# Unit-4

## SOURCE AND ERROR CONTROL CODING

**Information Theory** → is used for mathematical modeling and analysis of the communication systems.

**Measure of Information**

The amount of information transmitted through the signal (message) $m_K$ with probability $P_K$ is given as

$$\text{Amount of information } \boxed{I_K = -\log P_K \quad (\text{or}) \quad \log_2\left(\frac{1}{P_K}\right)}$$

$P_K \Rightarrow$ Probability of occurance of $m_K$.

Data compression → Source coding
Data transmission → Channel coding.

**Channel capacity:** is defined as the average rate of information transmission across the channel when probability of occurrence of the event is maximum.

**Differentiate: uncertainty, information & Entropy**

| Uncertainty | Information | Entropy |
|---|---|---|
| ① It is probability of occurrence of the event | ① It is the content received due to occurrence of event. | ① It is the average information received due to occurrence of multiple events |
| ② Uncertainty of event decides amount of information. | ② Information received in certain event is zero. But information received in rare event is maximum. | ② Entropy is zero if the event is sure (or) impossible |

## Types of Sources

- Sources
  - Analog Source
  - Discrete source
    - Discrete Memoryless Sources (DMS)
    - Stationary Source

**Analog Source:** The o/p of these sources are analog Eg. Radio and TV broadcasting.

**Discrete Source:** The o/p of these sources are discrete Eg. Digital computer / Storage device.

**DMS:** For a discrete source if the current o/p is independent from all the past and future o/p. Then the source is called as DMS.

Eg: Binary source generated in a random sequence.

**Stationary source:** If the o/p depends on the past and future o/ps then it is called as the discrete Stationary Source.

Eg: Source generating the English text.

## * Entropy (or) average Information (H)

* It is defined as the average information per message. Denoted by 'H' and its units are bits / message.

* Entropy must be as high as possible in order to ensure maximum transfer of information.

$$\boxed{H = \sum_{K=1}^{M} p_K \log_2 \left(\frac{1}{p_K}\right)}$$

$P_K \rightarrow$ is the Probability of $K^{th}$ message

$M \rightarrow$ is the total number of memories generated by the source.

## Properties of Entropy

* Entropy is zero if Event is sure (or) impossible (ie) $H = 0$ if $P_K = 0$ (or) 1.
* For equally likely symbols source Entropy is given as
$$H = \log_2 M \text{ if } P_K = \frac{1}{M}$$
* Upper bound on the Entropy is given as, $H_{max} = \log_2 M$.

---

Problem:

* A source Emits 4 symbols with probabilities, $P_0 = 0.4$, $P_1 = 0.3$, $P_2 = 0.2$ and $P_3 = 0.1$. Find out the amount of information obtained due to these 4 symbols.

$$H = \sum P_K \log_2\left(\frac{1}{P_K}\right)$$

$$= P_1 \log_2\left(\frac{1}{P_1}\right) + P_2 \log_2\left(\frac{1}{P_2}\right) + P_3 \log_2\left(\frac{1}{P_3}\right) + P_4 \log_2\left(\frac{1}{P_4}\right)$$

$$= 0.4 \log_2\left(\frac{1}{0.4}\right) + 0.3 \log_2\left(\frac{1}{0.3}\right) + 0.2 \log_2\left(\frac{1}{0.2}\right) + 0.1 \log_2\left(\frac{1}{0.1}\right)$$

$$\boxed{H = 1.846 \text{ bits/symbol}}$$

$$\log_2 \frac{1}{0.4} = \log_2 2.5 = \frac{\log 2.5}{\log 2} = 1.32192$$

$$\log_2 x = \frac{\log_{10} x}{\log_e 2} = 3.32 \log_{10} x$$

---

# Find the entropy of an Event of throwing a die

$$H = \sum P_K \log_2 \frac{1}{P_K} \qquad P_K = \frac{1}{6} \left\{\begin{matrix}\text{die has} \\ \text{6 faces}\end{matrix}\right\}$$

$$\therefore H = \frac{1}{6} \log_2 6 = \boxed{0.431 \text{ bits/symbol}}$$

## Source Coding Theorem

* The aim of the source coding is to reduce the data rate (to remove redundance information), that will improve the efficiency of the communication system.

**Shannon Fano algorithm** → used to Encode the message depending upon the Probabilities.

**Algorithm:**

**Step ①**: List the symbols in the decreasing probability order.

**Step ②**: Partition the symbol set into two such that the sum of the probabilities of each group are the same.

**Step ③**:
Assign '0' to each message in the upper set.
Assign '1' to each message in the lower set.

**Step 4**: Continue this processing until further partitioning is not possible.

**Problem**

* A discrete memory less source has 8 symbols with probability of occurrence as shown below. Construct the shannon Fano code and calculate the efficiency.

| $m_1$ | $m_2$ | $m_3$ | $m_4$ | $m_5$ | $m_6$ | $m_7$ | $m_8$ |
|---|---|---|---|---|---|---|---|
| $1/2$ | $1/8$ | $1/8$ | $1/16$ | $1/16$ | $1/16$ | $1/32$ | $1/32$ |

Solution:

| Message | Probability ($P_K$) | codeword | Length ($l_K$) |
|---|---|---|---|
| $m_1$ | 1/2 | 0 | 1 |
| $m_2$ | 1/8 | 1 0 0 | 3 |
| $m_3$ | 1/8 | 1 0 1 | 3 |
| $m_4$ | 1/16 | 1 1 0 0 | 4 |
| $m_5$ | 1/16 | 1 1 0 1 | 4 |
| $m_6$ | 1/16 | 1 1 1 0 | 4 |
| $m_7$ | 1/32 | 1 1 1 1 0 | 5 |
| $m_8$ | 1/32 | 1 1 1 1 1 | 5 |

Average Code length of the codeword (L)

$$L = \sum_{K=1}^{M} P_K l_K \quad \text{where } M = 8$$

$$= \frac{1}{2}\times 1 + \frac{1}{8}\times 3 + \frac{1}{8}\times 3 + \frac{1}{16}\times 4 + \frac{1}{16}\times 4 + \frac{1}{16}\times 4 + \frac{1}{32}\times 5 + \frac{1}{32}\times 5$$

$$\boxed{L = 2.3125 \text{ bits/symbol}}$$

Entropy of the source (H)

$$H = \sum_{K=1}^{M} P_K \log_2 \frac{1}{P_K}$$

$$= \frac{1}{2}\log_2 2 + \frac{1}{8}\log_2 8 + \frac{1}{16}\log_2 16 + \frac{1}{16}\log_2 16 + \frac{1}{16}\log_2 16 + \frac{1}{32}\log_2 32 + \frac{1}{32}\log_2 32$$

$$\boxed{H = 2.31 \text{ bits/symbol}}$$

$$\text{Efficiency} (\eta) = \frac{H}{L \log_2 D} = \frac{H}{L \underbrace{\log_2 2}_{=1}}$$

$D = 0$ & $1$
So $D = 2$

$$= \frac{2.31}{2.3125}$$

$$\boxed{\eta = 1}$$

## Huffman Coding

↳ * Performs better than Shannon-Fano Coding.

* This coding assigns a sequence of bits to each symbol of an alphabet roughly equal in length. It is one of the prefix code.

**Coding algorithms:**

Step ①: List the source symbols in the order of decreasing probabilities *

Step ②: The last 2 probabilities are added and combined into a new source symbol. The probability of the new symbol is place in the list in accordance with its value.

Step ③: Repeat the procedure until the final list of source symbol contains only two symbols

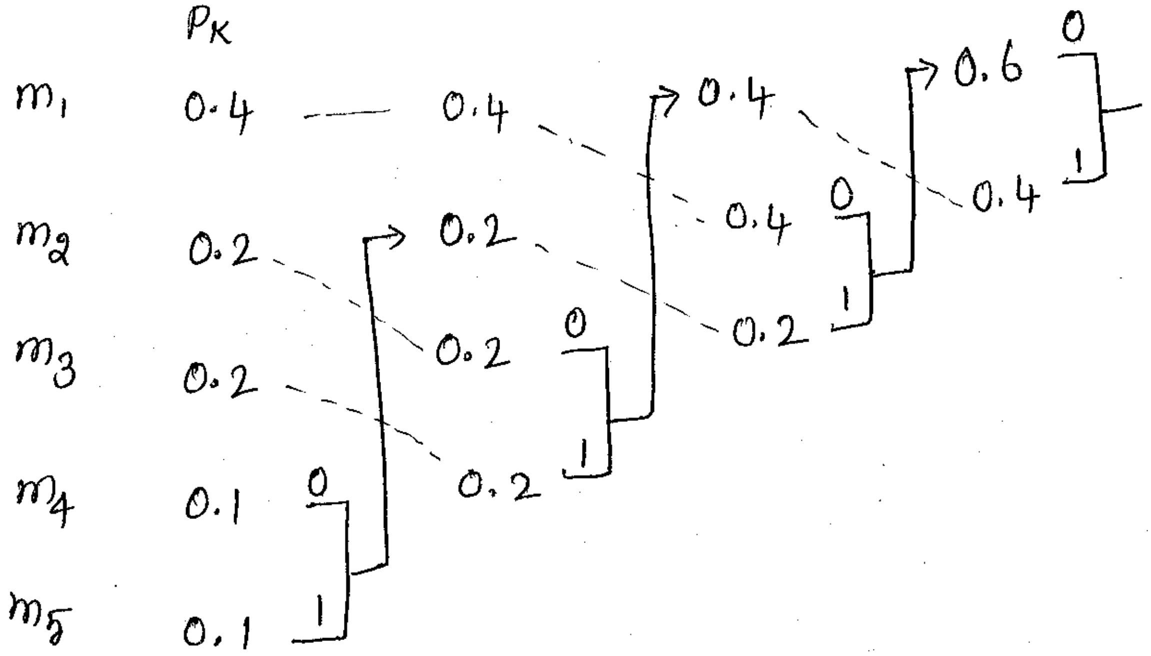
Step ④: Assign '0' and '1' to these two symbols

Step ⑤: Read the codewords of each symbol (Reverse) from the last stage

Problem :

| * Message | $m_1$ | $m_2$ | $m_3$ | $m_4$ | $m_5$ |
|---|---|---|---|---|---|
| Probability | 0.4 | 0.2 | 0.2 | 0.1 | 0.1 |

Huffman Coding




| message | code word | codeword length |
|---|---|---|
| $m_1$ | 00 | 2 |
| $m_2$ | 10 | 2 |
| $m_3$ | 11 | 2 |
| $m_4$ | 010 | 3 |
| $m_5$ | 011 | 3 |

Average codeword length $(L) = \sum_{k=1}^{M} P_k l_k$

$= 0.4 \times 2 + 0.2 \times 2 + 0.2 \times 2 + 0.1 \times 3 + 0.1 \times 3$

$\boxed{L = 2.2 \text{ bits/symbol}}$

## Entropy of the Source

$$H(x) = \sum_{K=1}^{M} P(x_K) \log_2 \frac{1}{P(x_K)}$$

$$= 0.4 \log_2 \frac{1}{0.4} + 0.2 \log_2 \frac{1}{0.2} + 0.2 \log_2 \frac{1}{0.2} + 0.1 \log_2 \frac{1}{0.1} + 0.1 \log_2 \frac{1}{0.1}$$

$$\boxed{H(x) = 2.1229 \text{ bits/symbol}}$$

Efficiency $(\eta)$ $= \dfrac{H(x)}{L \log_2 D} = \dfrac{2.1319}{2.2 \log_2 2} = 0.9636$

$\log_2 2 = 1$

$D = 0 \& 1$

$\therefore D = 2$

$$\boxed{\eta\% = 96.36\%}$$

Indicates that there are 4% redundant bits in the code.

Redundancy $(\gamma) = 1 - \eta$

$= 1 - 0.96$

$\boxed{\gamma = 0.04}$

## Mutual Information $I(x,y)$

↳ is defined as the uncertainty of channel i/p that is resolved by observing channel o/p.

$$\boxed{I(x,y) = H(x) - H(x/y)}$$

$H(x) \rightarrow$ is the uncertainty of channel i/p before the channel o/p is observed.

$H(x/y) \rightarrow$ is the uncertainty of channel i/p after the channel o/p is observed.

## Properties of Mutual Information

1. $I(x,y) = I(y,x)$

The mutual information of a channel is symmetric

$$I(x,y) = \sum_{i=1}^{m}\sum_{j=1}^{n} P(x_i, y_j)\log_2\left[\frac{P(x_i/y_j)}{P(x_i)}\right] \rightarrow ①$$

From Baye's rule for conditional Probabilities

$$\frac{P(x_i/y_j)}{P(x_i)} = \frac{P(y_j/x_i)}{P(y_j)} \rightarrow ②$$

Sub. ② in ① and interchanging the order of summation, we get

$$I(x,y) = \sum_{j=1}^{n}\sum_{i=1}^{m} P(x_i, y_j)\log_2\left[\frac{P(y_j/x_i)}{P(y_j)}\right]$$

$$I(x,y) = I(y,x)$$

2. $I(x,y) \geq 0$

The mutual information is always non negative

The joint Probability

$$P(x_i, y_j) = P\left(\frac{x_i}{y_j}\right)P(y_j)$$

$$P\left(\frac{x_i}{y_j}\right) = \frac{P(x_i, y_j)}{P(y_j)} \rightarrow ③$$

Sub. ③ in ①

$$I(x,y) = \sum_{i=1}^{m}\sum_{j=1}^{n} P(x_i, y_j)\log_2\left[\frac{P(x_i, y_j)}{P(x_i)P(y_j)}\right]$$

with fundamental inequality

$$I(x,y) \geq 0$$

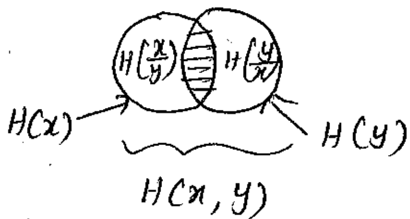
with equality, if and only if

$$P(x_i, y_j) = P(x_i) \cdot P(y_j) \text{ for all } i \,\&\, j.$$

$I(x,y)$ is zero when the i/p and o/p symbols are statistically independent.

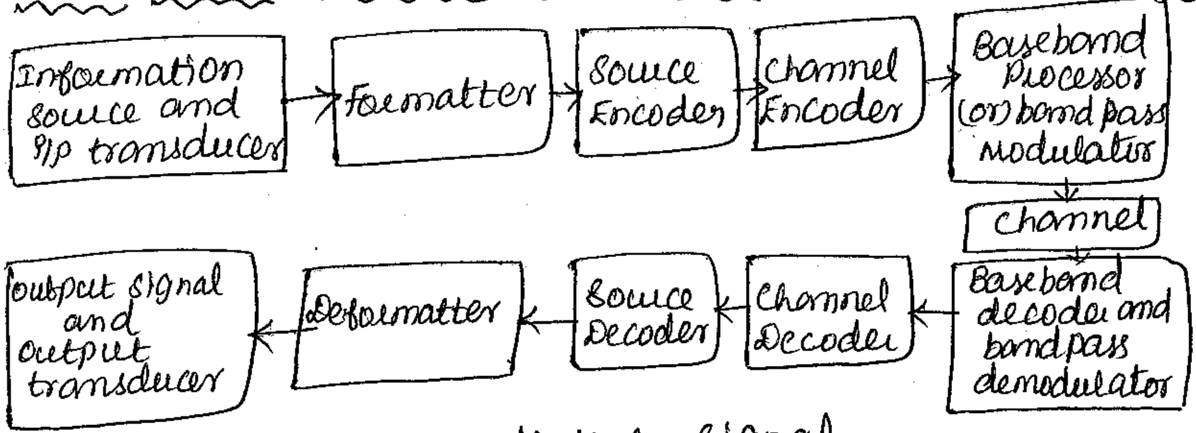
③ The mutual information of a channel is related to the joint Entropy of the channel i/p and channel o/p by

$$I(x,y) = H(x) + H(y) - H(x,y)$$




Relation Among various Channel Entropies.

Draw block diagram of digital communication system




i/p signal → digital signal

The block which converts the Electrical signals at the o/p of the transducer into a sequence of digital signals is known as Formatter

## Error control coding

↳ used for accurate transfer of information from one place to another.

**Advantages:**

* Reduces the Required transmitt power
* Reduces the Size of antenna
* Reduces the hardware cost.

**Disadvantages**

* Increases the transmission Bandwidth
* Increases the Complexity of decoder.

**Types of Error control codes**

Block codes | Convolutional codes

Block codes (Types): Linear Block codes | cyclic codes

**Block codes:** The codes which consists of $(n-k)$ Parity bits for every $k$-bit message block are known as block codes.

**Linear block codes:** Block code is the code in which to every 'k-bit' Message block $(n-k)$ Parity bits are appended to produce 'n' bit codeword. If the parity bits are the linear combination of 'k' message bits then the code is referred to as linear block codes.

**Structure of systematic block code**

| Parity bits | Message bits |
| --- | --- |
| $b_0, b_1, \ldots b_{n-k-1}$ | $m_0, m_1, \ldots m_{k-1}$ |

**Systematic codes:** Block codes in which the message bits are transmitted in unaltered form are called systematic codes.

Hamming weight : is defined as the number of non-zero Elements in the code vector.

$C_1 = 01001101 = \boxed{4}$

Hamming distance : is calculated between two codeword by, number of places (bits) the codeword differs.

$C_1 = 1011$ $\quad d(C_1, C_2) = \boxed{2}$
$C_2 = 0001$

differs.

Minimum distance ($d^*$ or $d_{min}$)

It is the smallest hamming distance in any pair of codewords in the code.

$$d^* = \min[d(C_i, C_j)] \quad (i \neq j)$$

Minimum weight ($W^*$) of the code is the smallest hamming weight of any non-zero codeword.

$C = [0100, 1001, 1100]$

$C_1 = 0100 \quad C_2 = 1001 \quad C_3 = 1100$

Hamming weight $\quad W_{C1} = 1 \quad W_{C3} = 2$
$W_{C2} = 2$

Minimum weight $\boxed{W^* = 1}$

Hamming distance
$d(C_1, C_2) = 3$
$d(C_2, C_3) = 2$
$d(C_1, C_3) = 1$

Minimum distance

$d^* = d_{min}[d(C_1, C_2), d(C_2, C_3), d(C_1, C_3)]$

$= \min[3, 2, 1]$

$\boxed{d^* = 1}$

Linear Block codes: A code is said to be linear when any two codewords can be added to produce a third codeword within the code.

Properties of Linear block codes

① The all zero word is always codeword

② The sum of 2 codewords belonging to the code, is also a codeword belonging to the same code.

③ Minimum distance = Minimum weight

---

Linear Block code Problem:

# A Generator matrix for (6,3) block code is given find (i) all code vectors & hamming weight of each code word

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

(ii) Check matrix and the Received code word is [100000]. (iii) Parity check matrix (iv) what is the minimum distance between Codewords (v) How many Errors can be detected and how many Errors can be corrected (vi) Find the transmitted information word.

Solution:

'G' → is used at the txr for encoding operation

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

n = 6 → size of codeword

k = 3 → message bit

r = n − k = 6 − 3 = 3 [Parity (or) check bit]

(i)

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

$I_K$ | $P_{K \times (n-K)}$

G → Generator matrix

$G = \left[ I_K \mid P_{K \times (n-K)} \right]_{K \times n}$

$I_K$ → Identity matrix

P → Submatrix (or) Coefficient Matrix

**Ex-or (or) mod 2 addition**

$0 \oplus 0 = 0$
$0 \oplus 1 = 1$
$1 \oplus 0 = 1$
$1 \oplus 1 = 0$

$I_{3\times3}$ $P_{3\times3}$

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad P = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

(ii) **Check matrix:**

$$C = MP$$

$$[C]_{1\times(n-k)} = [M]_{1\times k}\,[P]_{k\times(n-k)}$$

$$[C]_{1\times3} = [M]_{1\times3}\,[P]_{3\times3}$$

$$[c_1, c_2, c_3] = [a_1, a_2, a_3]\begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

$$c_1 = a_1 \cdot 1 \oplus a_2 \cdot 1 \oplus a_3 \cdot 0$$

$$\boxed{\begin{aligned} c_1 &= a_1 \oplus a_2 \\ c_2 &= a_2 \oplus a_3 \\ c_3 &= a_1 \oplus a_3 \end{aligned}}$$

**Code vector Table**

| Message bit | | | Check bits | | | Codeword | | | | | | Hamming weight |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $a_1$ | $a_2$ | $a_3$ | $c_1$ | $c_2$ | $c_3$ | $a_1$ | $a_2$ | $a_3$ | $c_1$ | $c_2$ | $c_3$ | W |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 3 |
| 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 3 |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 4 |
| 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 3 |
| 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 4 |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 4 |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 3 |

(iii) Parity check matrix (H)

$$H = [P^T_{(n-K)\times K} \; I_{(n-K)\times(n-K)}]_{(n-K)\times n}$$

$$H = [P^T_{3\times3} \; I_{3\times3}]_{3\times6}$$

$P^T \rightarrow$ obtained by interchanging the rows & columns.

$$H = \begin{array}{c} \begin{array}{cc} P^T & I \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

(iv) $\boxed{\text{Minimum weight} = \text{minimum distance}}$ .X.

Minimum hamming distance $d^*$ or $d_{min} = 3$
(from the property 3 of linear block codes)

(v) Number of Error can be detected

$$d^* - 1 = 3 - 1 = \boxed{2} \rightarrow 2 \text{ Error can be detected}$$

Number of Error corrected

$$d^* \text{ or } d_{min} \geq 2t+1$$

$$d^* \geq 2t+1$$

$$d^* - 1 \geq 2t$$

$$t \leq \frac{d^*-1}{2}$$

$$\leq \frac{2}{2}$$

$$\boxed{t \leq 1} \rightarrow 1 \text{ Error can be Corrected.}$$

'H' → is used at the Rxr for decoding operation

(vi) Received codeword $Y = [100000]$

Syndrome vector for Received code word is

$$[S] = [Y][H^T]$$

Syndrome → * Contains information about the Error pattern
* Used for Error detection.

$$H^T = \left[\frac{P_{K \times (n-K)}}{I_{(n-K) \times (n-K)}}\right]$$ (or) directly take transponse of H.

$$\therefore H^T = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

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

$$= 1.1 \oplus 0.1 \oplus 0.0 \oplus 0.1 \oplus 0.0 \oplus 0.0,$$
$$1.0 \oplus 0.1 \oplus 0.1 \oplus 0.0 \oplus 0.1 \oplus 0.0,$$
$$1.1 \oplus 0.0 \oplus 0.1 \oplus 0.0 \oplus 0.0 \oplus 0.1$$

$$= 1 \oplus 0 \oplus 0 \oplus 0 \oplus 0 \oplus 0,\ 0 \oplus 0 \oplus 0 \oplus 0 \oplus 0 \oplus 0,$$
$$1 \oplus 0 \oplus 0 \oplus 0 \oplus 0 \oplus 0$$

$[S] = [1\ 0\ 1]$ → indicate 5th row of $H^T$ is in Error. (ie) 5th bit position.

| Bit Error Position | Error vector 'e' Showing Single bit Error pattern $e_1$ $e_2$ $e_3$ $e_4$ $e_5$ $e_6$ | Syndrome [S] |
|---|---|---|
| 1 | 1 0 0 0 0 0 | 1 0 1 |
| 2 | 0 1 0 0 0 0 | 1 1 0 |
| 3 | 0 0 1 0 0 0 | 0 1 1 |
| 4 | 0 0 0 1 0 0 | 1 0 0 |
| Error → 5 | 0 0 0 0 1 0 | 0 1 0 |
| 6 | 0 0 0 0 0 1 | 0 0 1 |

Syndrome of Received codes.

The transmitted code word

$X = Y \oplus e$

5th bit Error

$$
\begin{array}{r l}
 & 1\,0\,0\,0\,0\,0 \leftarrow Y \\
\oplus & 0\,0\,0\,0\,1\,0 \leftarrow e \\
\hline
 & 1\,0\,0\,0\,1\,0
\end{array}
$$

5th bit Error

$\boxed{X = [1\,0\,0\,0\,1\,0]}$

## Cyclic Code

* is the subclass of linear block codes.
* It is a systematic method for correcting higher number of errors.
* There are 2 important reasons to prefer cyclic code.

(1) Encoding and syndrome calculation can be easily implemented by using simple shift registers with feedback connections.

(2) the mathematical structure of cyclic code is such that it is possible to design codes having useful error correcting properties.

**Definition**

A codeword 'C' is cyclic,

(i) if 'C' is a linear code

(ii) any cyclic shift of a codeword is also a codeword

Properties:

① Linearity → The sum of 2 codeword is also a codeword.

② cyclic property → Any cyclic shift of a codeword in either direction produce a new codeword.

Cyclic code – 2 types – → Systematic code
→ Non-systematic code.

Non systematic cyclic code

$$\boxed{C(x) = g(x)\,M(x)}$$

$M(x)$ is of degree $K-1$
$C(x)$ is of degree $n-1$

$g(x)$ → generating polynomial
$M(x)$ → ~~...~~ Message polynomial
$C(x)$ → Non-systematic cyclic code.

Systematic cyclic code

Given $(n, K)$ cyclic codes

Steps

① Multiply $M(x)$ by $x^{n-K}$

② Divide the result of the first step by $g(x)$ to give the remainder $r(x)$

③ Add $r(x)$ to the result of the first step to get systematic codeword

$$\boxed{C(x) = M(x)\,x^{n-K} + r(x)}$$

$M(x)$ → information polynomial
$C(x)$ → Systematic cyclic codeword.

## Non-systematic cyclic code Problem:

# construct a non-systematic (7, 4) cyclic code for the given generating polynomial $g(x) = x^3 + x + 1$ for the information vector [1 0 0 0]. Find the corresponding codeword.

Solution: $g(x) = x^3 + x + 1$

Message vectors $M(x) = [1\ 0\ 0\ 0]$

$\begin{array}{cccc} x^3 & x^2 & x^1 & x^0 \end{array}$

$\downarrow$

$K-1 \rightarrow$ coefficient polynomial of degree $4-1 = \boxed{3}$

$M(x) = 1 \cdot x^3 + 0 \cdot x^2 + 0 \cdot x^1 + 0 \cdot x^0$

$\boxed{M(x) = x^3}$

Codeword polynomial

$C(x) = M(x) \cdot g(x)$

$= x^3 \cdot (x^3 + x + 1)$

$C(x) = x^6 + x^4 + x^3$

codeword length $n = 7$

$\therefore$ Code word $= \boxed{1\ 0\ 1\ 1\ 0\ 0\ 0}$

$x^6\ x^5\ x^4\ x^3\ x^2\ x^1\ x^0$

## Systematic cyclic code Problem:

# construct a systematic (7, 4) cyclic code using generator polynomial $g(x) = x^3 + x + 1$ for the information vector [1 0 0 0]. Find the corresponding codeword.

Solution:

Message vector $M(x) = [1\ 0\ 0\ 0]$

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

$K-1 = 4-1 = 3 \leftarrow$ Polynomial of degree

$M(x) = 1.x^3 + 0.x^2 + 0.x^1 + 0.x$

$M(x) = x^3$

<u>Step ①</u>: Multiply $M(x)$ by $x^{n-k}$

$$M(x).x^{n-k} = x^3[x^{7-4}] = x^3[x^3] = x^6$$

<u>Step ②</u>: $M(x).x^{n-k} \div g(x)$ to get the remainder $r(x)$

$$
\begin{array}{r|l}
 & x^3+x+1 \\
x^3+x+1 & x^6 \\
 & x^6+x^4+x^3 \\ \hline
 & x^4+x^3 \\
 & x^4+x^2+x \\ \hline
 & x^3+x^2+x \\
 & x^3+x+1 \\ \hline
 & x^2+x+x+1
\end{array}
$$

$x^6 \oplus x^6 = 0$

$x^4 \oplus x^4 = 0$

$x^3 \oplus x^3 = 0$

$x \oplus x = 0$

$x^2 + x + x + 1$ ← Remainder $r(x)$

Division is carried out by <u>modulo 2 Addition</u>
↑
Ex-OR

<u>Step ③</u>: To get codeword, $r(x)$ is added to $M(x).x^{n-k}$

$$C(x) = M(x)x^{n-k} + r(x) = x^6 + x^2 + 1$$

<u>Codeword vector</u> = $\boxed{1000101}$ ← $n=7$ so 7 bits

∴ <u>Transmitted codeword</u> = $\boxed{1000101}$

## Decoding cyclic code Problem

\# construct the decoding table for single error correction for (7,4) systematic cyclic code from decoding table, determine the transmitted information code when received codeword is [1100000]

**Solution:**

Step ① : For (7,4) cyclic code, $g(x) = x^3 + x + 1$

Step ② : Received code word $Y(x) = \underset{x^6}{1}\,\underset{x^5}{1}\,\underset{x^4}{0}\,\underset{x^3}{0}\,\underset{x^2}{0}\,\underset{x^1}{0}\,\underset{x^0}{0}$

$$Y(x) = 1.x^6 + 1.x^5 + 0.x^4 + 0.x^3 + 0.x^2 + 0.x^1 + 0.x^0$$

$$Y(x) = x^6 + x^5$$

Step ③ : Syndrome for Received Codeword

$$\left\{ S = \text{rem}\left[\frac{Y(x)}{g(x)}\right]\right.$$

$$\begin{array}{r|l} & x^3 + x^2 + x \\ \hline x^3+x+1 & x^6 + x^5 \\ & x^6 + x^4 + x^3 \\ \hline & x^5 + x^4 + x^3 \\ & x^5 + x^3 + x^2 \\ \hline & x^4 + (x^3 + x^3) + x^2 \\ & x^4 \qquad + x^2 + x \\ \hline & x \end{array}$$

$x$ ← Error syndrome

(or)

$$S_{n-k-1} = S_{7-4-1} = x$$

$$S_2 = \underset{x^2}{0}\,\underset{x^1}{1}\,\underset{x^0}{0}$$

$x^1$ bit position is in Error

Step (4): Syndrome for Error Polynomial

$$S = rem\left[\frac{e(x)}{g(x)}\right]$$

| Bit Position | Error vector e(x) $e_1$ $x^6$ | $e_2$ $x^5$ | $e_3$ $x^4$ | $e_4$ $x^3$ | $e_5$ $x^2$ | $e_6$ $x^1$ | $e_7$ $x^0$ | e(x) | Degree Syndrome | $S(x) = \frac{e(x)}{g(x)}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | $x^6$ | 1 0 1 ($x^2$ $x^1$ 1) | $x^2+1$ |
| 2 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | $x^5$ | 1 1 1 ($x^2$ $x$ 1) | $x^2+x+1$ |
| 3 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | $x^4$ | 1 1 0 ($x^2$ $x$ 0) | $x^2+x$ |
| 4 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | $x^3$ | 0 1 1 | $x+1$ |
| 5 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | $x^2$ | 1 0 0 | $x^2$ |
| 6 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | $x^1$ | 0 1 0 | $x$ |
| 7 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | $x^0$ | 0 0 1 | 1 |
| | | | | | | | | $x^3+x+1$ | $\rightarrow g(x)$ | $\frac{e(x)}{g(x)}$ |

Corrected code word $C = Y \oplus e$
(or)
Transmitted code word

$$\begin{array}{rl} & 1100000 \leftarrow Y \\ \oplus & 0000010 \leftarrow e \\ \hline & 1100010 \leftarrow C \end{array}$$

$$\boxed{C = 1100010}$$

Disadvantages of Block codes
* It takes 'K' i/p bits and produces 'n' o/p
* If n and K are large it needs large storage space
* The decoding process starts only after entire block of information bits, (or) codeword Received.

Application
* It is only used in high data rate communication

These drawback can overcome by convolutional codes.

# Convolutional code

* It is a alternative form of Block Code.
* It is more efficient channel coding technique than Block code.
* It is widely used in practical communication system for Error correction.

## Advantages

* The message bits come serially rather than blocks.
* Encoder operates on the i/p message continuously on serial manner.

## Application

* It is used in 3G [Cellular Phone], INMARSAT and various wireless systems.

**Code Rate (R)**: is expressed as a ratio of the number of bits into the convolutional Encoder (k) to the number of channel symbols o/p by the convolutional encoder (n) in a given Encoder cycle.

$$\boxed{R = \frac{k}{n}}$$

## Constraint length (K)

The constraint length parameter, K denotes the "length" of the convolutional Encoder.

$$\boxed{K = m+1}$$

m ⇒ 'c' number of shift (or) k stage shift register

$\therefore m = K+1$

Constraint length: is the number of shift over which the single message bit can influence the encoder output. It is expressed in terms of message bits.

# Encoding and Decoding of Convolutional Code.

Convolutional code : Fixed number of i/p bits is stored in the shift register & they are combined with the help of mod 2 adders. This operation is equivalent to binary convolution coding

Encoding : Trellis Code
Decoding : Viterbi Algorithm [Maximum Likelihood Decoding]

**Encoding Problem**

\# For a $\frac{1}{2}$ Rate convolutional Encoder
$$g^{(1)} = \{1\ 0\ 1\} \text{ and } g^{(2)} = \{1\ 1\ 1\}$$

(a) Draw the Encoder block
(b) Draw trellis code
(c) Encode the message sequence 1 1 0

Solution :
$$\frac{k}{n} = \frac{1}{2}$$

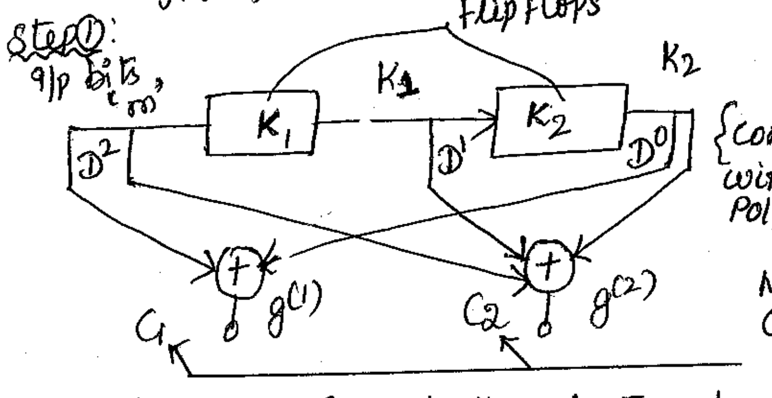
$k = 1 \Rightarrow$ number of message bits taken by the Encoder.
$n = 2 \Rightarrow$ number of Encoded o/p bits for one message bit.

$m = k+1 = 1+1 = 2$ {number of shift}

$\therefore$ Number of flip flops required ($k+1 = 1+1 = \boxed{2}$)

Length of the convolutional Encoder $K = m+1$
$= 2+1$
$= 3$

Step ①:




{Construct Encoder circuit with help of generator Polynomial.}

½ rate Convolutional Encoder $(2, 1, 3)$ $(n, k, K)$

$C_1 = m \oplus K_2 \qquad C_2 = m \oplus K_1 \oplus K_2$

**Step ②**: Define check bits ($C_1, C_2, \ldots$) from generator polynomial.

$$g^{(1)} = \{1\ 0\ 1\} \qquad g^{(2)} = \{1\ 1\ 1\}$$

$$\begin{array}{cccc} & x^2 & x^1 & x^0 \\ & \updownarrow & \updownarrow & \updownarrow \\ C_1 = & m & K_1 & K_2 \end{array} \qquad \begin{array}{cccc} & x^2 & x^1 & x^0 \\ & \updownarrow & \updownarrow & \updownarrow \\ C_2 = & m & K_1 & K_2 \end{array}$$

$\therefore C_1 = m \oplus K_2 \qquad \therefore C_2 = m \oplus K_1 \oplus K_2$

**Step ③**: Assign State of an Encoder

2 FlipFlops $K_1$ & $K_2$ $\rightarrow$ so 4 possible state and each state assigned a name as a, b, c, & d.

| $K_1$ | $K_2$ | State |
|---|---|---|
| 0 | 0 | a |
| 1 | 0 | b |
| 0 | 1 | c |
| 1 | 1 | d |

Next state

| m | $K_1$ | $K_2$ |
|---|---|---|
| 1 | 0 | 0 |

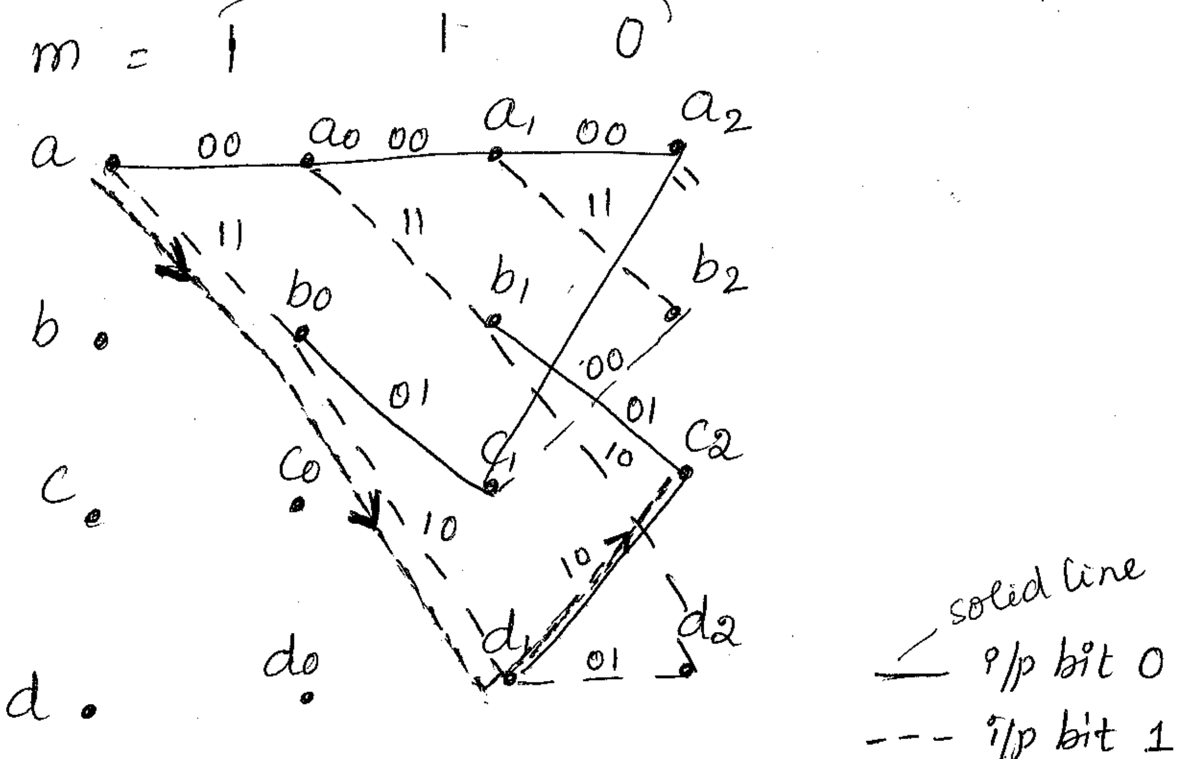
$\rightarrow$ 10

Shift Right by one bit position

**Step ④**: Construct State table

| Current State $K_1 K_2$ | Message Bit (m) | Check Bits $C_1 = m \oplus K_2$ | $C_2 = m \oplus K_1 \oplus K_2$ | Next state $K_1 K_2$ = State |
|---|---|---|---|---|
| a = 00 | 0 | 0 | 0 | 00 = a |
| | 1 | 1 | 1 | 10 = b |
| b = 10 | 0 | 0 | 1 | 01 = c |
| | 1 | 1 | 0 | 11 = d |
| c = 01 | 0 | 1 | 1 | 00 = a |
| | 1 | 0 | 0 | 10 = b |
| d = 11 | 0 | 1 | 0 | 01 = c |
| | 1 | 0 | 1 | 11 = d |

Step (5): From state table, draw trellis diagram representation to Encode the i/p Message m = {1, 1, 0}.




Encoded word {11 10 10} for given message 110.

Trellis diagram Representation

if m = {1 1 0}, then 3 stages are required to Encode the Message.

## Decoding - Viterbi Algorithm [Maximum Likelihood decoding]

* It is a self Error Correcting code [upto two bit Error detected and Corrected]

Problem:

# Consider 1/2 rate of convolutional Encoder with $g^{(1)} = \{1\ 0\ 1\}$ & $g^{(2)} = \{1\ 1\ 1\}$ and assume that the message data is m{1 1 0} and their encoded message

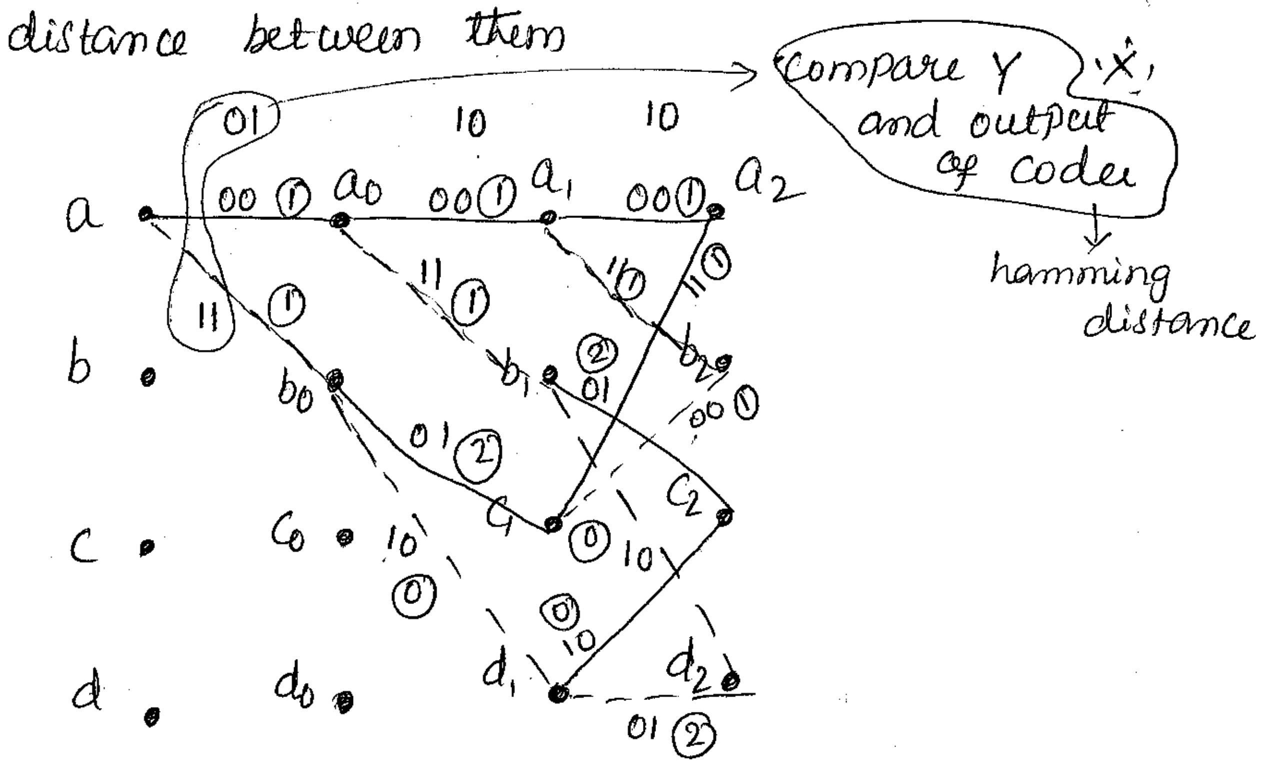
from coder is {11, 10, 10}. If 1st bit of Encoded sequence are affected during transmission Determine the error correcting capabilities using Viterbi algorithm

Solution : Step 1 ... Step 5 } Similar to Encoding Process.

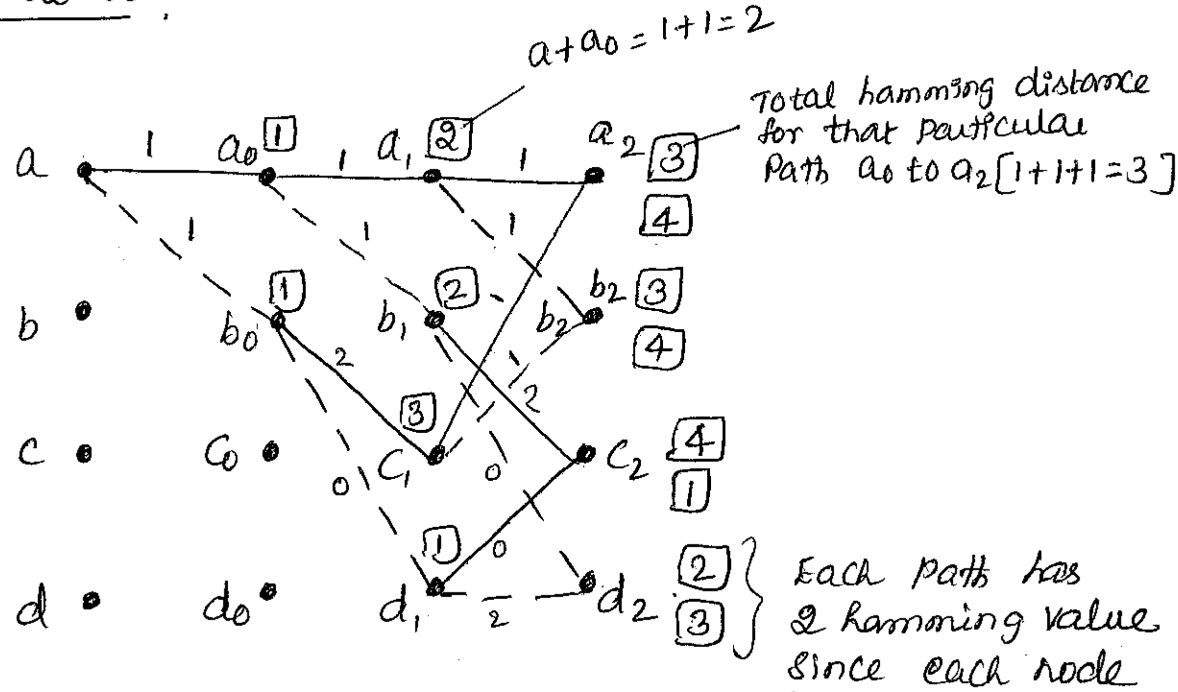
Assume $1^{st}$ bit is in Error, therefore Received codeword Y = {01, 10, 10}

Step 6 :

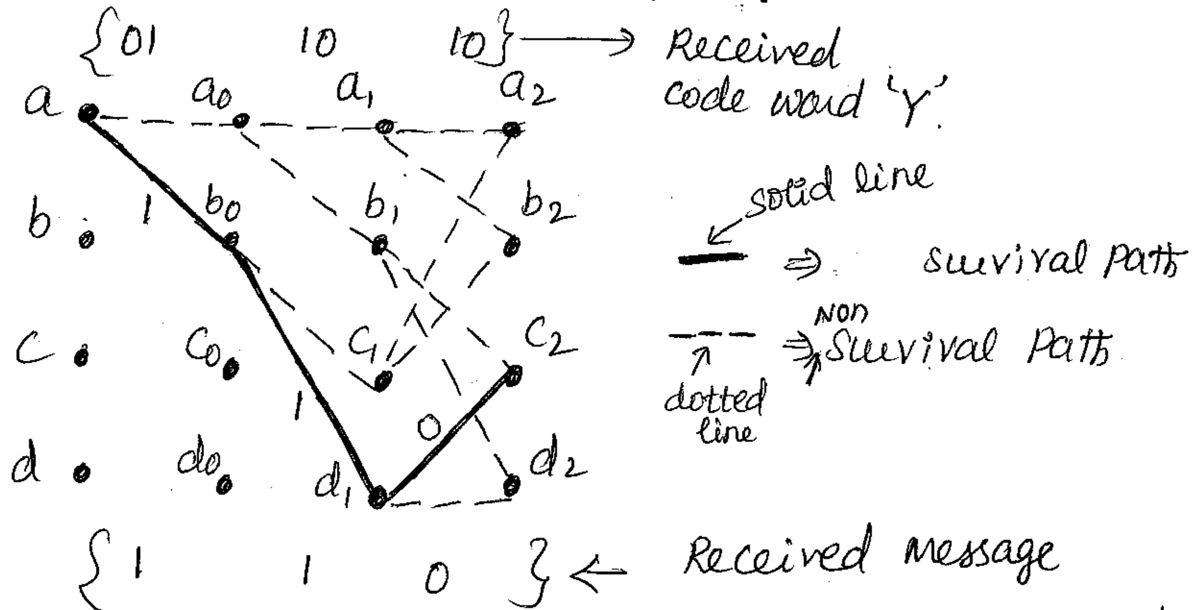
Compare each of these o/p sequence with the actual receive sequence, and determine hamming distance between them



**Step ⑦:** Re-draw the above diagram, only with hamming distance. For each path calculate total hamming distance with reference to initial state 'a'.




**Step ⑧:** The path having minimum hamming distance will be considered as survival path and other path considered as non-survival path.



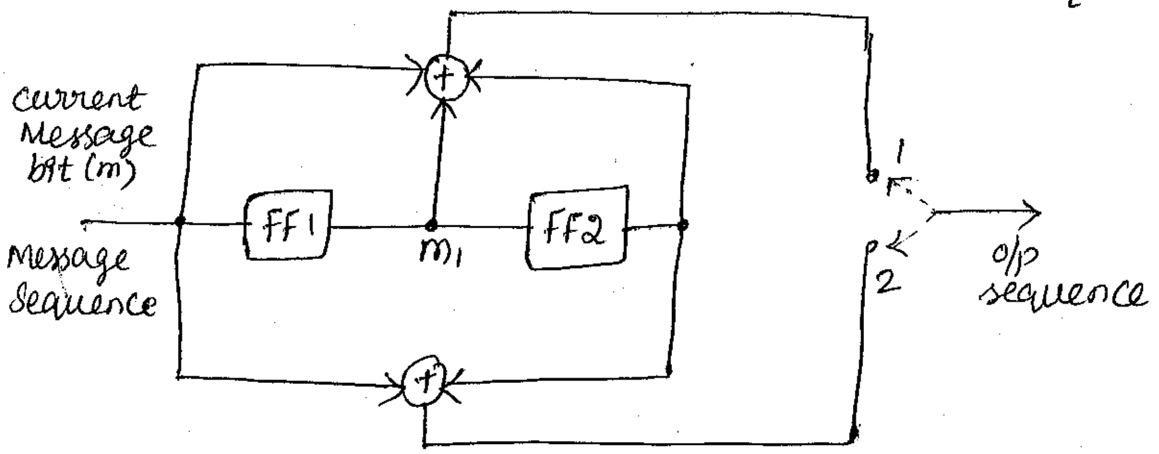

The Path abod1C2 has minimum distance and referred as survival path.

Trace the survival path {for message bit either '0' or '1'} to decode the information bit.

Decoded o/p = {1 1 0}

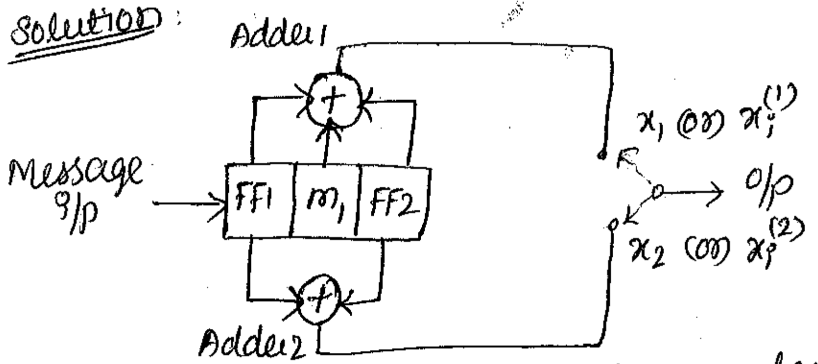
# Time Domain approach to analysis of convolutional Encoder

# For the convolutional encoder of fig. determine the following (i) Dimension of the code (ii) Code rate (iii) Constraint length (iv) Generating sequences [Impulse Response] (v) o/p sequence for message sequence of $m = \{1 0 0 1 1\}$




Convolutional Encoder

Solution:




Convolutional Encoder

(i) Dimension of the code → $x_1$ & $x_2$ so $\boxed{2}$

Dimension $= (n, k) = (2, 1)$ → For a convolutional Encoder it is always $\boxed{1}$

(ii) Code rate

$$r = \frac{k}{n} = \frac{1}{2}$$

$k$ ⇒ number of message bits taken by the encoder

$n$ ⇒ number of Encoded o/p bits for one message bit.

(iii) Constraint length

$K = m + 1 = 2 + 1 = 3$ bits [number of shifts (ie) one message bit will be shifted in 3 times].

$\therefore m = k + 1 = 1 + 1 = 2$

(iv) Generating sequences

$g_i^{(1)} = \{1 \; 1 \; 1\}$ (adder 1)

$g_i^{(2)} = \{1 \; 0 \; 1\}$ (adder 2)

## To obtain o/p sequence

$$m = (m_0\ m_1\ m_2\ m_3\ m_4) = (1\ 0\ 0\ 1\ 1)$$

### Due to adder 1

$$\boxed{x_i^{(1)} = \sum_{l=0}^{M} g_l^{(1)} m_{i-l}}$$

$i = 0, 1, 2 \ldots$ ($x_i$)

M = Message bits given ($m_0$ to $m_4$)

$\therefore \boxed{M = 4}$

$\boxed{i = 0 \text{ to } (M+3)-1} \rightarrow$ formula

(3 → No. of bits in generator sequence)

$i = 0$ to $(4+3)-1$

$\boxed{i = 0 \text{ to } 6}$

**i=0**

$$x_0^{(1)} = \sum_{l=0}^{M} g_l^{(1)} m_{0-l}$$

$$= g_0^{(1)} m_0$$

{$m_{-1}$, $m_{-2}$, $m_{-3}$ & $m_{-4}$ does not exist}

$$= 1 \times 1$$

$$\boxed{x_0^{(1)} = 1}$$

**i=1**

$$x_1^{(1)} = g_0^{(1)} m_1 \oplus g_1^{(1)} m_0 = 1\times 0 \oplus 1 \times 1 = 1$$

$$\boxed{x_1^{(1)} = 1}$$

{$m_{-1}$, $m_{-2}$ & $m_{-3}$ does not exist} ⇓ invalid terms (omit)

**i=2**

$$x_2^{(1)} = g_0^{(1)} m_2 \oplus g_1^{(1)} m_1 \oplus g_2^{(1)} m_0$$

$$= 1\times 0 \oplus 1 \times 0 \oplus 1 \times 1$$

$$\boxed{x_2^{(1)} = 1}$$

**i=3**

$$x_3^{(1)} = g_0^{1} m_3 \oplus g_1^{(1)} m_2 + g_2^{(1)} m_1$$

$$= (1\times 1) \oplus (1 \times 0) \oplus (1 \times 0)$$

$$\boxed{x_3^{(1)} = 1}$$

**i=4**

$$x_4^{(1)} = g_0^{(1)} m_4 \oplus g_1^{(1)} m_3 \oplus g_2^{(1)} m_2$$

$$= (1\times 1) \oplus (1 \times 1) \oplus (1 \times 0)$$

$$\boxed{x_4^{(1)} = 0}$$

**i=5**

$$x_5^{(1)} = g_0^{(1)} \underbrace{m_5}_{\text{is not available}} \oplus g_1^{(1)} m_4 \oplus g_2^{(1)} m_3$$

$$= g_1^{(1)} m_4 \oplus g_2^{(1)} m_3$$

$$= (1\times 1) \oplus (1 \times 1)$$

$$\boxed{x_5^{(1)} = 0}$$

**i=6**

not available

$$x_6^{(1)} = g_0^{(1)} m_6 \oplus g_1^{(1)} m_5 \oplus g_2^{(1)} m_4$$

$$= g_2^{(1)} m_4$$

$$= 1 \times 1$$

$$\boxed{x_6^{(1)} = 1}$$

The o/p of adder 1 is,

$$x_1 = x_i^{(1)} = \{1\ 1\ 1\ 1\ 0\ 0\}$$

Due to adder 2

$$x_2 = x_i^{(2)} = \sum_{l=0}^{M} g_l^{(2)} m_{i-l}$$

$i=0$

$$x_0^{(2)} = g_0^2 m_0 = 1 \times 1 = 1$$

$$\boxed{x_0^{(2)} = 1}$$

$i=1$

$$x_1^{(2)} = g_0^{(2)} m_1 \oplus g_1^{(1)} m_0$$

$$= (1 \times 0) \oplus (0 \times 1)$$

$$\boxed{x_1^{(2)} = 0}$$

$i=2$

$$x_2^{(2)} = g_0^{(2)} m_2 \oplus g_1^{(2)} m_1 \oplus g_2^{(2)} m_0$$

$$= (1 \times 0) \oplus (0 \times 0) \oplus (1 \times 1)$$

$$\boxed{x_2^{(2)} = 1}$$

$i=3$

$$x_3^{(2)} = g_0^2 m_3 \oplus g_1^{(2)} m_2 \oplus g_2^{(2)} m_1$$

$$= (1 \times 1) \oplus (0 \times 0) \oplus (1 \times 0)$$

$$\boxed{x_3^{(2)} = 1}$$

$i=4$

$$x_4^{(2)} = g_0^2 m_4 \oplus g_1^{(2)} m_3 \oplus g_2^{(2)} m_2$$

$$= (1 \times 1) \oplus (0 \times 1) \oplus (1 \times 0)$$

$$\boxed{x_4^{(2)} = 1}$$

$i=5$

$$x_5^{(2)} = g_1^{(2)} m_4 \oplus g_2^{(2)} m_3$$

$$= (0 \times 1) \oplus (1 \times 1)$$

$$\boxed{x_5^{(2)} = 1}$$

$i=6$

$$x_6^{(2)} = g_2^{(2)} m_4$$

$$= 1 \times 1$$

$$\boxed{x_6^{(2)} = 1}$$

The o/p of adder 2 is

$$x_2 = x_i^{(2)} = \{1\ 0\ 1\ 1\ 1\ 1\ 1\}$$

$x_1$ & $x_2$ are multiplexed

$$x_i = x_0^{(1)} x_0^{(2)}\ x_1^{(1)} x_1^{(2)}\ x_2^{(1)} x_2^{(2)}\ x_3^{(1)} x_3^{(2)}\ x_4^{(1)} x_4^{(2)}\ x_5^{(1)} x_5^{(2)}\ x_6^{(1)} x_6^{(2)}$$

$x_i = \{11, 10, 11, 11, 01, 01, 11\} \rightarrow$ o/p of Encoder

## Transform Domain Approach to analysis of Convolutional Encoder

$$\boxed{\begin{aligned} x^{(1)}(P) &= g^{(1)}(P) \cdot m(P) \\ x^{(2)}(P) &= g^{(2)}(P) \cdot m(P) \end{aligned}}$$

$m(P) \Rightarrow$ message polynomial

$g(P) \Rightarrow$ generating polynomial

[ Above problem for Transform Domain Approach ]

### Generating polynomial

\# $g_i^{(1)} = \{1\ 1\ 1\}$ Adder 1

$g^{(1)}(P) = 1\times1 + 1\times P + 1\times P^2$

$= 1+P+P^2$

$g_i^{(2)} = \{1\ 0\ 1\}$ Adder 2

$g^{(2)}(P) = 1\times1 + 0\times P + 1\times P^2$

$= 1+P^2$

variable 'P' $\Rightarrow$ unit delay operator

$\Rightarrow$ it represents the timedelay of the bits in the impulse Response.

### Message polynomial

$m = \{1\ 0\ 0\ 1\ 1\}$

$m(P) = 1\times1 + 0\times P + 0\times P^2 + 1\times P^3 + 1\times P^4$

$= 1+P^3+P^4$

### o/p due to adder 1

$x^{(1)}(P) = g^{(1)}(P) \cdot m(P) = (1+P+P^2)(1+P^3+P^4)$

$= 1+P+P^2+P^3+P^6$

$x_i^{(1)} = \{\underset{1}{1}\ \underset{P}{1}\ \underset{P^2}{1}\ \underset{P^3}{1}\ \underset{P^4}{0}\ \underset{P^5}{0}\ \underset{P^6}{1}\}$

### o/p due to adder 2

$x^{(2)}(P) = g^{2}(P) \cdot m(P) = (1+P^2)(1+P^3+P^4) = 1+P^2+P^3+P^4+P^5+P^6$

$x_i^{(2)} = \{\underset{1}{1}\ \underset{P}{0}\ \underset{P^2}{1}\ \underset{P^3}{1}\ \underset{P^4}{1}\ \underset{P^5}{1}\ \underset{P^6}{1}\}$

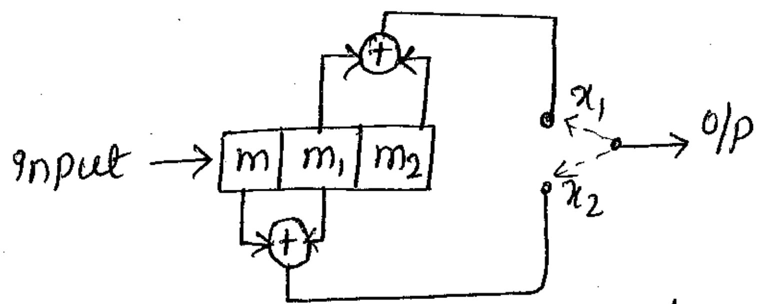
### Multiplexed o/p sequence [Encoder o/p]

$x_i = \{11, 10, 11, 11, 01, 01, 11\}$

# Comparison b/w Code tree and Trellis diagram.

| Code Tree | Trellis diagram |
|---|---|
| ① Indicates flow of the coded Signal along the nodes of the tree. | ① Indicates transitions from Current state to Next States |
| ② Lengthy way of Representing Coding Process | ② Code trellis diagram is Shorter (or) Compact way of representing Coding Process. |
| ③ Decoding is very Simple using code tree. | ③ Decoding is little complex using trellis diagram. |
| ④ Code tree repeats after number of stages used in the encoder. | ④ Trellis diagram repeats in every state. |
| ⑤ Complex to implement in Programming. | ⑤ Simpler to implement in Programming |

# Construct the code tree and the state diagram of rate $\frac{1}{2}$ convolutional Encoder given below.




Convolutional Encoder.

Solution: State transition table

| $m_2$ $m_1$ | State (assign) |
|---|---|
| 0 0 | a |
| 0 1 | b |
| 1 0 | c |
| 1 1 | d |

The o/ps of the encoder are

$$x_1 = m_1 \oplus m_2$$
$$x_2 = m \oplus m_1$$

State transition table

| Current State $m_2\ m_1$ | i/p $m$ | o/ps $x_1 = m_1 \oplus m_2$ | $x_2 = m \oplus m_1$ | Next state $m_2\ m_1$ |
|---|---|---|---|---|
| a = 0 0 | 0 | 0 | 0 | 0 0 = a |
| | 1 | 0 | 1 | 0 1 = b |
| b = 0 1 | 0 | 1 | 1 | 1 0 = c |
| | 1 | 1 | 0 | 1 1 = d |
| c = 1 0 | 0 | 1 | 0 | 0 0 = a |
| | 1 | 1 | 1 | 0 1 = b |
| d = 1 1 | 0 | 0 | 1 | 1 0 = c |
| | 1 | 0 | 0 | 1 1 = d |

Next state

$m\ m_1\ m_2$

0 0 0

shift right by one bit position so Next state $m_1 m_2 = 0\ 0$

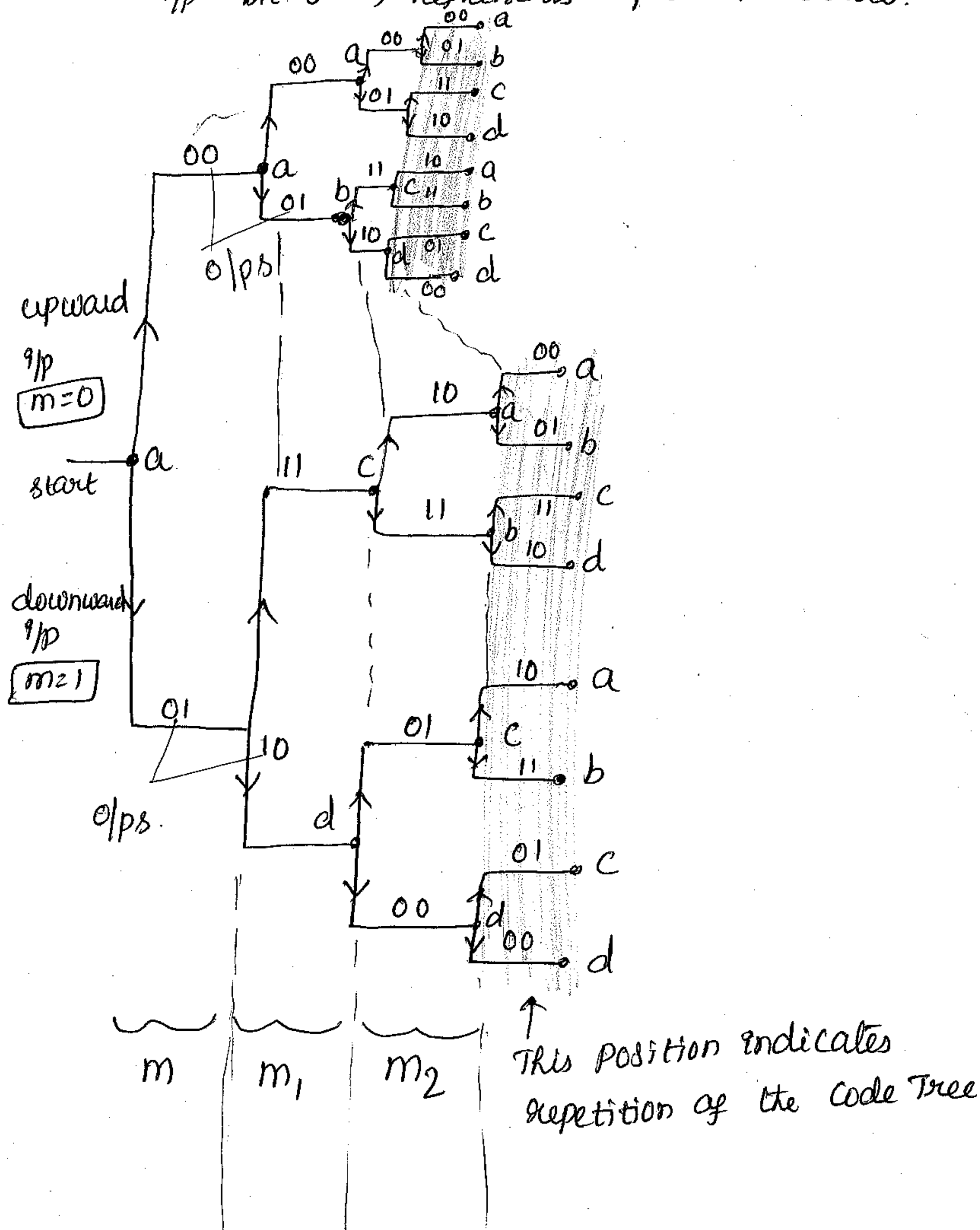
$m\ m_1\ m_2$

1 0 0 so Next state $m_1 m_2 = 1\ 0$

Code tree

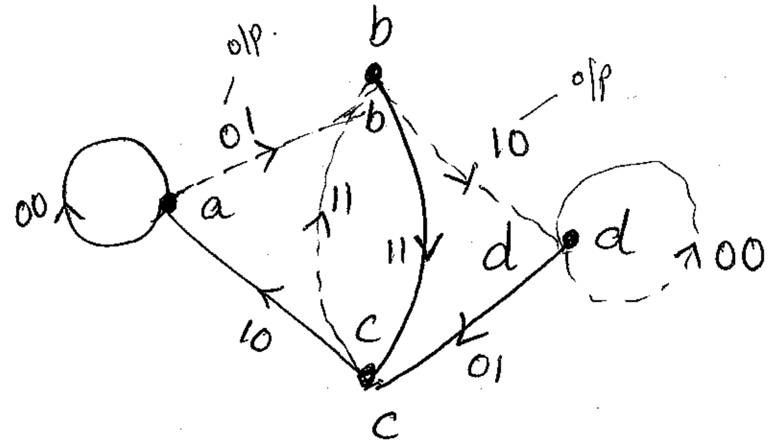
i/p bit 1 → represents downward arrow

i/p bit 0 → Represents upward arrow.




Message bit is stored in the shift Registers of the encoder for 3 bits $(m, m_1, m_2)$ after $3^{rd}$ bit repetition starts.

State diagram → combine current & Next states,
⇓
obtain state diagram.




State diagram

# Define channel capacity → is defined as the average rate of information transmission across the channel when Probability of occurrence of the event is Maximum.

# State Shannon's channel capacity theorem. Give an Example.

The channel capacity of the white band limited Gaussian channel is

$$C = B\log_2\left(1+\frac{S}{N}\right) \text{ bits/sec}$$

C ⇒ channel capacity
B ⇒ channel Bandwidth
$\frac{S}{N}$ ⇒ Signal to Noise Power Ratio.

# State channel coding theorem (Shannon's Second Theorem)

Given a source of M equally likely messages, with M>>1, which is generating information at a Rate R, Give channel with channel capacity 'C'. Then if, $\boxed{R \leq C}$, there exists a coding technique such that the o/p of the source may be transmitted over the channel with a probability of error in the Received message which may be made arbitrarily small.

* coding techniques are used to detect and correct the errors.