MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION, MUMBAI

GOVERNMENT POLYTECHNIC KARAD

SECOND YEAR DIPLOMA COMPUTER ENGINEERING (I-SCHEME)

A

MICRO-PROJECT REPORT

GENERATE FIBONACCI SERIES

UNDER THE SUBJECT

MICROPROCESSOR (22415)

SUBMITTED BY

Roll No	Name	Enrollment Number
1251	Pratik Pramod Shejwal	2100100053
1252	Yash Prashant Deshmukh	2100100054
1261	Riya Sunil Kharade.	2100100063

UNDER THE GUIDANCE OF

Mr.A.S.Chavan

(DEPARTMENT OF COMPUTER ENGINEERING)

2022-23

CERTIFICATE

This is to certify that,

Roll No	Name	Enrollment Number
1251	Pratik Pramod Shejwal	2100100053
1252	Yash Prashant Deshmukh	2100100054
1261	Riya Sunil Kharade.	2100100063

Have successfully completed microproject on **"Generate Fibonacci Series"** Of the subject **"Microprocessor (22415)"** prescribed by Maharashtra state board of technical education for 4th semester [Computer] during the year 2022-23.

Mrs. S. B.Patil
Head of the department.

Mr.A.S.Chavan
(Project Guide)

ACKNOWLEDGEMENT

We take it is an opportunity to thank all those who have directly and indirectly inspired, directed and assisted us towards successful completion of this microproject report.

We express our sincere thanks to the principal **Dr. R. K. Patil** for lecture allowed to submit this report as part of our academic learning our sincere thanks to **Mr.A.S.Chavan** In "Microprocessor(22415)" Department of COMPUTER ENGINEERING, Government Polytechnic Karad for this encouragement throughout this project report and guideline in designing and working out this project.

We are also grateful to learn of "Microprocessor (22415)" for their highly encouraging and co-operative attitude. We express our sense of gratitude towards our friend and parents for their constant moral support during microproject report.

Place: Government Polytechnic Karad
Date: 02/05/2023

Yours Sincerely,

1251 -Pratik Pramod Shejwal
1252 -Yash Prashant Deshmukh
1261 -Riya Sunil Kharade

MICROPROJEC REPORT

"GENERATE FIBONACCI SERIES"

❖ RATIONALE:

Assembly language is a low level programming language .It helps to understanding the programming language to machine code.In computers,there is an assembler that helps in converting the assembly code into machine executable code.Assembly language is designed to understand the instruction and provide it to machine language for further processing.It mainly depends on the architecture of the system,whether it is the operating system or computer architecture.

Assembly language helps programmers to write human-readable code that is almost similar to machine language.Machine language is difficult to understand and read as it is just a serias of numbers.Assembly language helps in providing full control of what tasks a computer is performing.

❖ AIM AND BENEFITS:

1)Understanding the steps of program development and execution.
2)Developing a code using assembly language programming.
3)Using relevant command of debugger to correct the programming errors.
4)To develop ability to generate Fibonacci series using assembly language program.

❖ COURSE OUTCOMES:

CO.b)Write assembly language program for the given problem.
CO.c)Use instruction for different addressing mode.
CO.d)Develop an assembly language program using assembler.

❖ LITERATURE REVIEW:

Assembly Language appeared in 1949 and soon saw wide use in Electronic Delay Storage Automatic Calculators. The Assembly was a low-level computer language that simplified the language of machine code ie. the specific instructions necessary to operate a computer.

The release of Intel's 8086 microprocessor in 1978 was a watershed moment for personal computing. The DNA of that chip is likely at the center of whatever computer—Windows, Mac, or Linux—you're using to read this, and it helped transform Intel from merely one of many chip companies to the world's largest.

❖ DIRECTIVES USED :

- DB (Data Byte):
  -The DB directive is used to declare a BYTE -2-BYTE variable – A BYTE is made up of 8 bits.
- ASSUME:
  -These directives ASSUME informs assembler the name of logical segment that should be used for specified segment.
  -Assume is a Program Organization Directives.
- SEGMENT
  -The SEGMENT directive is used to indicate the beginning of the logical segment.
- ENDS
  -The directive ENDS informs the assembler the end of the segment.
- END
  -The directive END is used to inform assembler the end of the program.

- MOV destination, source :
  -This instruction transfers data from source i.e. register/memory location/immediate data to destination i.e. another register/memory location.
- Instructions to transfer the address :
  -LEA − Used to load the address of operand into the provided register.
- Instructions to perform addition :
  -DAA − Used to adjust the decimal after the addition/subtraction operation.
  -INC − Used to increment the provided byte/word by 1.
- Instructions to perform subtraction :
  -SUB – Used to subtract the byte from byte/word from word.
- Iteration Control Instructions
  -LOOP − Used to loop a group of instructions until the condition satisfies, i.e., CX = 0.

❖ PROPOSED METHODOLOGY:

1)Discussion about given topic.
2)Selection of good leader and distribution of responsibilities.
3)Collection of information using different resources.
4)Analysis of given information.
5)Presentation of given report.
6)Completion and submission of given tasks.

❖ ACTION PLAN:

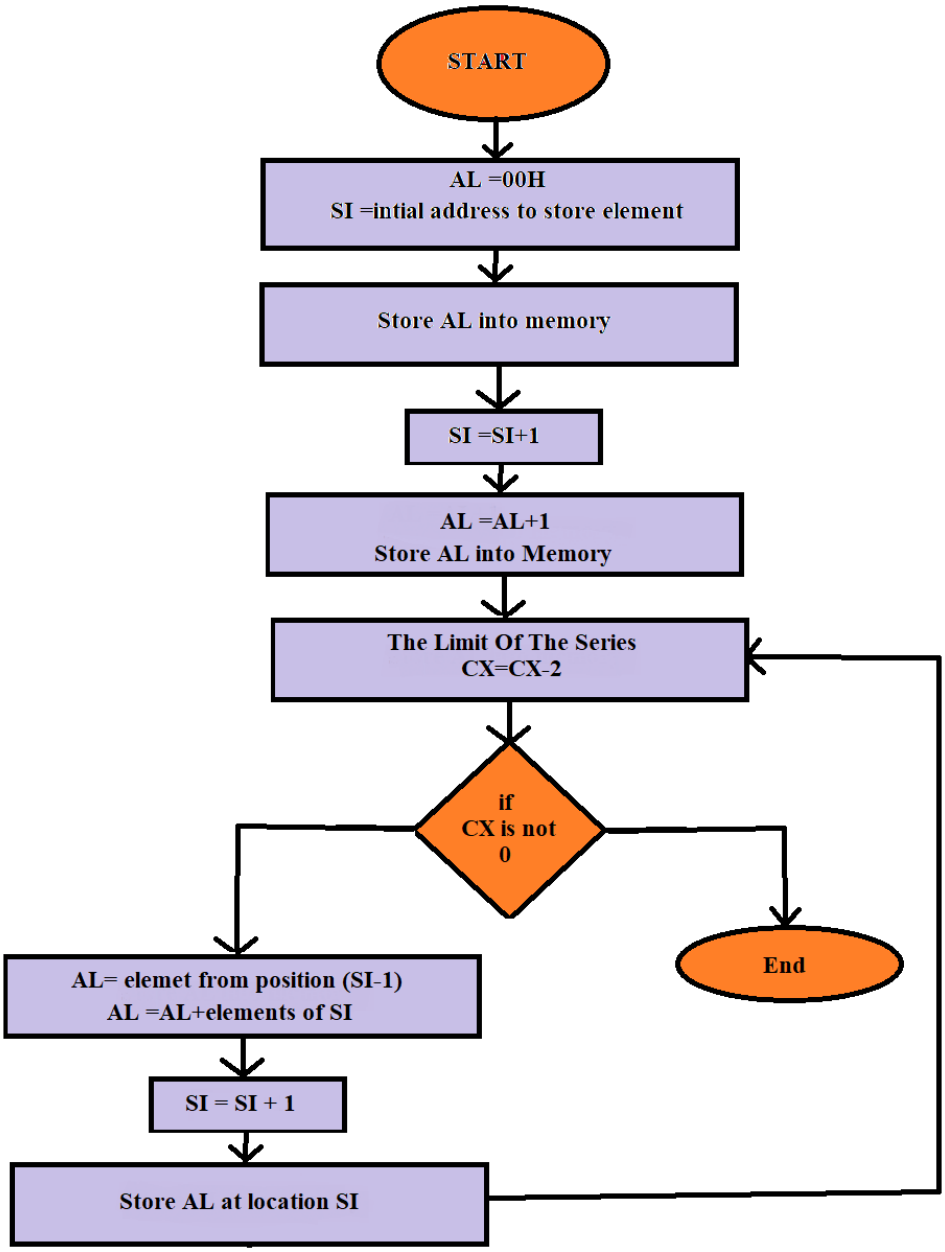
Sr.no	Details of activity	Planned start date	Planned finish date	Name of responsible team member
1.	Discussion And Finalazation Of Topic	21/02/23	23/02/23	All Team
2.	Completion of Proposal	25/02/23	05/03/23	All Team
3.	Preparation By Abstract	07/03/23	13/03/23	All Team
4.	Collection Of Data/Experiment	16/03/23	27/03/23	All Team
5.	Discussion And Outline	28/03/23	30/03/23	All Team
6.	Editing And Proof Reading Of Content	01/04/23	05/04/23	All Team
7.	Completion Of Report And Presentation	07/04/23	10/04/23	All Team
8.	Final Submission Of Microproject	15/04/23	15/04/23	All Team

❖ OUTPUT OF THE MICRO-PROJECT:

- **Algorithm:**

Step 01: Start
Step 02: Initialize Data Segment
Step 03: Load address of fib variable into source index
Step 04: Mov al to the source array
Step 05: increment SI
Step 06: Increment AL
Step 07: Mov al into the source array
Step 08:mov cnt into cl register
Step 09: mov 0 to ch
Step 10: Substract 2 from cx
Step 11: Mov value present at [SI-1]
Step 12: add value present at SI into al and store result into al
Step 13 : daa
Step 14 : Increment SI
Step 15 : Mov al into SI
Step 16 : If cnt<=7
Step 17 : Go to l1
Step 18: Load Address of String into dx
Step 19: Mov cnt into cl
Step 20: Mov offset of fib into SI
Step 21:Mov [SI] into al
Step 22: Add 30 into al
Step 23: Request to display the character
Step 24: Increment SI
Step 25: Go to step 20, If cnt<=7
Step 26: Stop

- **Flowchart:**


START
AL =00H
SI =intial address to store element
Store AL into memory
SI =SI+1
AL =AL+1
Store AL into Memory
The Limit Of The Series
CX=CX-2
if
CX is not
0
End
AL= elemet from position (SI-1)
AL =AL+elements of SI
SI = SI + 1
Store AL at location SI

- **CODE:**

```
data segment
    cnt db 07
    fib db 07 dup (0)
    msg db 'Series is $'
data ends
code segment
    start:assume cs:code,ds:data
    mov ax,data
    mov ds,ax
    mov al,00h
    lea si,fib
    mov [si],al
    inc si
    inc al
    mov [si],al
    mov cl,cnt
    mov ch,00h
    sub cx,02h
    l1:
    mov al,[si-1]
    add al,[si]
    daa
    inc si
    mov [si],al
    loop l1
    mov ah,09h
    lea dx,msg
    int 21h
    mov cl, cnt
    mov si, offset fib
   up:  mov al,[si]
    add al,30h
    mov ah,02h
    mov dl,al
    int 21h
    inc si
    loop up
    mov ah,4ch
    int 21h
code ends
end start
```

❖ **OUTPUT:**

```
C:\TASM>edit f2.asm

C:\TASM>tasm f2.asm
Turbo Assembler  Version 3.0  Copyright (c) 1988, 1991 Borland International

Assembling file:   f2.asm
Error messages:    None
Warning messages:  None
Passes:            1
Remaining memory:  476k


C:\TASM>tlink f2.obj
Turbo Link  Version 2.0  Copyright (c) 1987, 1988 Borland International
Warning: no stack

C:\TASM>f2.exe
Series is 0112358
```

❖ RESOURCES REQUIRED:

Sr.no	Name of Resources	Specification	Quantity
1.	Computer system	Laptop (HP) 8 GB RAM	1
2.	Software	Turbo Assembler(TASM)	-
3.	Ms Word	Microsoft Word 2021 MSO 64 bit	-
4.	Internet	4G,Wi-Fi	-

❖ Skill developed outcomes of the Micro-Project:

a) We acquired knowledge to use assembly language programming tools and functions.
b)We learned to calculate factorial of number using assembly language programming.
c)We understood to draw the flowchart of the program.
d)We learned to make flowchart of the program.
e)We have taken knowledge about leadership.

❖ Application Of The Micro-Project:

Today, assembly language is still used for direct hardware manipulation, access to specialized processor instructions, or to address critical performance issues. Typical uses are device drivers, low-level embedded systems, and real-time systems.
This project will used for calculating the factorial of the number.

❖ GROUP DETAILS:

Sr.no	Roll No	Enrollment No	Name of member
1.	1251	2100100053	Pratik Pramod Shejwal
2.	1252	2100100054	Yash Prashant Deshmukh
3.	1261	2100100063	Riya Sunil Kharade