Linear Block Codes (LBC):

Only systematic binary codes will be described. The ***r*** parity bits are obtained using a linear function of the **a**'s data. Mathematically, this can be described by the set of equations:

$$c_1=h_{11}a_1+h_{12}a_2+h_{13}a_3+\ldots\ldots+h_{1k}a_k$$
$$c_2=h_{21}a_1+h_{22}a_2+h_{23}a_3+\ldots\ldots+h_{2k}a_k$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \ldots\ldots\ldots\ldots\ldots\ldots(1)$$
$$c_r=h_{r1}a_1+h_{r2}a_2+h_{r3}a_3+\ldots\ldots .+h_{rk}a_k$$

Where + is mod-2 addition (EX-OR), product is the AND multiplication and h_{ij} coefficients are binary variables for a binary coding. The complete output codeword can be written in matrix form as:

[C]= [D][G] .....(1) ,

where:

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

$P_{k\times r}$] which is k × n matrix.

This matrix is called the **generator matrix** of the linear block code (LBC). Equation(1) can also be written in matrix form as:

[H] $[C]^T$=[o] .....(2)

where: [C]=[a_1 a_2 a_3 .....a_k c_1 c_2 c_3 .....c_r] and [H] matrix is in fact related with [G] matrix by:

[H]=[-$P^T_{r\times k}$: I_r], and for binary coding this – sign drops out. This r×n [H] matrix is called the **parity check matrix**. As will be shown, encoding can be done either using eq(1) ([G] matrix) or eq(2) ([H] matrix), but decoding is done using [H] matrix only.

Encoding of Linear Block codes:

Encoding can be done either using eq(1) ([G] matrix) or eq(2) ([H] matrix) using EX-OR gates.

Example: a given binary (7,4) Hamming code with a parity check matrix:

$$[H]=\begin{bmatrix}1&0&1&1&1&0&0\\1&1&0&1&0&1&0\\1&1&1&0&0&0&1\end{bmatrix}$$

Find: 1) encoder circuit 2) all possible codewords (code table) 3) error correction capability.

Solution : We know

$$[H]=\left[\begin{array}{cccc:ccc}1&0&1&1&1&0&0\\1&1&0&1&0&1&0\\1&1&1&0&0&0&1\end{array}\right]$$

$$-P^{T}_{r\times k} \qquad I_r$$

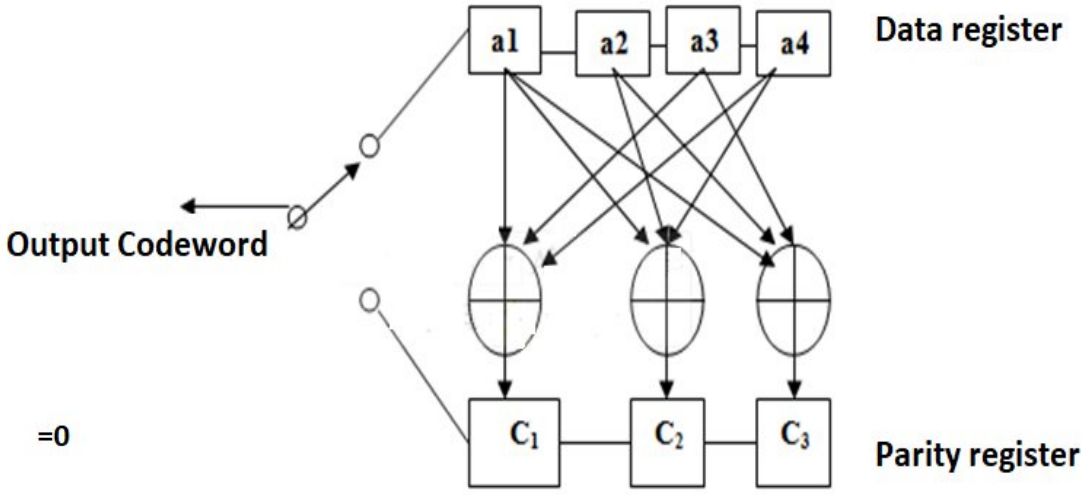
: [C]=[a1 a2 a3 a4 c1 c2 c3]

Using eq(2), $[H][C]^T=[0]$ will give

$$[H]=\left[\begin{array}{cccc:ccc}1&0&1&1&1&0&0\\1&1&0&1&0&1&0\\1&1&1&0&0&0&1\end{array}\right]\left|\begin{array}{c}a1\\a2\\a3\\a4\\c1\\c2\\c3\end{array}\right.=0$$

$$-P^{T}_{r\times k} \qquad I_r$$

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



Above equations for C's are used to find the code table for this code as $\ell_i(\min)=3=HD$, i.e. t= int(3-1)/2=1 bit. Hence, this is a single error correcting code(Hamming code).

Example: Find the generator matrix for the previous LBC.

Solution: r=3 , k=4 ,n=7

$$[H]=\begin{bmatrix}1&0&1&1&1&0&0\\1&1&0&1&0&1&0\\1&1&1&0&0&0&1\end{bmatrix}$$

($P_{r\times k}$ | $I_{r\times r}$)

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

($I_{k\times k}$ | $P_{k\times r}$)

$$[C]=[D][G]=[a_1\ a_2\ a_3\ a_4]\begin{bmatrix}1&0&0&0&1&1&1\\0&1&0&0&0&1&1\\0&0&1&0&1&0&1\\0&0&0&1&1&1&0\end{bmatrix}$$

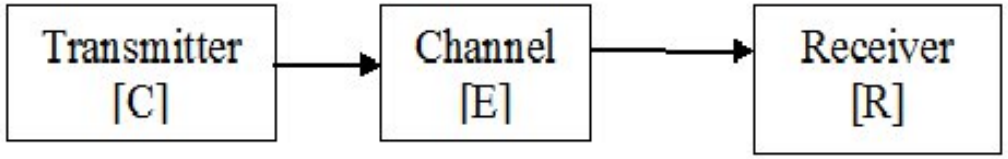
Note that the equation [C]=[D][G] gives:
$[C]=[a_1\ a_2\ a_3\ a_4\ (a_1+a_3+a_4)\ (a_1+a_2+a_4)\ (a_1+a_2+a_3)]=[a_1\ a_2\ a_3\ a_4\ c_1\ c_2\ c_3]$ as obtained before.

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

a_1	a_2	a_3	a_4	c_1	c_2	c_3	ω_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

The received codeword at Rx can be written as

$$[R]=[C]+[E]$$



Where [R] is the received word (***n***-vector)
[E] is the error word (***n***-vector)

- 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 codeword

Hence the corrected codeword is

$$[C]=[R]+[E]$$

Of course, [S] is calculated from [R]. The problem is now how to calculate the [E] from syndrome [S] ?

Simple decoding procedure for single error:

For single error Hamming codes, above mathematical solution is reduced into comparing the [S] ***r***-vector with all columns of the [H] matrix (2^r-1 non zero and non repeated columns). That column similar to [S] is the position of 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.

.

Example: For previous example, (7,4) code with [H] matrix given below [1]-Find the corrected word at the receiver if the received word [R]=[1001111]. [2]-Find the syndrome vector if double errors occur at 1st and last positions, comment. [3]- Draw the decoder circuit used to find the syndrome vector[S].

$$[H]=\begin{bmatrix}1&0&1&1&1&0&0\\1&1&0&1&0&1&0\\1&1&1&0&0&0&1\end{bmatrix}$$

Solution: the parity check matrix is:

If the received word is [R]=[1001111] then, **[S]= [H].[R]^T**

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

which is similar to the 4th column in [H].

Hence the corrected word [C]=[R]+[E]=[1001111]+[0001000]= [1 0 0 0 1 1 1]

0	0	0	0	0	0	0
0	0	0	1	1	1	0
0	0	1	0	1	0	1
0	0	1	1	0	1	1
0	1	0	0	0	1	1
0	1	0	1	1	0	1
0	1	1	0	1	1	0
0	1	1	1	0	0	0
1	0	0	0	1	1	1
1	0	0	1	0	0	1
1	0	1	0	0	1	0
1	0	1	1	1	0	0
1	1	0	0	1	0	0
1	1	0	1	0	1	0
1	1	1	0	0	0	1

Decoding procedure for single error:

[2]-To find the syndrome vector[S] for double errors, then $[S]=[H][E]^T$. Where [E]=[1000001] corresponding to double errors at 1st and last positions. Then:

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

Note that the syndrome for single error at the 4th position is the same as the syndrome for double errors at the 1st and last positions. This indicates that this code is only capable of correcting single error as expected.

[3]. To draw the decoder circuit, the syndrome equations must be found

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

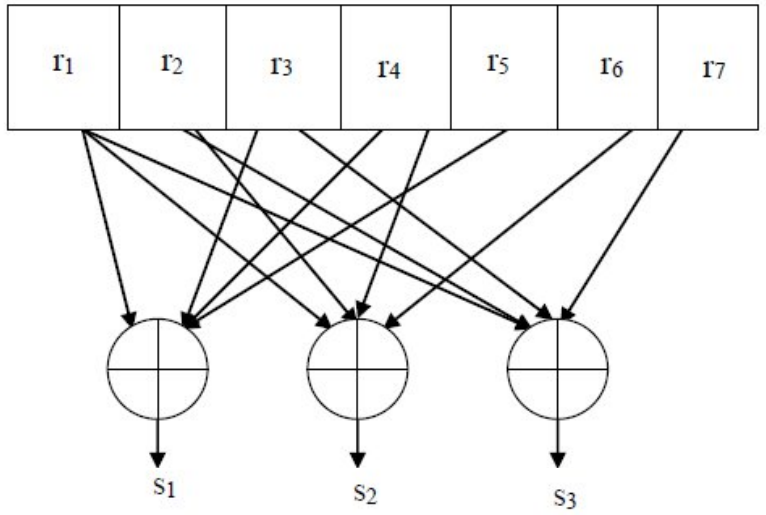
Which gives

$S_1 = r_1 + r_3 + r_4 + r_5$

$S_2 = r_1 + r_2 + r_4 + r_6$

$S_3 = r_1 + r_2 + r_3 + r_7$

Receiver register



Example:

The generator matrix of a LBC is given by:

$$[G]=\begin{bmatrix}1&0&0&1&1&0&1&1&0&0\\0&1&0&0&1&1&0&1&1&0\\0&0&1&0&0&1&1&1&0&1\end{bmatrix}$$

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 then compare with part(a). **d**-If the received word is [R]=[1011110011], find the corrected word at the Rx.

Solution: (a) n=10, **k**=3, **r**=7 , (10,3) code. Using Hamming bound for t=0, 1, 2, 3,.....

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

$$2^7 \geq C_0 10 + C_1 10 + C_2 10 + \ldots\ldots\ldots + C_t 10$$

That gives 128 > 1+10+(10*9/2), i.e t=2 double error correction. Note that single error (t=1)can be corrected here but triple errors (t=3) can not because the capability will exceed Hamming upper bound.

b. The parity check [**H**]matrix is found using the generator [**G**]matrix :

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

with no zero or repeated column

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	ω_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

ω_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:

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

Which is similar to the 9th column in [H] from the left, hence the corrected code word **[R]=[1011110001].**

These are subclass from the linear block codes. The name cyclic comes from the fact that any cyclic shift of a codeword is another codeword. i.e, if $[C_1]=[0011010]$ is a codeword then $[C_2]=[0001101]$ is another codeword obtained from $[C_1]$ by a right circular shift.

Cyclic codes can be classified into two types: **Systematic cyclic codes** and **Nonsystematic cyclic codes**

Generation of cyclic codes:

A) nonsystematic Cyclic Codes: (Multiplicative):
The output codeword is generated using polynomial multiplication.

Procedure:

(1) For $[D]=[a_1\ a_2\ .....\ a_k]$ data word, write the data word in terms of a power of a dummy variable $\boldsymbol{x}$ with a_1 weighted as MSB (Most Significant Bit) and $\boldsymbol{a_k}$ as LSB (Least Significant Bit).

x^{k-1}	x^{k-2}	x^2	x^1	x^0
a_1	a_2 ..... a_{k-2}	a_{k-1}	a_k	
MSB			LSB	

$$D(x)=a_k+a_{k-1}\,x+a_{k-2}\,x^2+.....+a_2\,x^{k-2}+a_1\,x^{k-1}$$

where "+" sign is mod-2 addition(Ex-OR)

$$x^4\ \ x^3\ \ x^2\ \ x^1\ \ x^0$$

For example if $[D]=[1\ 1\ 1\ 0\ 1]$, then $D(x)=1+x^2+x^3+x^4$ and

if $D(x)= x^6+x^2+1$ then $[D]=[1000101]$