Practical No. 5

Aim: Finding the 1’s Complement of an 8-bit Number

Objective: To determine the 1’s complement of an 8-bit number using binary operations.

Theory:

The **1’s complement** of a binary number is obtained by inverting all the bits, meaning:

- **0 becomes 1**
- **1 becomes 0**

For example, if the given 8-bit number is **11010011**, its 1’s complement is **00101100**.

1’s complement is useful in digital arithmetic operations, particularly in subtraction and signed number representation.

Materials/Tools Required:

- Microprocessor/microcontroller (e.g., 8085/8051)
- Assembler/Simulator
- Computer system with programming software
- Binary calculator (optional)

Procedure:

1. **Initialize Registers**: Load the given 8-bit number into a register (e.g., A register in an 8085 processor).
2. **Invert Bits**: Use the **CMA (Complement Accumulator)** instruction in an 8085 processor or manually invert each bit.
3. **Store the Result**: Store the 1’s complement in memory or display it on an output device.

Observations:

- Each bit of the number is flipped (0 → 1, 1 → 0).
- The 1’s complement is a useful intermediate step in the 2’s complement method.
- It is commonly used in binary subtraction and signed number representation.

Conclusion:

The 1’s complement of an 8-bit number is successfully determined by inverting all bits. This process is fundamental in arithmetic computations and digital logic design.

Applications (Optional):

- Used in digital subtraction (as part of the 2’s complement method).
- Essential in binary logic operations and error detection techniques.
- Applied in microprocessors for data manipulation and logical operations.