Channel Coding

The purpose of channel coding is:-

- To protect information from channel noise, distortion and jamming which is the subject of error detection and correction codes.
- To protect information from 3rd party "enemy" which is the subject of encryption scrambling.

Error Detecting and Correcting Codes:

The basic idea behind these codes is to add extra bits (digits) to information such that the receiver can use it to detect and correct errors with limited capabilities, these extra bits are called parity or check or correction bits.

If for k digits, r parity digits are added then the transmitted digits n = k+r. We will have r redundant digits and the code is called (n,k) code.

k = No. of information bits.
r = No. of check bits.
n = No. of transmitted bits.

With code efficiency or rate of ($\frac{k}{n}$). In general, the ability of detection or correction depends on :

1. The technique used.
2. n,k parameters.

Simple Error Detecting Codes:

The simplest error detection schemes are the well-known ***even & odd parity generators.***

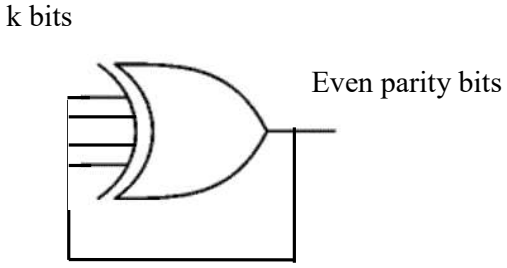
- For even parity generators, an extra bit is added for each k information bits such that the number of ones is even.
- At the receiver, an error is detected if the number of ones is odd. Either no error occurs or even number of errors occur.
- Hence, Prob. (detecting errors)=Prob. (odd number of errors).
  Prob. (Un detecting errors)=Prob. (even number of errors).

- For Odd parity generators, the same idea can be applied when number of ones is adjusted to be odd.

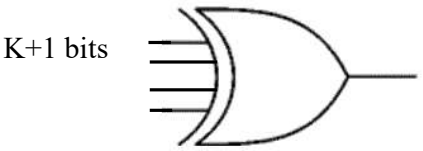
- The code efficiency of even and odd parity generators is $\frac{k}{k+1}$
- To implement these parity generators, simple Ex-OR gates are used at the transmitter and receiver as shown :

At transmitter

information bits	even parity
1010111	1
0110101	0
1011111	0
1101000	1



At receiver



- Hence, we can conclude the error detection is not ideal. It doesn't detect errors 100%.

- Note that the advantage of error detection is clear when (it is with Automatic Repeat Query ARQ systems).
- In these systems, two channels are used, the usual forward channel with error detection and backward channel is used to inform transmitter to retransmit the same data so that in the next transmission, data is received correctly.

Error Correcting Codes :

In order to make the receiver have the ability to detect and correct errors, not only a single checking (parity) bit is used but instead r bits are used giving what is called (n,k) code.

Basic Definitions:

1. **Systematic and non-systematic codes:-**
   If information bits $a_1, a_2, \ldots\ldots a_k$ are unchanged in their values and positions at the transmitted codeword then this code is called **systematic code.**

   Input data [D]=[$a_1, a_2, \ldots\ldots a_k$]

   Output systematic (n,k) codeword is

   [C]=[$a_1, a_2, \ldots\ldots a_k\ c_1, c_2, \ldots. c_r$]

   However, if data bits separated (spread) or changed at the output codeword. This **code** is said to be **non-systematic**.

   The output of non-systematic (7,4) code is

   [C]=[$c_2\ \ a_1\ \ c_3\ \ a_2\ \ c_1\ \ a_4\ \ a_3$]

2. **Hamming Distance**:-
   the ability of error detection and correction depends on this parameter, the hamming distance between any two codes c_i and c_j is denoted by d_{ij} which is the number of bits that differ between theses codes for a binary (n,k) code with 2^k possible codeword, then minimum hamming distance (H.D) is the minimum d_{ij}.

   Also note that $n \geq d_{ij} \geq 0$

   Ex: find hamming distance for the three code words
   [C_1]=[1011100] , [C_2]=[1011001] , [C_3]=[1011000]

   Sol:-
   $d_{12} = 2$ $d_{13} = 1$ $d_{23} = 1$

3. **Hamming weight:-**
   This is the number of ones in the non-zero codeword c_i. it is denoted by w_i.

   For linear block codes $w_{min} = H.D = \min(d_{ij})$
   - for example
     $[C_1]=[011110]$, $w_1 = 4$
     $[C_2]=[100001]$, $w_2 = 2$ and so on.

4. **Linear and Non-linear Codes:-**
   When the " r " parity bits are obtained from a linear function of the k information bits, then the code is said to be **linear**, other wise it is **non-linear**.

   For example
   $[C]=[\, a_1\ a_2\ a_3\ c_1\ c_2\ ]$, $c_1 = a_1 \,.\, a_2$

   $c_2 = a_1 \oplus a_2 \oplus a_3$

Linear Block Codes

The " r " parity bits are obtained using a linear function of a's data. Mathematically, this can be described by the set of equations:-

$$\begin{aligned} c_1 &= h_{11}.a_1 + h_{12}.a_2 + h_{13}.a_3 + h_{14}.a_4 + \ldots\ldots\ldots\ldots h_{1k}.a_k \\ c_2 &= h_{21}.a_1 + h_{22}.a_2 + h_{23}.a_3 + h_{24}.a_4 + \ldots\ldots\ldots\ldots h_{2k}.a_k \\ &\vdots \\ &\vdots \\ &\vdots \\ c_r &= h_{r1}.a_1 + h_{r2}.a_2 + h_{r3}.a_3 + h_{r4}.a_4 + \ldots\ldots\ldots\ldots h_{rk}.a_k \end{aligned} \quad \ldots..(1)$$

Note
[+] is mod-2 (EX-OR) addition and [.] is the AND gate multiplication & h_{ij} coefficients are binary variable for binary coding.

O/P codeword [C]= [D][G]

[D] : $[\, a_1, a_2, \ldots\ldots\ a_k\,]$

[G] : generator matrix

$$[G]=\begin{bmatrix} 1 & 0 & 0 & 0 & h_{11} & h_{21} & h_{31} & .... & h_{r1} \\ 0 & 1 & 0 & 0 & h_{12} & h_{22} & h_{32} & .... & h_{r2} \\ 0 & 0 & 1 & 0 & : & : & : & : & : \\ 0 & 0 & 0 & 1 & h_{1k} & h_{2k} & h_{3k} & .... & h_{rk} \end{bmatrix} = [I_k : P_{kxr}]$$

This matrix is called generator matrix of a Linear Block Code (LBC). Equation (1) can also be written in matrix form as:

$[H].[C]^T = [0]$ …….(2)

Where $[C]=[a_1, a_2, \ldots\ldots a_k\ c_1, c_2, \ldots c_r]$ and [H] matrix is related to [G] by :

$[H]=\left[-P_{kxr}^T : I_{rxr}\right]$ for binary coding the (-) signal drops out.

This rxn [H] matrix is called parity checking matrix as will be shown, encoding can be done either using eq. (1) [[G] matrix] or eq. (2) [[H] matrix but the decoding is done using [H] matrix only.

Hamming Bound :

The number of parity bits " r " added to have certain error correction capability is chosen by an inequality known as the humming bound. For binary codes

$$2^{n-k} = 2^r \geq \sum_{j=0}^{t} C_j^n$$

Where t is the number of bits to be corrected.

- For example if k=4, then to correct single error t=1 then:

$$2^r \geq \sum_{j=0}^{1} C_j^{4+r}$$

$$2^r \geq C_0^{4+r} + C_1^{4+r}$$

$$C_n^m = \frac{m!}{n!.(m-n)!}$$

This gives $2^r \geq 1+(4+r)$ and the minimum r is r=3 (take minimum r to have max code efficiency). This is (7,4) code note that the equality is satisfied in this example and the code is said to be perfect.

Another example if k=5 and t=3 where three errors are corrected then:

$$2^r \geq \sum_{j=0}^{3} C_j^{5+r}$$

$$2^r \geq C_0^{5+r} + C_1^{5+r} + C_2^{5+r} + C_3^{5+r}$$

$$2^r \geq 1 + (5+r) + \frac{(5+r)(4+r)}{2} + \frac{(5+r)(4+r)(3+r)}{6}$$

The minimum r here is 10, and the code is (15,5). Since the equality is not satisfied then the code is not perfect.

Hamming code:

It is a single error correcting perfect code with the following parameters: $n=2^r-1$, HD=3, t=1. example of hamming code are (7,4),(15,11),(31,26). The hamming codes are encoded and decoded as a linear block codes.

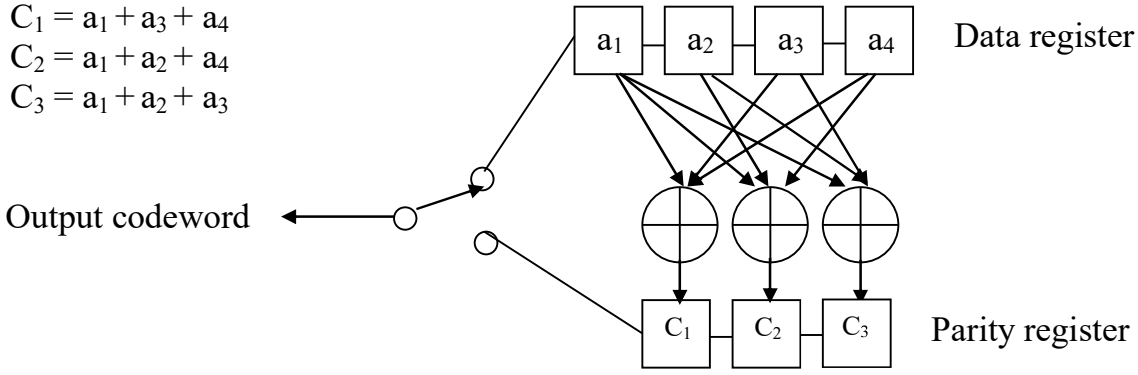
Encoding of Linear Block Codes:

Eq. (1) or eq. (2) can be implemented using EX-OR gates. Take for example a (7,4) hamming code with a parity checking matrix.

$$[H]=\begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$, and by using eq. (2) $[H][C]^T=[0]$ will give :

Where $[C]=[\, a_1 \;\; a_2 \;\; a_3 \;\; a_4 \; c_1 \; c_2 \;\; c_3]$

$C_1 = a_1 + a_3 + a_4$
$C_2 = a_1 + a_2 + a_4$
$C_3 = a_1 + a_2 + a_3$



The code table for this code can be found using the three equations of code word.

a_1	a_2	a_3	a_4	C_1	C_2	C_3	w_i
0	0	0	0	0	0	0	---
0	0	0	1	1	1	0	3
0	0	1	0	1	0	1	3
0	0	1	1	0	1	1	4
0	1	0	0	0	1	1	3
0	1	0	1	1	0	1	4
0	1	1	0	1	1	0	4
0	1	1	1	0	0	0	3
1	0	0	0	1	1	1	4
1	0	0	1	0	0	1	3
1	0	1	0	0	1	0	3
1	0	1	1	1	0	0	4
1	1	0	0	1	0	0	3
1	1	0	1	0	1	0	4
1	1	1	0	0	0	1	4
1	1	1	1	1	1	1	7

$W_{i\ min} = 3 = HD$

To find the number of errors that the (7,4) code can correct the following eq. can be used :

$$t = \frac{\text{int}(HD - 1)}{2}$$

i.e. t=int(3-1)/2 = 1 bit. Hence, this is a single error correcting code (Hamming code).

Ex: find the generator matrix for the previous LBC.

Sol:

$$[H]=\underbrace{\begin{bmatrix}1&0&1&1&1&0&0\\1&1&0&1&0&1&0\\1&1&1&0&0&0&1\end{bmatrix}}_{-P^T \quad I_r},$$

$$[G]=[I_k P^T]=\begin{bmatrix}1&0&0&0&1&1&1\\0&1&0&0&0&1&1\\0&0&1&0&1&0&1\\0&0&0&1&1&1&0\end{bmatrix}$$

Note that the equation [C]=[D][G] gives:

$[C]=[\,a_1 \;\; a_2 \;\; a_3 \;\; a_4 \;\; (a_1+a_3+a_4) \;\; (a_1+a_2+a_4) \;\; (a_1+a_2+a_3)]$

$[C]=[\,a_1 \;\; a_2 \;\; a_3 \;\; a_4 \;\; C_1 \;\; C_2 \;\; C_3]$ as obtained before

Decoding of Linear Block Codes :

If [R]=[C]+[E] is the received code, where [E] is the error word.

- If [E]=[0] then no error occurs.
- If [E]=[0 0 ….. 0 1 0], single error occurs at second position from the right.
- If [E]=[0 0 ……1 1 0], double errors occurs at second and third positions from the right, and so on.

The number of the errors can be corrected depend on (t) of the code.

If [R] is multiplied by [H] (the receiver must know [H]) then:

$$[H].[R]^T=[H][C]^T+[H][E]^T$$

Since

$[H][C]^T$ is set to [0] at the transmitter then define [S] vector:

$$[S]=[H].[R]^T=[H][E]^T$$

This [S] vector is called syndrome.

- If [S]=[0], the receiver decides on no errors.
- If [S]≠[0], then the receiver must use [S] to find [E] and use it to find the corrected code word.

[C]=[R]+[E] binary coding.

Decoding for single error:

For single error hamming code, [S] is fond by multiplying [H].$[R]^T$ and the result of the syndrome [S] is compared with each column in [H]. the column that matches the syndrome represent the position of the error.

Thereby, for single error correction, the parity checking matrix [H] must satisfies :

i. No all zero columns so as not to mix with no error case.
ii. No repeated columns so that the decoder can decode any received word correctly with single error only.

Ex: for (7,4) code, with [H]= $\begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$

a. Find the corrected word at the receiver, if the received word [R]=[1001111].
b. Find the syndrome vector if double errors occur at first and last positions
c. Draw the decoder circuit used to find [S].

Sol:

a. If [R]=[1001111]

$$[S]=[H][R]^T=\begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}=\begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

[S] is similar to the forth column in [H].
Then, [E]=[0001000] and the corrected code word

$$[C]=[R]+[E]=[1001111]+[0001000]=[1000111]$$

b. To find [S] for double errors, then $[S]= [H][E]^T$. where [E]=[1000001].

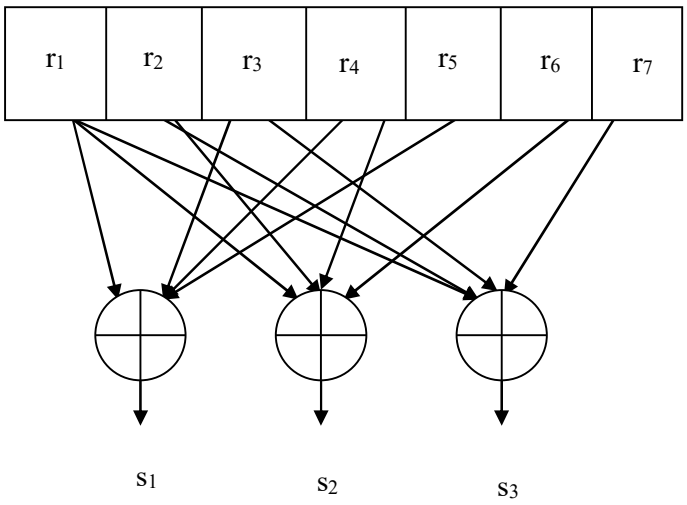
$$[S]=[H][E]^T=\begin{bmatrix}1&0&1&1&1&0&0\\1&1&0&1&0&1&0\\1&1&1&0&0&0&1\end{bmatrix}\begin{bmatrix}1\\0\\0\\0\\0\\0\\1\end{bmatrix}=\begin{bmatrix}1\\1\\0\end{bmatrix}$$

Note that the syndrome for double error is the same as the syndrome for double errors in this example. This assures that this code is capable of correcting single error only.

c. To draw the decoder circuit, the syndrome equations must be found

$$[S]=[H][R]^T=\begin{bmatrix}1&0&1&1&1&0&0\\1&1&0&1&0&1&0\\1&1&1&0&0&0&1\end{bmatrix}\begin{bmatrix}r_1\\r_2\\r_3\\r_4\\r_5\\r_6\\r_7\end{bmatrix}=\begin{bmatrix}s_1\\s_2\\s_3\end{bmatrix}$$

$s_1=r_1+ r_3+ r_{4+} r_5$
$s_2=r_1+ r_2+ r_{4+} r_6$
$s_3=r_1+ r_2+ r_{3+} r_7$



Ex:

[G]= $\begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}$ ***. Find the following:***

a. ***Use Hamming bound to find error correction capability.***
b. ***Find the parity check matrix.***
c. ***Find the code table, hamming weight and the error correction capability and compare it with a.***
d. ***If the received word is [R]=[1011110011], find the corrected word at the receiver.***

Sol:

a. n=10, k=3, r=7.
The code is (10,3). Using hamming bound
Let t=1

$$2^7 \geq \sum_{j=0}^{1} C_j^{10}$$

$$2^7 \geq C_0^{10} + C_1^{10}$$

$$128 \geq 11$$

Let t=2

$$2^7 \geq \sum_{j=0}^{2} C_j^{10}$$

$$2^7 \geq C_0^{10} + C_1^{10} + C_2^{10}$$

$$128 \geq 1 + 10 + (10 * 9 / 2)$$

Let t=3

$$2^7 \geq \sum_{j=0}^{3} C_j^{10}$$

$$2^7 \geq C_0^{10} + C_1^{10} + C_2^{10} + C_3^{10}$$

$$128 \leq 1 + 10 + (10 * 9 / 2) + (10 * 9 * 8 / 6)$$

When t=3 the hamming bound condition is disabled , there by we take t=2 where the condition is satisfied.

b. The parity check matrix is found using the generator matrix:

$$[H]=[P^T I]=\begin{bmatrix}1&0&0&1&0&0&0&0&0&0\\1&1&0&0&1&0&0&0&0&0\\0&1&1&0&0&1&0&0&0&0\\1&0&1&0&0&0&1&0&0&0\\1&1&1&0&0&0&0&1&0&0\\0&1&0&0&0&0&0&0&1&0\\0&0&1&0&0&0&0&0&0&1\end{bmatrix}$$

c. The equation $[H][C]^T=[0]$ gives:

$c_1=a_1$ $c_2=a_1+a_2$ $c_3=a_2+a_3$ $c_4=a_1+a_3$

$c_5=a_1+a_2+a_3$ $c_6=a_2$ $c_7=a_3$

a_1	a_2	a_3	c_1	c_2	c_3	c_4	c_5	c_6	c_7	w_i
0	0	0	0	0	0	0	0	0	0	----
0	0	1	0	0	1	1	1	0	1	5
0	1	0	0	1	1	0	1	1	0	5
0	1	1	0	1	0	1	0	1	1	6
1	0	0	1	1	0	1	1	0	0	5
1	0	1	1	1	1	0	0	0	1	6
1	1	0	1	0	1	1	0	1	0	6
1	1	1	1	0	0	0	1	1	1	7

$w_{i(min)}= 5 = HD$

i.e. t=int(5-1)/2 = 2 bit. Hence, this is a double error correcting code which agrees with part a.

d. If [R]=[1011110011], then:

$$[S]=[H][R]^T=\begin{bmatrix}1&0&0&1&0&0&0&0&0&0\\1&1&0&0&1&0&0&0&0&0\\0&1&1&0&0&1&0&0&0&0\\1&0&1&0&0&0&1&0&0&0\\1&1&1&0&0&0&0&1&0&0\\0&1&0&0&0&0&0&0&1&0\\0&0&1&0&0&0&0&0&0&1\end{bmatrix}\begin{bmatrix}1\\0\\1\\1\\1\\0\\0\\1\\1\end{bmatrix}=\begin{bmatrix}0\\0\\0\\0\\0\\1\\0\end{bmatrix}$$

Which is similar to the ninth column in [H] from the left, the corrected code word [R]=[1011110001].