Computer Organization and Microprocessor

LAB RECORD

For CSE/IT/CSIT

Prepared by

Mrs. G. Anitha
Assistant Professor
Department of CSE
Marri Laxman Reddy Institute of Technology and Management
Hyderabad - 500043

Mr. M. Rama Krishna
Assistant Professor
Department of CSE
Marri Laxman Reddy Institute of Technology and Management
Hyderabad - 500043

MARRI LAXMAN REDDY
Institute of Technology & Management
(Autonomous)
(Approved by AICTE-New Delhi, Accredited by NAAC with 'A' & Affiliated to JNTU, Hyderabad)
Recognised Under Section 2(f) & 12(B) of the UGC act, 1956
Dundigal, Quthbullapur (M), Hyderabad - 500043

Programme Educational Objectives (PEO's)

Civil Engineering	
PEO1:	Solving civil engineering problems in different circumstances.
PEO2:	Pursue higher education and research for professional development.
PEO3:	Inculcate qualities of leadership for technology innovation and entrepreneurship.
Computer Science and Engineering	
PEO1:	Establish a successful professional career in industry, government or academia.
PEO2:	Gain multidisciplinary knowledge providing a sustainable competitive edge in higher studies or Research.
PEO3:	Promote design, analyze, and exhibit of products, through strong communication, leadership and ethical skills, to succeed an entrepreneurial.
Electrical and Electronics Engineering	
PEO1:	Graduate will excel with a sound foundation in engineering fundamentals, to resolve the real time problems through technical knowledge and skills.
PEO2:	An Equip graduates to apply their technical knowledge and demonstrate the society with various trends in electrical engineering.
PEO3:	Build prospective career with effective communication skills, leadership qualities and team work with multi – disciplinary approach.
PEO4:	To Inculcate the need based requirement to the society blended with ethics and professionalism.

Programme Educational Objectives (PEO's)

Electronics and Communication Engineering	
PEO 1:	Have successful careers in Industry.
PEO 2:	Show excellence in higher studies/ Research.
PEO 3:	Show good competency towards Entrepreneurship.
Information Technology	
PEO1:	Establish a successful professional career in industry, government or academia.
PEO2:	Gain multidisciplinary knowledge providing a sustainable competitive edge in higher studies or Research.
PEO3:	Promote design, analyze, and exhibit of products, through strong communication, leadership and ethical skills, to succeed an entrepreneurial.
Mechanical Engineering	
PEO1:	Graduates shall emerge as successful Mechanical engineer's as their career progress
PEO2:	Graduates apply fundamentals of engineering, in practical applications and engage in active research.
PEO3:	Mechanical Graduates shall have the ability to design products with interdisciplinary skills.
PEO4:	Graduates will serve the society with their professional skills

II. PROGRAMME OUTCOMES (PO's)

PO1	**Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO2	**Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO3	**Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
PO4	**Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO5	**Modern tool usage :**Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
PO6	**The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
PO7	**Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of,and need for sustainable development.
PO8	**Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO9	**Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO10	**Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO11	**Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO12	**Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

MARRI LAXMAN REDDY
Institute of Technology & Management
(Autonomous)
(Approved by AICTE-New Delhi, Accredited by NAAC with 'A' & Affiliated to JNTU, Hyderabad)
Recognised Under Section 2(f) & 12(B) of the UGC act, 1956
Dundigal, Quthbullapur (M), Hyderabad - 500043

COURSE STRUCTURE, OBJECTIVES

COURSE STRUCTURE

Computer Organization & Microprocessor lab will have a continuous evaluation, during second semester of second year for 30 internal marks and 70 external examination marks.

Out of the 30 marks for internal evaluation, day-to-day work in the laboratory will be evaluatedfor 15 marks and internal practical examination shall be evaluated for 15 marks conducted by theconcerned faculty.

The end semester examination will be conducted with an external examiner and internal examiner. The external examiner will be appointed by the Principal.

COURSE OBJECTIVES

- To Work with 8086 microprocessor programs using MASAM software
- To understand the representation of data at the machine level and how computations are performed at machine level.
- To Work with machine level language programs.

Course Outcomes (CO's)

CO1: Understand apply the MASM software

CO2: Understand the representation of data at the machine level and how computations are performed at machine level.

CO3: Understand Work with machine level language programs.

CO4: Develop Applications such as: 8-bit Addition, Multiplication, Division, array operations, swapping, negative and positive numbers

Course Outcomes (CO's) – Program Outcomes (PO's) Mapping

CO's / Po's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	-	-	2	-	-	-	-	1	-	-
CO2	-	-	-	-	2	-	1	-	-	-	1	2
CO3	-	-	-	2	3	-	2	-	-	-	-	-
CO4	-	-	-	-	-	-	2	-	-	-	-	-
CO5	-	2	-	-	-	-	-	-	-	-	-	-
CO6	1	-	-	-	3	-	-	1	-	-	2	1

Simple -1 **Moderate - 2** **High - 3**

Content of Lab Experiments

MICROPROCESSOR

EXPERIMENT-1

ARITHMETIC OPERATIONS

Take the input in consecutive memory locations and results also Display the results by using "int xx" of 8086. Validate program for the boundary conditions.

Aim: To write ALP to evaluate the following expressions using

a) 8-bit,16-bit,32-bit
b) Considering 2 digit,4 digit and 8 digit BCD numbers

Expressions: i)a=b+c-d*e
ii)z=x*y+w-v+u/k

ALGORITHM:

FOR 8-BIT:

Step 1: Initialize code segment and data segment
Step 2: Initialize variable in data segment
Variable data size data
b DB 07H
c DB 06H
d DB03H
e DB 05H
RESULT DB ?
Step 3: end data segment
Step 4: start code segment
Begin code segment
MOVE E AX,DATA
MOVE DS,AX
MOVE AL,b
MOVE BL,c
ADDTION AL,BL
MOVE BL,d
SUBTRACT AL,BL
MOVE E BL,e
MULTIPLY BL
MOVE DI,OFFSET RESULT
MOVE [DI],AL
INT 21H
STEP 5: end code segment
STEP 6: END begin statement
STEP 7: END of the program

16-BIT:

Step 1: Initialize code segment and data segment

Step 2: Initialize variable in data segment

Variable data size data

```
a DW 0002H
c DW 000AH
d DW 001BH
e DW 002CH
RESULT  DW  ?
```

Step 3: end data segment

Step 4: start code segment

Begin code segment

```
START:
        MOVE AX,DATA
        MOVE DS,AX
        MOVE AX,b
        MOVE BX,c
        ADDITION AX,BX
        MOVE BX,d
        SUBTRACT AX,BX
        MOVE BX,e
        MULTIPLY BX
        MOVE DI,OFFSET RESULT
        MOVE [DI],AX
        INT 21H
```

STEP 5: end code segment

STEP 6:END begin statement

STEP 7:END of the program

ii) 8-bit:

Step 1: Initialize code segment and data segment

Step 2: Initialize variable in data segment

Variable data size data

```
x  DB 07H
y  DB 06H
w  DB03H
v  DB 05H
u DB 04H
k DB 08H
Z  DB  ?
```

Step 3: end data segment

Step 4: start code segment

Begin code segment

```
START:      MOVE AX,DATA
            MOVE DS,AX
            MOVE AL,x
            MOVE BL,y
            MULTIPLY BL
            MOVE BL,w
            ADD AL,BL
            MOVE BL,v
            SUBTRACT AL, BL
```

```
                MOVE BL,u
                ADDITION AL,BL
                MOVE BL,k
                DIVIDE BL
                MOVE DI,OFFSET  z
                MOVE [DI],AL
                INT 21H
```
STEP 5: end code segment
STEP 6:END begin statement
STEP 7:END of the program

16-bit:

Step 1: Initialize code segment and data segment
Step 2: Initialize variable in data segment
Variable data size data

```
ASSUME CS:CODE,DS:DATA
DATA SEGMENT
x DW 0002H
y DW 000AH
w DW 001BH
v  DW 002CH
u DB 0041H
z  DW  ?
```

Step 3: end data segment
Step 4: start code segment
Begin code segment

```
START:      MOVE AX,DATA
            MOVE DS,AX
            MOVE AX,x
            MOVE BX,y
            MULTIPLY BX
            MOVE BX,w
            ADDITION AX,BX
            MOVE BX,v
            SUBTRACT AX, BX
            MOVE BX,u
            ADDITON AX,BX
            MOVE BX,k
            DIVIDE BX
            MOVE DI,OFFSET z
            MOVE [DI],AX
            INT 21H
```

STEP 5: end code segment
STEP 6:END begin statement
STEP 7:END of the program

Viva Questions:

1. How many instructions can be executed per second in 8086/8088?
2. What are the features of Intel 8086?

3. What is Logical Address?
4. What is Effective Address?
5. What is data and address size in 8086?
6. Write the flags in 8086?
7. What is the function of BIU?
8. What is the function of CU?
9. What is the size of instruction queue in 8086?
10. What are the Interrupts of 8086

EXPERIMENT NO: 2

SORTING AN ARRAY

Aim: Write an ALP to sort the given 16-bit numbers in ascending and descending order

ALGORITHAM

a) ASCENDINGORDER

Step 1: Initialize code segment and data segment
Step 2: Initialize variable in data segment
Variable data size data
LIST DW 0125H, 0144H, 3001H, 0003H, 0002H
COUNT EQU 05H
Step 3: end data segment
Step 4: start code segment
Begin code segment

```
START:  MOVE        AX,DATA
        MOVE        DS,AX
        MOVE        DX,COUNT-1
BACK:   MOVE        CX,DX
        MOVE        SI,OFFSET LIST
AGAIN:  MOVE        AX,[SI]
        CMPARE      AX,[SI+2]
        JUMPCARRY   GO
        EXCHANGE    AX,[SI+2]
        EXCHANGE    AX,[SI]
GO:     INCREMENT   SI
        INCREMENT   SI
        LOOP        AGAIN
        DECREMENT   DX
                    JNZ     BACK

                INT     03H
```

STEP 5: end code segment
STEP 6:END begin statement
STEP 7:END of the program

a) DESCENDINGORDER

Step 1: Initialize code segment and data segment
Step 2: Initialize variable in data segment

Variable data size data

LIST DW 0125H, 0144H, 3001H, 0003H, 0002H

COUNT EQUAL 05H

Step 3: end data segment
Step 4: start code segment

Begin code segment

START: MOVE EAX,DATA

MOVE E DS, AX

MOVE E DX,COUNT-1

BACK: MOVE E CX,DX

MOVE E SI, OFFSET LIST

AGAIN: MOVE E AX,[SI]

CMP AX,[SI+2] JNC GO

XCHG AX, [SI+2] XCHG

AX, [SI]

GO: INC SI INC SI

LOOP AGAIN

DEC DX

JNZ BACK

INT 03H

STEP 5: end code segment
STEP 6: END begin statement
STEP 7: END of the program

VIVA QUESTIONS:

1. How many address lines are there in 8086 micro Processor?
2. Give the number of flags in 8086 micro Processor?
3. How to divide 16 bit data by 16 bit data using DIV instruction?
4. What is the use of INT 03instruction?
5. Which other registers can be used with DIV instruction?
6. What is difference between instructions MUL and IMUL?
7. What is difference between instructions DIV and IDIV?
8. What is difference between JMP and CALL Instruction?
9. How clock signal is generated in 8086?
10. What are special registers available in 8086?

EXPERIMENT NO:3
MIN, MAX&AVERAGE

Aim: Write an ALP to 8086 to take N numbers as input and do the following

I) Maximum and minimum
II) Average

ALP:

```
Step 1: Initialize code segment and data segment
Step 2:  Initialize variable in data segment
                    Variable data size data
          LIST DW 0012H,0011H,0009H
         MOVE CX,COUNT-1
        MOVE AX,[SI]
Step 3: end data segment
Step 4: start code segment
          Begin code segment
   AGAIN:   CMPARE AX,[SI+2]
            JUMPLABLE GO
            MOVE AX,[SI+2]
GO:          ADDITION SI,02H
            DECREMENT CX
             JUMP NON ZERO AGAIN
              MOVE [SI+2],AX
               INT 03H
        STEP 5: end code segment
STEP 6: END begin statement
STEP 7: END of the program
```

II)MAXIMUM:

```
Step 1: Initialize code segment and data segment
Step 2:  Initialize variable in data segment
                    Variable data size data
          LIST DW 0012H,0011H,0009H
         COUNT DW 03H
Step 3: end data segment
Step 4: start code segment
          Begin code segment
  START:MOVE AX,DATA
  MOVE DS,AX
  MOVE SI,OFFSET LIST
           MOVE CX,COUNT
                MOVE AX,[SI]
AGAIN:          CMPARE AX,[SI+2]
                JUMPNOTLESSTHAN GO
                MOVE AX,[SI+2]
GO:            ADDITION SI,02H
               DECREMENT CX
```

JUMPNONZERO AGAIN
MOVE [SI],AX
INT 03H

STEP 5: end code segment
STEP 6: END begin statement
STEP 7: END of the program

II) AVERAGE:
Step 1: Initialize code segment and data segment
Step 2: Initialize variable in data segment
Variable data size data
A DB 1,2,3,4,5,6,7,8,9,10
SUM DB ?
Step 3: end data segment
Step 4: start code segment
Begin code segment
START: MOVE AX,DATA
MOVE DS,AX
LOAD EFFECTIVE ADDRESS BX,A
MOVE CL,10
MOVE AX,0000
L1: ADDITION AL,BYTE PTR[BX]
INCREMENT BX
DECREMENT CL
CMPARE CL,00
JUMP NOT ZERO L1
MOVE SUM,AL
MOVE BH,10
DIVIDE BH
MOVE AH,4CH
INT 21H
STEP 5: end code segment
STEP 6: END begin statement
STEP 7: END of the program

viva question

1. What are the flags in 8086?
2. Which flags are called as conditional flags and control flags?
3. What is the difference between 8086 and 8086?
4. Give example for Non-Maskable interrupts?
5. What is meant by a bus?
6.What are the various registers in 8085?
7.Name 5 different addressing modes?
8.What is Program counter?
9.What is the RST for the TRAP?
10. What is Stack Pointer?

Experiment No. 4

Aim: Write an ALP program to print the Fibonacci series

ALP:

Step 1: Initialize code segment and data segment
Step 2: Initialize variable in data segment
Variable data size data

```
DATA1 DB 08H
```

Step 3: end data segment
Step 4: start code segment

```
Begin code segment
        MOVE AX,DATA
        MOVE DS, AX
        MOVE SI, OFFSET DATA1
        MOVE CL,[SI]
        INCREMENT SI
        MOVE AL,00H
        MOVE [SI],AL
        INCREMENT SI
        INCREMENT AL
        MOVE [SI],AL
        SUBTRACT CL,02H
      LABEL G: DECREMENT SI
      MOVE AL,[SI]
      INCREMENT SI
        MOVE BL,[SI]
        INCREMENT SI
        MOVE BL,[SI]
         ADDITION AL,BL
      INCREMENT SI
      MOVE [SI],AL
      LOOP LABEL G
        CALL INTERRUPT      03H
```

STEP 5: end code segment
STEP 6: END begin statement
STEP 7: END of the program

Experiment No. 5

Aim: Write an ALP Program to find even or odd number using macros.

ALP:

Step 1: Initialize code segment and data segment
Step 2: Initialize variable in data segment
Variable data size data

```
MESSAGE DB 10,13,'ENTER A NUMBER = $'
MESSAGE1 DB 10,13,'NUMBER IS EVEN $'
MESSAGE2 DB 10,13,'NUMBER IS ODD $'
```

Step 3: end data segment
Step 4: start code segment
Begin code segment

```
MOVE BX,DATA
MOVE DS,BX

PRINT MACRO MESSAGE
LOAD EA DX,MESSAGE
MOVE AH,09H
CALL INTERRUPT 21H
```

Step 5: end Macro

```
PRINT Message
MOVE AH,01H
CALL INTERRUPT 21H

SAR AL,01
JUMP CARRY ODD

PRINT MESSAGE1

JUMP TERMINATE

ODD:
PRINT MESSAGE2

TERMINATE:
MOVE AH,4CH
CALL INTERRUPT 21H
```

STEP 5: end code segment
STEP 6: END begin statement
STEP 7: END of the program

Experiment No. 6

Aim: Write a simple program in ALP using procedures with arguments.

ALP:

Step 1: Initialize code segment and data segment

Step 2: Initialize variable in data segment

Variable data size data

```
DATA1 DB 08H
```

Experiment No. 7

Experiment No. 8

I) LENGTH OF STRING
II) STRING REVERSAL AND PALINDROME

Aim: To write an ALP of 8086 to take a string as input (in 'C' format) and do the following

1. Length of the given string.
2. Palindrome.

Algoritham

Step 1: Initialize code segment and data segment
Step 2: Initialize variable in data segment

Variable data size data
STR DB 01H, 03H, 08H, 09H, 05H,

07H, 02H LENGTH DB?

Step 3: end data segment
Step 4: start code segment

```
START:  MOVE            AX, DATA
        MOVE            DS, AX
        MOVE            AL, 00H
        MOVE            CL, 00H
        MOVE            SI, OFFSET STR
BACK:   CMPARE          AL, [SI]
        JUMP NO CARRY   GO
        INCREMENT       CL
        INCREMENT       SI
        JUMP NON ZERO   BACK
GO:     MOVE            LENGTH, CL
        INT             03H
```

STEP 5: end code segment
STEP 6: END begin statement
STEP 7: END of the program

Viva-Voce Questions:

1. Which instruction is used for indicating the direction for string operations?
2. What is a linker?
3. How an XCHG instruction works?

4. What is the role of REP?

5. What is the difference between base address and offset address of a word?

II) STRING REVERSAL AND PALINDROME

Aim: To write and execute an ALP to 8086 processor to reverse the given string and verify whether it is a palindrome.

ALGORITHM:

Step 1: Initialize code segment and data segment

Step 2: Initialize variable in data segment

Variable data size data

```
CR EQU 0DH LF EQU 0AH
INS DB 40 DUP (0)
RES DB 40 DUP (0)
MSG1 DB 'ENTER THE STRING.INPUT STRING= ','$'
MSG2 DB CR, LF,'REVERSE OF STRING= ','$'
MSG3 DB CR, LF, 'INPUT STRING IS PALINDROME','$'
MSG4 DB CR, LF, 'INPUT STRING IS NOT A PALINDROME','$'
```

Step 3: end data segment

```
DISPLAY MACRO MSG
        MOVE AH, 09H
        MOVE DX, OFFSET
        MSG INT 21H
END MACRO
```

Step 4: start code segment

```
START:   MOVE AX,DATA
         MOVE DS, AX
         MOVE SI, OFFSET
         INSERT MOV DI,
         OFFSET RES DISP
         MSG1 MOVE CX, 00H
RDCHAR:  MOVE AH,01H
         INTEREPT 21H
         CMPARE AL, CR
         JUMP EQUAL AHEAD
         MOVE [SI], AL
         INCREMENT SI
         INCREMENT CX
         JUMP
RDCHAR AHEAD:
      MOVE BX, CX
REVERSE: DECREMENT
SI
         MOVE AL, [SI]
         MOVE [DI], AL
         INCREMENT DI
         LOOP REVERSE
```

MOVE AL, '$'
MOVE [DI], AL
DISPLAY MSG2
DISPLAY RES
MOVE SI, OFFSET INS
MOVE DI, OFFSET RES
MOVE CX, BX

CHECK: MOVE AL,[SI]
CMPARE AL, [DI]
JUMP NOT EQUAL FALSE INC SI
INCREMENT DI
LOOP CHECK DISPLAY MSG3
JUMP EXIT

FALSE: DISPLAY MSG4

EXIT: INT03H

STEP 5: end code segment
STEP 6: END begin statement
STEP 7: END of the program

Viva questions

1. What happens when HLT instruction is executed in processor?
2. How many interrupts are there in 8085?
3.Which interrupt has the highest priority?
4**.What are Hardware interrupts?**
5.What is a segment in memory?
6.Define code segment
7.Define data segment.
8.Define extra segment
9. What is a Microprocessor?
10.What is Program counter?

EXPERIMENT NO-9

Aim: To write an ALP to 8086 to take a string of as input (in 'C' format) and do the following.

1. **W**hether given string is a substring or not.

ALGORITHAM

Step 1: Initialize code segment and data segment
Step 2: Initialize variable in data segment
Variable data size data

```
STR DB 'AXYBCSDEF$'
SUBSTR DB 'BCS$'
LEN1 DB 0
LEN2 DB 0
MSG1 DB 10,13,'STRING IS : $'
MSG2 DB 10,13,'SUBSTRING IS : $'
MSG3 DB 10,13,'SUBSTRING IS FOUND AT POSITION : $'
POS DB -1
RTN DB '-1$'
```

Step-3: END OF THE DATA SEGMENT

STEP-4: MACRO INITILAZATION
DISPLAY MACRO MACRONAME
MOVE AH,9
LOAD EFFECTIVE ADDRESS DX,MSG
INTERRUPT 21H
STEP-4 : END OF THE MACRO

STEP-5: START CODE SEGMENT
START:
MOVE AX,DATA
MOVE DS,AX

DISPLAY MACRO MESSAGE MSG1
DISPLAY MACRO MESSAGE STR
DISPLAY MESSAGE MSG2
DISPLAY MESSAGE SUBSTR

LOAD EFFECTIVE ADDRESS SI,STR
LABEL :NEXT1:
COMPARE [SI],'$'
JUMPEQUAL DONE1
INCREMENT LEN1
INCREMENT SI
JUMP LABLE NXT1
LABLE DONE1:
LOAD EFFECTIVE ADDRESS DI,SUBSTR

LABEL NXT2:
COMPARE [DI],'$'
JUMPEQUAL DONE2
INCREMENT LEN2
INCREMENT DI
JUMP LABEL NXT2
DONE2:
DISPLAY MSG3

LOAD EFFECTIVE ADDRESS SI,STR
MOVE AL,LEN1
SUBTRACT AL,LEN2
MOVE CL,AL
MOVE CH,0
FIRST:
INCREMENT POS
MOVE AL,[SI]
COMPARE AL,SUBSTR[0]
JUMPEQUAL CMPR
INCREMENT SI
LOOP FIRST

COMPARE: INCREMENT SI
MOVE AL,[SI]
COMPARE AL,SUBSTR[1]
JUMPNOTEQUAL NOTEQUAL
INCREMENT SI
MOVE AL,[SI]
COMPARE AL,SUBSTR[2]
JUMPEQUL EQUAL

LABEL NOTEQUAL:
MOVE POS,-1
DISPLAY RTN
JUMP EXIT

LABEL EQUAL:
MOVE DL,POS
ADDITION DL,30H
MOVE AH,2
INTERREPT 21H

LABLE: EXIT: MOVE AH,4CH
INTERRUPT 21H
STEP 6: end code segment

STEP 7: end start

Viva-Voce Questions:

1. Explain the Logic in your program?
2. Explain about LOOP statement?
3. .Explain Dup?
4. Define Variable?
5. What is the main use of ready pin?
6. What is a macro?
7. What is the difference between macro and procedure?
8. Define bit, byte and word.
9. Explain the use of INT 0 and INT 4.
10. Classify the assembler directives available in 8086.

EXPERIMENT NO-10

CALLING THE FUNCTIONS FROM MAIN FUNCTION

Aim: To write an ALP to 8086 to implement the functions using call function.

ALGORITHAM

Step 1: Initialize code segment and data segment

Step 2: Initialize variable in data segment

Variable data size data

```
NUM DB 0FFH
RES DB 10 DUP ('$')
```

Step 3: end data segment

Step 4: start code segment

```
START:MOVE AX,DATA
      MOVE DS,AX
      MOVE AH,0
      MOVE AL,NUM
      LOADEFFECTIVEADDRESS SI,RES
      CALL HEX2DEC        LOADEFFECIVEADDRESS
      DX,RES
      MOVE AH,9
      INTERRUPT 21H
    MOVE AH,4CH
    INTERRUPT 21H
```

Step 5: end of code segment

Step 6: start procedure

```
HEX2DEC PROC NEAR
MOVE CX,0
    MOVE BX,10
    LABLE-LOOP1: MOVE DX,0
    DIVIDE BX
    ADDITION DL,30H
    PUSH DX
    INCREMENT CX
    CAMPARE AX,9
    JUMP IF GRATER  LABLEL LOOP1
    ADDITION AL,30H
    MOVE [SI],AL
LABEL 2: LOOP2: POP AX
        INCREMENT  SI
        MOVE [SI],AL
        LOOP 2
```

RET HEX2DEC

STEP-7: END OF THE PROCEDURE

STEP-8 END OF THE PROGRAM

Viva-Voce Questions:

1. What is the logic in your program?
2. What are the registers used in your program?
3. What are assembler directives?
4. Name the arithmetic instructions which won't affect CY flag.
5. Discuss the syntax of procedure w.r.t assembler and w.r.t processor.
6. Discuss the syntax of macro.
7. Discuss the use of following interrupts: INT0, INT1, INT2, INT3
8. What is meant by bootstrap loader
9. Explain the use of AAA and AAM instructions?
10. What is the use of CBW and CWD instructions?

EXPERIMENT NO:11

FACTORIAL

Aim: To write an ALP to 8086 to find out factorial of a given number using GNU ASSEMBLER

ALGORITHAM

Step 1: Initialize code segment and data segment
Step 2: Initialize variable in data segment
Variable data size data
A DB 05H
Step 3: end data segment
Step 4: start code segment
START:
MOVE AX,DATA
MOVE DS,AX
MOVE AH,00H
MOVE AL,A
LABLE- L1: DECREMENT A
MULTIPLY A
MOVE CL,A
COMPSRE CL,01
JUMPNOTZERO L1
MOVE AH,4CH
INTERRUPT 21H

STEP-5: END OF THE START

STEP-6 END OF THE PROGRAM

viva questions

1. What are most common registers present in a microprocessor?
2. Why is address bus unidirectional?
3. Why is data bus bidirectional?
4. What is Program counter?
5. What is meant by a bus?
6 Give an example of one address microprocessor?
7. What are Software interrupts?
8. How many interrupts are there in 8085?
9. In 8086 which is called as High order / Low order Register?
10. What are input & output devices?