CNL Roll No: 31147

Assignment 2

Problem Statement:

Write a program for error detection and correction for 7/8 bits ASCII codes using Hamming code. Demonstrate the packets captured traces using Wireshark Packet Analyzer Tool for peer to peer mode.

Objectives:

1) Study the error detection codes (Hamming code & CRC).
2) Study packets using wireshark packet analyzer

Requirements:

Object oriented programming concept.
Wireshark Pack Analyzer Tool.

Theory:

What is Errors?

The data when send from sender to receiver may get corrupted during transmission this is called error.

Error detecting codes:

In digital communication system errors are transferred from one communication system to another, along with the data. If these errors not detected and corrected, data will be lost. For effective communication, data should transferred with high accuracy. This can be achieved by first detecting the errors and then correcting them. Error detection is the process of detecting the errors that are present in the data transmitted from transmitter to receiver in a communication system. We use some redundancy codes to detect these errors, by adding to the data while it is transmitted from source.

Types of Error detection:

1) Parity checking.
2) Cyclic Redundancy Check (CRC)
3) Longitudinal Redundancy Check (LRC)

1) Cyclic Redundancy Check:

A cyclic code is a linear (n,k) block code with the property that every cyclic shift of a codeword results in another word. CRC is a method of detecting is a method of detecting accidental changes/errors in the communication channel.

CRC uses Generator Polynomial which is available on both sender & receiver side. An example generator polynomial is of the form like x^3+x+1. This generator polynomial represents key 1011. Another example is x^2+1 that represents key 101.

n: Number of bits in data to be sent from sender side.

k: Number of bits in the key obtained from generator polynomial.

Eg Data to be sent : 100100.

key = 1101 [generator polynomial x^3+x^2+1].

```
          11101
      ____________
1101 | 100100000
       1101
       ----
        1000
        1101
        ----
         1010
         1101
         ----
          1110
          1101
          ----
          0110
          0000
          ----
           1100
           1101
           ----
           0 01
```

Remainder is 001 and hence the encoded data sent is 100100001.

If Remainder of 100100001 by 1101 is zero then data is errorless.

Page No.
Date

Hamming Code:

Redundant bits are extra binary bits that are generated and added to the information-carrying bits of data transfer to ensure that no bits were lost during the data transfer.

The no: of redundant bits can be calculated

$2^r \geq m+r+1$ m = data bit r = redundant bit.

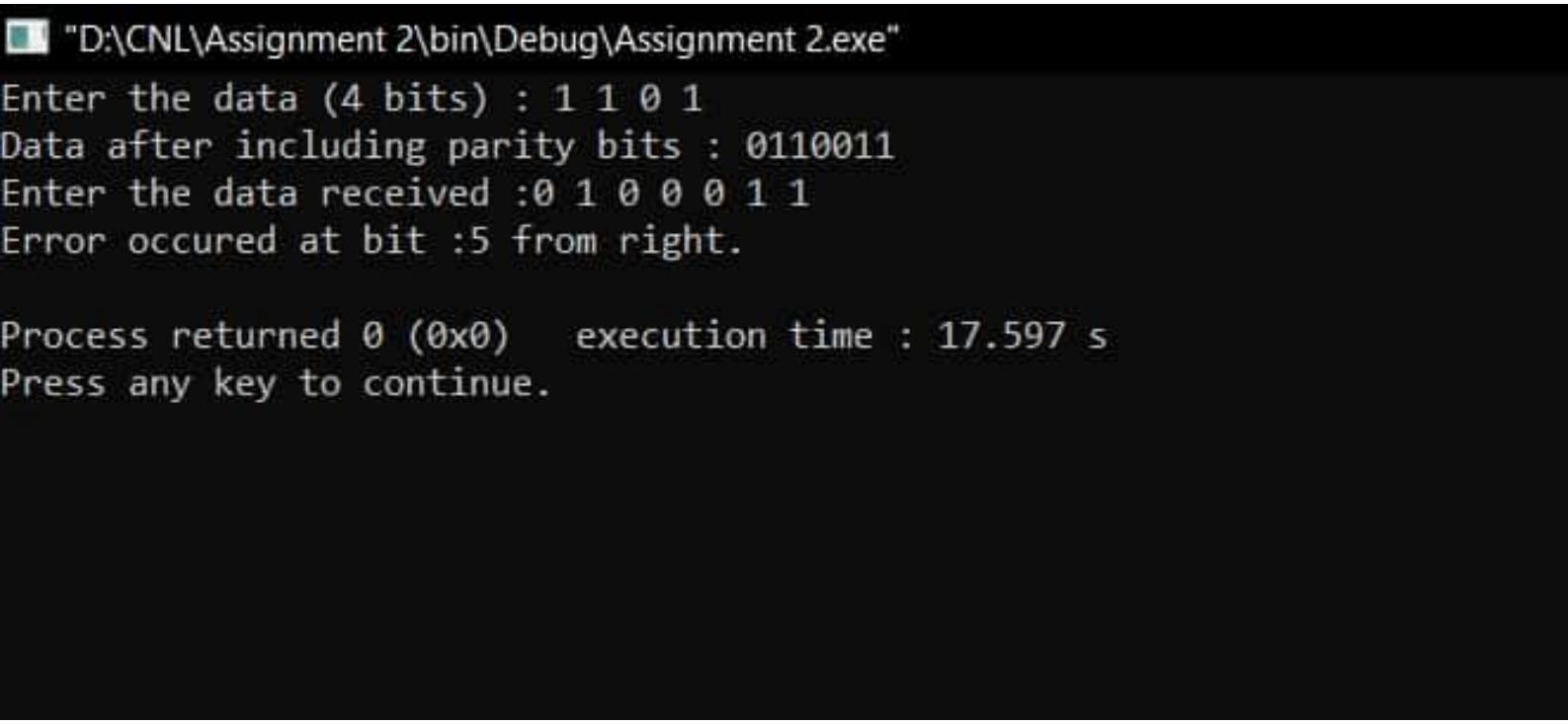
Suppose no: of data bit is 7, then.

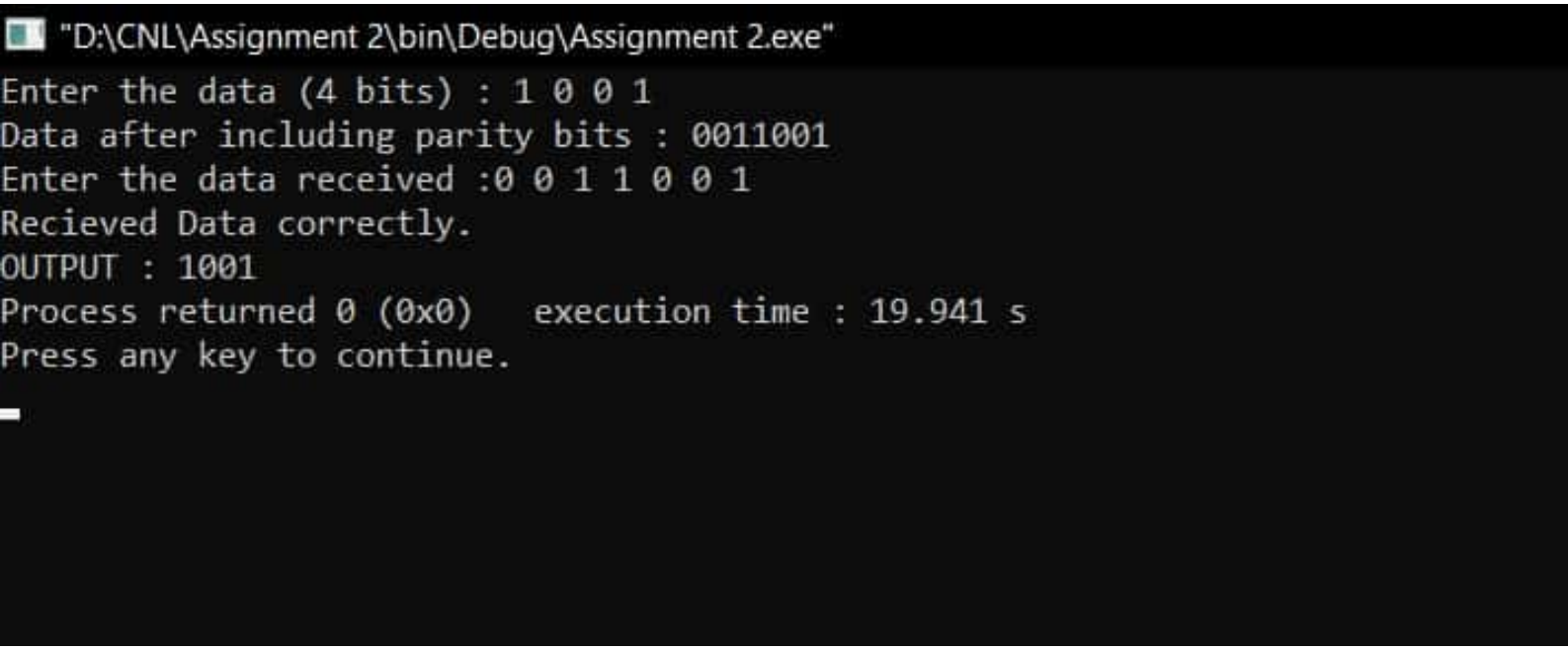
$2^4 \geq 7+4+1$ $r=4$.

Algorithm:

1) Write bit position starting from 1 in binary form.
2) All bits position that are a power of 2 are marked as parity bits.
3) All the other bit positions are marked as data bits.
4) Each data bit is included in a unique set of parity bits, as determined its bit position in binary form.
   a) Parity bit 1 covers all the bits position whose binary representation includes a 1 in least significant (1,3,5,7 etc)
   b) Parity bit 2 covers all bits positions whose binary representation include a 1 in the second position from least significant bit (2,3,6,7,10,11 etc).
   c) Parity bit 4 covers all bits position whose third position from least significant bit includes 1 (4,5,6,7,12,13,14,15).
5) Since we check for even parity set a parity bit to 1 if the number of ones in positions it checks is odd.
6) Set parity bit to 0 if total number of ones in the position it checks is even.

Redundant bits are set at power of 2.





Page No.
Date

Test Case:

1) Enter the data (4 bits): 1 1 0 1
Data after including parity: 0110011
Enter the data received: 0 1 0 0 0 1 1
Error occured at bit : 5 bit from right.

2) Enter the data (4 bits): 1 0 0 1
Data after including parity: 0011001
Enter the data received: 0 0 1 1 0 0 1
Recieved Data correctly:
Output: 1001.

Conclusion:

Learnt about Error detection technique and implemented Hamming code using C++.