

Assignment - 2

Title - CRC (Cyclic Redundancy Check)

Problem Statement - write a program for error detection and correction for 7/8 bits ASCII codes using CRC. Demonstrate the packets captured using Wireshark Packet Analyzer Tool for peer to peer mode.

Software and Hardware Requirements - C++ compiler, Wireshark Packet Analyzer Tool, IDE, Processor.

Theory -

In digital systems, the analog signals will change into digital sequence (in form of bits). This sequence of bits is called as "Data Stream". The change in position of single bit also leads to catastrophic (major) error in data output.

• Error -

- i) The data can be corrupted during transmission (from source to receiver).
- ii) It may be affected by external noise or some other physical imperfections.
- iii) In this case, the input data is not same as the received output data.
- iv) This mismatched data is called 'Error'.

* Types of Error

1] Single Bit Errors -

- i) The change in one bit in whole data sequence, is called single bit error.
- ii) Occurrence of single bit error is very rare in serial communication system.
- iii) This type of error occurs only in parallel communication system, as data is transferred bit wise in single line, there is chance that single line to be noisy.

2] Multiple Bit Data Errors -

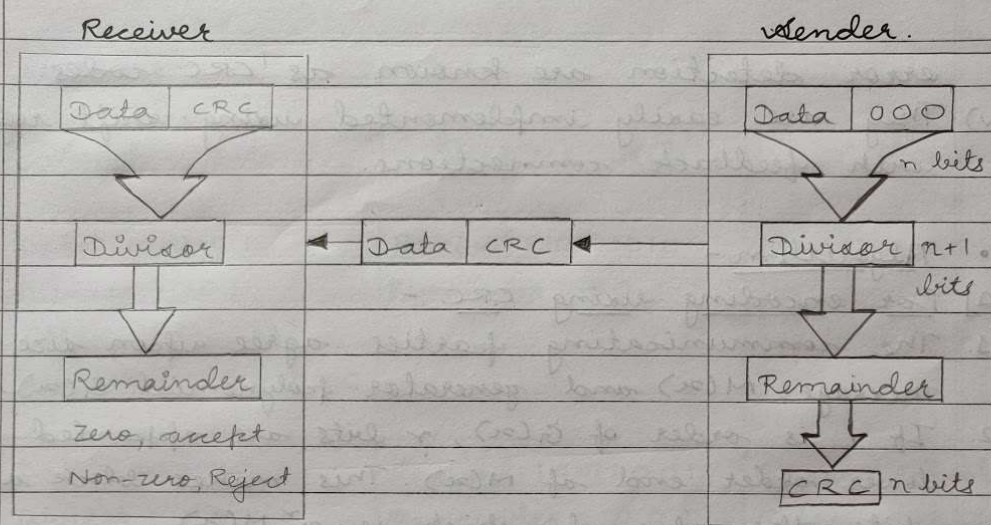
- i) If there is change in two or more bits of data sequence of transmitter to receiver, it is called "Multiple Bit Error".
- ii) This type of error occurs in both serial type and parallel type data communication networks.

3] Burst Errors -

- i) The change of set of bits in data sequence is called 'Burst Error'.
- ii) The burst error is calculated in from first bit change to last bit change.

* Error Detecting Codes

- i) In digital communication system errors are transferred from one communication system to another, along with the data.
- ii) If these errors are not detected & corrected, data will be lost.



CRC (Cyclic Redundancy Check)

* Error Detecting Codes -

1. Parity Checking
2. Cyclic Redundancy Check (CRC)
3. Longitudinal Redundancy Check (LRC)
4. Check sum

• Cyclic Redundancy Check (CRC)

- i) A cyclic code is a linear (n, k) block code with the property that every cyclic shift of a codeword results in another code word.
- ii) Here, k indicates the length of the message at transmitter (the number of information bits).
- iii) n is the total length of the message after adding check bits (actual data & check bits)
- iv) The codes used for cyclic redundancy check there by

error detection are known as CRC codes.

- v) They are easily implemented using shift registers with feedback connections.

• Algorithm -

A] For encoding using CRC -

1. The communicating parties agree upon size of message, $M(x)$ and generator polynomial $G(x)$.
2. If r is order of $G(x)$, r bits are appended to lower order end of $M(x)$. This makes block size bits, the value of which is $x^r M(x)$.
3. The block $x^r M(x)$ is divided by $G(x)$ using Modulo 2 division.
4. The remainder after division is added to $x^r M(x)$. The result is frame to be transmitted, $T(x)$.

B] Decoding using CRC -

1. The receiver divides the incoming data frame $T(x)$ with $G(x)$ using Modulo 2 Division.
2. Mathematically, if $E(x)$ is error, then modulo 2 division of $[M(x) + E(x)]$ by $G(x)$ is done.
3. If there is no remainder, data frame is accepted.
4. A remainder indicates an error, and data frame is ~~accepted~~ rejected. The receiver may send an erroneous ack, back to sender for retransmission.

* Testing -

| | <u>Sender Site</u> | <u>Receiver Site</u> | <u>Result</u> |
|----|--|--------------------------------------|---|
| 1. | Data: 1010101 Divisor: 101 CRC: 00 Transmitted: 101010100 | Received: 101010100 Remainder: 00 | No error received in message. |
| 2. | Data: 10011 Divisor: 1101 CRC: 011 Transmitted: 10011011 | Received: 10011001 Remainder: 001 | Error detected! Retransmit the data. |

• Conclusion -

Error Detection & Correction using CRC was studied & implemented successfully.