

Modern Education Society's  
Wadia College of Engineering, Pune

NAME OF STUDENT:	CLASS:
SEMESTER/YEAR:	ROLL NO:
DATE OF PERFORMANCE:	DATE OF SUBMISSION:
EXAMINED BY:	EXPERIMENT NO:

**Assignment No. 3(Group - A)**

Title: To demonstrate error correction and detection using CRC or Hamming Code

Objectives : To implement error detection and correction techniques

Problem Statement: Write a program for error detection and correction for 7/8 bits ASCII codes using Hamming Codes or CRC.

Outcome : Demonstrate Hamming Codes or CRC with example.

Software Requirements :Jdk

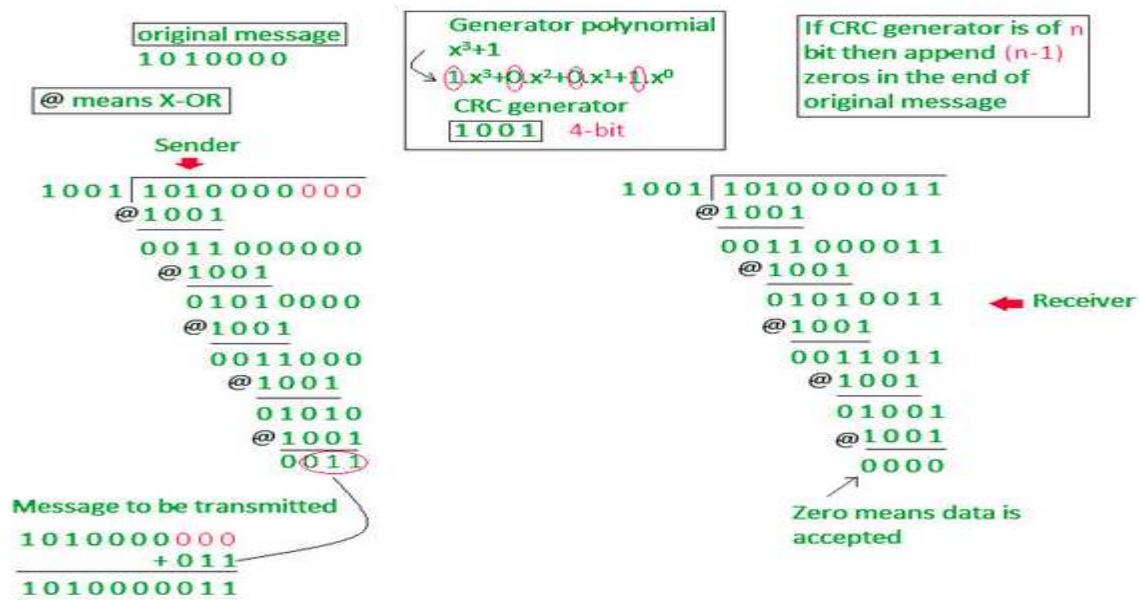
Hardware Requirements :Open source linux operating system.

**THEORY:**

Cyclic Redundancy Check: CRC

- Given a k-bit frame or message, the transmitter generates an n-bit sequence, known as *aframe check sequence (FCS)*,so that the resulting frame, consisting of (k+n) bits, is exactly divisible by some predetermined number.
- The receiver then divides the incoming frame by the same number and, if there is no remainder, assumes that there was no error.

Example:



Hamming code

- Hamming codes are a family of [linear error-correcting codes](#) that generalize the [Hamming\(7,4\)-code](#)
- Invented by [Richard Hamming](#) in 1950

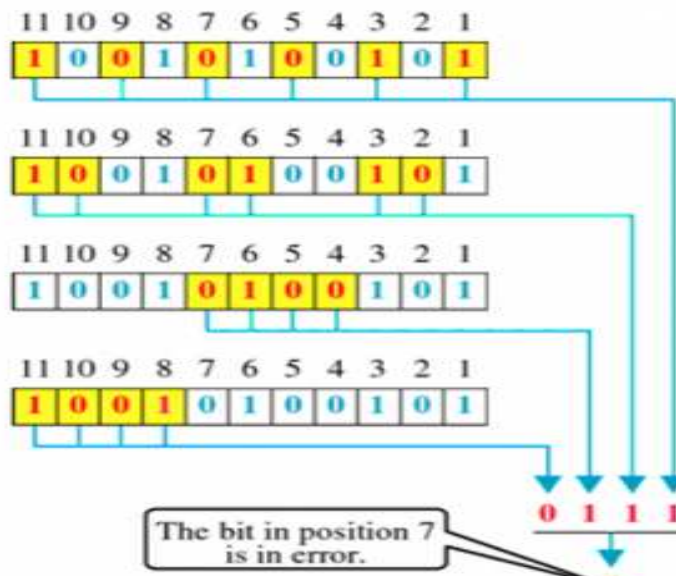
Hamming codes can detect up to two-bit errors or correct one-bit errors without detection of uncorrected errors.

General algorithm

- The following general algorithm generates a single-error correcting (SEC) code for any number of bits.
- Number the bits starting from 1: bit 1, 2, 3, 4, 5, etc.
- Write the bit numbers in binary: 1, 10, 11, 100, 101, etc.
- All bit positions that are powers of two (have only one 1 bit in the binary form of their position) are parity bits: 1, 2, 4, 8, etc. (1, 10, 100, 1000)
- All other bit positions, with two or more 1 bits in the binary form of their position, are data bits.
- Each data bit is included in a unique set of 2 or more parity bits, as determined by the binary form of its bit position. Each data bit is included in a unique set of 2 or more parity bits, as determined by the binary form of its bit position.
- Parity bit 1 covers all bit positions which have the least significant bit set: bit 1 (the parity bit itself), 3, 5, 7, 9, etc.

- Parity bit 2 covers all bit positions which have the second least significant bit set: bit 2 (the parity bit itself), 3, 6, 7, 10, 11, etc.
- Parity bit 4 covers all bit positions which have the third least significant bit set: bits 4–7, 12–15, 20–23, etc.
- Parity bit 8 covers all bit positions which have the fourth least significant bit set: bits 8–15, 24–31, 40–47, etc.
- In general each parity bit covers all bits where the bitwise AND of the parity position and the bit position is non-zero.
- Each data bit is included in a unique set of 2 or more parity bits, as determined by the binary form of its bit position.
- Parity bit 1 covers all bit positions which have the least significant bit set: bit 1 (the parity bit itself), 3, 5, 7, 9, etc.
- Parity bit 2 covers all bit positions which have the second least significant bit set: bit 2 (the parity bit itself), 3, 6, 7, 10, 11, etc.
- Parity bit 4 covers all bit positions which have the third least significant bit set: bits 4–7, 12–15, 20–23, etc.
- Parity bit 8 covers all bit positions which have the fourth least significant bit set: bits 8–15, 24–31, 40–47, etc.
- In general each parity bit covers all bits where the bitwise AND of the parity position and the bit position is non-zero.

Example:-



Conclusion:

Hence we have implemented Erroro Detection and correction method.

Questions

1)What is CRC? Explain CRC generator and checker with example.

2)What is hamming code? Generate hamming code for 7/8 bit data word.

3) Explain checksum in detail