*LAB # 06*

*LAB task*

* *Explore all the addressing modes with examples with respect to assembly language*

*Addressing modes in assembly language specify how the processor can access the data required for an instruction. Each mode provides a unique way to locate operands, enhancing programming flexibility. Here’s a detailed overview:*

***1. Immediate Addressing Mode***

* ***Explanation****: In this mode, the operand is directly embedded within the instruction as a constant value.*
* ***Example****:*
* *MOV AL, 5h ; Load the value 5h directly into the AL register.*

***2. Register Addressing Mode***

* ***Explanation****: The operand is stored in a CPU register, and the instruction identifies which register to use.*
* ***Example****:*
* *MOV AX, BX ; Transfer the content of register BX to register AX.*

***3. Direct Addressing Mode***

* ***Explanation****: The instruction contains the memory address where the operand is located.*
* ***Example****:*
* *MOV AX, [1234h] ; Load the value from memory address 1234h into AX.*

***4. Indirect Addressing Mode***

* ***Explanation****: The instruction specifies a register or memory location containing the address of the operand.*
* ***Example****:*
* *MOV AX, [BX] ; Use the address in BX to fetch data and move it into AX.*

***5. Indexed Addressing Mode***

* ***Explanation****: A base address is combined with an offset (index) to access the operand.*
* ***Example****:*
* *MOV AX, [SI+10h] ; Add 10h to the address in SI to locate the data for AX.*

***6. Base-Indexed Addressing Mode***

* ***Explanation****: Combines a base register and an index register to calculate the effective address.*
* ***Example****:*
* *MOV AX, [BX+SI] ; Use the sum of BX and SI as the address to fetch the data for AX.*

***7. Relative Addressing Mode***

* ***Explanation****: The operand’s address is determined by adding a displacement to the current instruction pointer (IP).*
* ***Example****:*
* *JMP SHORT LABEL ; Jump to a location relative to the current IP.*

***8. Implicit Addressing Mode***

* ***Explanation****: The instruction implicitly uses specific registers or memory locations without explicitly mentioning them.*
* ***Example****:*
* *CLC ; Clear the carry flag, which is an implicit operand.*

***Conclusion***

*Addressing modes are essential for efficient programming, as they provide different ways to access operands depending on the situation. By leveraging the appropriate addressing mode, developers can optimize code for speed, memory usage, or simplicity. Understanding these modes is fundamental for working effectively with assembly language.*