

Assignment-02

Page No.

Date

- Title :- Cyclic Redundancy Check (CRC)
- Problem Statement :- Write a program in C/C++ for error detection & correction for 7/8 bits ASCII codes using Hamming codes or CRC. Demonstrate the packets using Wireshark packet analyzer tool for peer to peer mode.
- Software & Hardware PC, 64 bit Open Source OS, Wireshark Packet analyzer tool
- Related Theory

1. Cyclic Redundancy Check (CRC)

- CRC is a different approach to detect if the received frame contains valid data.
- This technique involves binary division of the data bits being sent.
- The divisor is generated using polynomial.
- The sender performs a division operation on the bits being sent & calculates the remainder.
- Before sending the actual bits, the sender adds the remainder at the end of the actual bits. Actual data bits plus the remainder is called as codeword. The sender transmits data bits as codewords.

Eg:

Data bits : 1 1 0 0 1

Divisor : 1 0 1

at Sender's side

$$\begin{array}{r}
 101 \overline{) 11001} \\
 \underline{101} \\
 110 \\
 \underline{101} \\
 111 \\
 \underline{101} \\
 10 \leftarrow \text{CRC}
 \end{array}$$

CRC bit : 10

 \therefore Codeword : 1 1 0 0 1 1 0At Receiver side :

$$\begin{array}{r}
 101 \overline{) 1100110} \\
 \underline{101} \\
 110 \\
 \underline{101} \\
 111 \\
 \underline{101} \\
 101 \\
 \underline{101} \\
 000 \rightarrow \text{No Error.}
 \end{array}$$

\rightarrow At other end the receiver performs division operation on Codewords using the same CRC divisor.

→ If remainder contains all zeroes the data bits are accepted, otherwise it is considered as there some data corruption occurred in transit.

• Atto Algorithms

1) Consider

n → number of bits in data to be sent from sender side.

k → number of bits in the key obtained from generator polynomial.

2) At sender side (Generation of Codeword)

(i) The binary data is first augmented by adding $k-1$ zeroes in the end of the data.

(ii) Use modulo-2 binary division to divide binary data by the key & store remainder of division.

(iii) Append the remainder at the end of the data to form the encoded data (codeword) & send the same.

3) At Receiver side (Check if errors occurred in transmission)

→ Perform modulo-2 division again & if the remainder is 0, then there are no errors.

→ Test Cases

Operation	Input	Expected O/p	Actual o/p
1> Input	data: 100100 key: 1101	—	—
2> Send	—	remainder: 001 codeword: 100100001	Success
3> Receive	Codeword: 100100001	remainder: 000 No error	remainder: 000 no error
4> Receive	Codeword: 100100001	remainder: 00 ¹ 0 error	remainder: 010 Error

→ Conclusion

Successfully, studied the cyclic Redundancy Check Error detection method & implemented it using C++ Program.