

EXPERIMENT 05

AIM :-

Program for 16-bit BCD Addition

LO No :- L03

LO : Build a program on a microprocessor using arithmetic & logical instruction set of 8086.

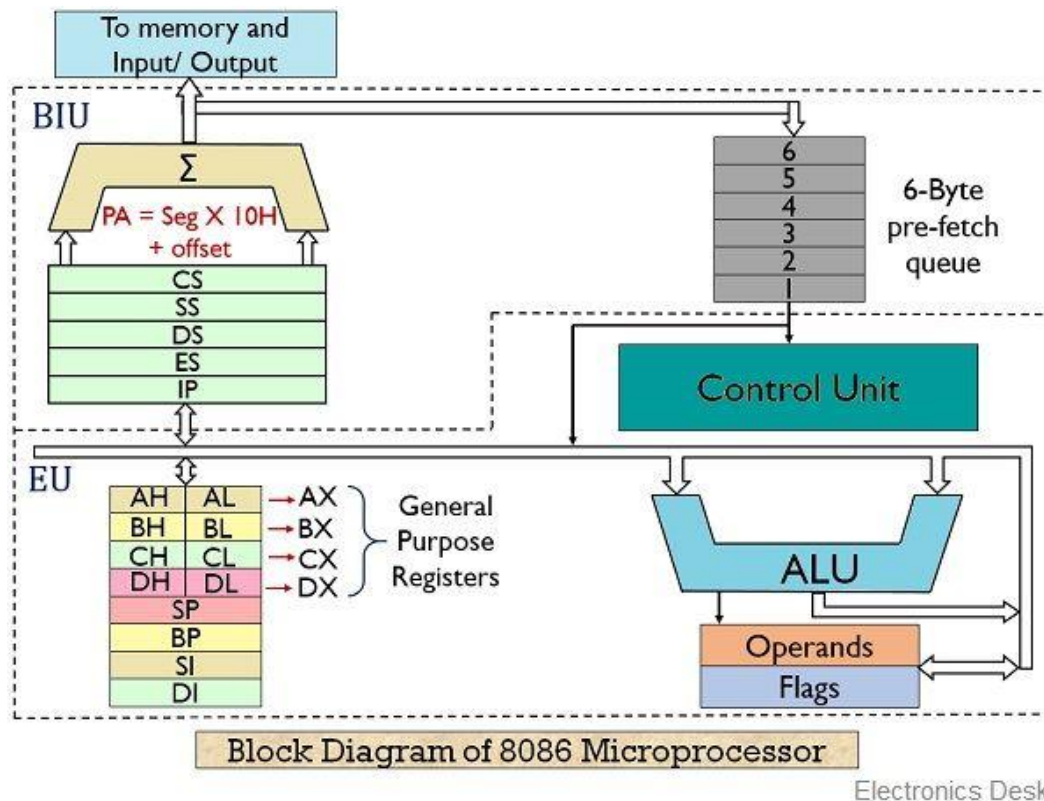
SOFTWARE :- Tasm Software

Theory :-

8086 Microprocessor is an enhanced version of 8085 Microprocessor that was designed by Intel in 1976. It is a 16-bit Microprocessor having 20 address lines and 16 data lines that provides up to 1MB storage. It consists of powerful instruction set, which provides operations like multiplication and division easily.

Features

- It has an instruction queue, which is capable of storing six instruction bytes from the memory resulting in faster processing.
- It was the first 16-bit processor having 16-bit ALU, 16-bit registers, internal data bus, and 16-bit external data bus resulting in faster processing.



MOV

The MOV instruction is the most important command in the 8086 because it moves data from one location to another.

Syntax: Mov source, destination

Example: Mov Ax,1234H

ADD

The ADD instruction performs an addition on both the first source register's contents and the second source. register's contents, and stores the result in the destination register.

Syntax: ADD Source, Destination

Example: Add Ax,Bx

INTERRUPT

int 21h means, call the interrupt handler 0x21 which is the DOS Function dispatcher. the "mov ah,01h" is setting AH with 0x01, which is the Keyboard Input with Echo handler in the interrupt.

Syntax: int 21H

Example: int 21H

Code :-

Assume CS: code , DS: data

Data Segment

n1 dw 1234H

n2 dw 5678H

ans dw ?

data ends

Code Segment

Start: MOV Ax,data

MOV Ds,Ax

MOV Ax,n1

MOV Bx,n2

ADD Ax,Bx

MOV ans,Ax

MOV AH,4CH

INT 21H

code ends

end Start

Output :-

The screenshot displays the MPLAB IDE interface for an 80486 processor. The main window shows assembly code with addresses and instructions. A status window is open, indicating the program has terminated with exit code 172. The code includes instructions like `mov ax, 48AD`, `mov ds, ax`, `mov ax, [0000]`, `mov bx, [0002]`, `add ax, bx`, `mov [0004], ax`, `mov ah, 4C`, and `int 21`. The status window has 'OK' and 'Help' buttons.

Address	Instruction
48AE:0000	<code>mov ax, 48AD</code>
48AE:0003	<code>mov ds, ax</code>
48AE:0005	<code>mov ax, [0000]</code>
48AE:0008	<code>mov bx, [0002]</code>
48AE:000C	<code>add ax, bx</code>
48AE:000E	<code>mov [0004], ax</code>
48AE:0011	<code>mov ah, 4C</code>
48AE:0013	<code>int 21</code>

Terminated, exit code 172

OK Help

Conclusion : We built a program on microprocessor using arithmetic and logical instructions and performed BCD addition on 16-bit values.