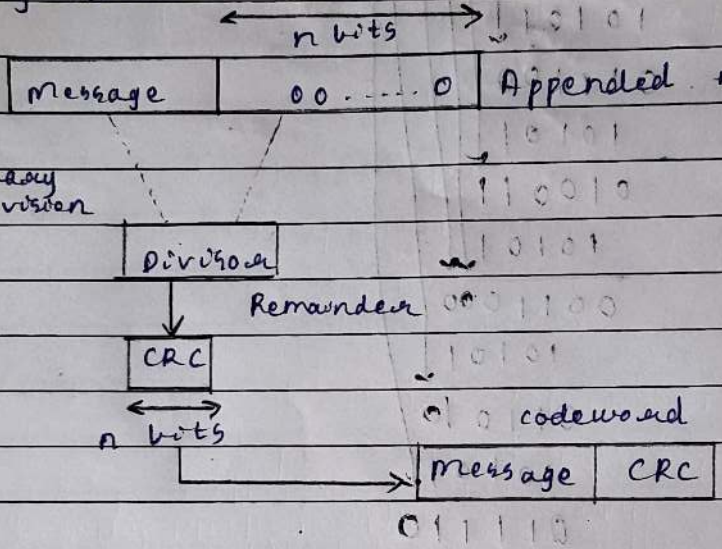
Q1. What is CRC? Explain CRC generator and checker with example.

Ans. CRC stands for Cyclic Redundancy Check, which is a mathematical algorithm that detects errors in data transmission.

1. CRC generator:-



eg: Generate CRC code for the data word 1100 10101.
The divisor is 10101.

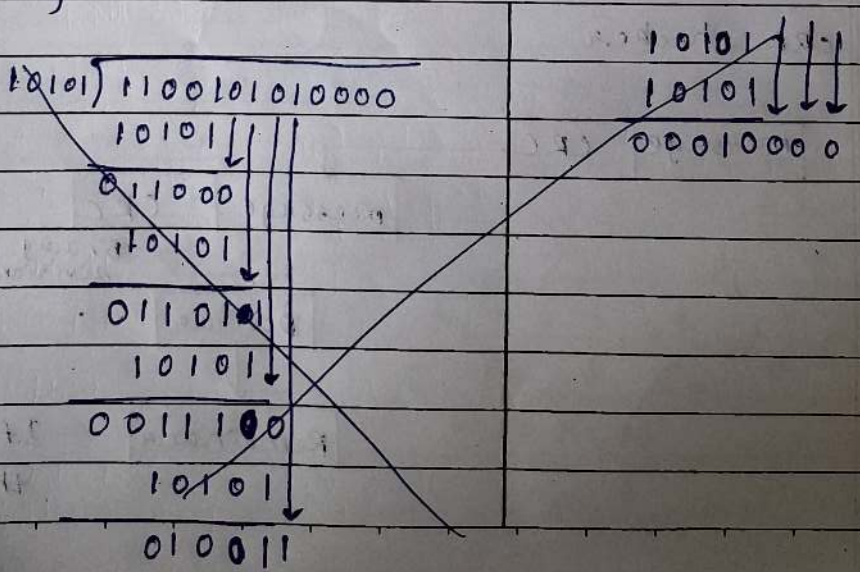
Soln: number of bits of codeword = 5

Step 1: Obtain the dividend

Dividend = data word + (n-1) zeroes

Dividend = 1100101010000

Step 2: Carry out division



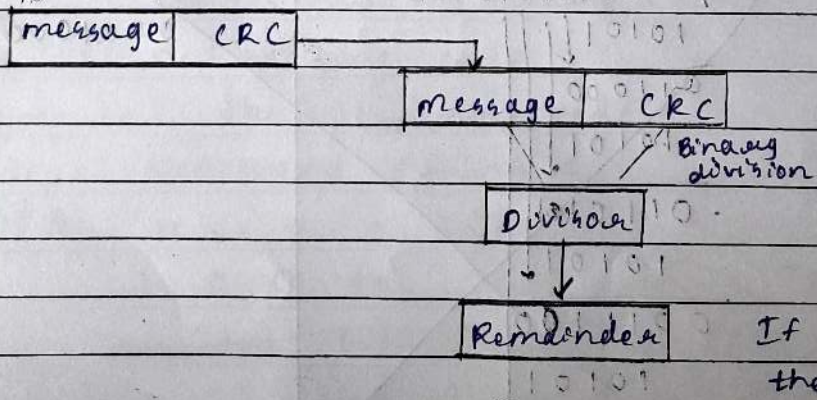
```
10101 | 1100 1010 10000
        10101
        -----
        01100
         1010 1
         -----
         01100
          1010 1
          -----
          01100
           1010 1
           -----
           01100
            1010 1
            -----
            01100
             1010 1
             -----
             01100
              1010 1
              -----
              01110
               1010 1
               -----
               01111
                1010 1
                -----
                0101 1   ← Remainder
```

Step 3: Obtain codeword

∴ codeword = 1 1 0 0 1 0 1 0 1 1 0 1 1

(1 1 0 0 1 0 1 0 1 : data word; 1 0 1 1 : Remainder)

2. CRC checker



eg: The codeword is received as 1100 1001 01011. Check whether there are errors in the received codeword, if divisor is 10101.

Soln:
Data word : 1100 1001 01011
~~Code~~ Divisor : 10101

```
10101 ) 1100100101011
        10101
        011000
         10101
         011010
          10101
          011111
           10101
           010100
            10101
            000011011
                10101
                01110  ← Remainder
```

- The non-zero remainder shows there are errors in received codeword.

Q2. What is hamming code? Generate hamming code for 7/8 bit data word.

Ans. • Hamming codes are linear block codes. It is an error correcting code.

→ Compute the Hamming code for the data 1001101.

Soln: Step 1: codeword format:

11	10	9	8	7	6	5	4	3	2	1
1	0	0	P8	1	1	0	P4	1	P2	P1

Step 2: ~~consider bits~~ Find P1, P2, P4, P8:

1. P1:

- consider bits 1, 3, 5, 7, 9, 11
  1 0 1 0 1 P1
  ∴ For ~~even~~ odd parity P1 = 1

2. P2 :
- consider bits 2, 3, 6, 7, 10, 11
  1 0 1 1 1 P2
  ∴ For ~~odd~~ even parity P2 = 0

3. ~~P4~~ P4 :
- Consider bits 4, 5, 6, 7
  1 1 0 P4
  ∴ For even parity P4 = 0

4. P8 :
- Consider bits 8, 9, 10, 11
  1 0 0 P8
  ∴ For odd parity P8 = 1

→ Step 3 : Write codeword :
  code word = 1 0 0 1 1 1 0 0 1 0 1

Q3 Explain checksum in detail.

Ans → Defination :-

- A checksum is a small-sized datum derived from a block of digital data for the purpose of detecting errors that may have been introduced during its transmission or storage

→ Calculation :-

- As each word is ~~added~~ transmitted, it is added to the previously sent word and the sum is retained at the transmitter.
- Each successive word is added in this manner to

the previous sum.

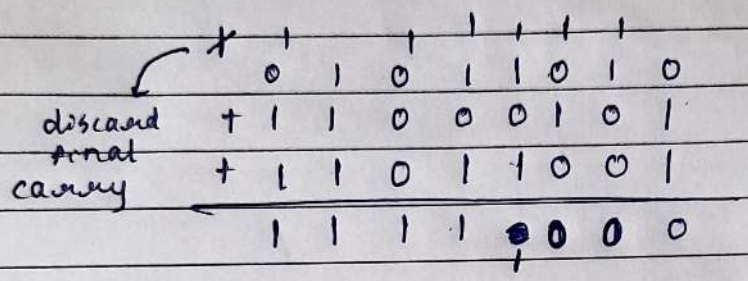
- At the end of the transmission the sum (called checksum) up to that time is sent.

→ eg:

What is the checksum of the following characters?

01011010, 11000101, 11011001

Soln:



∴ 11111000 is the checksum