Name - Ayesha Nautiyal
SAP ID - 500122813
Batch - 5

__/__/____

Experiment - 03

CRC & Hamming code

CRC code :-

1) Compute the CRC for the message 1110001 using the polynomial 1001.

⇒ The polynomial 1001 can be represented as

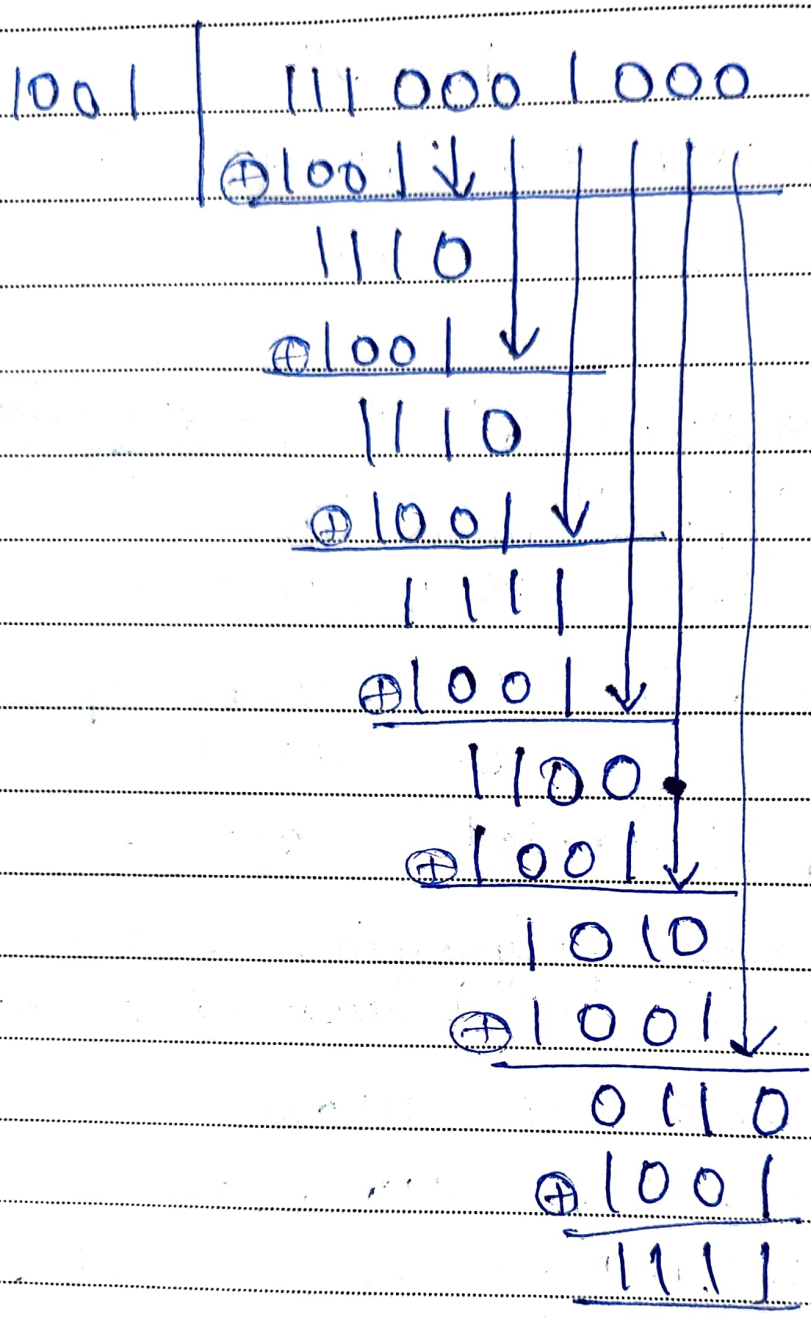
$$P(x) = x^3 + 1$$

this is 3 degree polynomial, so there is 3bit CRC.

original data bits : 1110001

updated data bits : 1110001000

Now, we can divide original data bits x updated data bits by XOR operator.



So, the CRC code → 111000111

Ans.

2) A sender the polynomial 111 to encode the message 1001101. Find the CRC and the transmitted code.

⇒ The polynomial can be represented as-

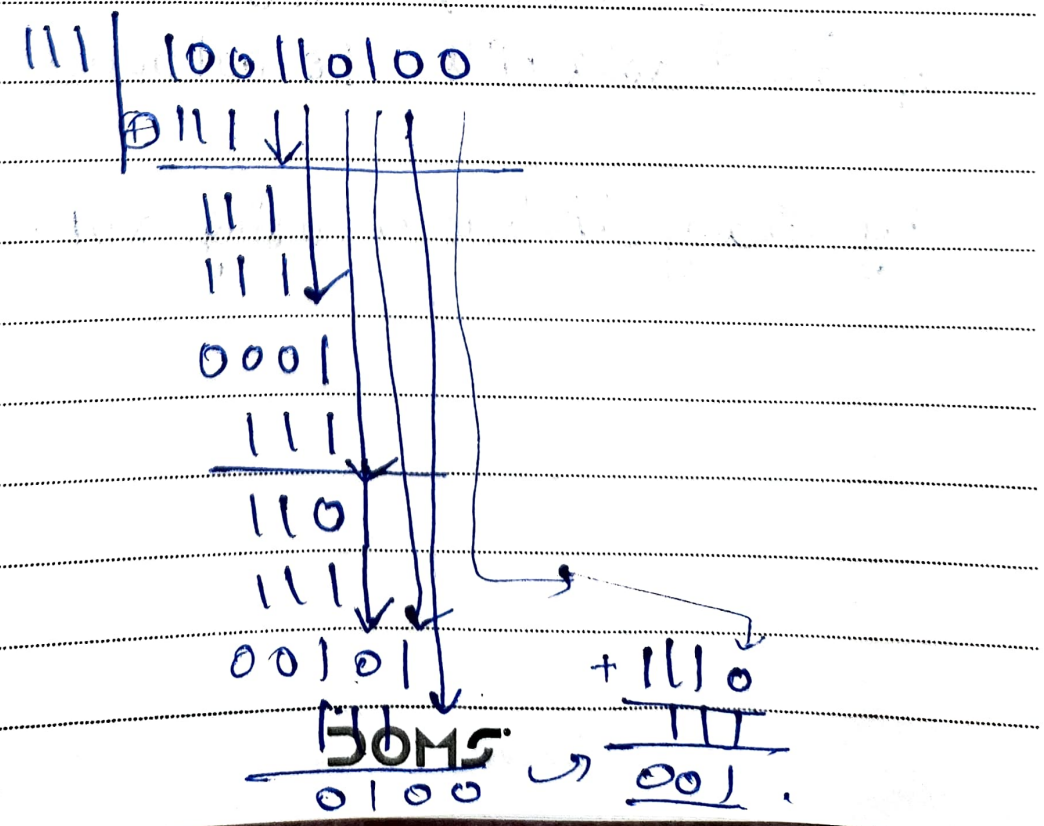
$111 \rightarrow x^2+x+1$.

There is 2 degree polynomial so 2 bit CRC.

original data bits: 1001101

updated data bits: 100110100.

performing binary divisions using the bits using XOR.



so, the CRC : 01
transmitted bits are : 10011010 1.

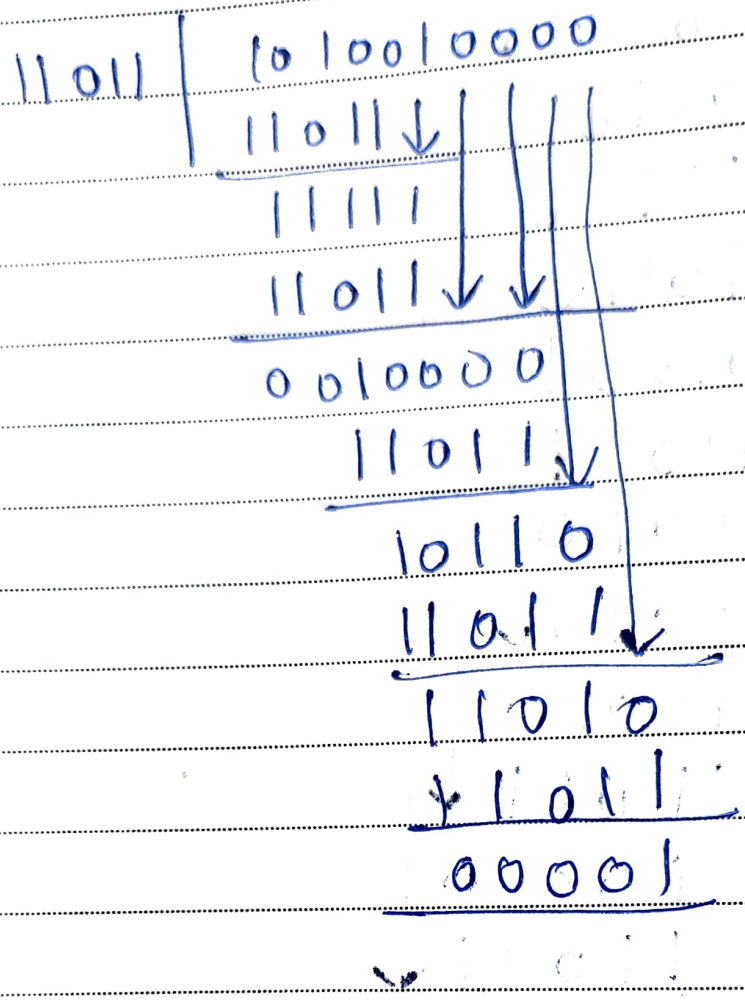
3.) If the message is 10100 1 and the polynomial is 11011. Determine the CRC and check if the received message 10100110 01 is error free.

→ The polynomial is:
11011 → $x^4 + x^3 + x + 1$.

degree of polynomial is 4, so 4 bit CRC.

original data bits : 10100 1.
updated data bits : 10100 10000

by binary division using XOR.



so, the transmitted code is :-

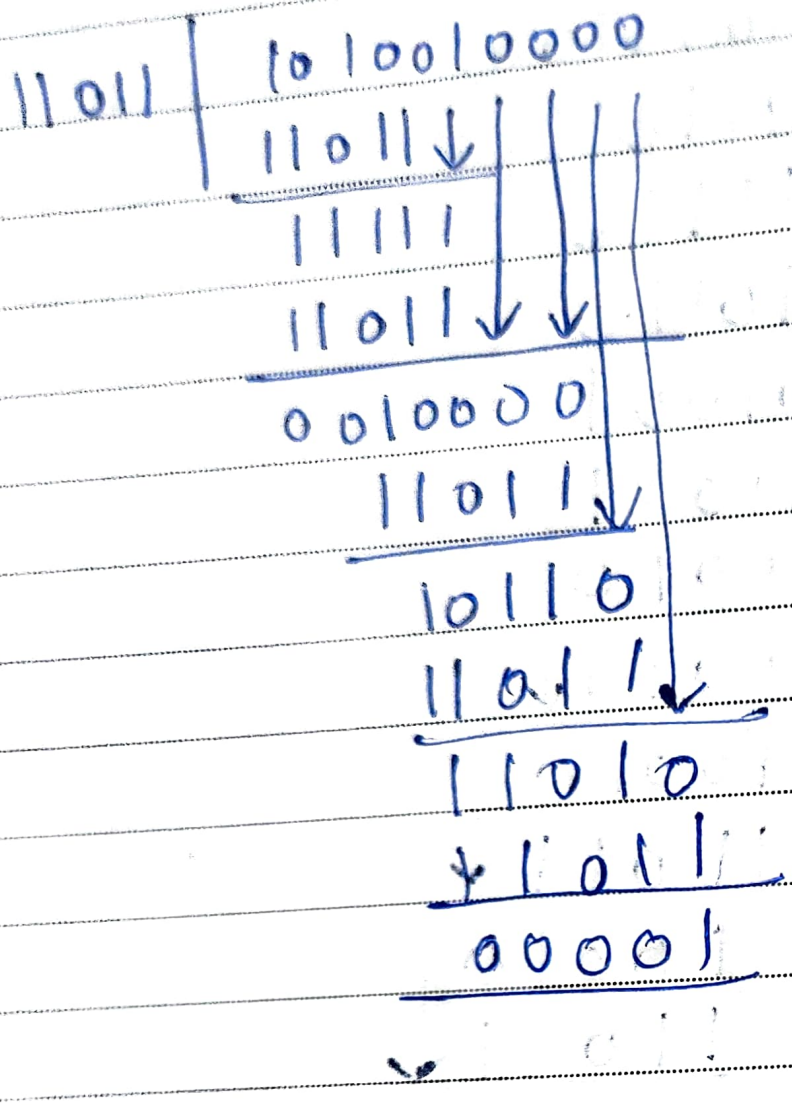
1010010001

CRC → 0001.

checking the error :-

so, the received message is 10100110001.

we can do binary division using XOR.



So, the transmitted code is :-

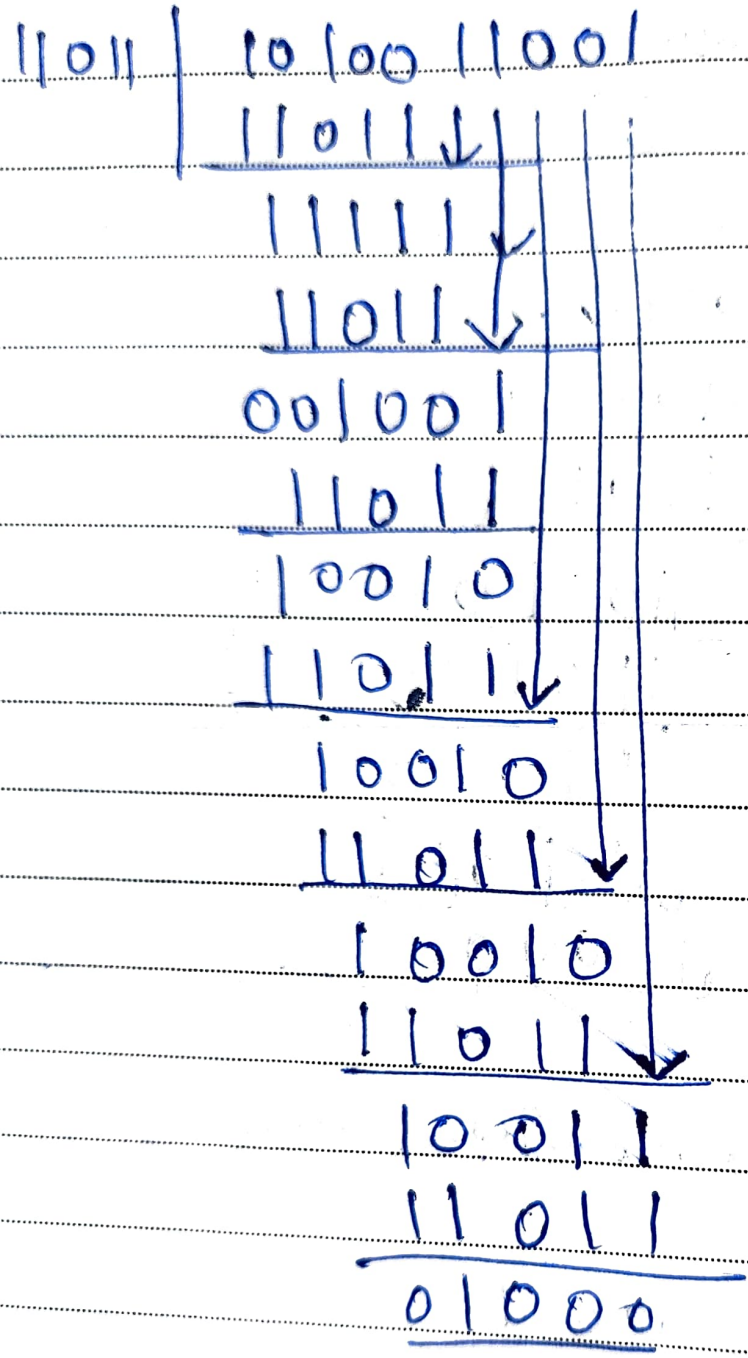
10100 10001

CRC → 0001.

Checking the error :-

So the received message is 10100 11001

division using XOR



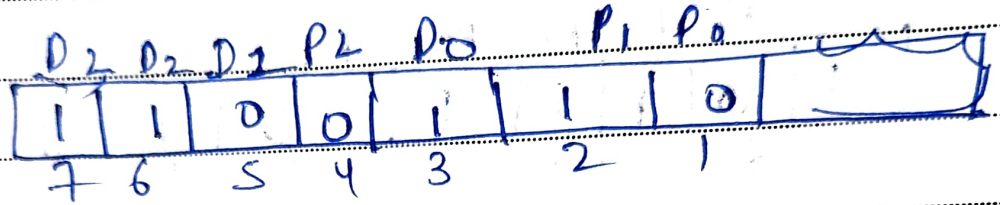
So, 01000 we received this (remainder $\neq 0$).

So, its an error.

4) Hamming code :-

↳ generate the hamming code for the data bits 1011 using given pair.

data bits : 1011



Pn → even parity bits {2^n}

Dn → data bits { remaining after 2^n}

So, we need to calculate even parity bit using XOR.

P_0 : * ⊕ 1 ⊕ 0 ⊕ 1 = 0

P_1 : * ⊕ 1 ⊕ 1 ⊕ 1 = 1

P_2 : * ⊕ 0 ⊕ 1 ⊕ 1 = 0

So, $P_0 = 0$

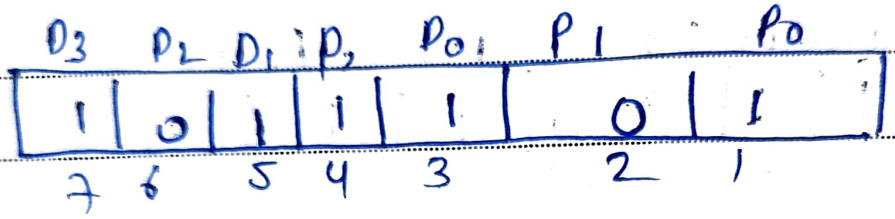
$P_1 = 1$

$P_2 = 0$.

So, final Hamming code for even parity is → 0 1 1 0 0 1 1.

2) A 7 bit Hamming code 1011101 is received. Detect and calculate any error.

⇒ 7-bit Hamming code → 1011101.



so, there is received so, we need to rectify and calculate the errors.

P_0: $1 \oplus 1 \oplus 1 \oplus 1 = 0$ (correct)

P_1: $0 \oplus 1 \oplus 0 \oplus 1 = 0$ (correct)

P_2: $1 \oplus 1 \oplus 0 \oplus 1 \neq 1$ (it is not correct with the received bits).

so, after checking the received hamming code P_2 is getting unmatched so, it is not correct.

the error is in position → 4.

correct hamming code: 1010101.

3.) Construct the Hamming code for the data 11001 using odd parity.

→ data bits → 11001 (given)
constructing Hamming code using odd parity:

D_4	P_3	D_3	D_2	D_1	P_2	D_0	P_1	P_0
1	0	0	0	1	0	1	0	0
9	8	7	6	5	4	3	2	1

P_0 : * ⊕ 1 ⊕ 1 ⊕ 0 ⊕ 1 = 1 it is odd so [0]

P_1 : * ⊕ 1 ⊕ 0 ⊕ 0 = 1
↳ it is odd so [0]

P_2 : * ⊕ 1 ⊕ 0 ⊕ 0 = 1
↳ it is odd so [0]

P_3 : * ⊕ 1 = 1
↳ it is odd so, [0].

P_0 : 0
P_1 : 0
P_2 : 0
P_3 : 0

final Hamming code is : 00100001

DOMS