ITC - Mod 4

- Convolutional Codes

Block Codes: 'n' digits generated depends on 'k' input message within a particular time-unit

Convolutional: also depends on preceding $(m-1)$ blocks of msg digits $(m>1)$. Usually k & n = small

Message blocks $\longrightarrow$ | Convolutional Encoder | $\longrightarrow$ Code Block

(n, k, m)

n: no. of outputs / mod-2 adders

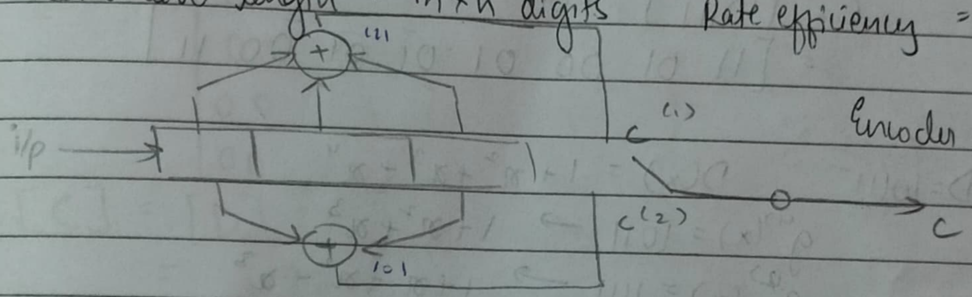
k: no. of i/p bits entering at any time

m: no. of stages of shift register / no. of FF.

L: no. of message bits in sequence

Constraint length = $m \times n$ digits

Rate efficiency = k/n



Encoder diagram

Encoding → Time domain → Conventional method

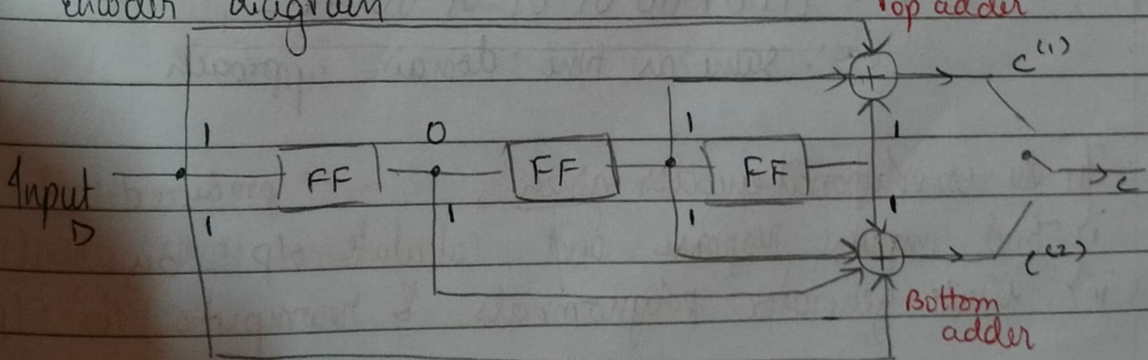
Encoding → Transfer domain → Generator matrix

- Questions

1. (2,1,3) convolution encoder, find o/p sequence for i/p d = 10111 using i) Time domain approach
   ii) Transfer-domain approach, for $g^{(1)} = 1011$, $g^{(2)} = 1111$

a) n = 2 modulo adders k = 1 = no. of inputs
   m = 3 FFs L = 5

Encoder diagram



Page No.
Date

Generator sequences :- $g^{(1)} = g_1^{(1)}\ g_2^{(1)}\ g_3^{(1)}\ g_4^{(1)} = 1011$
$g^{(2)} = g_1^{(2)}\ g_2^{(2)}\ g_3^{(2)}\ g_4^{(2)} = 1111$

generator matrix G is of order
$\boxed{L \times [n(L+m)]} = 5\times(2(5+3)) = 5\times 16$

i.e. $G = \begin{bmatrix} 11 & 01 & 11 & 11 & 00 & 00 & 00 & 00 \\ 00 & 11 & 01 & 11 & 11 & 00 & 00 & 00 \\ 00 & 00 & 11 & 01 & 11 & 11 & 00 & 00 \\ 00 & 00 & 00 & 11 & 01 & 11 & 11 & 00 \\ 00 & 00 & 00 & 00 & 11 & 01 & 11 & 11 \end{bmatrix}$

To find code vector $C = DG$
$C = [10111] \times G$
$= [11\ 01\ 00\ 01\ 01\ 01\ 00\ 11]$

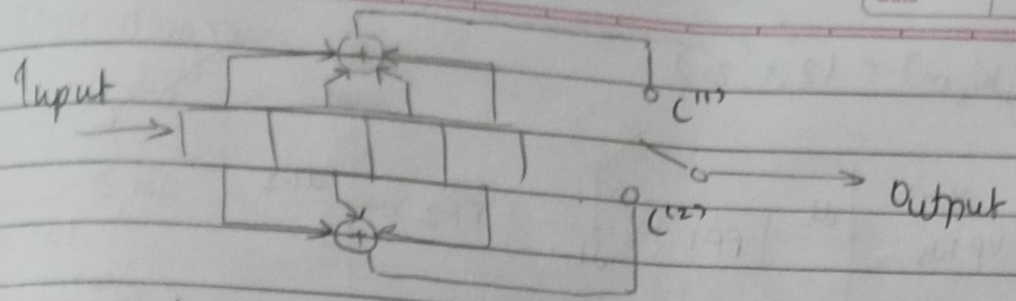
b) $D = 10111$ $\quad D(x) = 1 + x^2 + x^3 + x^4$
$g^{(1)}(x) = 1011 \rightarrow 1 + x^2 + x^3$
$g^{(2)}(x) = 1111 \rightarrow 1 + x + x^2 + x^3$

Bottom adder $c^2(x) = d(x)\, g^{(2)}(x) = (1 + x^2 + x^3 + x^4)(1 + x + x^2 + x^3)$
$= 1 + x + x^3 + x^4 + x^5 + x^7$

Top adder $c'(x) = d(x)\, g^{(1)}(x) = (1 + x^2 + x^3 + x^4)(1 + x^2 + x^3) = 1 + x^7$

Final encoder output :-
$C(x) = c'(x^n) + x c^{(2)}(x^n) + x^2 c^3(x^n) + \dots + x^{n-1}(c^n)(x^n)$
$C(x) = c^{(1)}(x^2) + x c^2(x^2)$
$= (1 + x^{7\times 2}) + x(1 + x^2 + x^{3\times 2} + x^{4\times 2} + x^{5\times 2} + x^{7\times 2})$
$= 1 + x + x^3 + x^7 + x^9 + x^{11} + x^{14} + x^{15}$
$C(x) = [11\ 01\ 00\ 01\ 01\ 01\ 00\ 11]$
$\therefore$ same as time-domain approach

2 For the convolutional encoder given,
i) Find impulse response and calculate o/p produced by 10111
ii) Write generator polynomials & recompute the o/p



→ no connection from i/p to adders so, $g^{(1)} = [01111]$
and $g^{(2)} = [01101]$

0 1 1 0 1
00 11 11 10 11

$L \xrightarrow{} D$ $L = 5$ $n \xrightarrow{} +$ $n = 2$ $m \xrightarrow{} FF$ $m = 4$ $k = 1$ → no. of i/p

$[G] = L \times [n(L+m)] = 5 \times [2(5+4)] = 5 \times 18$

$$[G] = \begin{bmatrix} 00 & 11 & 11 & 10 & 11 & 00 & 00 & 00 & 00 \\ 00 & 00 & 11 & 11 & 10 & 11 & 00 & 00 & 00 \\ 00 & 00 & 00 & 11 & 11 & 10 & 11 & 00 & 00 \\ 00 & 00 & 00 & 00 & 11 & 11 & 10 & 11 & 00 \\ 00 & 00 & 00 & 00 & 00 & 11 & 11 & 10 & 11 \end{bmatrix}$$

$[C] = [D][G] = [10111][G]$
$= [00\ 11\ 11\ 01\ 11\ 10\ 10\ 01\ 11]$

ii) $d(x) = 1 + x^2 + x^3 + x^4$
$g^{(1)}(x) = x + x^2 + x^3 + x^4$ top (+)
$g^{(2)}(x) = x + x^2 + x^4$ bottom (+)

o/p polynomial for adders
$c^{(1)}(x) = d(x)\, g^{(1)}(x) = x + x^2 + x^4 + x^5 + x^6 + x^8$
$c^{(2)}(x) = d(x)\, g^{(2)}(x) = x + x^2 + x^3 + x^4 + x^7 + x^8$

Final o/p $C(x) = c^{(1)}[x^2] + x\, c^{(2)}(x^2)$
$= (x^2 + x^{2\times 2} + x^8 + x^{10} + x^{12} + x^{16}) + x(x^2 + x^4 + x^6 + x^8 + x^{14} + x^{16})$
$= x^2 + x^3 + x^4 + x^5 + x^7 + x^8 + x^9 + x^{10} + x^{12} + x^{15} + x^{16} + x^{17}$

if there are 2 same variables, it'll be 0. eg. $x^2 + x^2 = 0$

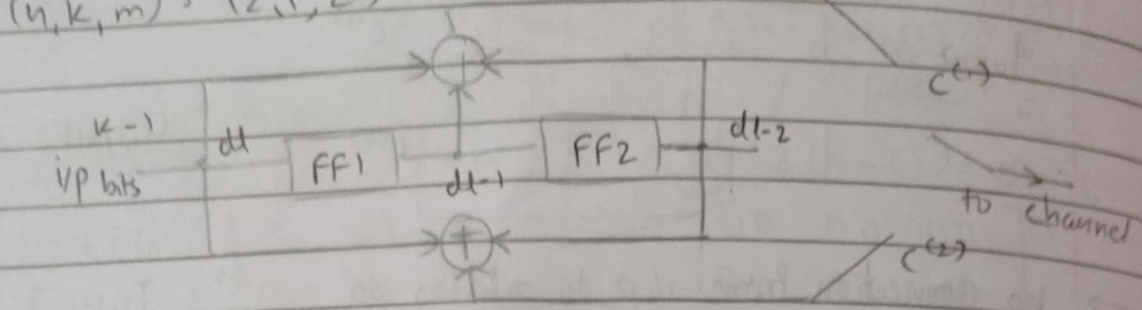
$[C] = [00\ 11\ 11\ 01\ 11\ 10\ 10\ 01\ 11]$

- State table, State transition table, State diagram, Code tree

1. Draw state transition table, state diagram & code tree for (2, 1, 2) convolution encoder with impulse response $g^{(1)} = 111$, $g^{(2)} = 101$ and input $D = [10111]$.

mod o/p wo. → 2 FF

→ $(n, k, m) = (2, 1, 2)$



State table : 2 FFs in the shift register

$\boxed{2^2 = 4 \text{ states}}$ with binary data

State :	S_0	S_1	S_2	S_3
Binary Description:	00	10	01	11

State Transition Table :

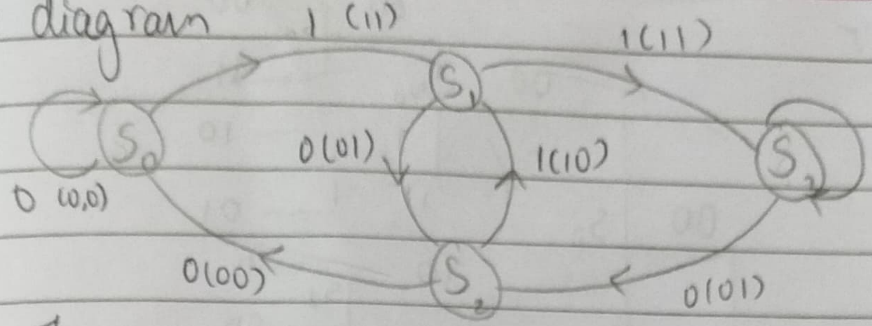
Indicates transition b/w states and also corresponding code bits during each transition.

$c^{(1)} = d_l + d_{l-1} + d_{l-2}$

$c^{(2)} = d_l + d_{l-2}$

Present State	Input bit	Next State	d_l	d_{l-1}	d_{l-2}	Output $c^{(1)}$	$c^{(2)}$
$S_0 = 00$	0	$S_0 = 00$	0	0	0	0	0
	1	$S_1 = 10$	1	0	0	1	1
$S_1 = 10$	0	$S_2 = 01$	0	1	0	1	0
	1	$S_3 = 11$	1	1	0	0	1
$S_2 = 01$	0	$S_0 = 00$	0	0	1	1	1
	1	$S_1 = 10$	1	0	1	0	0
$S_3 = 11$	0	$S_2 = 01$	0	1	1	0	1
	1	$S_3 = 11$	1	1	1	1	0

State diagram



Code Tree :-

input = 0 = up 1 = down ⟶ to node ↓ to branch

D=[10111]

$$S_0 \xrightarrow[11]{1} S_1 \xrightarrow[01]{0} S_2 \xrightarrow[10]{1} S_1 \xrightarrow[11]{1} S_3 \xrightarrow[10]{1} S_3 \underbrace{\xrightarrow[01]{0} S_2 \xrightarrow[11]{0}}_{} S_0$$

that's why we add 2 extra bits

$n(L+m) = 2(5+2) = 14$ bits

$[C] = [11\ 01\ 10\ 11\ 10\ 01\ 11]$

↓

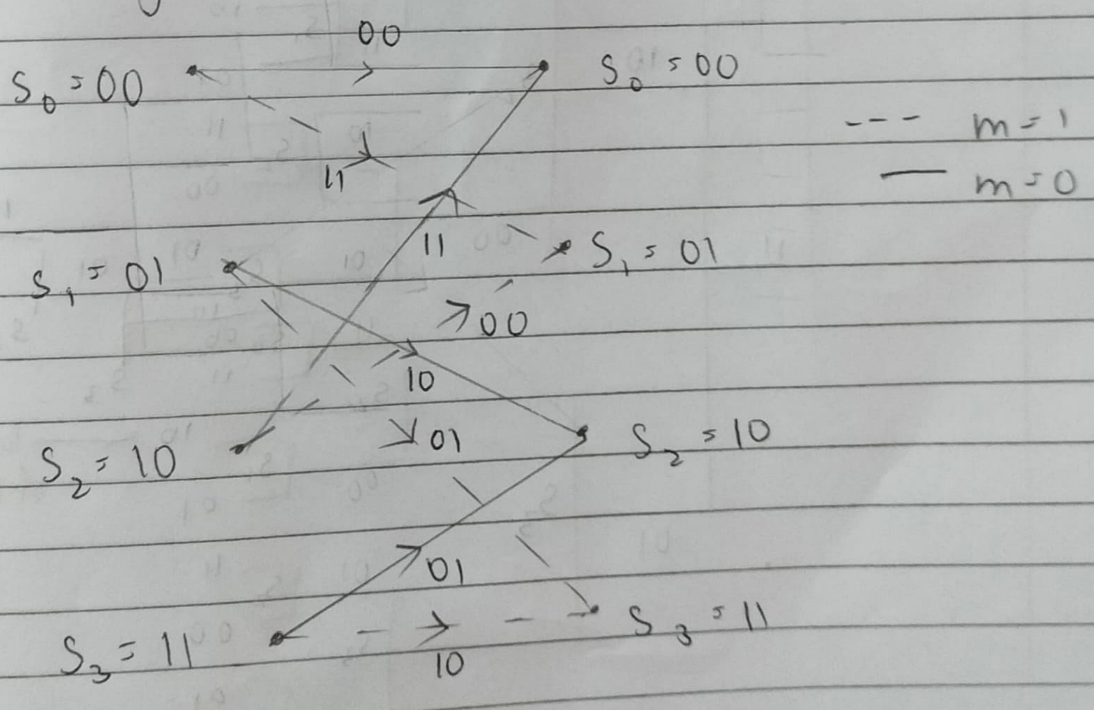
discard last m bits

Decode :-

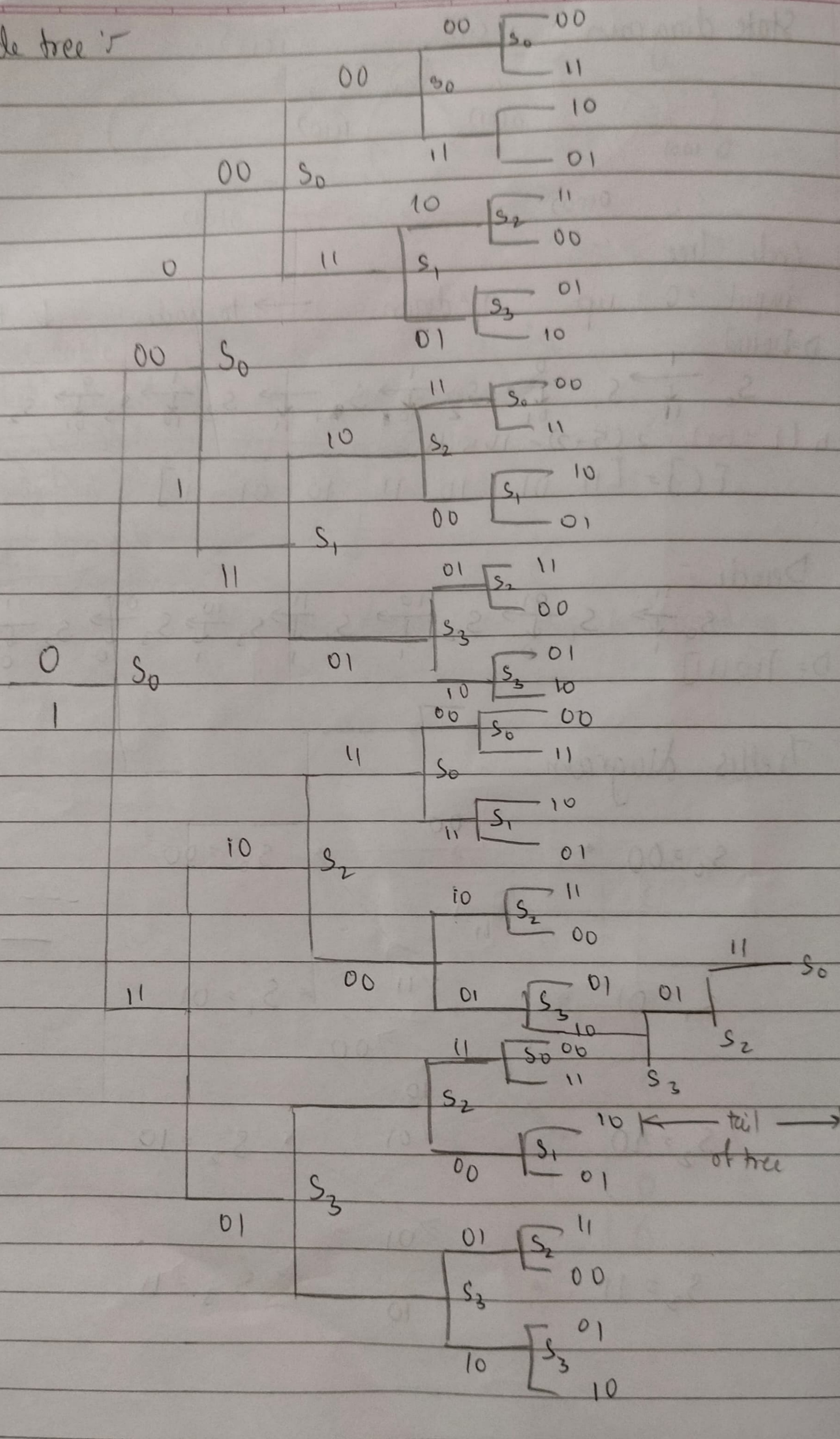
$$S_0 \xrightarrow[1]{11} S_1 \xrightarrow[0]{01} S_2 \xrightarrow[1]{10} S_1 \xrightarrow[1]{11} S_3 \xrightarrow[1]{10} \overbrace{S_3 \xrightarrow[0]{01} S_2 \xrightarrow[0]{11} S_0}$$

D= [10111]

Trellis diagram



Code tree :-



0
1
S0
00
S0
0
1
00
S0
11
S1
10
11
01
S2
S3
00
11
10
01
S0
S1
S2
S3
tail
of tree