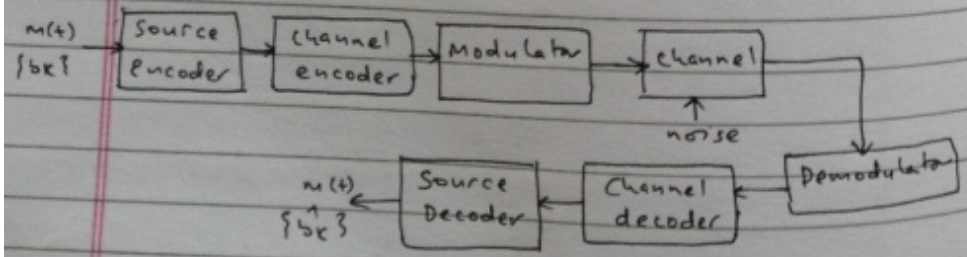
8. Error control coding techniques.

We now know that the error probability in digital communication is a function of input SNR and data rate r_b. In practice, maximum signal power as well as the channel bandwidth are restricted to some upper limit by government regulation.

But to reduce the error probability we either need to increase the signal power or reduce the noise. So, a practical method for decreasing the bit error rate or error probability is by using an error control coding technique. This error control coding is performed at by the channel encoder at the transmitting end and respective decoding is performed by channel decoder at receiver end.



The input message bits are first fed into source encoder where it removes any redundancy and produces a k-bits message codeword. Now this source encoded k-bits is fed to channel encoder which then gives out a n-bit codeword.

This n-bit codeword contains the k-bits message codeword as well as (n-k) bits of check bits. These extra bits are also known as parity bits.

So, data bits along with parity bits form a codeword which will be the output of a channel encoder.

Basically there are two types of error control techniques.

i) Automatic repeat request (ARQ)
→ receiver asks for retransmission of signal.

ii) Forward error correction (FEC)
→ adds parity bits for the decision making.

Thus FEC being more suitable we can further classify error correcting codes as,

i) Block codes
ii) Convolutional codes

i) Block codes.

In block codes, for a k-number of message bits we define a block code of size (n,k) where 'n' is the length of codeword produced for k-bits message and 'n-k' parity bits.

i.e. a block of k message bits is followed by (n-k) parity or check bits of a 'n' bit codeword.

These parity bits donot convey any message or information.

In block codes we basically deal with linear block codes.

Some important definitions related to codes.

a. codeword :

→ It is the n-bit encoded block of bits. It consists of message bits and parity bits.

b. code rate :

It is the ratio of number of message bits (k) to the total number of bits (n) in a codeword.

i.e. code rate (r) = k/n.

c. code vector :

It is the ~~number~~ ~~different~~ possible combinations of the n-bit codeword.

i.e. for 3 bit codeword we can have 8 possible codewords which are the code vectors for 3 bit codeword.

d. Hamming weight (HW).

It is the number of non-zero components in the codeword.

i.e. for 1100101, HW = 4.

e. Hamming distance (HD).

It is the number of locations in which the respective elements differs for two different code words of same code vector.

codeword (1) : 1 0 0 1 0 1 1
codeword (2) : 1 1 0 0 1 0 1

We see that the respective elements differ at 4 locations. therefore HD = 4.

f. Minimum distance (dm)

It is the smallest distance between any pair of codewords in the code vector. ie. smallest Hamming distance between any pair of codewords in the code vector.

The minimum distance is always the smallest Hamming weight of the non-zero code words in a code vector.

Now,

When $d_m \geq (s+1)$ we can detect upto 's' errors per codeword.

When $d_m \geq (2t+1)$, we can detect and correct upto 't' errors per codeword.

When $d_m \geq (t+s+1)$, then we can correct upto 't' errors and detect 's' errors per word. Here $s > t$.

So, for a 3-bit code word, the combination of codes with dm = 2 are,

000, 101, 110 and 011.

Now,

$d_m = 2$ and $d_m \geq (s+1)$

and i.e. $2 \geq (s+1)$

Therefore for dm = 2, we can detect maximum '1' error.

So, if we use the above given codewords any single error can easily be detected.

ie. if 000 was sent and there is error in any one position of bit we may receive,

001 or 010 or 100 which can easily be recognised as erroneous codeword as none of these exist in our list of codeword vector.

Linear Block codes.

It is one of the form of block codes where a block of 'k' number of bits represent the message bits and 'n-k' number of bits are used for parity such that the channel encoder produces a 'n' bit codeword.

A channel encoder thus adds the parity bits to the block of message bits and these parity bits are computed from the message bits following some prescribed encoding rule.

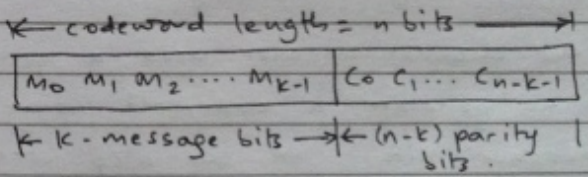


fig. Structure of the codeword for a linear block code.

i.e. message bits [group of 'k' bits] → channel encoder → n-bit codewords

let [X] be the code vector such that the elements of the codewords are,

$x_0, x_1, x_2, \ldots, x_{n-1}$

So,

$x_i = m_i$ for $i = 0, 1, \ldots, k-1$

$= c_{i-k}$ for $i = k, k+1, \ldots, n-1$.

And thus,

$[X] = [M : C]$

where, M = k - message vectors

C = (n-k) - parity vectors.

Now, as the channel encoder uses a generator matrix to generate a codeword using block of message bits, we can also write,

$X = M \cdot G$

or $[x]_{1\times n} = [M]_{1\times k} \cdot [G]_{k\times n}$

where,

$[x]_{1\times n}$ = code vector of size $1\times n$

$[M]_{1\times k}$ = message vector of size $1\times k$

& $[G]_{k\times n}$ = Generator vector of $k\times n$ size.

Now, this generator matrix is generally represented as,

$[G] = [I_k \mid P]$

where,

I_k = $k\times k$ identity matrix

& P = $k \times (n-k)$ coefficient matrix that can have 1 or 0 as its value.

Therefore,

$$I_k = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}_{k\times k}$$

$$P = \begin{bmatrix} P_{00} & P_{10} & \cdots & P_{n-k-1,0} \\ P_{01} & P_{11} & & P_{n-k-1,1} \\ \vdots & & & \\ P_{0(k-1)} & P_{1(k-1)} & & P_{n-k-1,(k-1)} \end{bmatrix}_{k\times(n-k)}$$

Now,

$[X] = [M : C] = MG = [M] \cdot [G] = [M][I_k \mid P]$

or $[M : C] = P[M \cdot [I_k : M \cdot P]$

$\therefore$ The parity vector,

$C = M \cdot P$

i.e.

$$[c_0, c_1, \cdots c_{n-k-1}]_{1\times n-k} = [m_0\ m_1 \cdots m_{k-1}] \cdot \begin{bmatrix} P_{00} & P_{10} & P_{(n-k-1)0} \\ P_{01} & P_{11} & P_{(n-k-1)1} \\ P_{0(k-1)} & P_{1(k-1)} & P_{(n-k-1)(k-1)} \end{bmatrix}_{k\times n-k}$$

$\therefore c_0 = m_0 \cdot P_{00} \oplus m_1 \cdot P_{10} \oplus \cdots \oplus m_{k-1} \cdot P_{0(k-1)}$

$c_1 = m_0 P_{10} \oplus m_1 \cdot P_{11} \oplus \cdots \oplus m_{k-1} \cdot P_{1(k-1)}$

$c_{n-k-1} = m_0 \cdot P_{(n-k-1)0} + m_1 \cdot P_{(n-1-k)1} \oplus \cdots \oplus m_{k-1} \cdot P_{(n-1-k)(k-1)}$

In the above equation the addition is a modulo-2 addition.

(#) The generator matrix for a (6,3) block code is given below. Find all the code vectors.

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

Here,

(n,k) block code = (6,3) block code,

i.e. $n = 6$ and $k = 3$

i.e. the codeword has 6 bits of which 3 bits are message bits.

Now, $G = [I_k \mid P]$

$= [I_3 \mid P]$

i.e. $I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ & $P = P_{3\times 3} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$

Now,

$C = M \cdot P.$

or $[c_0, c_1, c_2] = [m_0, m_1, m_2] \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$

$\therefore$

$c_0 = m_0 \cdot 0 \oplus m_1 * 1 \oplus m_2 \cdot 1 = m_1 \oplus m_2$

$c_1 = m_0 \cdot 1 \oplus m_1 \cdot 0 \oplus m_2 \cdot 1 = m_0 \oplus m_2$

$c_2 = m_0 \cdot 1 \oplus m_1 \cdot 1 \oplus m_2 \cdot 0 = m_0 \oplus m_1$

So, if $(m_0, m_1, m_2) = (0,0,0)$

$(c_0, c_1, c_2) = (0,0,0)$.

$\therefore$ codeword [X] = ~~0 0 0~~ [0 0 0 0 0 0]

if $(m_0, m_1, m_2) = (1,1,1)$

$(c_0, c_1, c_2) = (0,0,0)$

$\therefore$ codeword [X] = [111000]

(#) Syndrome calculation :

Now at the receiver end, for error detection and correction we use parity check matrix [H], such that [H] is defined as,

$$H = [P^T \mid I_{n-k}]$$

where,

P^T = Transpose of coefficient matrix [P]
= [n-k] x k size matrix.

I_{n-k} = (n-k) x (n-k) identity matrix.

Now,

$$H^T = \begin{bmatrix} P \\ I_k \end{bmatrix}$$

Also, $X = [M][G] = [M] \cdot [I_k : P_k]$

$\therefore$ $X \cdot H^T = 0$. indicates that the codeword received is error free.

Suppose we receive a codeword R such that $R = X + E$

where E is the error vector.

The receiver in such case won't know among R which is error E.

The decoding is thus done by multiplying R with H^T and the result is called syndrome S, such that,

$$S = R \cdot H^T$$
$$= [X+E] \cdot H^T$$
$$= X \cdot H^T + E \cdot H^T$$

$\therefore$ $S = E \cdot H^T$ as $X \cdot H^T = 0$.

(#) An error control code has the following parity check matrix.

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

i) Determine the generator matrix
ii) Find the codeword for message [101].
iii) Decode the received codeword [100110].

We know, $H = [P^T \mid I_{n-k}]$

and from question,

$$[H]_{3\times 6} = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

$$\therefore P^T = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \quad \& \quad I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

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

(i) Now, generator matrix,

$$G = [I_k \mid P]$$

$$= \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

ii) Given, $M = [1\ 0\ 1]$.

and $C = M \cdot P$.

$$\text{or } [C_0\ C_1\ C_2] = [1\ 0\ 1] \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

$\therefore C_0 = 1\cdot 1 \oplus 0\cdot 0 \oplus 1\cdot 1 = 0$

$C_1 = 1\cdot 1 \oplus 0\cdot 1 \oplus 1\cdot 0 = 1$

$C_2 = 1\cdot 0 \oplus 0\cdot 1 \oplus 1\cdot 1 = 1$

$\therefore X = [M : C]$

$= [1\ 0\ 1\ 0\ 1\ 1]$

iii) The received codeword $R = [1\ 0\ 0\ 0\ 1\ 1]$.

The syndrome is given by,

$S = R \cdot H^T$

or

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

$\therefore S = 1\cdot 1 \oplus 0\cdot 0 \oplus 0\cdot 1 \oplus 0\cdot 1 \oplus 1\cdot 0 \oplus 1\cdot 0,$

$1\cdot 1 \oplus 0\cdot 1 \oplus 0\cdot 0 \oplus 0\cdot 0 \oplus 1\cdot 1 \oplus 1\cdot 0,$

$1\cdot 0 \oplus 0\cdot 1 \oplus 0\cdot 1 \oplus 0\cdot 0 \oplus 1\cdot 0 \oplus 1\cdot 1$

$= [1\ 0\ 1]$

Therefore there is an error received.

Now, if a linear block code of size (n, k) has $d_{min} = 3 = q$, then such the resultant codewards are called Hamming code where,

$$n = 2^{d_{min}} - 1 \quad \Leftarrow \quad 2^q - 1$$
$$= 7$$

and, $k = 2^q - q - 1$
$= 8 - 3 - 1 = 4.$

and no. of parity bits $= q = 3.$

(#) The parity check matrix is given as,

$$H = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}_{3 \times 6}.$$

Calculate the syndrome vector for single bit error.

We have,

$$S = E \cdot H^T.$$

$$S = [E]_{1\times 6} [H^T]_{6\times 3}$$

for single bit error, the possible error vectors are,

1 0 0 0 0 0 → first bit error
0 1 0 0 0 0 → 2nd bit error
0 0 1 0 0 0 → 3rd bit error
0 0 0 1 0 0 → 4th bit error
0 0 0 0 1 0 → 5th bit error
0 0 0 0 0 1 → 6th bit error

so for error at first bit;

$$S_1 = [1\,0\,0\,0\,0\,0] \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = [1\,1\,0]$$

$$S_2 = [0\,1\,0\,0\,0\,0] \cdot [H^T] = [0\,1\,1]$$

$$\vdots$$

$$S_6 = [0\,0\,0\,0\,0\,1] \cdot [H^T] = [0\,0\,1]$$

$\therefore$ Syndrom vectors are,

$$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

which is same as H^T. Therefore the syndrom vector can indicate the position of error if there exists one.

Binary cyclic codes.

Binary cyclic codes are special form of linear block codes when encoding, and syndrome calculations can easily be implemented using simple shift registers.

A ~~linear~~ block code is said to be cyclic if it exhibits two fundamental properties.

1. linearity : The sum of two codewords is also a codeword.
2. Cyclic : Any cyclic shift of a codeword is also a codeword.

let $[x_0, x_1, x_2 x_3 \ldots x_{n-1}]$ be a codeword of an $[n, k]$ linear block code. Then it can be expressed in a form of codeword polynomial as,

$$X[p] = x_0 \cdot p^0 + x_1 p^1 + x_2 \cdot p^2 + \ldots + x_{n-1} p^{n-1}$$

p = any arbitrary real variable '1' or '0'.

Now the shifted codeword is,

$$X_1[P] = x_{n-1} + x_0 P^1 + x_1 \cdot P^2 + x_2 \cdot P^3 + \ldots + x_{n-2} \cdot P^{n-1}$$

i.e. $X_1 = [x_{n-1}\ x_0\ x_1\ x_2 \cdots x_{n-2}]$

Now, the generator polynomial for cyclic code is represented by $G[P]$.

In an (n,k) cyclic code there exists one and only generator polynomial given by,

$$G[P] = 1 + \sum_{i=1}^{n-k-1} G_i P^i + P^{n-k}$$

such that the degree of generator polynomial is equal to the number of parity bits in the codeword.

So we have,

$$X[P] = M[P] \cdot G[P]$$

where, $M(P)$: Message polynomial of degree less than or equal to K.

These codevectors generated directly from the multiplication of message and generator ~~polynom~~ polynomial is known an unsystematic codewords.

Now to generate systematic codeword first we need to equate,

$$\frac{P^{n-k} \cdot M(P)}{G(P)} = Q(P) \oplus \frac{C(P)}{G(P)}$$

where,

$Q[P]$: Quotient polynomial

$C[P]$: Remainder polynomial

Finally, the systematic codeword can be generated as,

$$X[P] = P^{n-k} \cdot M[P] \oplus C[P].$$

(#) All the addition are modulo-2 addition. Also, the division requires ex-or calculation.

(#) Let $G[p] = 1 + p + p^3$ for a polynomial of (7,4) cyclic code. Find the systematic and non-systematic code vectors for message vector {1010}.

Here, $(n, k) = (7, 4)$ i.e. $n = 7$, $k = 4$.

Now,

$M = [1\ 0\ 1\ 0]$

$\therefore M[p] = 1 + 0 \cdot p + 1 \cdot p^2 + 0 \cdot p^3$

$= 1 + p^2$

And

$G[p] = 1 + p + p^3$

$\therefore X[p] = M[p] \cdot G[p]$

$= [1 + p^2][1 + p + p^3]$

$= 1 + p + p^3 + p^2 + p^3 + p^5$

$= 1 + p + p^2 + p^3 + p^3 + p^5$

$= 1 + p + p^2 + p^5 \quad \because [p^3 + p^3 = 0]$

$= 1 + p + p^2 + 0 \cdot p^3 + 0 \cdot p^4 + p^5 + 0 \cdot p^6$

$\therefore X = [1\ 1\ 1\ 0\ 0\ 1\ 0]_{1 \times 7}$

which is the unsystematic codeword.

Now, $p^{n-k} \cdot M[p]$

$= p^3 \cdot [1 + p^2]$

$= p^3 + p^5$

And, $\dfrac{p^{n-k} \cdot M[p]}{G[p]} = \dfrac{p^3 + p^5}{1 + p + p^3}$

$$p^3 + p + 1 \;\big)\; p^5 + p^3 \;\big(\; p^2$$
$$\oplus p^5 \oplus p^3 + p^2$$
$$+ p^2$$

i.e. $\dfrac{p^{n-k} \cdot M[p]}{G[p]} = p^2 \oplus \dfrac{p^2}{p^3 + p + 1}$ ($Q[p]$, $C[p]$)

$\therefore X[p] = p^{n-k} \cdot M[p] \oplus C[p]$

$= p^3 + p^5 + p^2$

$= 0 + 0 \cdot p + 1 \cdot p^2 + 1 \cdot p^3 + 0 \cdot p^4 + 1 \cdot p^5 + 0 \cdot p^6$

$\therefore X = [0\ 0\ 1\ \underbrace{1\ 0\ 1}_{\text{msg.}}\ 0]_{1 \times 7}$

which is the systematic codeword.

Generator and parity check matrices for cyclic codes.

We know that the cyclic codes are linear block codes. Therefore we can define the generator and parity check matrix for cyclic codes as well.

The generator matrix has a size of $k \times n$ ie. it has 'k' rows and 'n' columns.

We have

$$G[p] = 1 + \sum_{i=1}^{n-k-1} G_i p^i + p^{n-k}$$

Multiplying both sides by p^i we get,

$$p^i \cdot G[p] = p^i + p^i \cdot \sum_{i=1}^{n-k-1} G_i p^i + p^{n+i-k+i}$$

where,

$i = k-1, k-2, \ldots, 2, 1, 0$.

So, for different values of 'i' we get the polynomial for rows of generator matrix.

For a (7,4) cyclic code, determine the generator matrix if $G[p] = 1 + p + p^3$.

Here,

$n = 7, \quad k = 4 \quad$ hence $\quad n-k = 3$.

Now,

$G[p] = 1 + p + p^3$

multiplying both sides by p^i, we get,

$$p^i \cdot G[p] = p^i + p^{1+i} + p^{3+i}$$

Also,

$i = (k-1), \ldots, 1, 0 \quad = \quad 3, 2, 1, 0$.

So,

for $i = 3$, $\quad p^3 \cdot G[p] = p^3 + p^4 + p^6$

for $i = 2$ $\quad p^2 \cdot G[p] = p^2 + p^3 + p^5$

for $i = 1$ $\quad p^1 \cdot G[p] = p + p^2 + p^4$

for $i = 0$ $\quad p^0 \cdot G[p] = 1 + p + p^3$

$\therefore$ Row 1 : $i = 3 \rightarrow p^6 + 0p^5 + p^4 + p^3 + 0 \cdot p^2 + 0 \cdot p^1 + 0 \cdot p^0$

Row 2 : $i = 2 \rightarrow 0 \cdot p^6 + p^5 + 0 \cdot p^4 + p^3 + p^2 + 0 \cdot p^1 + 0 \cdot p^0$

Row 3 : $i = 1 \rightarrow 0 \cdot p^6 + 0 \cdot p^5 + p^4 + 0 \cdot p^3 + p^2 + p^1 + 0 \cdot p^0$

Row 4 : $i = 0 \rightarrow 0 \cdot p^6 + 0 \cdot p^5 + 0 \cdot p^4 + p^3 + 0 \cdot p^2 + 1 \cdot p^1 + 1$

$\therefore G = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$ → unsystematic

Now for systematic form of generator matrix

We have,

$G = [I_k : P]_{k \times n-k}$

i.e. there are 'k' rows in generator matrix.

The i^{th} row of 'G' can be represented as,

i^{th} row of $G = p^{(n-i)} + R_i(p)$ where $i = 1, 2, \cdots k$

Now,

$$\frac{p^{(n-i)}}{G(p)} = Q_i[p] \oplus \frac{c_i(p)}{G[p]}$$

where, $Q_i(p)$: quotient

$c_i(p)$: remainder

or,

$$p^{(n-i)} = G[p] \cdot Q_i(p) \oplus c_i[p]$$

∴

And $p^{(n-i)} \oplus c_i[p] = G[p] \cdot Q_i[p]$.

The equation above gives the i^{th} row of systematic generator matrix.

(#) For a systematic (7,4) cyclic code, determine the generator matrix and hence parity check matrix. Given $G[p] = p^3 + p + 1$

Here, $n = 7$, $k = 4$ and $n-k = 3$.

Now,

$$p^{(n-i)} \oplus c_i[p] = Q_i[p] \cdot G[p]$$

where $i = 1, 2, \cdots, k$.

Now,

for $i = 1$, we get,

$$p^{[7-1]} \oplus c_1[p] = Q_1[p] \cdot (p^3 + p + 1)$$

Now, $\frac{p^6}{1+p+p^3}$

$p^3+p+1 \;)\; p^6 \;(\; p^3+p+1$

$\oplus\; p^6 \oplus p^4 \oplus p^3$

$p^4 \oplus p^3$

$\oplus\; p^4 \oplus p^2 \oplus p$

$p^3 \oplus p^2 \oplus p$

$p^3 \oplus p \oplus 1$

$p^2 + 1$

∴ $Q_1(p) = p^3 + p + 1$

$c_1(p) = p^2 + 1$

∴ $p^6 + c_1(p) = p^6 + p^2 \times 1 \;(p^3+p+1)(p^3+p+1)$

$= p^6 + p^4 + p^3 + p^4 + p^2 + p + p^3 + p + 1$

$= p^6 + 0 \cdot p^5 + 0 \cdot p^4 + 0 \cdot p^3 + p^2 + 0 \cdot p + 1$

∴ 1^{st} Row = $[1\ 0\ 0\ 0\ 1\ 0\ 1]$

Now for i = 2,

$$\frac{p^5}{1+p+p^3} = p^3+p+1 \;\big)\; p^5 \;\big(\; p^2+1$$

$$\oplus p^5 \oplus p^3 \oplus p^2$$

$$p^3 \oplus p^2$$

$$p^3 \oplus p \oplus 1$$

$$p^2 \oplus p \oplus 1$$

$\therefore$

$p^5 \oplus p^3 \oplus p \oplus 1 =$

$p^5 \oplus C_2(p) = (p^2+1)(p^3+p+1)$

$= p^5 + p^3 + p^2 + p^3 + p + 1$

$= 0\cdot p^6 + 1\cdot p^5 + 0\cdot p^4 + 0\cdot p^3 + 1\cdot p^2 + 1\cdot p + 1$

$\therefore$ 3rd row = [0 1 0 0 1 1 1]

Similarly, for i = 3, $p^4 \oplus C_3(p) = p^4 + p^2 + p$

3rd row = [0 0 1 0 1 1 0]

for i = 4, $p^3 \oplus C_4(p) = p^3 + p + 1$

4th row = [0 0 0 1 0 1 1]

$\therefore$

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

I_k | P

Now, $H = [P^T : I_r]$

Now, $P^T = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{bmatrix}$

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

(#) Syndrome calculation.

We have $X[p] = x_0 + x_1 p + x_2 p^2 + \dots + x_{n-1} p^{n-1}$

Now, let the received polynomial,

$Y[p] = y_0 + y_1 p + y_2 p^2 + \dots + y_{n-1} p^{n-1}$

So,

$$\frac{Y[p]}{G[p]} = Q[p] + \frac{c[p]}{G[p]}.$$

or $Y[p] = Q[p] \cdot G[p] \oplus c[p]$.

The remainder $c[p]$ is now known as the syndrome polynomial. i.e. $S[p]$ = remainder of $\frac{Y[p]}{G[p]}$

$= c[p]$.

Now, if $S[p] = c[p] = 0$, there is no error.

Convolutional codes.

In block codes, the channel encoder accepts a k-bit message block and generates an n-bit codeword.

The codewords are thus produced on a block-by-block basis.

Whereas in convolutional coding, the message bits come in serially and passes through a required number of shift registers in channel encoder. The codewords are then generated from the incoming bit with respect to previous bits placed at the shift registers.

Thus in convolutional codes, the codewords depend on the previous bits of message unlike in block codes where the codeword is determined by the present value of i^{th} bits.

In convolutional codes we use 'N' number of shift registers and 'n' number of modulo-2 adders to generate the codewords.

For convolutional codes, 'n x N' product is known as constraint length.

So, for 'k' input message bit, as per the number of modulo-2 adder, we have 'n' number of code bits.

Suppose we have M = [101] and code block size (n,k) = (2,1) having constraint length n x N: 2 x 3 = 6.

let the modulo-2 outputs be defined as,

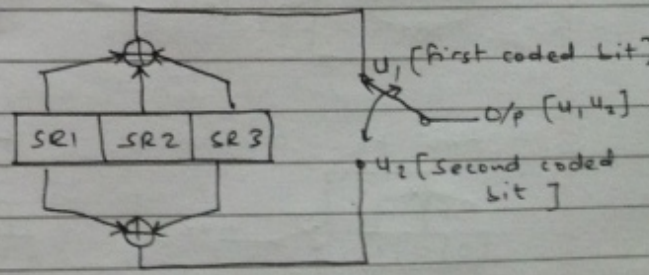
u_1 = SR1 ⊕ SR2 ⊕ SR3

u_2 = SR1 ⊕ SR3

Given, n = 2 : no. of modulo-2 adders

N = 3 : no. of shift registers.

The convolution diagram can now be drawn as,



As from the diagram there are three shift registers SR1, SR2 and SR3.

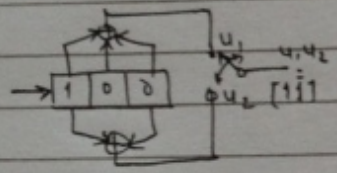
Here, SR1 takes the incoming data whereas the rest form the memory of encoder.

Now,

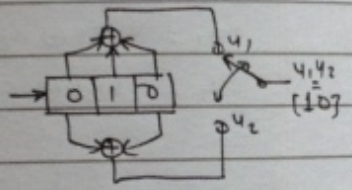
$u_1 = SR1 \oplus SR2 \oplus SR3$ &

$u_2 = SR1 \oplus SR3$

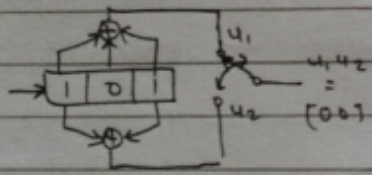
So, at time t_1,



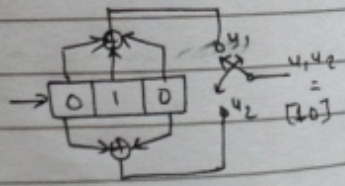
time t_2



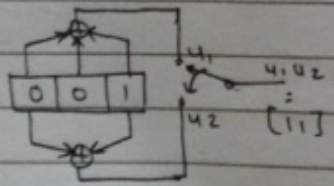
time t_3



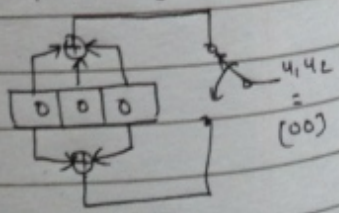
time t_4



time t_5



time t_6



$\therefore$ M[101] → [Encoder] → U = [11100010111].

Now, the two modulo-2 adders can be defined in ~~a~~ ~~poly~~ generator ~~polynomia~~ polynomial as,

$g_1 = 111 = 1 + x + x^2 : g_1(x)$

$g_2 = 101 = 1 + x^2 = g_2(x)$.

such that encoder output,

U[x] = output sequence

= M[x]. g_1[x] interlaced with M(x). $g_2(x)$

i.e. It is more like transform domain approach

Now, for M = 101

$M[x] = 1 + x^2$

And

$M[x] \cdot g_1[x] = [1+x^2][1+x+x^2]$

$= 1 \oplus x \oplus x^2 \oplus x^2 \oplus x^3 \oplus x^4$

$= 1 + x + x^3 + x^4$

$M[x] \cdot g_2[x] = [1+x^2][1+x^2]$

$= 1 \oplus x^2 \oplus x^2 \oplus x^4$

$= 1 + x^4$

$\therefore M[x]\,g_1[x] = 1 + x + 0 \cdot x^2 + x^3 + x^4$

$M[x]\,g_2[x] = 1 + 0 \cdot x + 0 \cdot x^2 + 0 \cdot x^3 + 1 \cdot x^4$

$\therefore U[x] = (11) + (10)x + (00)x^2 + (10)x^3 + (11)x^4$

$\therefore U = [11\ 10\ 00\ 10\ 11]$

Graphical representation for convolutional encoding.

1. Code tree:
2. Code trellis
3. State diagram.

Code tree.

Following the example provided for convolutional coding, we have three shift registers and two Modulo 2 adders such that,

$u_1 = SR1 \oplus SR2 \oplus SR3$

$u_2 = SR1 \oplus SR3$

Initially these shift registers are cleared such that it contains all zeros.

i.e.

SR3:0	SR2:0	SR1:0	← input message bit.

Now,

if input message bit is '0', the register contents will be,

0	0	0	← i/p bit '0' & will be,

0	0	1	← i/p bit '1'.

So, for i/p bit '0', $u_1 = 0$, $u_2 = 0$

$\therefore$ output codeword,

$u_1u_2 = 00$.

And for i/p bit '1', $u_1 = 1$, $u_2 = 1$

$\therefore u_1u_2 = 11$

So, for initial state '00', the next state can be '00' or '01' and the corresponding output codeword [00] & [11] respectively.

Now say we had '1' as input such that registers contents are,

SR3	SR2	SR1	← i/p bit.
0	0	1	

Now, if the second i/p is 0, we have,

SR3	SR2	SR1	→ $u_1u_2 = 10$
0	1	0	

and if second i/p is 1, we have,

SR3	SR2	SR1	→ $u_1u_2 = 01$
0	1	1	

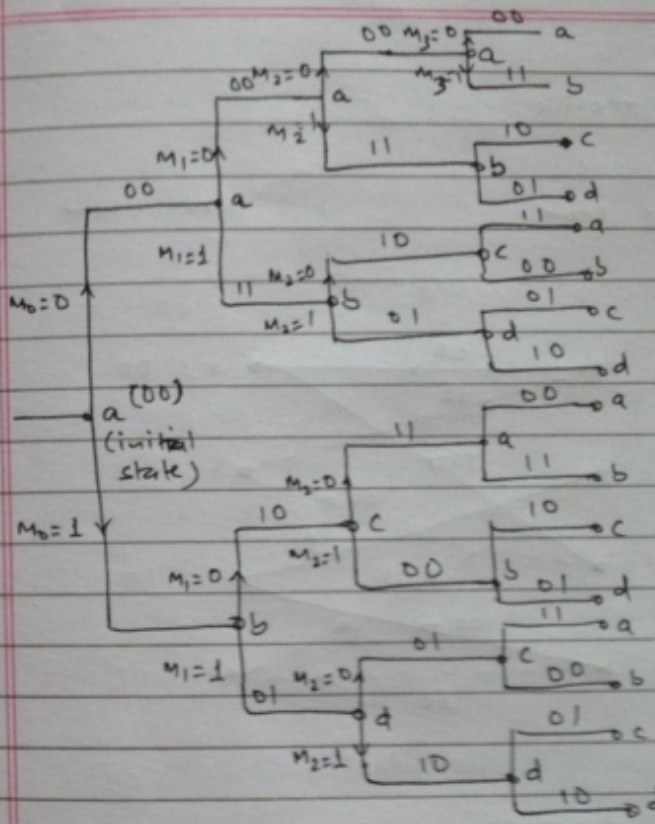
So, if the first input message is '1' then in such case the initial/current state is '01' and the next states are '10' and '11' with coded outputs [10] and [01] respectively.

Thus we can see that there are four possible states, namely,

States	States
00	a
01	b
10	c
11	d

So, we start a branch from the initial state de on the basis that if a '0' comes as input we take upper branch and if a '1' comes we take the lower branch.

So for $M = [m_0\ m_1\ m_2]$



The initial state or current state can be decided by the bits in SR3 SR2 and next state by bits in SR_2 SR_1.

ie.

SR3	SR2	SR1
0	1	0

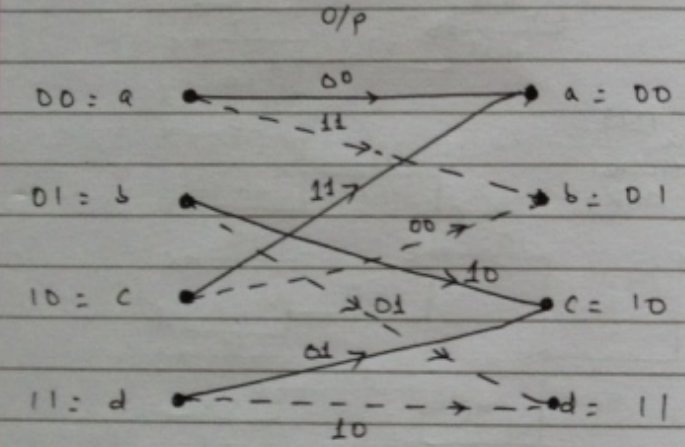
01 10

01 → current state

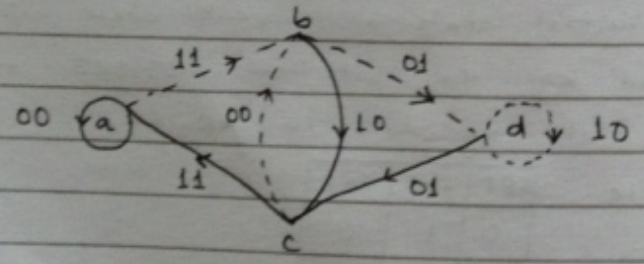
10 → next state.

Code trellis

Current state → Next state.



State diagram:



Decoding methods of convolutional codes.

Viterbi Algorithm.

let the transmitted code

$X = U_1 U_2 = [\,11\ 10\ 00\ 10\ 11\,]$.

Also the received signal be $Y = X$.

So, we now derive the path and path metrices for Viterbi algorithms.

We always start with initial state 'a' such that there are two possible next state. Graphically,