

EXPERIMENT: 32

IMPLEMENTATION OF BIT STUFFING MECHANISM USING JAVA/C

Aim: To implement Bit stuffing mechanism using C program.

Bit suffering: It is a technique used in communication system to prevent data loss or corruption during transmission.

It involves inserting one or more extra bits into a data packet to differentiate it from the control characters.

Bit suffering is implemented using bitwise operators in c programming language.

In this code, the `bit Stuffing` function takes an input byte array, its length, an output byte array, and a pointer to the output length variable. It performs bit stuffing on the input data and stores the stuffed data in the output array.

The main logic of the bit stuffing is implemented using bitwise operations. The input data is processed byte by byte, and each bit is checked for consecutive 1's. If five consecutive 1's are found, a 0 bit is stuffed into the output frame. The flag sequence (01111110) is added at the beginning and end of the output frame.

In the `main` function, an example input frame is provided, and the bit stuffing is performed by calling the `bit Stuffing` function. The input and output frames are then printed for verification.

Note that in this example, the input frame is hard-coded, and the output frame is printed in hexadecimal format for better readability. You can modify the input frame and test the code with different inputs.

```
Output
Enter number of bits: 12
Enter bit stream: 0 1 0 1 1 1 1 1 0 0 1
Stuffed Bit Stream: 0101111101001

=== Code Execution Successful ===
```

Result: Therefore bit suffering mechanism has been successfully implemented using c program.