

29/12/22

LAB-8

Write a program for error detection using CRC-16 bit

Data: 1011101

$G(x)$ : 10001000000100001

Modified data: Data + 16 zero's

Codeword = Data + Checksum.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
char g[] = "100010000001", p[50], c[50];
int m1, m2;
-> void xor() { int i;
for(i=0; i<m1; i++) {
if(c[i] == g[i]) {
c[i] = '0';
}
else {
c[i] = '1';
}
}
```

```
-> void crc() { int i, count;
for(count=0; count<m1; count++) {
c[
```

```
-> void crc(int u) { int i, j;
for(i=0; i<u; i++)
temp[j] = m[j];
for(i=0; i<16; i++)
r[j] = m[i];
for(j=0; j<u-16; j++) {
if(r[0] == '1') {
```

```

q[j] = '1';
calcrun(); }
else { q[i] = '0';
shiftl(); }
r[16] = m[17+i];
r[17] = '10';
for (j=0; j<=17; j++)
temp[j] = r[j]; }
q[u-16] = '10'; }

```

```

void calcrun() { int i, j;
for (i=1; i<=16; i++)
r[i-1] = ((int) temp[i]-48) ^ ((int) q[i]-48) + 48;
void swiftl() { int i;
for (i=1; i<=16; i++)
r[i-1] = r[i]; }

```

```

void caltrans(int n) {
int i, K = 0;
for (i=n-16; i<=n; i++)
u[i] = ((int) n[i]-48) ^ ((int) r[K++]-48) + 48;
m[i] = '10'; }

```

O/P: Enter data = 1011101

g(x) = 10001000000100001

Modified data = 101110100000000000000000

Transmitted data: 10111011000101101011000

Enter transmitted data: 10111011000101101011000

CR check

Reminder: 0000000000000000000

Received data is correct.

~~N~~  
5/1/2023