Module 4

Data Communication and Networking

Error Detection and Correction

⇒ Applied, when there are chances of corruption and data loss.

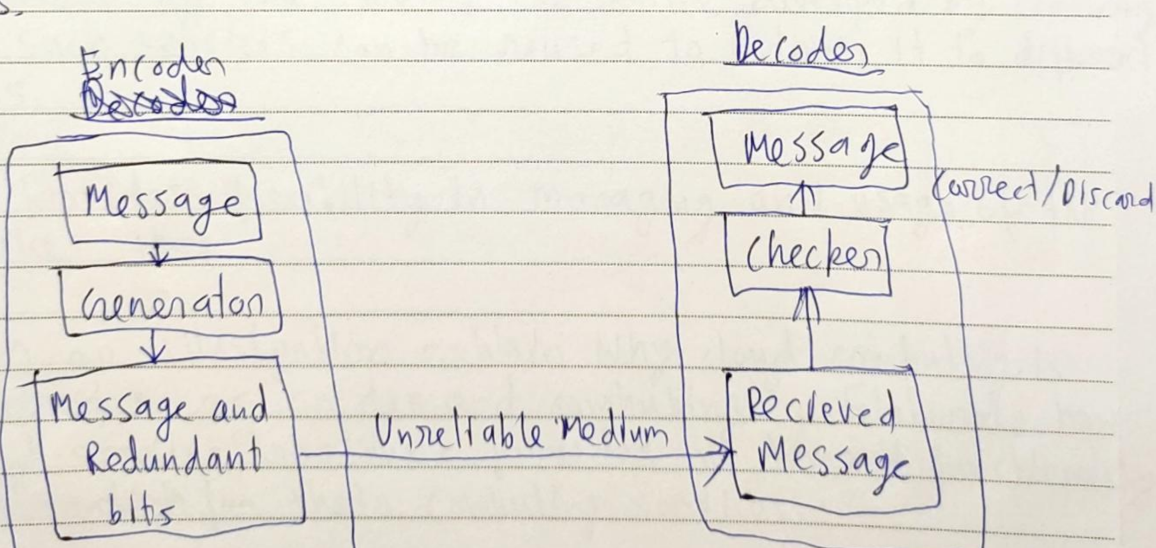
Types of Errors :-

1) Single Bit error :- One data bit is changed.

2) Burst error :- When two or more bits are changed.

⇒ length of Burst Errors :- From first corrupt bit till last one.

⇒ For detecting and correcting the errors, we use redundant bits.



⇒ Modulo N - Arithematic uses integers in range of 0 to N-1.

Block Coding

⇒ A message is divided into no. of blocks of size k bits called Datawords. Then, r redundant bits are added to each block to make length $\boxed{m = k + r}$ resulting in the m-bit blocks called codewords.

$\boxed{2^m \text{ codewords}}$ ⇐ in a redundant message

$\boxed{2^k \text{ valid codewords}}$

⇒ It can only detect one bit errors only.

⇒ Dataword and Codeword table is present to sender and receiver both.

Hamming Distance

⇒ Differences between corresponding bits, by utilizing XOR ⊕ operation

Example :- distance b/w 000, 011 ⇒ 000 ⊕ 011 = 011.

⇒ Minimum hamming Distance :- Smallest hamming distance between all possible pairs of words.

8) Using minimum Hamming distance (d_{min}), the guaranteed s errors to be detected can be calculated by

$$\boxed{d_{min} = s + 1}$$

8) For guaranteed correction of up to t errors,

$$\boxed{d_{min} = 2t + 1}$$

Revision

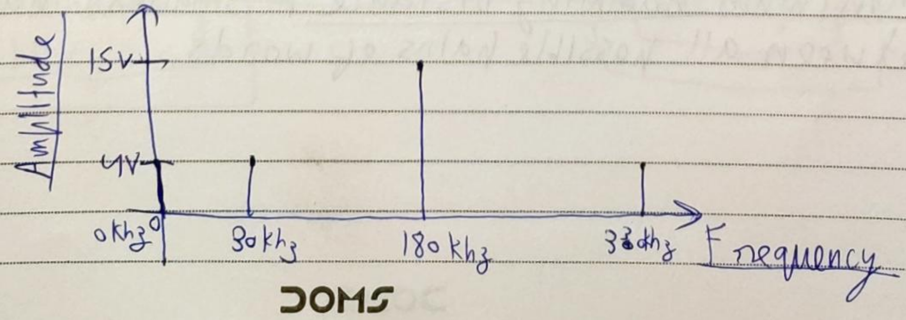
Q1 Identify and describe the layers of OSI model for the following activities:-

1) Host to Host Flow Control → Data Link
2) Route Determination → Network
3) Reliable Process-to-Process Delivery → Transport
4) Mechenical, Electrical Interface → Physical
5) Data Compression → Presentation

~~Q3 Identify the switching technique~~

Q A non periodic signal has a bandwidth of 330 khz with middle freq. of 180khz and peak amplitude of 15V. The two extreme freq. have amplitude of 4V, Draw Frequency domain of the signal.

Ans



Linear Block Coding

⇒ All block codes belong to a subset of linear block codes.

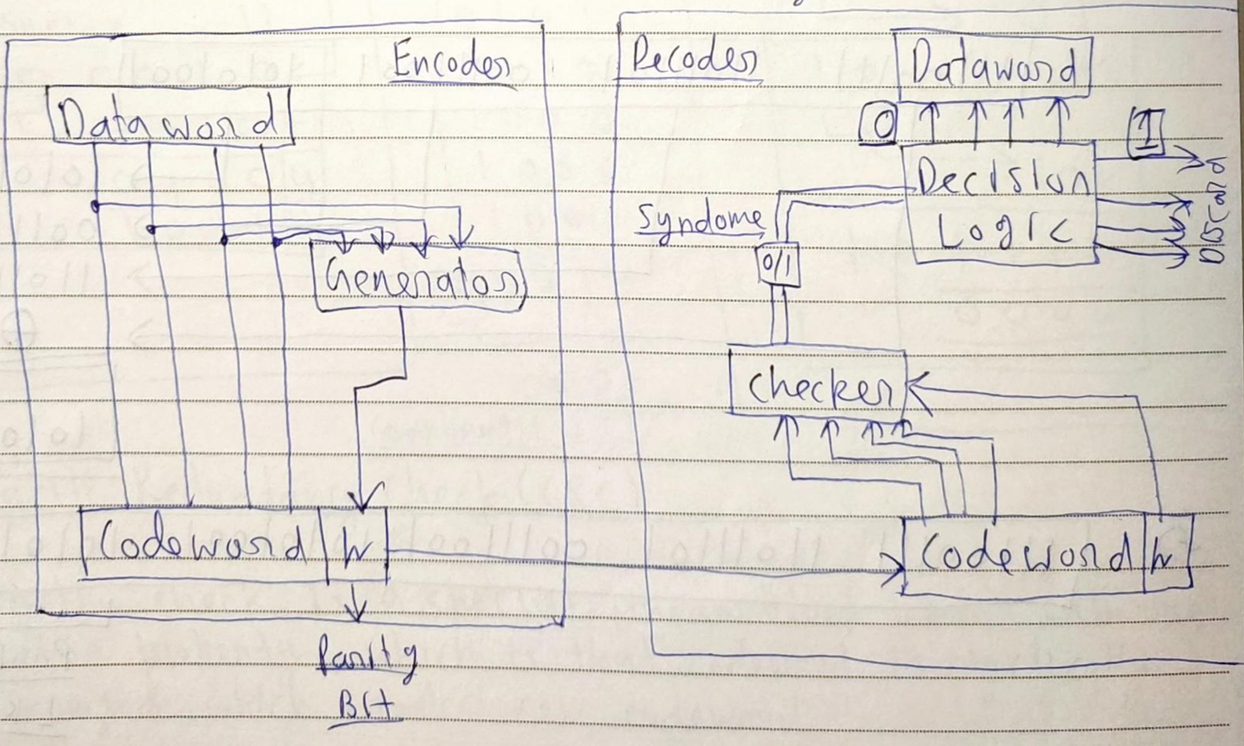
Simple Parity Check

⇒ It detects single bit errors. (Can only detect odd bits error)

Error detecting Codes

1) Parity Check 2) Hamming Code 3) CRC

Encoder & Decoder in Simple Parity Check



Vertical Redundandy Check (VRC)

- It also known as Parity check
- Works with even and odd parity, but usually works in even parity.

Longitudnal Redundancy Check (LRC)

- It creates a table to organize the data and creates a parity for each column.
- It can detect all burst errors upto length n (No. of columns).

11100111 11011101 00111001 10101001

10101001
00111001
11011101
$\oplus$
11100111

10101010

- 11100111 11011101 00111001 10101001 10101010

$\underbrace{10101010}_{\text{Parity LRC}}$

- If two bits in one data unit are damaged and two bits at exactly same position in another data unit are damaged, then LRC can't detect error.

40
4

32
16
48

__/__/__

Checksum

Data = 5 2 8 9 10

Sender Side

5
2
8
9
10

Sum = 34
Wrapped Sum = 4
Checksum = 11

100010
10 +
0100 | 4
1011 | 11

5	2	8	9	10	11

Packet

8 4 2 1
0101
0010
1000
1001
10010
100010
10
00100
Compliment 1011

Reciever Side

5
2
8
9
10
11

Sum = 45
Checksum = 15 ← Wrapped sum
0

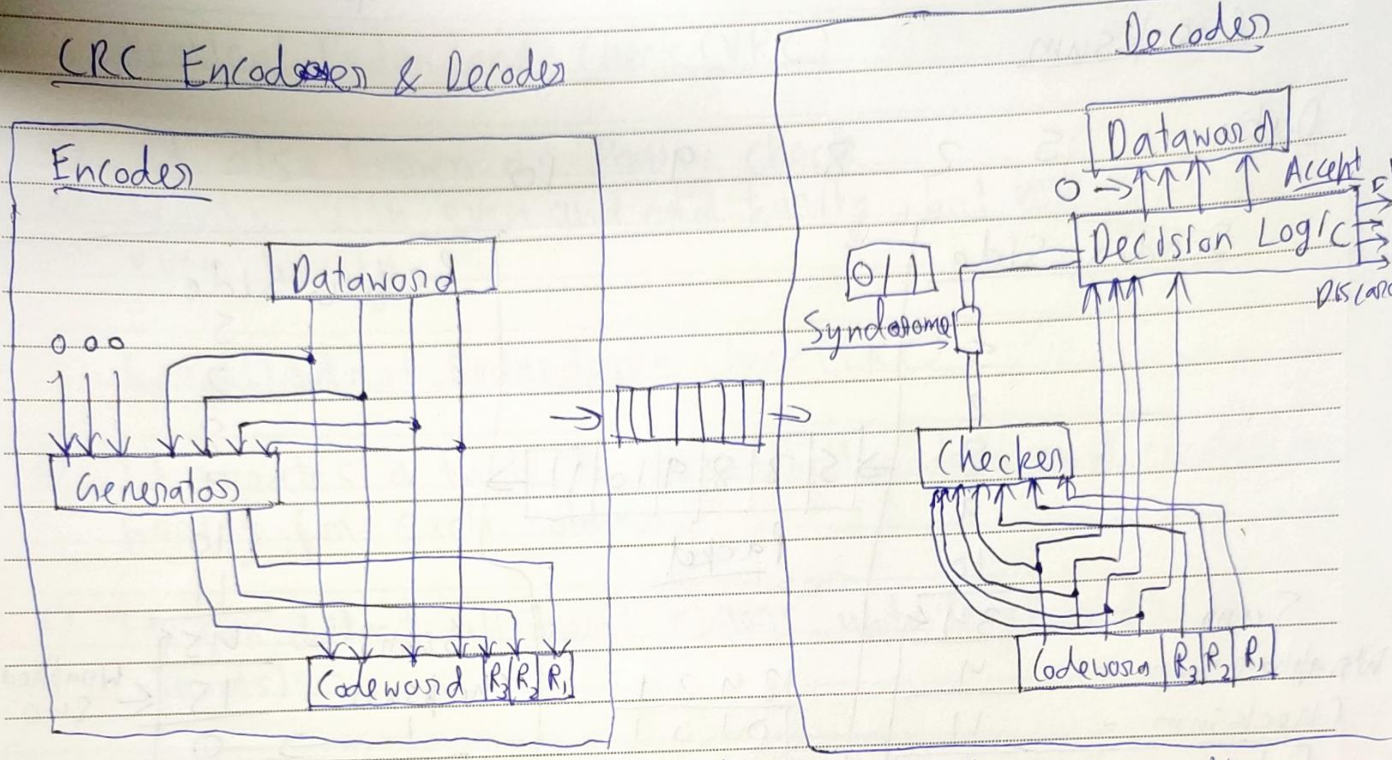
101101
10 +
1111
0000

Cyclic Redundancy Check (CRC)

- Cyclic check is a special linear block codes with one extra property, which is that codeward is cyclically changed, which creates new codeword.

- In CRC, the reminder size will be n-1, where n is no. of bits in ~~dataword~~ divison and 0 bits are added in dataword corresponding to no. of bits in remainder

CRC Encoder & Decoder



→ CRC uses XOR operation during checksum generation and verification process (during division).

Q D.W. ⇒ 1001010110   No. of bits added = 5 − 1 = 4
Divisor ⇒ 10101   (where 5 is no. of bits in divisor)

```
             1011001000
10101 ) 1001010110 0000
        ⊕10101
         X01111
         ⊕00000
          X11110
          ⊕10101
           X10111
           ⊕10101
            X00101
             ⊕00000
              X01010
              ⊕00000
               X1010
```

Remainder = 1010

```
 10100
⊕10101
 X00010
  00000
  X00100
   00000
   X01000
    00000
    X1000
```

~~10100~~
~~10101~~
~~00001~~

Polynomial representation of Binary Word

a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0
1	0	1	1	0	1	0	1

$1x^7 + 0x^6 + 1x^5 + 1x^4 + 0x^3 +$ $1x^2 + 0x^1 + 1x^0$

CRC division using Polynomial

Dataword = 1001 = $x^3 + x^0 = x^3 + 1$

Divisor = 1011 = $x^3 + x^1 + x^0 = x^3 + x + 1$

$x^3 + x$

$x^3 + x + 1 \,\big|\, x^6 + x^3$ ← $1001\boxed{000}$ $(x^6 + x^3)$

$\oplus\ \cancel{x^6} + \cancel{x^3} + x^4$

$\cancel{x^4}$

$\oplus\ \cancel{x^4} + x^2 + x$

$x^2 + x$ Remainder

Checking

$x^3 + x$

$x^3 + x + 1 \,\big|\, \cancel{x^6} + \cancel{x^3} + x^2 + x$

$\oplus\ \cancel{x^6} + \cancel{x^3} + x^4$

$\cancel{x^4} + \cancel{x^2} + \cancel{x}$

$\oplus\ \cancel{x^4} + \cancel{x^2} + \cancel{x}$

0 0 0

Q Dataword ⇒ 1 0 1 1 0 1

$x^5 \; x^3 x^2 \; x^0 = x^5 + x^3 + x^2 + 1$

Division ⇒ 1 0 1 1 $= x^3 + x + 1$

$x^3 \; x^1 x^0$

⇒ $101101 + 000 = 101101000 = x^8 + x^6 + x^5 + x^3$

$x^8 \; x^6 \; x^5 \; x^3$

⇒

$$\begin{array}{r|l} & x^5 + 1 \\ \hline x^3+x+1 & \cancel{x^8} + \cancel{x^6} + \cancel{x^5} + x^3 \\ & \oplus \cancel{x^8} + \cancel{x^6} + \cancel{x^5} \downarrow \\ \hline & \cancel{x^3} \\ & \oplus \cancel{x^3} + x + 1 \\ \hline & x + 1 \end{array}$$

Checking

$$\begin{array}{r|l} & x^5 + 1 \\ \hline x^3+x+1 & \cancel{x^8} + \cancel{x^6} + \cancel{x^5} + x^3 + x + 1 \\ & \oplus \cancel{x^8} + \cancel{x^6} + \cancel{x^5} \downarrow \quad \downarrow \downarrow \\ \hline & \cancel{x^3} + \cancel{x} + \cancel{1} \\ & \oplus \cancel{x^3} + \cancel{x} + \cancel{1} \\ \hline & 0 \end{array}$$

Ans

Computer Networks

Error Correction Codes

1) Hamming Code 2) Binary Convolutional Codes
3) Reed-Solomon Codes 4) Low Density Parity Code check

Hamming Code

⇒ It can only correct one bit error.

⇒ No. of redundant bits $= 2^r >= m + r + 1$

where, m is no. of bits in dataword and r is no. of redundant bits.

⇒ Position of redundant bits are in the power of 2 i.e. for 4 redundant bits positions can be $2^0, 2^1, 2^2$ and 2^3.

Example :- 10011010

⇒ $2^r \geq m+r+1$ put $r = 4$
⇒ $16 \geq 8+4+1$
⇒ $16 \geq 13$

⇒ Position of redundant bits are $2^0, 2^1, 2^2, 2^3$

_ _ 1 _ 0 0 1 _ 1010
1 2 3 4 5 6 7 8

⇒ _ _ 1 _ 0 0 1 _ 1010
R_1 R_2 R_4 R_8

$\Rightarrow R_1 = 1, \frac{1}{3}, \frac{0}{5}, \frac{1}{7}, \frac{1}{9}, \frac{1}{11} = 0$

$\Rightarrow R_2 = 2, \frac{1}{3}, \frac{0}{6}, \frac{1}{7}, \frac{0}{10}, \frac{1}{11} = 1$

$\Rightarrow R_4 = \overline{4}, \frac{0}{5}, \frac{0}{6}, \frac{1}{7}, \frac{1}{12} = 1$

$\Rightarrow R_8 = \overline{8}, \frac{1}{9}, \frac{0}{10}, \frac{1}{11}, \frac{0}{12} = 0$

$\Rightarrow$ For R_m bits, to calculate values use first m bits starting from R_m bit, then leave m bits and used next bit. i.e. use first m bits, then use the bits in difference of m and again use m bits.

Example $R_1 = 1, 3, 5, 7, 9, 11$ ① ~~2~~ ③ ~~4~~ ⑤ ~~6~~ ⑦ ~~8~~ ⑨ ~~10~~ ⑪
$R_2 = 2, 3, 6, 7, 10, 11$

Q Dataword 10111100101001

Ans $m = 14$

For $r = 5$
$2^5 \geq 14 + 5 + 1$

$\Rightarrow 32 \geq 20$

$\Rightarrow$ Redundant bits are $2^0, 2^1, 2^2, 2^3, 2^4$

_	_	1	_	0	1	1	_	1	1	0	0	1	0	1	_	0	0	1
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

__/__/__

$$R_1 = \frac{}{1}, \frac{1}{3}, \frac{0}{5}, \frac{1}{7}, \frac{1}{9}, \frac{1}{11}, \frac{1}{13}, \frac{1}{15}, \frac{0}{17}, \frac{1}{19} = 0$$

$$R_2 = \frac{}{2}, \frac{1}{3}, \frac{1}{6}, \frac{1}{7}, \frac{1}{10}, \frac{0}{11}, \frac{0}{14}, \frac{1}{15}, \frac{0}{18}, \frac{1}{19} = 0$$

$$R_4 = \frac{}{4}, \frac{0}{5}, \frac{1}{6}, \frac{1}{7}, \frac{0}{12}, \frac{1}{13}, \frac{0}{14}, \frac{1}{15} = 0$$

$$R_8 = \frac{}{8}, \frac{1}{9}, \frac{1}{10}, \frac{0}{11}, \frac{0}{12}, \frac{1}{13}, \frac{0}{14}, \frac{1}{15} = 0$$

$$R_{16} = \frac{}{16}, \frac{0}{17}, \frac{0}{18}, \frac{1}{19} = 1$$

0	1	1	0	0	1	1	1	1	0	0	0	1	0	1	1	0	0	1
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

(In reciever side when 10 bit has error)

0 1 0 1 0
16 8 4 2 1
10