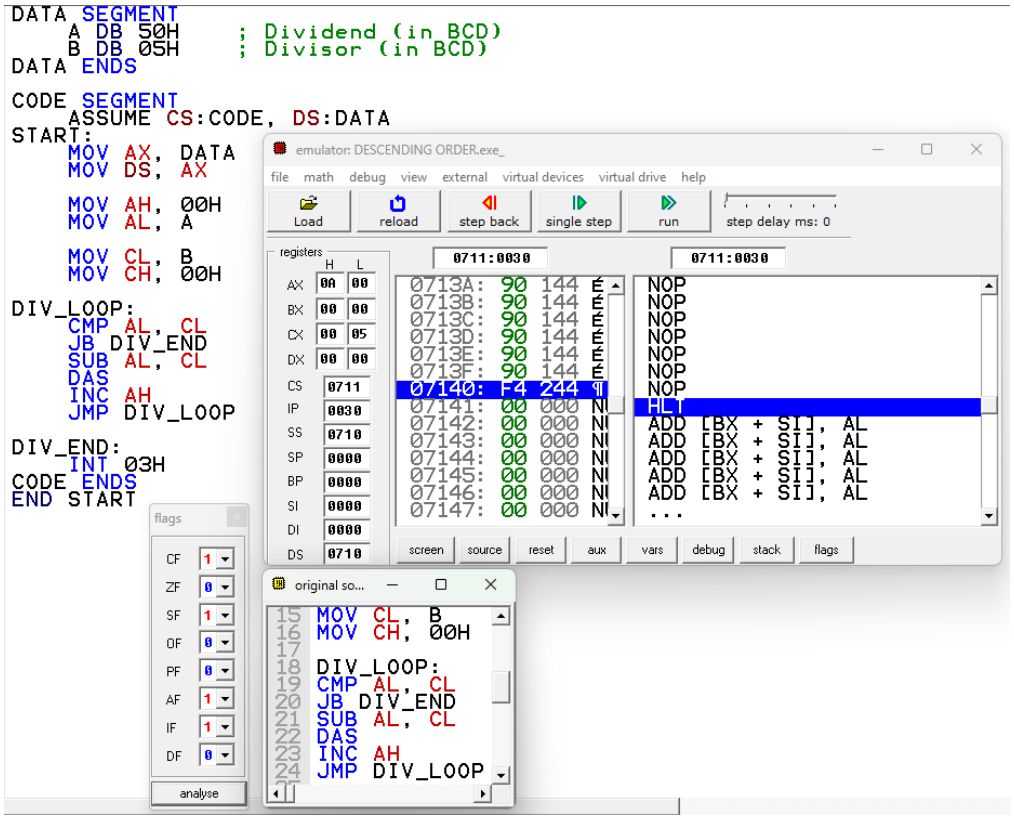


4. Write an ALP to BCD divide two given 8-bits numbers.

ANS:

```
DATA SEGMENT
    A DB 12H
    B DB 09H
DATA ENDS

CODE SEGMENT
    ASSUME CS:CODE, DS:DATA
    START:
    MOV AX, DATA
    MOV DS, AX
    MOV AX, 0000H
    MOV CL, B
    UP:
    ADD AL, A
    DAA
    DEC   CL
    JNZ   UP
    INT 03H
CODE ENDS
END START
```

Output:



- **Conclusion :**

In above practical we learn how to do the BCD addition ,subtraction ,division , multiplicatino of given numbers

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: Microprocessor	Subject Code:22415
Semester:4th Semester	Course: Computer Engineering
Laboratory No: L004	Name of Subject Teacher: Pragati Mali
Name of Student: Aditya Makwana	Roll Id: 22203A0042

Experiment No:	7
Title of Experiment	a) Write an ALP to perform block data transfer using string instructions. b) Write an ALP to perform block data transfer without using string instructions

1. Write an ALP to perform block data transfer without using string instructions?

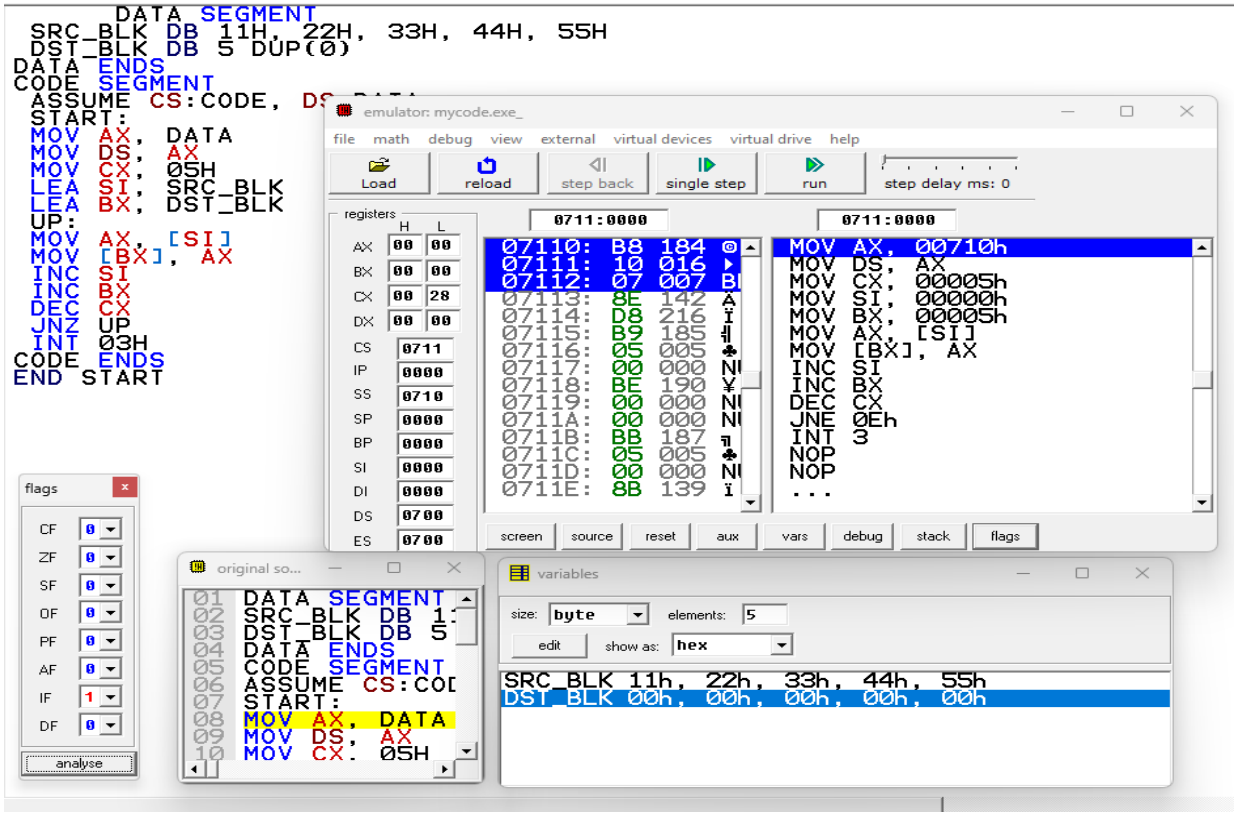
ANS:

PROGRAM:

```
DATA SEGMENT
SRC_BLK DB 11H, 22H, 33H, 44H, 55H
DST_BLK DB 5 DUP(0)
DATA ENDS
CODE SEGMENT
ASSUME CS:CODE, DS:DATA
```

```
START:
MOV AX, DATA
MOV DS, AX
MOV CX, 05H
LEA SI, SRC_BLK
LEA BX, DST_BLK
UP:
MOV AX, [SI]
MOV [BX], AX
INC SI
INC   BX
DEC   CX
JNZ   UP
INT 03H
CODE ENDS
END START
```

OUTPUT:



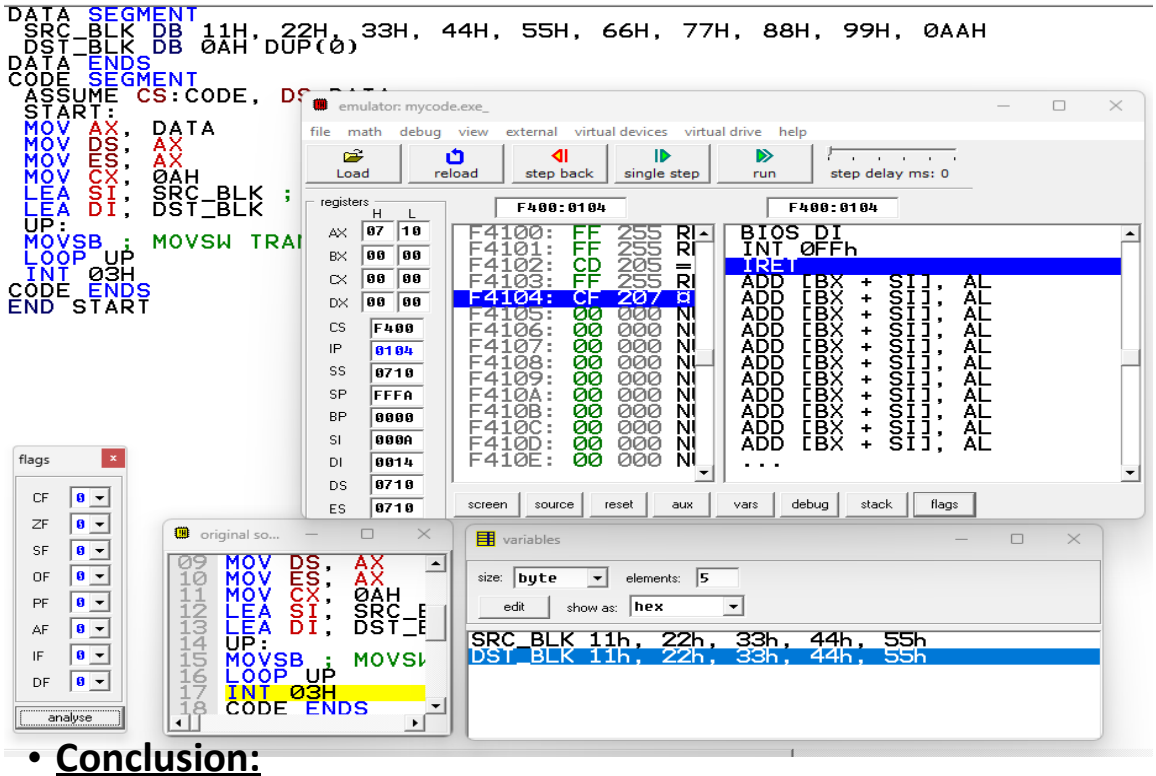
2. Write an ALP to perform block data transfer using string instructions ?

ANS:

PROGRAM:

```
DATA SEGMENT
 SRC_BLK DB 11H, 22H, 33H, 44H, 55H, 66H, 77H, 88H, 99H, 0AAH
 DST_BLK DB 0AH DUP(0)
DATA ENDS
CODE SEGMENT
 ASSUME CS:CODE, DS:DATA
 START:
 MOV AX, DATA
 MOV DS, AX
 MOV ES, AX
 MOV CX, 0AH
 LEA SI, SRC_BLK ; MOV SI, OFFSET SRC_BLK
 LEA DI, DST_BLK
 UP:
 MOVSB ; MOVSW TRANSFER TWO BYTES
 LOOP UP
 INT 03H
CODE ENDS
END START
```

OUTPUT:



- **Conclusion:**

In this practical we learnt how to operate on array to sort it in ascending as well as descending order.

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: Microprocessor	Subject Code:22415
Semester:4th Semester	Course: Computer Engineering
Laboratory No: L004	Name of Subject Teacher: Pragati Mali
Name of Student: Aditya Makwana	Roll Id: 22203A0042

Experiment No:	8
Title of Experiment	a) Write an ALP to find sum of series of Hexadecimal numbers b) Write an ALP to find sum of series of BCD numbers

1. **Write an ALP to find sum of series of Hexadecimal numbers.**

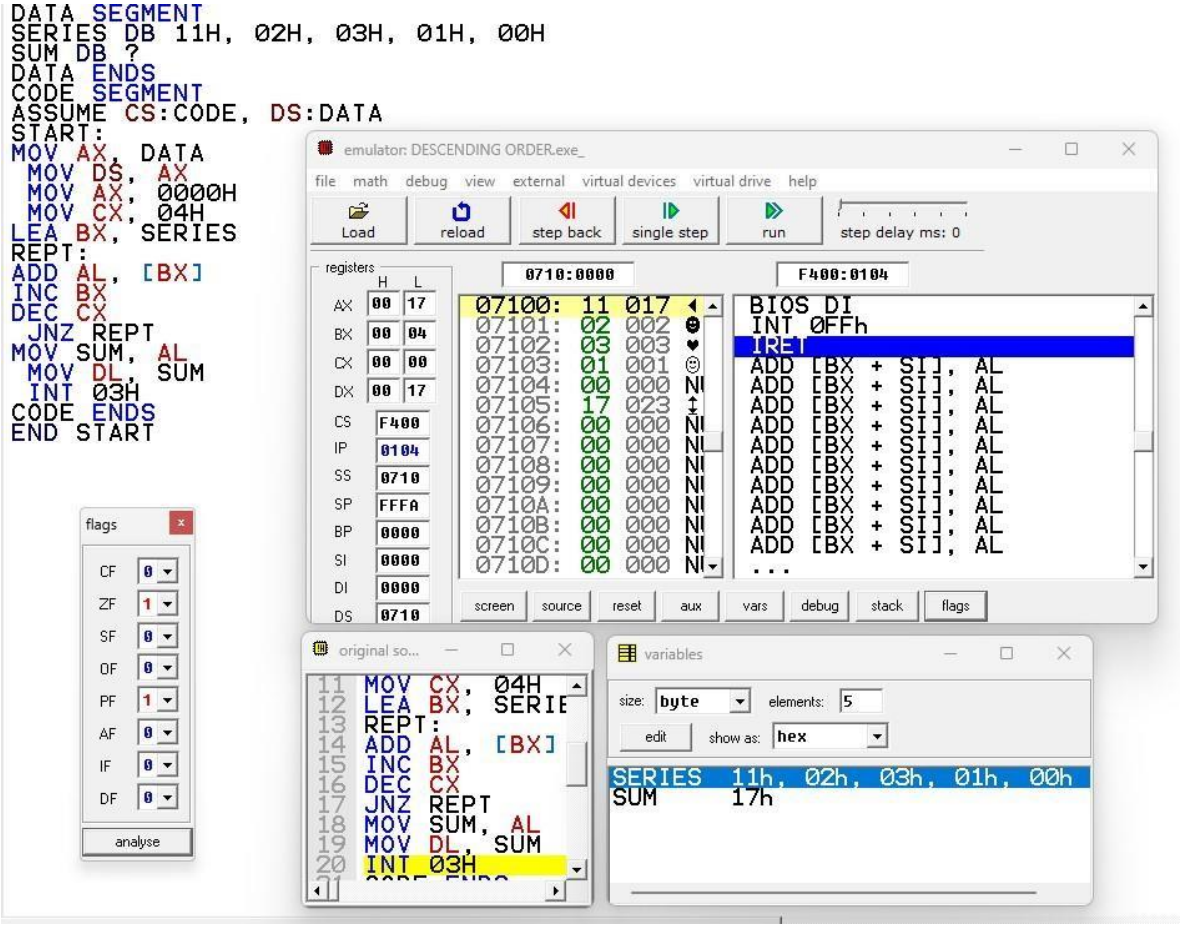
ANS:

PROGRAM:

```
DATA SEGMENT
SERIES DB 11H, 02H, 03H, 01H, 00H
SUM DB ?
DATA ENDS
CODE SEGMENT ASSUME
CS:CODE, DS:DATA START:
MOV AX, DATA
 MOV DS,  AX
MOV AX, 0000H
MOV CX, 04H LEA
BX, SERIES REPT:
ADD AL, [BX]
INC BX
```

```
DEC     CX
 JNZ REPT
MOV  SUM,  AL
 MOV DL, SUM
 INT 03H
CODE ENDS
END START
```

OUTPUT:



2. Write an ALP to find sum of series of BCD numbers.

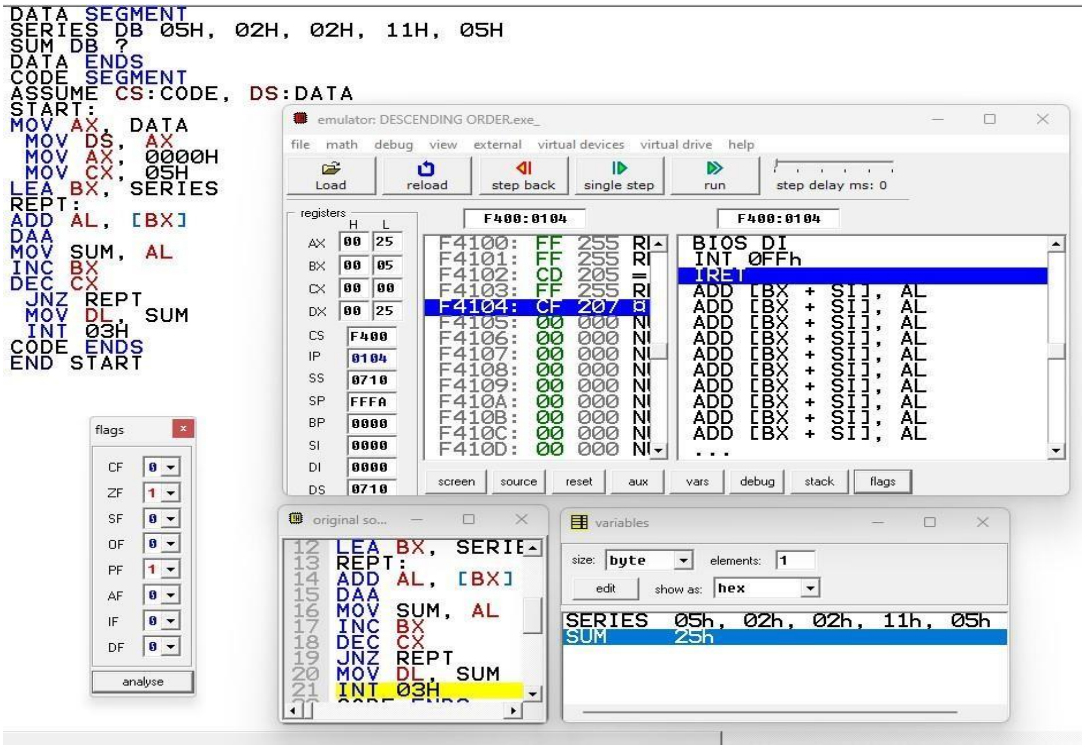
ANS:

PROGRAM:

```
DATA SEGMENT
```

```
SERIES DB 05H, 02H, 02H, 11H, 05H SUM
DB ?
DATA ENDS
CODE SEGMENT
ASSUME CS:CODE, DS:DATA START:
MOV AX, DATA
 MOV DS,  AX
MOV AX, 0000H
MOV CX, 05H LEA
BX, SERIES REPT:
ADD AL, [BX]
DAA
MOV SUM, AL
INC BX
DEC CX
 JNZ REPT
 MOV DL, SUM
 INT 03H
CODE ENDS
END START
```

OUTPUT:



- **Conclusion:**

In this practical we learned how to find sum of series of Hexadecimal and BCD numbers.

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: Microprocessor	Subject Code:22415
Semester:4th Semester	Course: Computer Engineering
Laboratory No: L004	Name of Subject Teacher: Pragati Mali
Name of Student: Aditya Makwana	Roll Id: 22203A0042

Experiment No:	9
Title of Experiment	a) Write an ALP to find smallest number in an array. b) Write an ALP to find largest number in an array.

1. Write an ALP to find smallest number in an array?

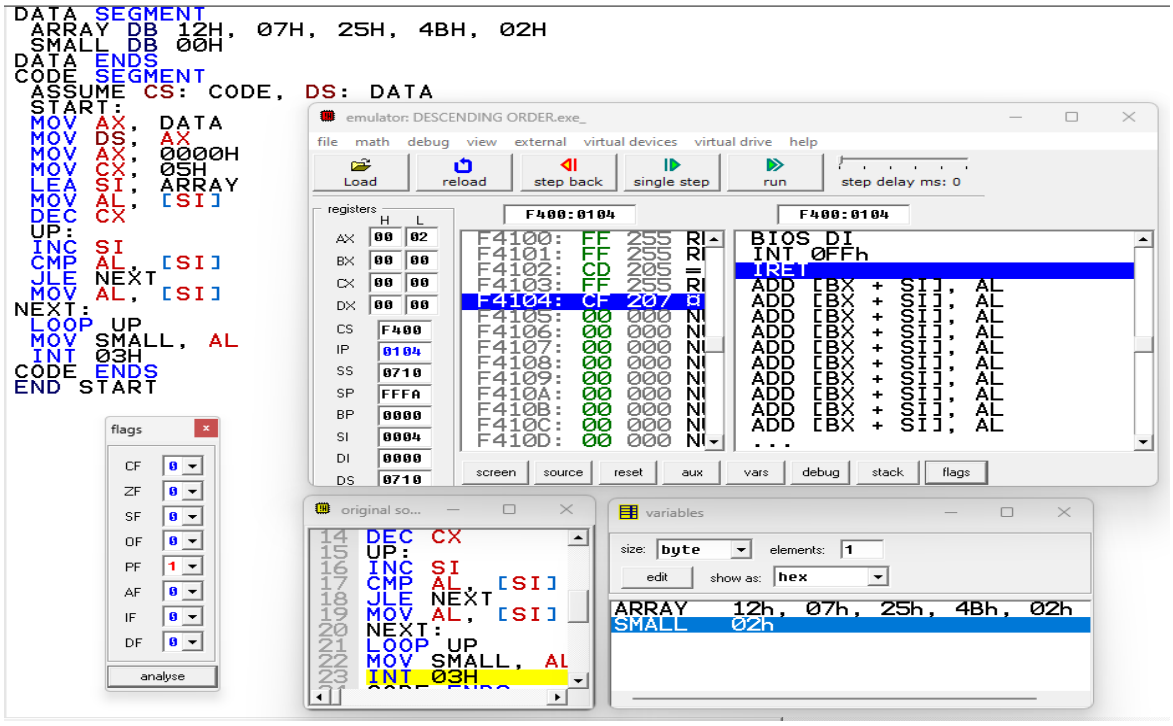
ANS:

PROGRAM:

```
DATA SEGMENT
 ARRAY DB 12H, 07H, 25H, 4BH, 02H
 SMALL DB 00H
DATA ENDS
CODE SEGMENT
 ASSUME CS: CODE, DS: DATA
 START:
 MOV AX, DATA
 MOV DS, AX
 MOV AX, 0000H
 MOV CX, 05H
 LEA SI, ARRAY
 MOV AL, [SI]
 DEC CX
```

```
 UP:
 INC SI
 CMP AL, [SI]
JLE NEXT
MOV AL, [SI]
NEXT:
 LOOP UP
 MOV SMALL, AL
 INT 03H
CODE ENDS
END START
```

OUTPUT:



2. **Write an ALP to perform block data transfer using string instructions.**

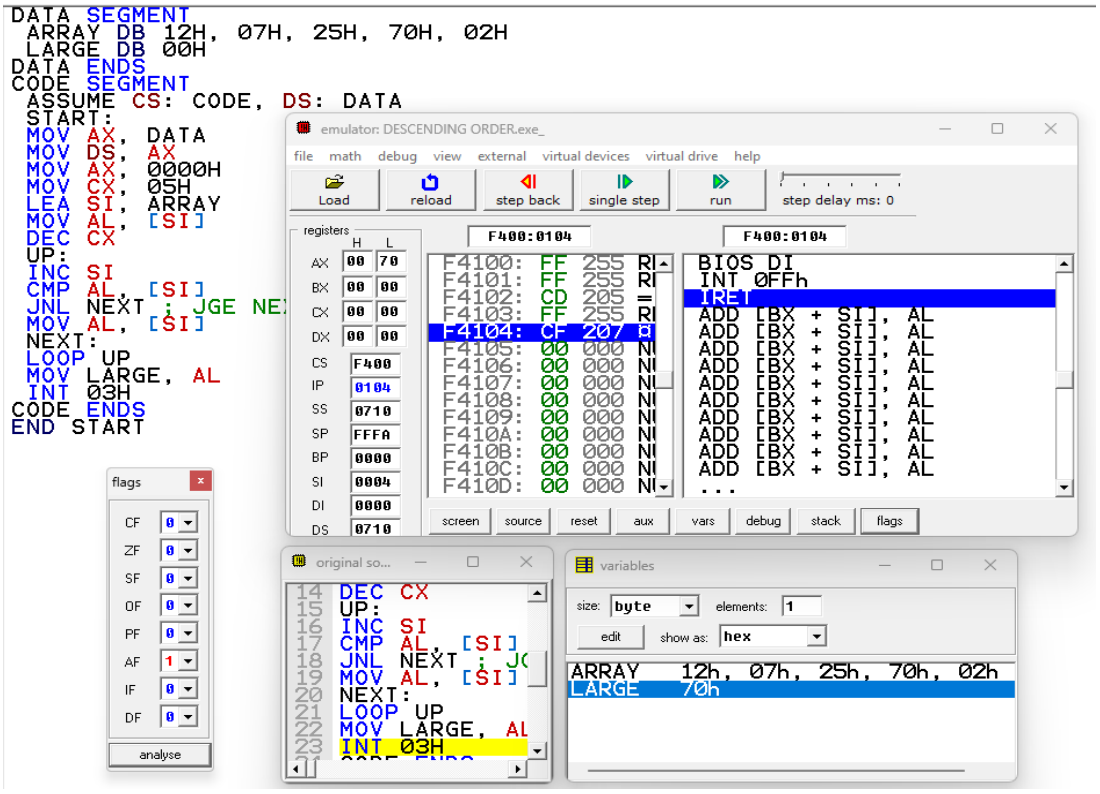
ANS:

PROGRAM:

```
DATA SEGMENT
SERIES DB 05H, 02H, 02H, 11H, 05H
SUM DB ?
DATA ENDS
```

```
CODE SEGMENT
ASSUME CS:CODE, DS:DATA
START:
MOV AX, DATA
MOV DS, AX
MOV AX, 0000H
MOV CX, 05H
LEA BX, SERIES
REPT:
ADD AL, [BX]
DAA
MOV SUM, AL
INC BX
DEC CX
 JNZ REPT
 MOV DL, SUM
 INT 03H
CODE ENDS
END START
```

OUTPUT:



- **Conclusion:**

With this practical we learned how to find largest and smallest number in an array.

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: Microprocessor	Subject Code:22415
Semester:4th Semester	Course: Computer Engineering
Laboratory No: L004	Name of Subject Teacher: Pragati Mali
Name of Student: Aditya Makwana	Roll Id: 22203A0042

Experiment No:	10
Title of Experiment	a) Write an ALP to arrange numbers in array in ascending order b) Write an ALP to arrange numbers in array in descending order.

1. **Write an ALP to arrange numbers in array in ascending order ?**

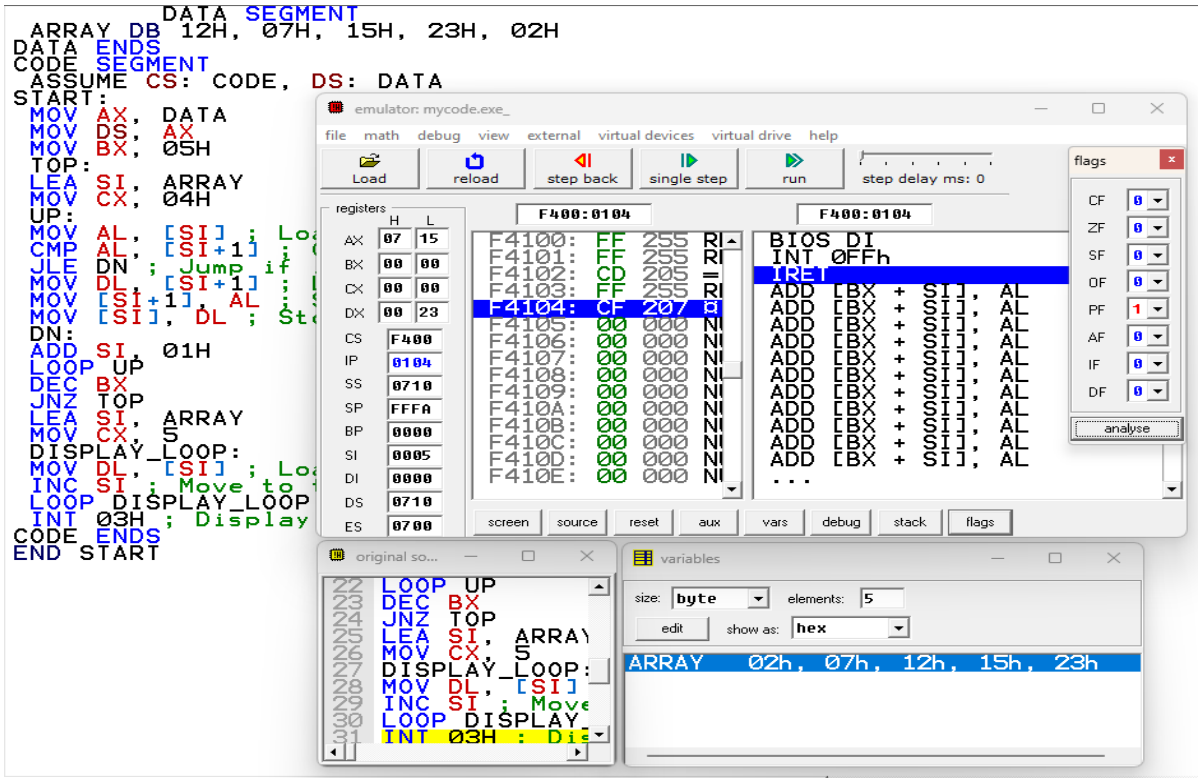
ANS:

PROGRAM:

```
DATA SEGMENT
 ARRAY DB 12H, 07H, 15H, 23H, 02H
DATA ENDS
CODE SEGMENT
 ASSUME CS: CODE, DS: DATA
START:
 MOV AX, DATA
 MOV DS, AX
 MOV BX, 05H
 TOP:
 LEA SI, ARRAY
 MOV CX, 04H
 UP:
 MOV AL, [SI] ; Load the current element into AL?
```

```
CMP AL, [SI+1] ; Compare with the next element?
JLE DN ; Jump if less than or equal, skip swap?
MOV DL, [SI+1] ; Load the next element into DL?
MOV [SI+1], AL ; Store AL into the next position?
MOV [SI], DL ; Store DL into the current position?
DN:
ADD SI, 01H
LOOP UP
DEC BX
JNZ TOP
LEA SI, ARRAY
MOV CX, 5
DISPLAY_LOOP:
MOV DL, [SI] ; Load the value to be displayed?
INC SI ; Move to the next position?
LOOP DISPLAY_LOOP ; Loop until all elements are displayed ?
INT 03H ; Display the character ?
CODE ENDS
END START
```

OUTPUT:



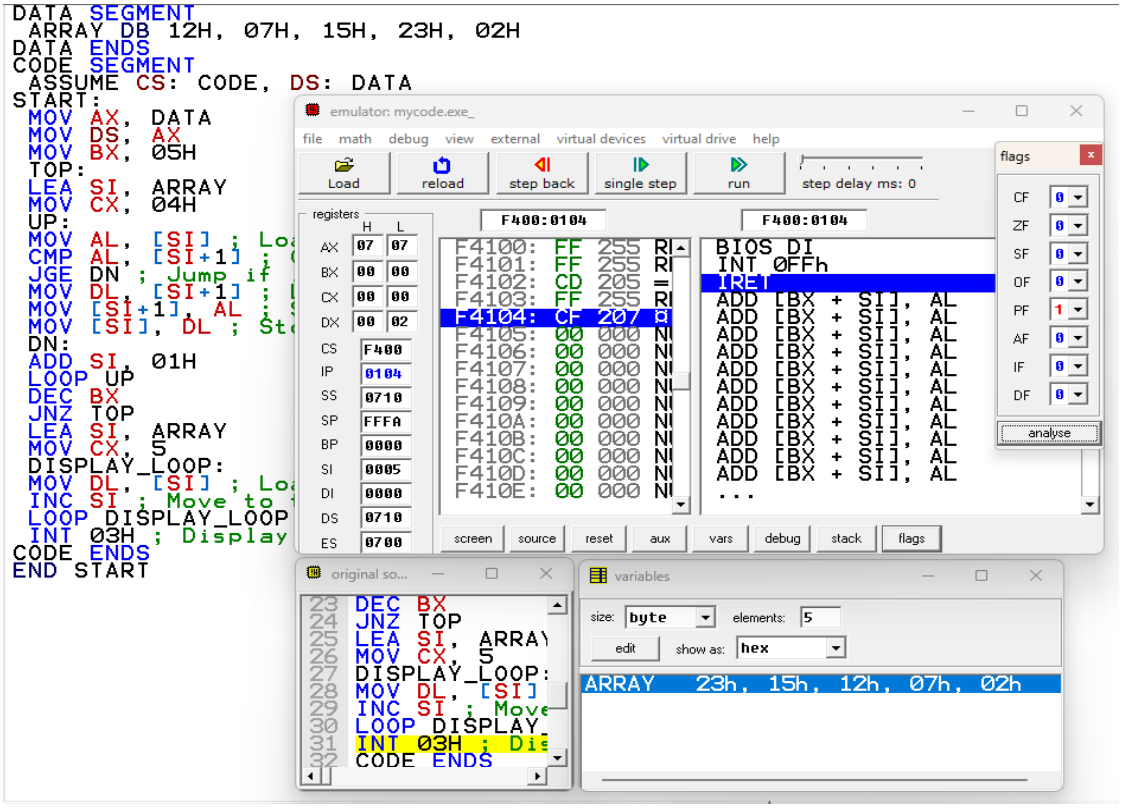
2. **Write an ALP program to arrange array in Descending order ?**

ANS:

PROGRAM:

```
DATA SEGMENT
 ARRAY DB 12H, 07H, 15H, 23H, 02H
DATA ENDS
CODE SEGMENT
 ASSUME CS: CODE, DS: DATA
START:
 MOV AX, DATA
 MOV DS, AX
 MOV BX, 05H
 TOP:
 LEA SI, ARRAY
 MOV CX, 04H
 UP:
 MOV AL, [SI] ; Load the current element into AL?
 CMP AL, [SI+1] ; Compare with the next element?
 JGE DN ; Jump if less than or equal, skip swap?
 MOV DL, [SI+1] ; Load the next element into DL?
 MOV [SI+1], AL ; Store AL into the next position?
 MOV [SI], DL ; Store DL into the current position?
 DN:
 ADD SI, 01H
 LOOP UP
 DEC BX
 JNZ TOP
 LEA SI, ARRAY
 MOV CX, 5
 DISPLAY_LOOP:
 MOV DL, [SI] ; Load the value to be displayed?
 INC SI ; Move to the next position?
 LOOP DISPLAY_LOOP ; Loop until all elements are displayed ?
 INT 03H ; Display the character ?
CODE ENDS
END START
```

OUTPUT:



- ## Conclusion:

In this practical we learnt how to operate on array to sort it in ascending as well as descending order.

Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: MICROPROCESSOR	Subject Code:22415
Semester:4th Semester	Course: Computer Engineering
Laboratory No:L004	Name of Subject Teacher: Pragati Mali
Name of Student: Aditya Makwana	Roll Id: 22203A0042

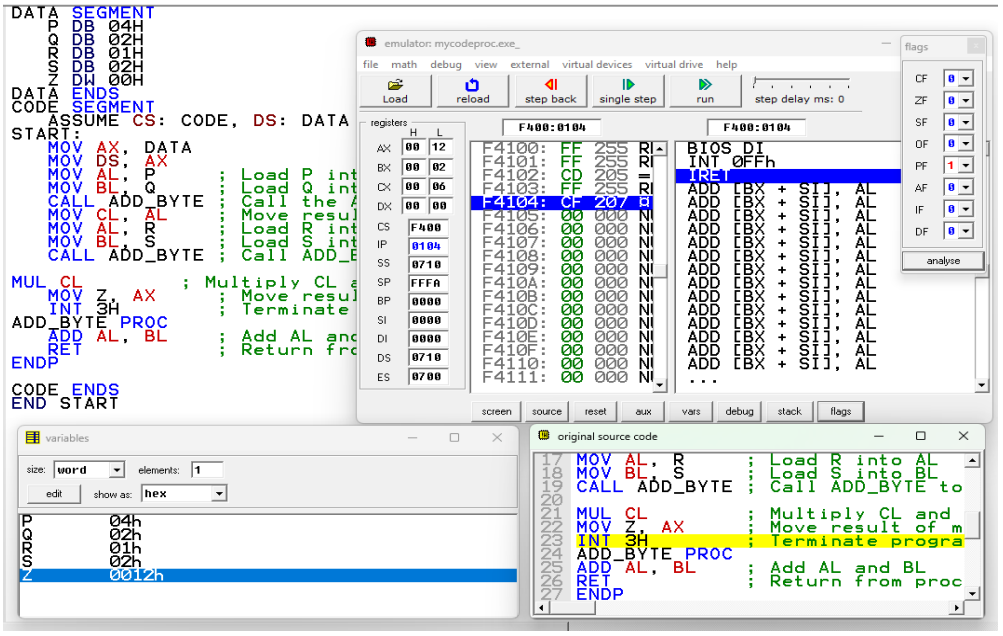
Experiment No:	11
Title of Experiment	Write an ALP for Z=(P+Q)*(R+S) using procedure.

- **PROGRAM:-**

```
DATA SEGMENT
  P DB 04H
  Q  DB  02H
  R  DB  01H
  S  DB  02H
  Z DW 00H
DATA        ENDS
CODE SEGMENT
  ASSUME CS: CODE, DS: DATA
START:
  MOV AX, DATA
  MOV DS, AX
  MOV AL, P     ; Load P into AL
  MOV BL, Q     ; Load Q into BL
  CALL ADD_BYTE ; Call the ADD_BYTE procedure to add AL and BL
```

```
    MOV CL, AL    ; Move result of addition into CL
    MOV AL, R     ; Load R into AL
    MOV BL, S     ; Load S into BL
    CALL ADD_BYTE ; Call ADD_BYTE to add AL and BL
  MUL CL       ; Multiply CL and AL
    MOV Z, AX     ; Move result of multiplication into Z
    INT 3H        ; Terminate program
  ADD_BYTE PROC
    ADD AL, BL    ; Add AL and BL
    RET           ; Return from procedure
  ENDP
  CODE ENDS
  END START
```

- **OUTPUT: -**



Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: MICROPROCESSOR	Subject Code:22415
Semester:4th Semester	Course: Computer Engineering
Laboratory No:L004	Name of Subject Teacher: Pragati Mali
Name of Student: Aditya Makwana	Roll Id: 22203A0042

Experiment No:	12
Title of Experiment	Write an ALP for Z=(A+B)*(C+D) using macro.

- **PROGRAM:-**

```
DATA SEGMENT
  A DB 04H
  B DB 04H
  C DB 01H
  D DB 02H
  R1 DB 00H
  R2 DB 00H
  Z DW 00H
DATA ENDS
SUM_BYTE MACRO A1, A2, RES
  MOV AL, A1     ; Move A1 into AL
  ADD AL, A2     ; Add A2 to AL
  MOV RES, AL    ; Move the result in AL to RES
ENDM
```

```
CODE SEGMENT
  ASSUME CS: CODE, DS: DATA
START:
  MOV AX, DATA
  MOV DS, AX
SUM_BYTE A, B, R1 ; Call SUM_BYTE macro to add A and B, store result in R1
SUM_BYTE C, D, R2  ; Call SUM_BYTE macro to add C and D, store result in R2
  MOV AL, R1     ; Move R1 into AL
  MUL R2         ; Multiply R2 with AL, result in AX
  MOV Z, AX      ; Move result of multiplication into Z
  INT 3H
CODE ENDS
END START
```

- **OUTPUT: -**