```cpp
// SL.No.: - 31
// Admission No.: - 21JE0269
// Name: - Chotaliya Zeel Vijaybhai

#include <iostream>
#include <string>
#include <bitset>
#include <limits>
#include <vector>

using namespace std;

bitset<8> mod2Division(const bitset<40>& dividend, const bitset<9>& divisor) {
    bitset<40> dividendCopy = dividend; // Copy of the dividend for division

    // Perform the division process, starting from the most significant bit
    for (int i = 39; i >= 8; i--) {
        // If the current bit is 1, perform XOR with the divisor
        if (dividendCopy[i] == 1) {
            // XOR each bit of the divisor with the corresponding bit in the dividend
            for (int j = 0; j < 9; ++j) {
                dividendCopy[i - j] = dividendCopy[i - j] ^ divisor[8 - j]; // XOR operation
            }
        }
    }

    // The remainder is in the least significant 8 bits of the dividendCopy
    bitset<8> remainder;
    for (int i = 0; i < 8; ++i) {
        remainder[i] = dividendCopy[i]; // Copying remainder
    }

    return remainder; // Returning the 8-bit remainder
}

string crc8encode(string bitstream){


    // converting string to bitset variable.
    bitset<40> dataword(bitstream);

    // appending 8 zeros at last.
    dataword <<= 8;

    // creating divisor.
    bitset<9> divisor("100011101");

    // performing modulo - 2 - division.
    bitset<8> remainder;
    remainder = mod2Division(dataword, divisor);
    string remainderString = remainder.to_string();

    // returning encoded bitsream.
    return bitstream + remainderString;
}

void crc8decode(string bitstream){
    // converting string to bitset variable.
    bitset<40> dataword(bitstream);

    // creating divisor.
    bitset<9> divisor("100011101");

    // performing modulo - 2 - division.
    bitset<8> remainder;
    remainder = mod2Division(dataword, divisor);
```

```
    string remainderString = remainder.to_string();

    // checking for error
    bool iserror = false;
    for(char c : remainderString){
        if(c == '1'){
            iserror = true;
            break;
        }
    }

    cout << "Decoded Bitstream" << endl;

    if(iserror){
        cout << "Error Detected. Ask for Retransmission." << endl;
    }else{
        cout << "No error;" << endl;
        cout << "Data: - " << bitstream.substr(0, bitstream.size() - 8) << endl;
    }
    return;
}

int main(){

    // Checking, whether user wants to send or receive message.
    cout << "Are you sender (1) or receiver (0) ?" << endl;
    bool x;
    cin >> x;

    string bitstream;

    if(x){
        cout << "Enter bitstream to be encoded :" << endl;
        cin >> bitstream;

        string encoded_bitstream;
        encoded_bitstream = crc8encode(bitstream);

        cout << "Encoded bitstream : " << endl;
        cout << encoded_bitstream << endl;
    }else{

        cout << "Enter bitstream to be decoded :" << endl;
        cin >> bitstream;

        crc8decode(bitstream);
    }

    return 0;
}
```