|  |  |
| --- | --- |
| Name: | Prerna Sunil Jadhav |
| Sap  Id: | 60004220127 |
| Class: | S. Y. B.Tech (Computer Engineering) |
| Course: | Computer Networks (DJ12CEL405) |
| Date of Performance: |  |
| Date of Submission: |  |
| Experiment  No.: | 04 |
| Aim: | Error Detection and Correction Mechanism |

**AIM: Error Detection and Correction Mechanism**

1. HAMMING CODE

CODE:

#include <stdio.h>

void main()

{

    int data[10];

    int dataatrec[10], c, c1, c2, c3, i;

    printf("Enter 4 bits of data one by one\n");

    scanf("%d", &data[0]);

    scanf("%d", &data[1]);

    scanf("%d", &data[2]);

    scanf("%d", &data[4]);

    // Calculation of even parity

    data[6] = data[0] ^ data[2] ^ data[4];

    data[5] = data[0] ^ data[1] ^ data[4];

    data[3] = data[0] ^ data[1] ^ data[2];

    printf("\nEncoded data is\n");

    for (i = 0; i < 7; i++)

        printf("%d", data[i]);

    printf("\n\nEnter received data bits one by one\n");

    for (i = 0; i < 7; i++)

        scanf("%d", &dataatrec[i]);

    c1 = dataatrec[6] ^ dataatrec[4] ^ dataatrec[2] ^ dataatrec[0];

    c2 = dataatrec[5] ^ dataatrec[4] ^ dataatrec[1] ^ dataatrec[0];

    c3 = dataatrec[3] ^ dataatrec[2] ^ dataatrec[1] ^ dataatrec[0];

    c = c3 \* 4 + c2 \* 2 + c1;

    if (c == 0)

    {

        printf("\nNo error while transmission of data\n");

    }

    else

    {

        printf("\nError on position %d", c);

        printf("\nData sent : ");

        for (i = 0; i < 7; i++)

            printf("%d", data[i]);

        printf("\nData received : ");

        for (i = 0; i < 7; i++)

            printf("%d", dataatrec[i]);

        printf("\nCorrect message is\n");

        if (dataatrec[7 - c] == 0)

            dataatrec[7 - c] = 1;

        else

            dataatrec[7 - c] = 0;

        for (i = 0; i < 7; i++)

        {

            printf("%d", dataatrec[i]);

        }

    }

}

OUTPUT:

Text

Description automatically generated

1. CRC

CODE:  
#include <stdio.h>

#include <string.h>

#define N strlen(gen\_poly)

char data[28];

char check\_value[28];

char gen\_poly[10];

int data\_length, i, j;

void XOR()

{

    for (j = 1; j < N; j++)

        check\_value[j] = ((check\_value[j] == gen\_poly[j]) ? '0' : '1');

}

void receiver()

{

    printf("Enter the received data: ");

    scanf("%s", data);

    printf("\n-----------------------------\n");

    printf("Data received: %s", data);

    crc();

    for (i = 0; (i < N - 1) && (check\_value[i] != '1'); i++);

    if (i < N - 1)

        printf("\nError detected\n\n");

    else

        printf("\nNo error detected\n\n");

}

void crc()

{

    for (i = 0; i < N; i++)

        check\_value[i] = data[i];

    do

    {

        if (check\_value[0] == '1')

            XOR();

        for (j = 0; j < N - 1; j++)

            check\_value[j] = check\_value[j + 1];

        check\_value[j] = data[i++];

    } while (i <= data\_length + N - 1);

}

int main()

{

    printf("\nEnter data to be transmitted: ");

    scanf("%s", data);

    printf("\n Enter the Generating polynomial: ");

    scanf("%s", gen\_poly);

    data\_length = strlen(data);

    for (i = data\_length; i < data\_length + N - 1; i++)

        data[i] = '0';

    printf("\n----------------------------------------");

    printf("\n Data padded with n-1 zeros : %s", data);

    printf("\n----------------------------------------");

    crc();

    printf("\nCRC or Check value is : %s", check\_value);

    for (i = data\_length; i < data\_length + N - 1; i++)

        data[i] = check\_value[i - data\_length];

    printf("\n----------------------------------------");

    printf("\n Final data to be sent : %s", data);

    printf("\n----------------------------------------\n");

    receiver();

    return 0;

}

OUTPUT:  
Text

Description automatically generated