Conclusion:

For obj 1

It can be concluded that for immediate addressing the operand is specified in the instruction itself.

For obj 2

It can be concluded that for direct addressing offset is given in the instruction as a 16 bit displacement element.

For obj 3

It can be concluded that for indirect addressing the operand offset is placed SI registers as specified in the instructions.

For obj 4

It can be concluded that for indirect addressing the offset is the sum of the content of the SI registers and a 16 bit displacement element.

For obj 5

It can be concluded that for base indexing addressing the offset is the sum of content of BV and SI registers.

IV. POST LAB:

1. Discuss different general-purpose registers used in 8086 microprocessor.

Ans :- Execution limit has 8 general purpose registers :-

AX : word multiply, word divide, word I/O

BAL : Byte multiply, byte divide, byte I/O

AH : Byte multiply, Byte divide

BX : Store address information

CX : String operation, Loops

CL : Variable shift and rotate

bx : Variable shift and rotate.

2. Explain the concept of segmented memory. What are its advantages?

Ans It is the process in which main memory of computer is divided into different segment and each segment has its own base address

Advantages

1. It provides a powerful memory managment mechanism
2. code related operation can be done in separate code segments.
3. Code related or stack related operations can be performed in different segments.
4. It allows to process to easily passed data.

Lab1

3. Explain the physical address formation in 8086.

physical address = Base/ Segment Address X 1011 + Offset address.

4. Write a program to add two 16 bit numbers 1F12H and 08H, and store the sum using indirect addressing modes.

```
MOV SI, 0520H
MOV AX, 1F12H
MOV BX, 0008H
ADD AX, BX
MOV [SI], AX
HLT
```

Aasif

V. Conclusion:

For obj 1

It can be concluded that the sum of numbers when dry run and executed in system found to be same. Thus the program to add two 16 bit number was executed.

For obj 2

It can be concluded that the difference of numbers when dry run and executed in system found to be same. Thus the program to subtract two 16 bit number was executed.

For obj 3

It can be concluded that the product of numbers when dry run and executed in the system found to be same. Thus program to multiply two 16 bit number was executed.

For obj 4

It can be concluded that the division of number when dry run and executed in system found to be same. Thus the program to divide two 16 bit number was executed.

For obj 5

It can be concluded that the multiplication of two 16 bit number can be done without using MUL instruction, by using repetive addition.

For obj 6

It can be concluded that the division of two 16 bit numbers can be done without using DIV instruction by using repetative subtraction.

IV. POST LAB:

1. Write the assembly language program to subtract two 16 bit numbers 20H and 06H, and store the difference using register indirect addressing mode.

Ans

```
MOV AX, 0020H;
MOV BX, 0006H;
SUB AX, BX;
MOV [BP], AX;
HLT
```

2. Explain briefly any five arithmetic instructions.

Ans ADD:- used to add the provided byte to byte / word to word.

SUB:- used to subtract the byte from byte / word from word.

MUL:- used to multiply unsigned word by byte ~~word~~ word by word.

DIV:- used to divide the unsigned word by byte or unsigned double word by word.

INC:- used to increment the provided byte/word by 1

3. Which of the following instruction is not valid

a) MOV AX, BX

b) MOV DS,5000H

c) MOV AX, 5000H

d) MOV AX, [5000H]

b) MOV DS, 5000H

4. Write the function of the following instructions

a) JMP b) HLT c) INC d)ADC AL,01H

a) JMP - unconditionally transfers the control of execution to the specified address

(b) HLT - Halt CPU

(c) INC -

IV. POST LAB:

1. Enlist the advantages of assembly language programming over machine language.

Advantages

a) It allows complex jobs to run in a simpler way
b) It is memory efficient as it require less memory
(c) It is faster in speed as its execution time is less.
d) It is mainly hardware oriented
e) It is used for critical jobs
f) It is low level embedded system

2. Write the function of the following logical instructions.
   a) SHL/SAL b) SHR c) SAR d) ROR e) ROL

a) SHL/SAL
It is used to shift bits of a byte/words to words towards left and put zeroes in LSBs.

b) SHR
It is used to shift bits of a byte/words towards right and copy the old MSBs into new MSB.

(c) SAR
It is used to ~~rotate~~ shift bits of bytes towards the right ie LSB to MSB and to Carry Flag.

(e) ROL:
It is used to rotate bits of byte towards the left ie MSB to LSB and to Carry flag.

3. Write a assembly program to multiply 05H and 04H without using arithmetic instruction.

```
MOV AX, 0000H
MOV DS, AX
MOV AX, 0000H
MOV BX, [2000H]
MOV CX, [2002H]
BACK: ADD AX, BX
      DEC CX
      JNZ BACK
MOV [2004H], AX
HLT
```

IV. Conclusion:

For objective 1

It can be concluded that the sum and average of N numbers when dry run and executed in system found to be same. Thus the program to find the sum and average of N numbers was executed.

For objective 2

It can be concluded that in order to calculate the number of 1's in 8 bit number we can rotate the number 8 times and count the number of times carry flag changed.

Lab4

III. POST LAB:

1. Analyze the following code and find out the value of registers.

```
MOV AX, 4246H
MOV BX, 123FH
AND AX, BX
ADD AX, BX
ROR AX, 02H
INC BX
INC BX
MOV [BX], AX
HLT
```

1. AX = 4246H

BX = 123FH

AX AND BX = 0206H

AX = 0206H

BX = 123FH

AX ADD BX = 1445H

AX = 1445H

ROR AX, 02H ; AX 4511

∴ [BX] = 4511H

Date

Conclusion:

For obj1

It can be concluded to determine the largest number in array when executed in system found to be same. Thus the program to determine the largest number in the array was executed.

For obj2

It can be concluded to determine the smallest number in array when executed in system found to be same. Thus the program to determine the smallest number in array was executed.

For obj3

It can be concluded to determine ascending order of an array when executed in system found to be same. Thus the program to determine the ascending order of an array was executed.

For obj4

It can be concluded to determine descending order of an array when executed in system found to be same. Thus the program to determine the descending order of an array was executed.

Lab 5

Conclusion:

For obj 1

It can be concluded to determine the largest number in array when executed in system found to be same. Thus, the program to determine the largest number in an array was executed.

For obj 2

It can be concluded to determine the smallest number in array when executed in system found to be same. Thus the program to determine the smallest number in array was executed.

For obj 3

It can be concluded to determine ascending order of an array when executed in system found to be same. Thus the program to determine the ascending order of an array was executed.

For obj 4

It can be concluded to determine descending order of an array when executed in system found to be same. Thus the program to determine the descending order of an array was executed.

Lab5

VI. POST LAB:

1. What are the directives available for data declaration in 8086 microprocessor?

Ans: The direction available for data declaration

- DB - define bytes
- DW → define word
- DD → define double
- DQ → Define Quad word
- DT → Define Ten Bytes.

2.State the difference between END, ENDP, ENDS directives.

Ans: END - END of a program
The END directive marks the end of an --- ALP
ENDP - End of procedure
The ENDP directive is used to indicate the end of procedure
ENDS - End of Segment
The END directive is used to indicate the end of Segment.

Lab 5

3.Translate the following C code to assembly language code using 8086 instruction set. Assume that the values of i, A are available in CL,DL. The base address of array RESULT is available in register SI.

```
For(i=0;i<A;i++)
 Result[i] = I * A;
```

Ans:

```
MOV AX, 0000H
MOV DS, AX
MOV SI, 2000H
MOV [SI], CL  ; value of i
MOV DL, 10H   ; value of A.

L1: MUL CL
    DEC CX
    MOV [SI], CL
    INC SI
    JNZ L1
    HLT
```

Lab 6

V.Conclusion:

For obj1

It can be concluded to determine addition of two ~~bit~~ 32 bit nos using load/ store addressing mode when executed in System found to be same. Thus the addition of two 32 bit no was executed.

For obj2

It can be concluded to determine substraction of two 32 bit nos using load/ store addressing mode when executed in System found to be same. Thus the ~~multiplication~~ Subtraction of two 32 bits no was executed.

For obj3

It can be concluded to determine multiplication of two 32 bit nos using load/ store addressing mode when executed in System found to be same. Thus multiplication of two 32 bits no was executed.

For obj4

It can be concluded to determine AND, OR, XOR and NOT on two 32 bits nos using load/ store addressing mode when executed in System found to be same. Thus logical operation of two 32 bit nos was executed.

Lab6

IV. POST LAB:

1. Explain briefly any five arithmetic instructions.

- ADD
  add r_3, r_2, r_1
  → It adds the content of r_2 and r_1 and stores in r_3.
- SUB
  sub r_3, r_2, r_4
  → It subtracts the content of r_2 and r_1 and stores in r_3
- MUL
  → It multiplies the content of r_2 and r_1 and stores in r_3
- RSB
  RSb r_3, r_2, r_1
  → It reverse subtract the content of r_2 and r_1 and stores in r_3
- ADC
  It adds with carry.

2. Which of the following instruction is not valid
   a) MOVN R7.R2
   b) LDR R1,=LABEL

Ans a) MOVN R7. R_2 is not valid.

Conclusion:

For obj 1

It can be concluded to determine the largest number in a given array of size N when executed in system found to be same. Thus the program to find the largest number in a given array of size N was executed.

For obj 2

It can be concluded to determine the smallest number in a given array of size N when executed in system found to be same. Thus the program to find the smallest number in a given array of size N was executed.

For obj 3

It can be concluded to separate even number and odd number in an array of size N when executed in system found to be same. Thus the program to separate them even number and odd numbers in an array of size N was executed.

IV. POST LAB:

1. Explain briefly condition codes (flags) of ARM processor.

Ans

i) N: Negative

The N flag is set by an instruction if the result is negative. In practise N is set to the two's complement sign bit of the result (bit 31)

ii) Z: zero

The Z flag is set if the result of the flag setting instruction is zero

iii) C: Carry (for unsigned overflow)

The C flag is set if the result of an unsigned operation overflows the 32 bit instruction result register. This bit can be used to implement 64 bit unsigned arithmetic

iv) V (signed) overflow:-

The V flag works the same as the C flag but for signed operations.

2. Which condition codes (flags) is considered for the following branch instructions?

a) B Label

b) BEQ label

c) BLT label

Ans

a) The branch instruction has no condition code

b) The condition code used here is EQ, which checks if Z flags its equals to zero or not.

c) The condition code used here is LT, which checks if the above comparision is signed less than or not by seeing if $N \neq V$