

PART-B

Program 14

Write a program for error detecting code using CRC-CCITT (16-bits).

Code :

1/14/24 (13)

Aim: Implementation of CRC.

code:

```
def xor(a, b):
    result = []
    for i in range(1, len(b)):
        if a[i] == b[i]:
            result.append('0')
        else:
            result.append('1')
    return ' '.join(result)
```

```
def modDiv(dividend, divisor):
    pick = len(divisor)
    temp = dividend[0:pick]
    while pick < len(dividend):
        if temp[0] == '1':
            temp = xor(divisor, temp) + dividend[pick]
        else:
            temp = xor('0' * pick, temp) + dividend[pick]
        pick += 1
    if temp[0] == '1':
        temp = xor(divisor, temp)
    else:
        temp = xor('0' * pick, temp)
    checkword = temp
    checkword = temp
    return temp checkword.
```

(44)

```

def encodeData (data, key):
    l = len(key)
    append_data = data + '0' * (l - len(data))
    remainder = mod2div(append_data, key)
    codeword = data + remainder
    print("Remainder", remainder)
    print("EncodeData (data + remainder):",
          codeword)
  
```

data = "100100"

key = "1101"

EncodeData (data, key).

Output :

Sender side

Remainder : 001

EncodeData (data + Remainder) : 100100001

Receiver side

Correct message received.

Output

Enter data: 1100110

Enter generator polynomial: 1101

CRC: 100

Transmitted Data: 1100110100

Enter received data: 1100110100

No Error

=== Code Execution Successful ===