CHAPTER 8

Problem 8.1 :

(a) Interchanging the first and third rows, we obtain the systematic form :

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

(b)

$$\mathbf{H} = \left[\mathbf{P}^T | \mathbf{I}_4\right] = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

(c) Since we have a (7,3) code, there are $2^3 = 8$ valid codewords, and 2^4 possible syndromes. From these syndromes the all-zero one corresponds to no error, 7 will correspond to single errors and 8 will correspond to double errors (the choice is not unique) :

Error pattern	Syndrome
0 0 0 0 0 0 0	0 0 0 0
0 0 0 0 0 0 1	0 0 0 1
0 0 0 0 0 1 0	0 0 1 0
0 0 0 0 1 0 0	0 1 0 0
0 0 0 1 0 0 0	1 0 0 0
0 0 1 0 0 0 0	1 1 0 1
0 1 0 0 0 0 0	0 1 1 1
1 0 0 0 0 0 0	1 1 1 0
1 0 0 0 0 0 1	1 1 1 1
1 0 0 0 0 1 0	1 1 0 0
1 0 0 0 1 0 0	1 0 1 0
1 0 0 1 0 0 0	0 1 1 0
1 0 1 0 0 0 0	0 0 1 1
1 1 0 0 0 0 0	1 0 0 1
0 1 0 0 0 1 0	0 1 0 1
0 0 0 1 1 0 1	1 0 1 1

(d) We note that there are 3 linearly independent columns in $\mathbf{H}$, hence there is a codeword $\mathbf{C}_m$ with weight w_m =4 such that $\mathbf{C}_m\mathbf{H}^T = 0$. Accordingly : $d_{\min} = 4$. This can be also obtained by generating all 8 codewords for this code and checking their minimum weight.

(e) 101 generates the codeword : $101 \rightarrow \mathbf{C} = 1010011$. Then : $\mathbf{CH}^T = [0000]$.

Problem 8.2 :

$$\mathbf{G}_a = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \qquad \mathbf{G}_b = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Message $\mathbf{X}_{\mathrm{m}}$	$\mathbf{C}_{\mathrm{ma}} = \mathbf{X}_{\mathrm{m}}\mathbf{G}_{\mathrm{a}}$	$\mathbf{C}_{\mathrm{mb}} = \mathbf{X}_{\mathrm{m}}\mathbf{G}_{\mathrm{b}}$
0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 0
0 0 0 1	0 0 0 1 0 1 1	0 0 0 1 0 1 1
0 0 1 0	0 0 1 0 1 1 0	0 0 1 0 1 1 0
0 0 1 1	0 0 1 1 1 0 1	0 0 1 1 1 0 1
0 1 0 0	0 1 0 1 1 0 0	0 1 0 0 1 1 1
0 1 0 1	0 1 0 0 1 1 1	0 1 0 1 1 0 0
0 1 1 0	0 1 1 1 0 1 0	0 1 1 0 0 0 1
0 1 1 1	0 1 1 0 0 0 1	0 1 1 1 0 1 0
1 0 0 0	1 0 1 1 0 0 0	1 0 0 0 1 0 1
1 0 0 1	1 0 1 0 0 1 1	1 0 0 1 1 1 0
1 0 1 0	1 0 0 1 1 1 0	1 0 1 0 0 1 1
1 0 1 1	1 0 0 0 1 0 1	1 0 1 1 0 0 0
1 1 0 0	1 1 1 0 1 0 0	1 1 0 0 0 1 0
1 1 0 1	1 1 1 1 1 1 1	1 1 0 1 0 0 1
1 1 1 0	1 1 0 0 0 1 0	1 1 1 0 1 0 0
1 1 1 1	1 1 0 1 0 0 1	1 1 1 1 1 1 1

As we see, the two generator matrices generate the same set of codewords.

Problem 8.3 :

The weight distribution of the (7,4) Hamming code is ($n = 7$) :

$$\begin{aligned} A(x) &= \tfrac{1}{8}\left[(1+x)^7 + 7(1+x)^3(1-x)^4\right] \\ &= \tfrac{1}{8}\left[8 + 56x^3 + 56x^4 + 8x^7\right] \\ &= 1 + 7x^3 + 7x^4 + x^7 \end{aligned}$$

Hence, we have 1 codeword of weight zero, 7 codewords of weight 3, 7 codewords of weight 4, and one codeword of weight 7. which agrees with the codewords given in Table 8-1-2.

Problem 8.4:

(a) The generator polynomial for the (15,11) Hamming code is given as $g(p) = p^4 + p + 1$. We will express the powers p^l as : $p^l = Q_l(p)g(p) + R_l(p)$ $l = 4, 5, ...14$, and the polynomial $R_l(p)$ will give the parity matrix $\mathbf{P}$, so that $\mathbf{G}$ will be $\mathbf{G} = [\mathbf{I}_{11}|\mathbf{P}]$. We have :

$$\begin{aligned}
p^4 &= g(p) + p + 1 \\
p^5 &= pg(p) + p^2 + p \\
p^6 &= p^2 g(p) + p^3 + p^2 \\
p^7 &= (p^3 + 1)g(p) + p^3 + p + 1 \\
p^8 &= (p^4 + p + 1)g(p) + p^2 + 1 \\
p^9 &= (p^5 + p^2 + p)g(p) + p^3 + p \\
p^{10} &= (p^6 + p^3 + p^2 + 1)g(p) + p^2 + p + 1 \\
p^{11} &= (p^7 + p^4 + p^3 + p)g(p) + p^3 + p^2 + p \\
p^{12} &= (p^8 + p^5 + p^4 + p^2 + 1)g(p) + p^3 + p^2 + p + 1 \\
p^{13} &= (p^9 + p^6 + p^5 + p^3 + p + 1)g(p) + p^3 + p^2 + 1 \\
p^{14} &= (p^{10} + p^7 + p^6 + p^4 + p^2 + p + 1)g(p) + p^3 + 1
\end{aligned}$$

Using $R_l(p)$ (with $l = 4$ corresponding to the last row of $\mathbf{G}$,... $l = 14$ corresponding to the first row) for the parity matrix $\mathbf{P}$ we obtain :

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

(b) In order to obtain the generator polynomial for the dual code, we first factor $p^{15} + 1$ into : $p^{15+1} = g(p)h(p)$ to obtain the parity polynomial $h(p) = (p^{15} + 1)/g(p) = p^{11} + p^8 + p^7 + p^5 + p^3 + p^2 + p + 1$. Then, the generator polynomial for the dual code is given by :

$$p^{11}h(p^{-1}) = 1 + p^3 + p^4 + p^6 + p^8 + p^9 + p^{10} + p^{11}$$

Problem 8.5 :

We can determine $\mathbf{G}$, in a systematic form, from the generator polynomial $g(p) = p^3 + p^2 + 1$:

$$\begin{array}{l} p^6 = (p^3+p^2+p)g(p) + p^2 + p \\ p^5 = (p^2+p+1)g(p) + p + 1 \\ p^4 = (p+1)g(p) + p^2 + p + 1 \\ p^3 = g(p) + p^2 + 1 \end{array} \quad \mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \quad \mathbf{H} = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Hence, the parity check matrix for the extended code will be (according to 8-1-15) :

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

and in systematic form (we add rows 1,2,3 to the last one) :

$$\mathbf{H}_{es} = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \Rightarrow \mathbf{G}_{es} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Note that $\mathbf{G}_{es}$ can be obtained from the generator matrix $\mathbf{G}$ for the initial code, by adding an overall parity check bit. The code words for the extended systematic code are :

Message $\mathbf{X}_{\mathrm{m}}$	Codeword $\mathbf{C}_{\mathrm{m}}$
0 0 0 0	0 0 0 0 0 0 0 0
0 0 0 1	0 0 0 1 1 0 1 1
0 0 1 0	0 0 1 0 1 1 1 0
0 0 1 1	0 0 1 1 0 1 0 1
0 1 0 0	0 1 0 0 0 1 1 1
0 1 0 1	0 1 0 1 1 1 0 0
0 1 1 0	0 1 1 0 1 0 0 1
0 1 1 1	0 1 1 1 0 0 1 0
1 0 0 0	1 0 0 0 1 1 0 1
1 0 0 1	1 0 0 1 0 1 1 0
1 0 1 0	1 0 1 0 0 0 1 1
1 0 1 1	1 0 1 1 1 0 0 0
1 1 0 0	1 1 0 0 1 0 1 0
1 1 0 1	1 1 0 1 0 0 0 1
1 1 1 0	1 1 1 0 0 1 0 0
1 1 1 1	1 1 1 1 1 1 1 1

An alternative way to obtain the codewords for the extended code is to add an additional check bit to the codewords of the initial (7,4) code which are given in Table 8-1-2. As we see, the minimum weight is 4 and hence : $d_{\min} = 4$.

Problem 8.6 :

(a) We have obtained the generator matrix $\mathbf{G}$ for the (15,11) Hamming code in the solution of Problem 8.4. The shortened code will have a generator matrix $\mathbf{G}_s$ obtained by $\mathbf{G}$, by dropping its first 7 rows and the first 7 columns or :

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

Then the possible messages and the codewords corresponding to them will be :

Message $\mathbf{X}_{\rm m}$	Codeword $\mathbf{C}_{\rm m}$
0 0 0 0	0 0 0 0 0 0 0 0
0 0 0 1	0 0 0 1 0 0 1 1
0 0 1 0	0 0 1 0 0 1 1 0
0 0 1 1	0 0 1 1 0 1 0 1
0 1 0 0	0 1 0 0 1 1 0 0
0 1 0 1	0 1 0 1 1 1 1 1
0 1 1 0	0 1 1 0 1 0 1 0
0 1 1 1	0 1 1 1 1 0 0 1
1 0 0 0	1 0 0 0 1 0 1 1
1 0 0 1	1 0 0 1 1 0 0 0
1 0 1 0	1 0 1 0 1 1 0 1
1 0 1 1	1 0 1 1 1 0 1 0
1 1 0 0	1 1 0 0 0 1 1 1
1 1 0 1	1 1 0 1 0 1 0 0
1 1 1 0	1 1 1 0 0 0 0 1
1 1 1 1	1 1 1 1 0 0 1 0

(b) As we see the minimum weight and hence the minimum distance is 3 : $d_{\min} = 3$.

Problem 8.7 :

(a)

$$g(p) = (p^4 + p^3 + p^2 + p + 1)(p^4 + p + 1)(p^2 + p + 1) = p^{10} + p^8 + p^5 + p^4 + p^2 + p + 1$$

Factoring p^l, $l = 14, ...10$, into $p^l = g(p)Q_l(p) + R_l(p)$ we obtain the generator matrix in systematic form :

$$\left.\begin{array}{l} p^{14} = (p^4+p^2+1)g(p) + p^9+p^7+p^4+p^3+p+1 \\ p^{13} = (p^3+p)g(p) + p^9+p^8+p^7+p^6+p^4+p^2+p \\ p^{12} = (p^2+1)g(p) + p^8+p^7+p^6+p^5+p^3+p+1 \\ p^{11} = pg(p) + p^9+p^6+p^5+p^3+p^2+p \\ p^{10} = g(p) + p^8+p^5+p^4+p^2+p+1 \end{array}\right\} \Rightarrow$$

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

The codewords are obtained from the equation : $\mathbf{C}_m = \mathbf{X}_m\mathbf{G}$, where X_m is the row vector containing the five message bits.

(b)

$$d_{\min} = 7$$

(c) The error-correcting capability of the code is :

$$t = \left\lfloor \frac{d_{\min}-1}{2} \right\rfloor = 3$$

(d) The error-detecting capability of the code is : $d_{\min} - 1 = 6$.

(e)

$$g(p) = (p^{15}+1)/(p^2+p+1) = p^{13}+p^{12}+p^{10}+p^9+p^7+p^6+p^4+p^3+p+1$$

Then :

$$\begin{array}{l} p^{14} = (p+1)g(p) + p^{12}+p^{11}+p^9+p^8+p^6+p^5+p^3+p^2+1 \\ p^{13} = g(p) + p^{12}+p^{10}+p^9+p^7+p^6+p^4+p^3+p+1 \end{array}$$

Hence, the generator matrix is :

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

and the valid codewords :

$\mathbf{X}_\mathrm{m}$	Codeword $\mathbf{C}_\mathrm{m}$
0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1	0 1 1 0 1 1 0 1 1 0 1 1 0 1 1
1 0	1 0 1 1 0 1 1 0 1 1 0 1 1 0 1
1 1	1 1 0 1 1 0 1 1 0 1 1 0 1 1 0

The minimum distance is : $d_{\min} = 10$

Problem 8.8 :

The polynomial p^7+1 is factors as follows : $p^7+1 = (p+1)(p^3+p^2+1)(p^3+p+1)$. The generator polynomials for the matrices $\mathbf{G}_1, \mathbf{G}_2$ are : $g_1(p) = p^3+p^2+1,\ g_2(p) = p^3+p+1$. Hence the parity polynomials are : $h_1(p) = (p^7+1)/g_1(p) = p^4+p^3+p^2+1,\ h_2(p) = (p^7+1)/g_2(p) = p^4+p^2+p+1$. The generator polynomials for the matrices $\mathbf{H}_1, \mathbf{H}_2$ are : $p^4h_1(p^{-1}) = 1+p+p^2+p^4,\ p^4h_2(p^{-1}) = 1+p^2+p^3+p^4$. The rows of the matrices $\mathbf{H}_1, \mathbf{H}_2$ are given by : $p^ip^4h_{1/2}(p^{-1}),\ i = 0,1,2$, so :

$$\mathbf{H}_1 = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix} \quad \mathbf{H}_2 = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}$$

Problem 8.9 :

We have already generated an extended (8,4) code from the (7,4) Hamming code in Probl. 8.5. Since the generator matrix for the (7,4) Hamming code is not unique, in this problem we will construct the extended code, starting from the generator matrix given in 8-1-7 :

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

Then :

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

We can bring this parity matrix into systematic form by adding rows 1,2,3 into the fourth row :

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

Then :

$$G_{e,s} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Problem 8.10 :

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

Then the standard array is :

000	001	010	011	100	101	110	111
000000	001101	010011	011110	100110	101011	110101	111000
000001	001100	010010	011111	100111	101010	110100	111001
000010	001111	010001	011100	100100	101001	110111	111010
000100	001001	010111	011010	100010	101111	110001	111100
001000	000101	011011	010110	101110	100011	111101	110000
010000	011101	000011	001110	110110	111011	100101	101000
100000	101101	110011	111110	000110	001011	010101	011000
100001	101100	110010	111111	000111	001010	010100	011001

For each column, the first row is the message, the second row is the correct codeword corresponding to this message, and the rest of the rows correspond to the received words which are the sum of the valid codeword plus the corresponding error pattern (coset leader). The error patterns that this code can correct are given in the first column (all-zero codeword), and the corresponding syndromes are :

E_i	$S_i = E_i H^T$
000000	000
000001	001
000010	010
000100	100
001000	101
010000	011
100000	110
100001	111

We note that this code can correct all single errors and one two-bit error pattern.

Problem 8.11 :

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

Then, the standard array will be :

000	001	010	011	100	101	110	111
0000000	0010111	0101110	0111001	1001011	1011100	1100101	1110010
0000001	0010110	0101111	0111000	1001010	1011101	1100100	1110011
0000010	0010101	0101101	0111011	1001001	1011110	1100111	1110000
0000100	0010011	0101010	0111101	1001111	1011000	1100001	1110110
0001000	0011111	0100110	0110001	1000011	1010100	1101101	1111010
0010000	0000111	0111110	0101001	1011011	1001100	1110101	1100010
0100000	0110111	0001110	0011001	1101011	1111100	1000101	1010010
1000000	1010111	1101110	1111001	0001011	0011100	0100101	0110010
1100000	1110111	1001110	1011001	0101011	0111100	0000101	0010010
1010000	1000111	1111110	1101001	0011011	0001100	0110101	0100010
1001000	1011111	1100110	1110001	0000011	0010100	0101101	0111010
1000100	1010011	1101010	1111101	0001111	0011000	0100001	0110110
1000010	1010101	1101100	1111010	0001001	0011110	0100111	0110000
1000001	1010110	1101111	1111001	0001010	0011101	0100100	0110011
0010001	0000110	0111111	0101001	1011010	1001101	1110100	1100011
0001101	0011010	0100011	0110101	1000110	1010001	1101000	1111111

For each column, the first row is the message, the second row is the correct codeword corresponding to this message, and the rest of the rows correspond to the received words which are the sum of the valid codeword plus the corresponding error pattern (coset leader). The error patterns that this code can correct are given in the first column (all-zero codeword), and the corresponding syndromes are :

E_i	$S_i = E_i H^T$
0000000	0000
0000001	0001
0000010	0010
0000100	0100
0001000	1000
0010000	0111
0100000	1110
1000000	1011
1100000	0101
1010000	11000
1001000	0011
1000100	1111
1000010	1001
1000001	1010
0010001	0110
0001101	1101

We note that this code can correct all single errors, seven two-bit error patterns, and one three-bit error pattern.

Problem 5.12 :

The generator matrix for the systematic (7,4) cyclic Hamming code is given by (8-1-37) as :

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

Then, the correctable error patterns E_i with the corresponding syndrome $S_i = E_i H^T$ are :

S_i	E_i
000	0000000
001	0000001
010	0000010
011	0001000
100	0000100
101	1000000
110	0100000
111	0010000

Problem 8.13 :

We know that : $\mathbf{e}_1 + \mathbf{e}_2 = \mathbf{C}$, where $\mathbf{C}$ is a valid codeword. Then :

$$\mathbf{S}_1 + \mathbf{S}_2 = \mathbf{e}_1\mathbf{H}^T + \mathbf{e}_2\mathbf{H}^T = (\mathbf{e}_1 + \mathbf{e}_2)\mathbf{H}^T = \mathbf{C}\mathbf{H}^T = \mathbf{0}$$

since a valid codeword is orthogonal to the parity matrix. Hence : $\mathbf{S}_1 + \mathbf{S}_2 = \mathbf{0}$, and since modulo-2 addition is the same with modulo-2 subtraction :

$$\mathbf{S}_1 - \mathbf{S}_2 = \mathbf{0} \Rightarrow \mathbf{S}_1 = \mathbf{S}_2$$

Problem 8.14 :

(a) Let $g(p) = p^8 + p^6 + p^4 + p^2 + 1$ be the generator polynomial of an (n, k) cyclic code. Then, $n - k = 8$ and the rate of the code is

$$R = \frac{k}{n} = 1 - \frac{8}{n}$$

The rate R is minimum when $\frac{8}{n}$ is maximum subject to the constraint that R is positive. Thus, the first choice of n is $n = 9$. However, the generator polynomial $g(p)$ does not divide $p^9 + 1$ and therefore, it can not generate a $(9, 1)$ cyclic code. The next candidate value of n is 10. In this case

$$p^{10} + 1 = g(p)(p^2 + 1)$$

and therefore, $n = 10$ is a valid choice. The rate of the code is $R = \frac{k}{n} = \frac{2}{10} = \frac{1}{5}$.

(b) In the next table we list the four codewords of the $(10, 2)$ cyclic code generated by $g(p)$.

Input	$X(p)$	Codeword
00	0	0000000000
01	1	0101010101
10	p	1010101010
11	$p+1$	1111111111

As it is observed from the table, the minimum weight of the code is 5 and since the code is linear $d_{\min} = w_{\min} = 5$.

(c) The coding gain of the $(10, 2)$ cyclic code in part (a) is

$$G_{\text{coding}} = d_{\min} R = 5 \times \frac{2}{10} = 1$$

Problem 8.15 :

(a) For every n

$$p^n + 1 = (p + 1)(p^{n-1} + p^{n-2} + \cdots + p + 1)$$

where additions are modulo 2. Since $p + 1$ divides $p^n + 1$ it can generate a (n, k) cyclic code, where $k = n - 1$.

(b) The i^{th} row of the generator matrix has the form

$$\mathbf{g}_i = [\ 0 \ \ \cdots \ \ 0 \ \ 1 \ \ 0 \ \ \cdots \ \ 0 \ \ p_{i,1}\ ]$$

where the 1 corresponds to the i-th column (to give a systematic code) and the $p_{i,1}$, $i = 1, \ldots, n-1$, can be found by solving the equations

$$p^{n-i} + p_{i,1} = p^{n-i} \bmod p + 1, \qquad 1 \le i \le n - 1$$

Since $p^{n-i} \bmod p + 1 = 1$ for every i, the generator and the parity check matrix are given by

$$\mathbf{G} = \left(\begin{array}{ccc|c} 1 & \cdots & 0 & 1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & 1 \end{array} \right), \qquad \mathbf{H} = [\ 1 \ \ 1 \ \ \cdots \ \ 1 \ \ | \ \ 1\ ]$$

(c) A vector $\mathbf{c} = [c_1, c_2, \ldots, c_n]$ is a codeword of the $(n, n-1)$ cyclic code if it satisfies the condition $\mathbf{c}\mathbf{H}^t = 0$. But,

$$\mathbf{c}\mathbf{H}^t = 0 = \mathbf{c}\begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} = c_1 + c_2 + \cdots c_n$$

Thus, the vector $\mathbf{c}$ belongs to the code if it has an even weight. Therefore, the cyclic code generated by the polynomial $p+1$ is a simple parity check code.

Problem 8.16 :

(a) The generator polynomial of degree $4 = n-k$ should divide the polynomial p^6+1. Since the polynomial p^6+1 assumes the factorization

$$p^6 + 1 = (p+1)^3(p+1)^3 = (p+1)(p+1)(p^2+p+1)(p^2+p+1)$$

we find that the shortest possible generator polynomial of degree 4 is

$$g(p) = p^4 + p^2 + 1$$

The i^{th} row of the generator matrix $\mathbf{G}$ has the form

$$\mathbf{g}_i = \begin{bmatrix} 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & p_{i,1} & \cdots & p_{i,4} \end{bmatrix}$$

where the 1 corresponds to the i-th column (to give a systematic code) and the $p_{i,1}, \ldots, p_{i,4}$ are obtained from the relation

$$p^{6-i} + p_{i,1}p^3 + p_{i,2}p^2 p_{i,3}p + p_{i,4} = p^{6-i}(\bmod p^4 + p^2 + 1)$$

Hence,

$$\begin{aligned} p^5 \bmod p^4 + p^2 + 1 &= (p^2+1)p \bmod p^4 + p^2 + 1 = p^3 + p \\ p^4 \bmod p^4 + p^2 + 1 &= p^2 + 1 \bmod p^4 + p^2 + 1 = p^2 + 1 \end{aligned}$$

and therefore,

$$\mathbf{G} = \left(\begin{array}{cc|cccc} 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \end{array}\right)$$

The codewords of the code are

$$\begin{aligned} \mathbf{c}_1 &= [\,0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\,] \\ \mathbf{c}_2 &= [\,1\ \ 0\ \ 1\ \ 0\ \ 1\ \ 0\,] \\ \mathbf{c}_3 &= [\,0\ \ 1\ \ 0\ \ 1\ \ 0\ \ 1\,] \\ \mathbf{c}_4 &= [\,1\ \ 1\ \ 1\ \ 1\ \ 1\ \ 1\,] \end{aligned}$$

(b) The minimum distance of the linear $(6,2)$ cyclic code is $d_{\min} = w_{\min} = 3$. Therefore, the code can correct

$$e_c = \frac{d_{\min} - 1}{2} = 1 \text{ error}$$

Problem 8.17 :

Consider two n-tuples in the same row of a standard array. Clearly, if $\mathbf{Y}_1, \mathbf{Y}_2$ denote the n-tuples, $\mathbf{Y}_1 = \mathbf{C}_j + \mathbf{e}$, $\mathbf{Y}_2 = \mathbf{C}_k + \mathbf{e}$, where $\mathbf{C}_k, \mathbf{C}_j$ are two valid codewords, and the error pattern e is the same since they are in the same row of the standard array. Then :

$$\mathbf{Y}_1 + \mathbf{Y}_2 = \mathbf{C}_j + \mathbf{e} + \mathbf{C}_k + \mathbf{e} = \mathbf{C}_j + \mathbf{C}_k = \mathbf{C}_m$$

where $\mathbf{C}_m$ is another valid codeword (this follows from the linearity of the code).

Problem 8.18 :

From Table 8-1-6 we find that the coefficients of the generator polynomial for the (15,7) BCH code are $721 \rightarrow 111010001$ or $g(p) = p^8 + p^7 + p^6 + p^4 + 1$. Then, we can determine the l-th row of the generator matrix $\mathbf{G}$, using the modulo $R_l(p)\ : p^{n-l} = Q_l(p)g(p) + R_l(p),\ l = 1, 2, ..., 7$. Since the generator matrix of the shortened code is obtained by removing the first three rows of $\mathbf{G}$, we perform the above calculations for $l = 4, 5, 6, 7$, only :

$$\begin{array}{l}
p^{11} = (p^3 + p^2 + 1)g(p) + p^4 + p^3 + p^2 + 1 \\
p^{10} = (p^2 + p)g(p) + p^7 + p^6 + p^5 + p^2 + p \\
p^9 = (p+1)g(p) + p^6 + p^5 + p^4 + p + 1 \\
p^8 = (p+1)g(p) + p^7 + p^6 + p^4 + 1
\end{array}$$

Hence :

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

Problem 8.19 :

For M-ary FSK detected coherently, the bandwidth expansion factor is :

$$\left(\frac{W}{R}\right)_{FSK} = \frac{M}{2\log_2 M}$$

For the Hadamard code : In time T (block transmission time), we want to transmit n bits, so for each bit we have time : $T_b = T/n$. Since for each bit we use binary PSK, the bandwidth requirement is approximately : $W = 1/T_b = n/T$. But $T = k/R$, hence :

$$W = \frac{n}{k}R \Rightarrow \frac{W}{R} = \frac{n}{k}$$

(this is a general result for binary block-encoded signals). For the specific case of a Hadamard code the number of waveforms is $M = 2n$, and also $k = \log_2 M$. Hence :

$$\left(\frac{W}{R}\right)_{Had} = \frac{M}{2\log_2 M}$$

which is the same as M-ary FSK.

Problem 8.20 :

From (8-1-47) of the text, the correlation coefficient between the all-zero codeword and the l-th codeword is $\rho_l = 1 - 2w_l/n$, where w_l is the weight of the l-th codeword. For the maximum length shift register codes : $n = 2^m - 1 = M - 1$ (where m is the parameter of the code) and $w_l = 2^{m-1}$ for all codewords except the all-zero codeword. Hence :

$$\rho_l = 1 - \frac{22^{m-1}}{2^m - 1} = -\frac{1}{2^m - 1} = -\frac{1}{M-1}$$

for all l. Since the code is linear if follows that $\rho = -1/(M-1)$ between any pair of codewords. Note : An alternative way to prove the above is to express each codeword in vector form as

$$s_l = \left[\pm\sqrt{\frac{\mathcal{E}}{n}}, \pm\sqrt{\frac{\mathcal{E}}{n}}, ..., \pm\sqrt{\frac{\mathcal{E}}{n}}\right] \quad \text{(n elements in all)}$$

where $\mathcal{E} = n\mathcal{E}_b$ is the energy per codeword and note that any one codeword differs from each other at exactly 2^{m-1} bits and agrees with the other at $2^{m-1} - 1$ bits. Then the correlation coefficient is :

$$Re\left[\rho_{mk}\right] = \frac{s_l \cdot s_k}{|s_l|\,|s_k|} = \frac{\frac{\mathcal{E}}{n}\left(2^{m-1}\cdot 1 + (2^{m-1} - 1)\cdot(-1)\right)}{\frac{\mathcal{E}}{n}n} = -\frac{1}{n} = -\frac{1}{M-1}$$

Problem 8.21 :

We know that the (7,4) Huffman code has $d_{\min} = 3$ and weight distribution (Problem 8.3) : w=0 (1 codeword), w=3 (7 codewords), w=4 (7 codewords), w=7 (1 codeword).

Hence, for soft-decision decoding (8-1-51) :

$$P_M \leq 7Q\left(\sqrt{\frac{24}{7}\gamma_b}\right) + 7Q\left(\sqrt{\frac{32}{7}\gamma_b}\right) + Q\left(\sqrt{8\gamma_b}\right)$$

or a looser bound (8-1-52) :

$$P_M \leq 15Q\left(\sqrt{\frac{24}{7}\gamma_b}\right)$$

For hard-decision decoding (8-1-82):

$$P_M \leq \sum_{m=2}^{7} \binom{7}{m} p^m (1-p)^{7-m} = 1 - \sum_{m=0}^{1} \binom{7}{m} p^m (1-p)^{7-m} = 1 - 7p(1-p)^6 - (1-p)^7$$

where $p = Q\left(\sqrt{2R_c\gamma_b}\right) = Q\left(\sqrt{\frac{8}{7}\gamma_b}\right)$ or (8-1-90) :

$$P_M \leq 7\left[4p(1-p)\right]^{3/2} + 7\left[4p(1-p)\right]^2 + \left[4p(1-p)\right]^{7/2}$$

or (8-1-91) :

$$P_M \leq 14\left[4p(1-p)\right]^{3/2}$$

Problem 8.22 :

We assume that the all-zero codeword is transmitted and we determine the probability that we select codeword C_m having weight w_m. We define a random variable $X_i,\ i = 1, 2, ...w_m$ as :

$$X_i = \left\{ \begin{array}{cc} 1, & \text{with probability p} \\ -1, & \text{with probability } 1-\text{p} \end{array} \right\}$$

where p is the error probability for a bit. Then, we will erroneously select a codeword C_m of weight w_m, if more than $w_m/2$ bits are in error or if $\sum_{i=1}^{w_m} X_i \geq 0$. We assume that $p < 1/2$; then, following the exact same procedure as in Example 2-1-7 (page 60 of the text), we show that :

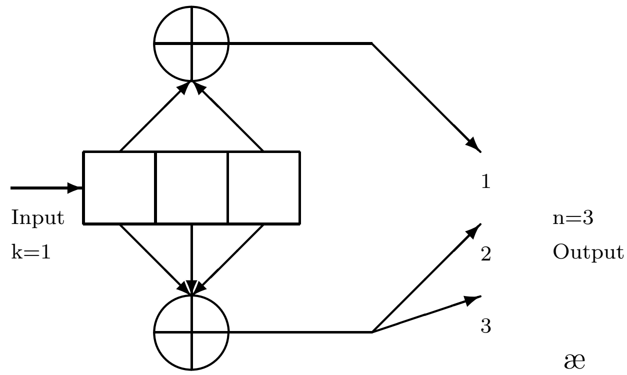
$$P\left(\sum_{i=1}^{w_m} X_i \geq 0\right) \leq \left[4p(1-p)\right]^{w_m/2}$$

By applying the union bound we obtain the desired result :

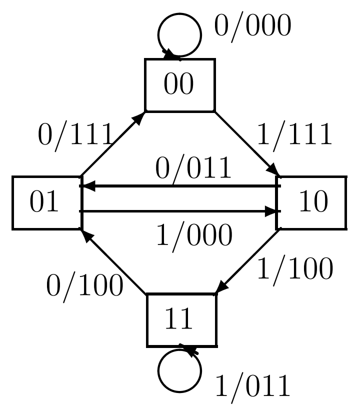
$$P_M \leq \sum_{m=2}^{M} \left[4p(1-p)\right]^{w_m/2}$$

Problem 8.23 :

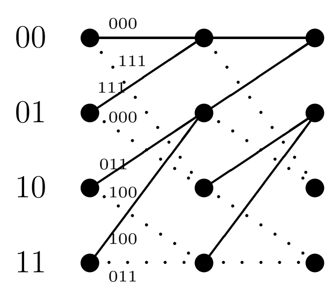
(a) The encoder for the $(3,1)$ convolutional code is depicted in the next figure.



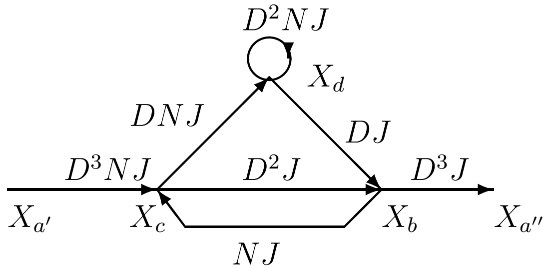
(b) The state transition diagram for this code is depicted in the next figure.



(c) In the next figure we draw two frames of the trellis associated with the code. Solid lines indicate an input equal to 0, whereas dotted lines correspond to an input equal to 1.



(d) The diagram used to find the transfer function is shown in the next figure.



Using the flow graph results, we obtain the system

$$\begin{aligned} X_c &= D^3NJX_{a'} + NJX_b \\ X_b &= D^2JX_c + DJX_d \\ X_d &= DNJX_c + D^2NJX_d \\ X_{a''} &= D^3JX_b \end{aligned}$$

Eliminating X_b, X_c and X_d results in

$$T(D,N,J) = \frac{X_{a''}}{X_{a'}} = \frac{D^8NJ^3(1+NJ-D^2NJ)}{1-D^2NJ(1+NJ^2+J-D^2J^2)}$$

To find the free distance of the code we set $N = J = 1$ in the transfer function, so that

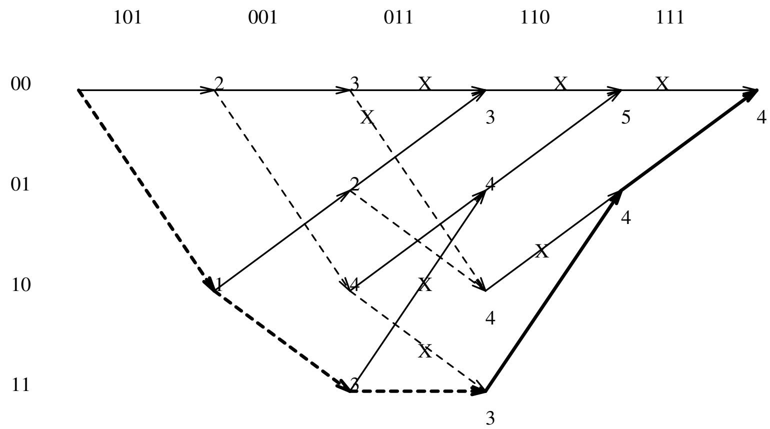
$$T_1(D) = T(D,N,J)|_{N=J=1} = \frac{D^8(1-2D^2)}{1-D^2(3-D^2)} = D^8 + 2D^{10} + \cdots$$

Hence, $d_{\text{free}} = 8$

(e) Since there is no self loop corresponding to an input equal to 1 such that the output is the all zero sequence, the code is not catastrophic.

Problem 8.24 :

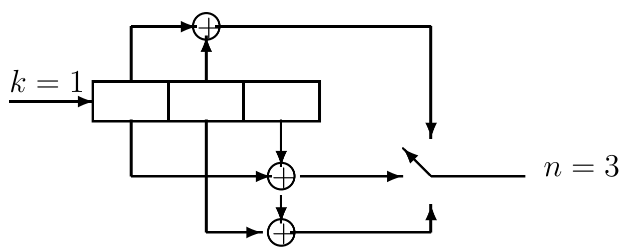
The code of Problem 8-23 is a $(3,1)$ convolutional code with $K = 3$. The length of the received sequence $\mathbf{y}$ is 15. This means that 5 symbols have been transmitted, and since we assume that the information sequence has been padded by two 0's, the actual length of the information sequence is 3. The following figure depicts 5 frames of the trellis used by the Viterbi decoder. The numbers on the nodes denote the metric (Hamming distance) of the survivor paths (the non-survivor paths are shown with an X). In the case of a tie of two merging paths at a node, we have purged the upper path.



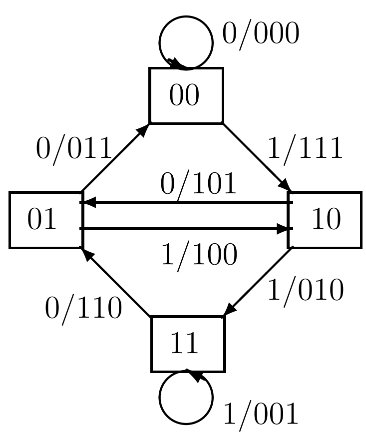
The decoded sequence is $\{111, 100, 011, 100, 111\}$ (i.e the path with the minimum final metric - heavy line) and corresponds to the information sequence $\{1, 1, 1\}$ followed by two zeros.

Problem 8.25 :

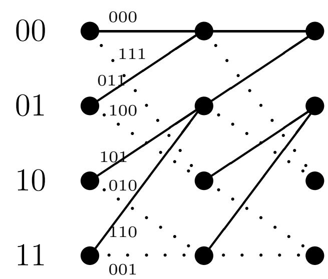
(a) The encoder for the $(3, 1)$ convolutional code is depicted in the next figure.



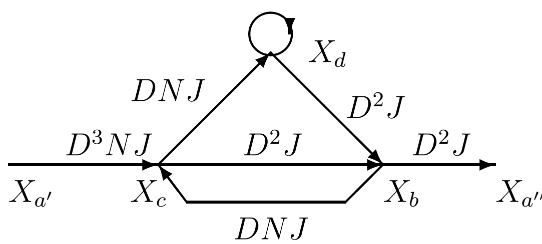
(b) The state transition diagram for this code is shown below



(c) In the next figure we draw two frames of the trellis associated with the code. Solid lines indicate an input equal to 0, whereas dotted lines correspond to an input equal to 1.



(d) The diagram used to find the transfer function is shown in the next figure.



Using the flow graph results, we obtain the system

$$\begin{aligned}
X_c &= D^3NJX_{a'} + DNJX_b \\
X_b &= D^2JX_c + D^2JX_d \\
X_d &= DNJX_c + DNJX_d \\
X_{a''} &= D^2JX_b
\end{aligned}$$

Eliminating X_b, X_c and X_d results in

$$T(D,N,J) = \frac{X_{a''}}{X_{a'}} = \frac{D^7NJ^3}{1 - DNJ - D^3NJ^2}$$

To find the free distance of the code we set $N = J = 1$ in the transfer function, so that

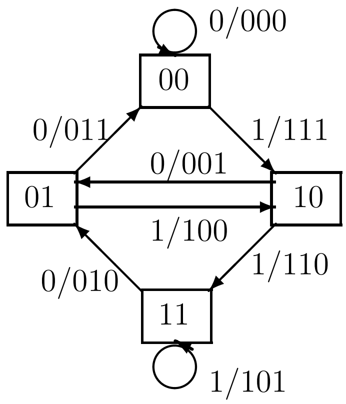
$$T_1(D) = T(D,N,J)|_{N=J=1} = \frac{D^7}{1 - D - D^3} = D^7 + D^8 + D^9 + \cdots$$

Hence, $d_{\text{free}} = 7$

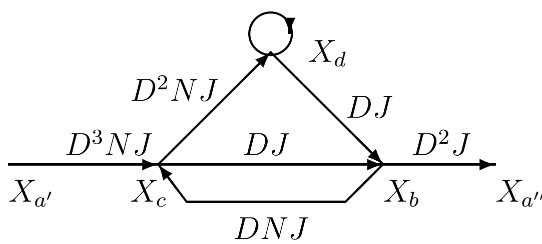
(e) Since there is no self loop corresponding to an input equal to 1 such that the output is the all zero sequence, the code is not catastrophic.

Problem 8.26 :

(a) The state transition diagram for this code is depicted in the next figure.



(b) The diagram used to find the transfer function is shown in the next figure.



Using the flow graph results, we obtain the system

$$\begin{aligned} X_c &= D^3NJX_{a'} + DNJX_b \\ X_b &= DJX_c + DJX_d \\ X_d &= D^2NJX_c + D^2NJX_d \\ X_{a''} &= D^2JX_b \end{aligned}$$

Eliminating X_b, X_c and X_d results in

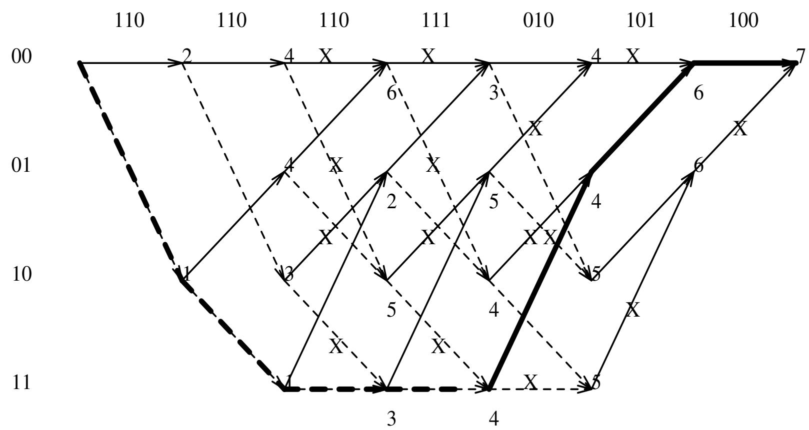
$$T(D,N,J) = \frac{X_{a''}}{X_{a'}} = \frac{D^6NJ^3}{1 - D^2NJ - D^2NJ^2}$$

(c) To find the free distance of the code we set $N = J = 1$ in the transfer function, so that

$$T_1(D) = T(D,N,J)|_{N=J=1} = \frac{D^6}{1-2D^2} = D^6 + 2D^8 + 4D^{10} + \cdots$$

Hence, $d_{\text{free}} = 6$

(d) The following figure shows 7 frames of the trellis diagram used by the Viterbi decoder. It is assumed that the input sequence is padded by two zeros, so that the actual length of the information sequence is 5. The numbers on the nodes indicate the Hamming distance of the survivor paths. The deleted branches have been marked with an X. In the case of a tie we deleted the upper branch. The survivor path at the end of the decoding is denoted by a thick line.



The information sequence is 11110 and the corresponding codeword 111 110 101 101 010 011 000...

(e) An upper to the bit error probability of the code is given by

$$P_b \leq \left.\frac{dT(D, N, J=1)}{dN}\right|_{N=1, D=\sqrt{4p(1-p)}}$$

But

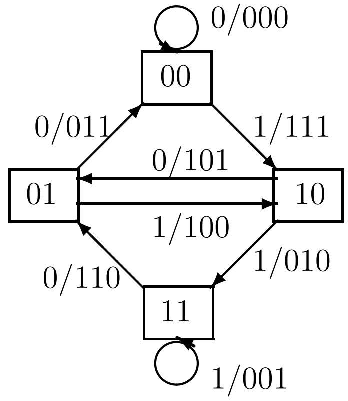
$$\frac{dT(D, N, 1)}{dN} = \frac{d}{dN}\left[\frac{D^6 N}{1-2D^2 N}\right] = \frac{D^6 - 2D^8(1-N)}{(1-2D^2N)^2}$$

and since $p = 10^{-5}$, we obtain

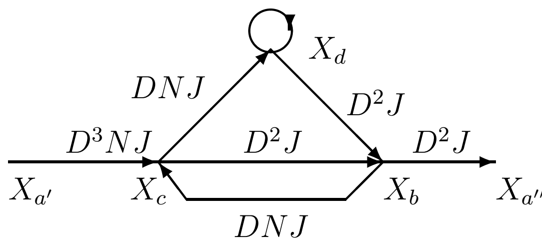
$$P_b \leq \left.\frac{D^6}{(1-2D^2)^2}\right|_{D=\sqrt{4p(1-p)}} \approx 6.14 \cdot 10^{-14}$$

Problem 8.27 :

(a) The state transition diagram for this code is shown below



(b) The diagram used to find the transfer function is shown in the next figure.



Using the flow graph results, we obtain the system

$$\begin{aligned} X_c &= D^3NJX_{a'} + DNJX_b \\ X_b &= D^2JX_c + D^2JX_d \\ X_d &= DNJX_c + DNJX_d \\ X_{a''} &= D^2JX_b \end{aligned}$$

Eliminating X_b, X_c and X_d results in

$$T(D,N,J) = \frac{X_{a''}}{X_{a'}} = \frac{D^7NJ^3}{1 - DNJ - D^3NJ^2}$$

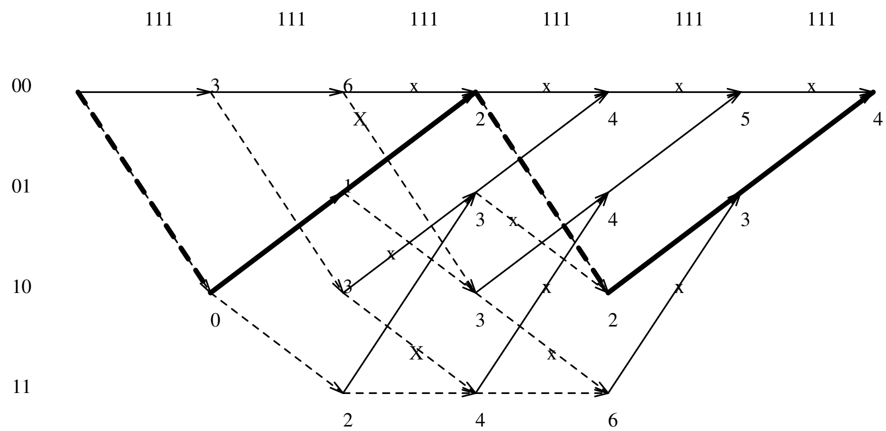
(c) To find the free distance of the code we set $N = J = 1$ in the transfer function, so that

$$T_1(D) = T(D,N,J)|_{N=J=1} = \frac{D^7}{1 - D - D^3} = D^7 + D^8 + D^9 + \cdots$$

Hence, $d_{\text{free}} = 7$. The path, which is at a distance d_{free} from the all zero path, is the path $X_a \to X_c \to X_b \to X_a$.

(d) The following figure shows 6 frames of the trellis diagram used by the Viterbi algorithm to decode the sequence $\{111, 111, 111, 111, 111, 111\}$. The numbers on the nodes indicate the

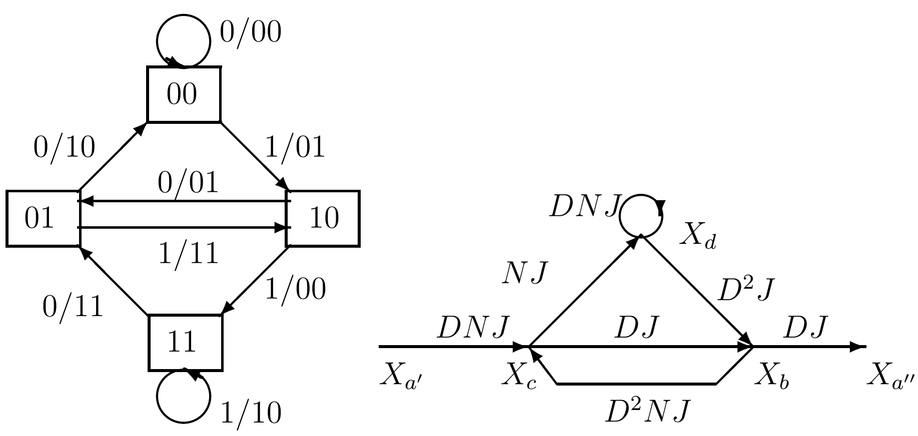
Hamming distance of the survivor paths from the received sequence. The branches that are dropped by the Viterbi algorithm have been marked with an X. In the case of a tie of two merging paths, we delete the upper path.



The decoded sequence is $\{111, 101, 011, 111, 101, 011\}$ which coresponds to the information sequence $\{x_1, x_2, x_3, x_4\} = \{1, 0, 0, 1\}$ followed by two zeros.

Problem 8.28 :

(a) The state transition diagram and the flow diagram used to find the transfer function for this code are depicted in the next figure.



Thus,

$$\begin{aligned} X_c &= DNJX_{a'} + D^2NJX_b \\ X_b &= DJX_c + D^2JX_d \\ X_d &= NJX_c + DNJX_d \end{aligned}$$

$$X_{a''} = DJX_b$$

and by eliminating X_b, X_c and X_d, we obtain

$$T(D,N,J) = \frac{X_{a''}}{X_{a'}} = \frac{D^3NJ^3}{1 - DNJ - D^3NJ^2}$$

To find the transfer function of the code in the form $T(D,N)$, we set $J = 1$ in $T(D,N,J)$. Hence,

$$T(D,N) = \frac{D^3N}{1 - DN - D^3N}$$

(b) To find the free distance of the code we set $N = 1$ in the transfer function $T(D,N)$, so that

$$T_1(D) = T(D,N)|_{N=1} = \frac{D^3}{1 - D - D^3} = D^3 + D^4 + D^5 + 2D^6 + \cdots$$

Hence, $d_{\text{free}} = 3$

(c) An upper bound on the bit error probability, when hard decision decoding is used, is given by (see (8-2-34))

$$P_b \leq \frac{1}{k} \frac{dT(D,N)}{dN}\Big|_{N=1,D=\sqrt{4p(1-p)}}$$

Since

$$\frac{dT(D,N)}{dN}\Big|_{N=1} = \frac{d}{dN} \frac{D^3N}{1 - (D + D^3)N}\Big|_{N=1} = \frac{D^3}{(1 - (D + D^3))^2}$$

with $k = 1$, $p = 10^{-6}$ we obtain

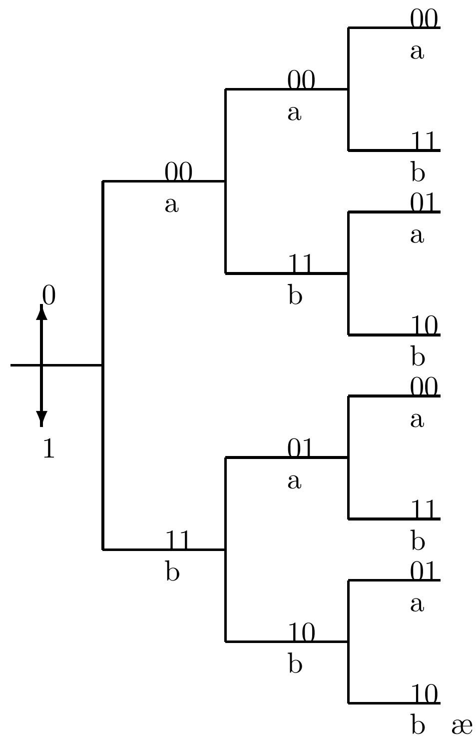
$$P_b \leq \frac{D^3}{(1 - (D + D^3))^2}\Big|_{D=\sqrt{4p(1-p)}} = 8.0321 \times 10^{-9}$$

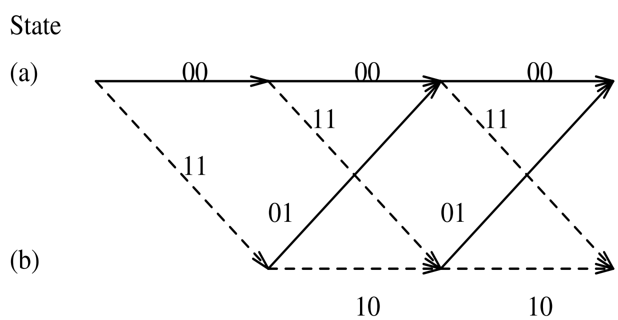
Problem 8.29 :

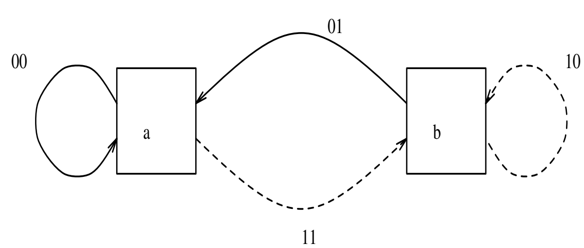
(a)

$$g_1 = [10],\ g_2 = [11], \quad \text{states : (a)} = [0],\ \text{(b)} = [1]$$

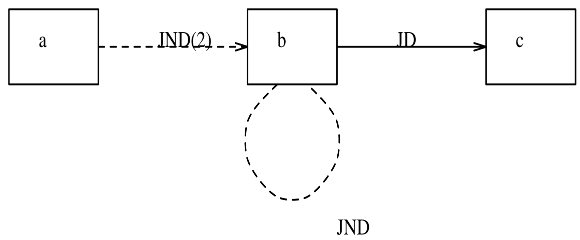
The tree diagram, trellis diagram and state diagram are given in the following figures :







(b) Redrawing the state diagram :



$$X_b = JND^2X_a + JNDX_b \Rightarrow X_b = \frac{JND^2}{1-JND}X_a$$

$$X_c = JDX_b \Rightarrow \frac{X_c}{X_a} = T(D,N,J) = \frac{J^2ND^3}{1-JND} = J^2ND^3 + J^3N^2D^4 + ...$$

Hence :

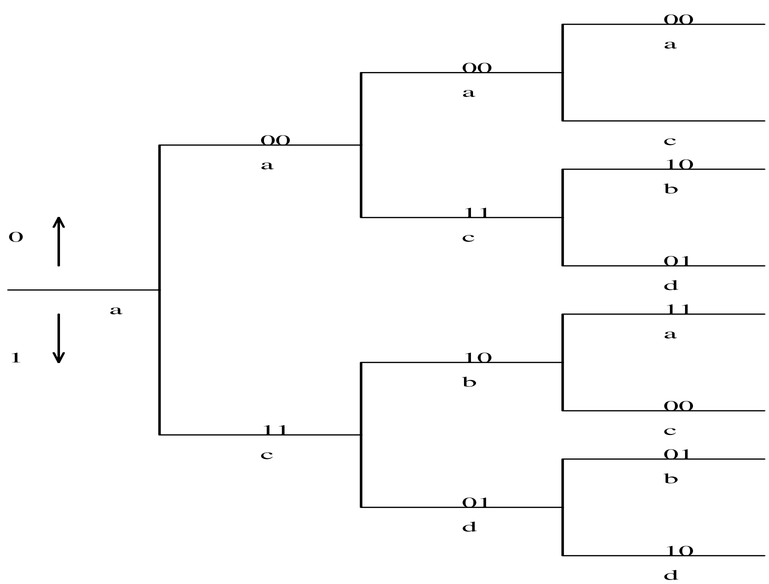
$$d_{\min} = 3$$

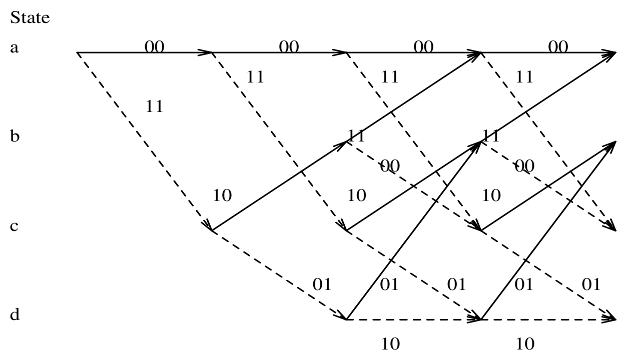
Problem 8.30 :

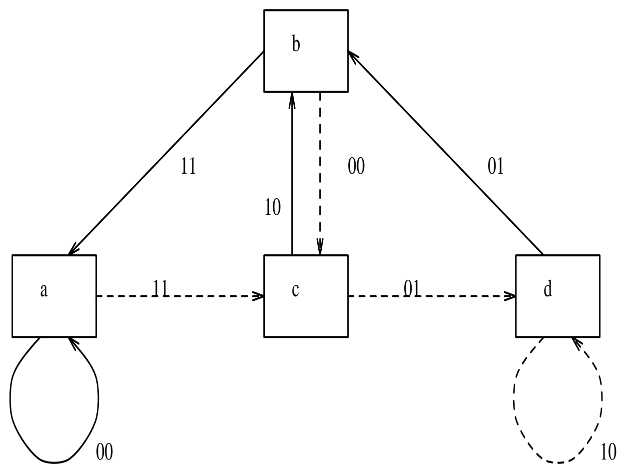
(a)

$$g_1 = [111],\ g_2 = [101], \quad \text{states} \ : \ (\text{a}) = [00],\ (\text{b}) = [01],\ (\text{c}) = [10],\ (\text{d}) = [11]$$

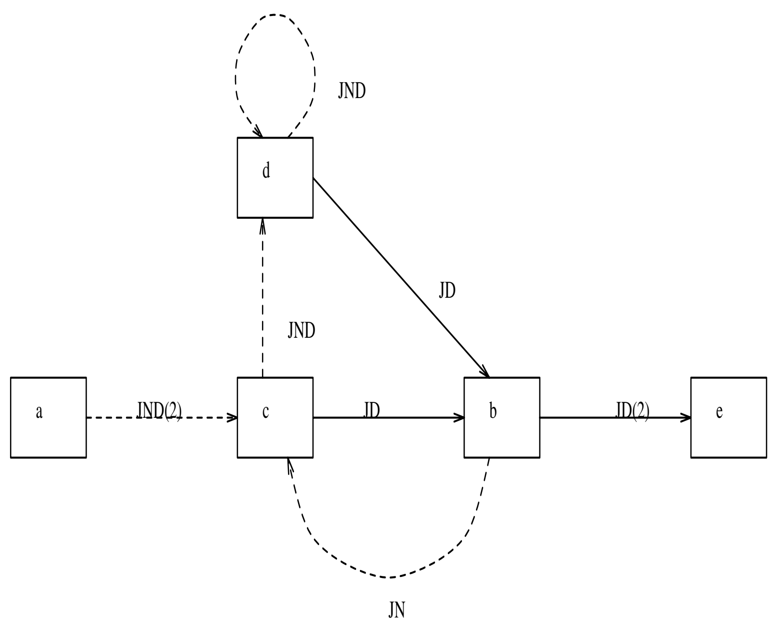
The tree diagram, trellis diagram and state diagram are given in the following figures :







(b) Redrawing the state diagram :



$$\left.\begin{cases} X_c = JND^2X_a + JNX_b \\ X_b = JDX_c + JDX_d \\ X_d = JNDX_d + JNDX_c = NX_b \end{cases}\right\} \Rightarrow X_b = \frac{J^2ND^3}{1-JND(1+J)}X_a$$

$$X_e = JD^2X_b \Rightarrow \frac{X_e}{X_a} = T(D,N,J) = \frac{J^3ND^5}{1-JND(1+J)} = J^3ND^5 + J^4N^2D^6(1+J) + ...$$

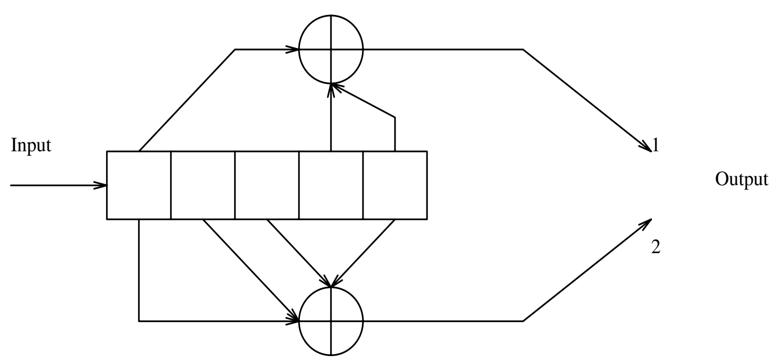
Hence :

$$d_{\min} = 5$$

Problem 8.31 :

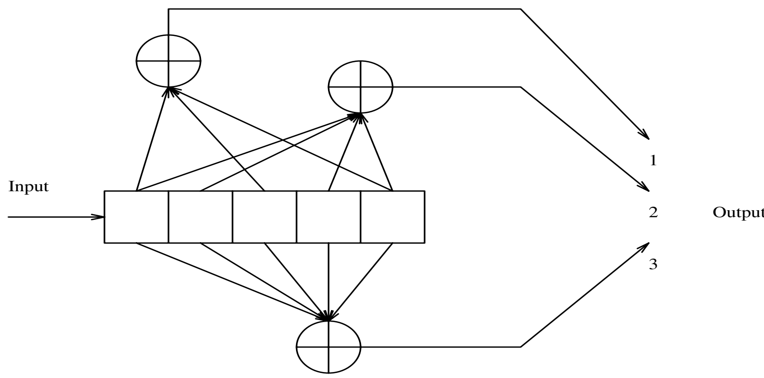
(a)

$$g_1 = [23] = [10011], \; g_2 = [35] = [11101]$$



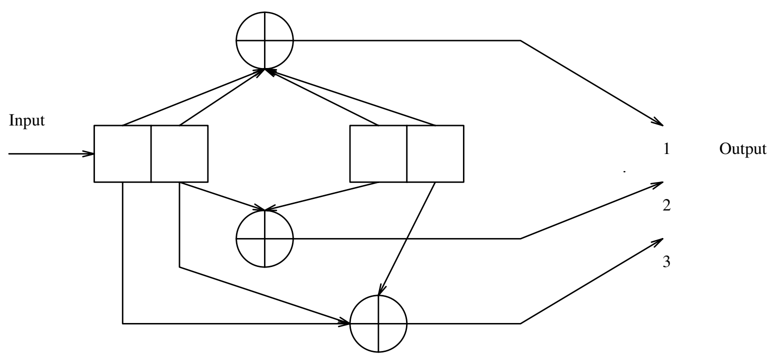
(b)

$$g_1 = [25] = [10101], \; g_2 = [33] = [11011], \; g_3 = [37] = [11111]$$



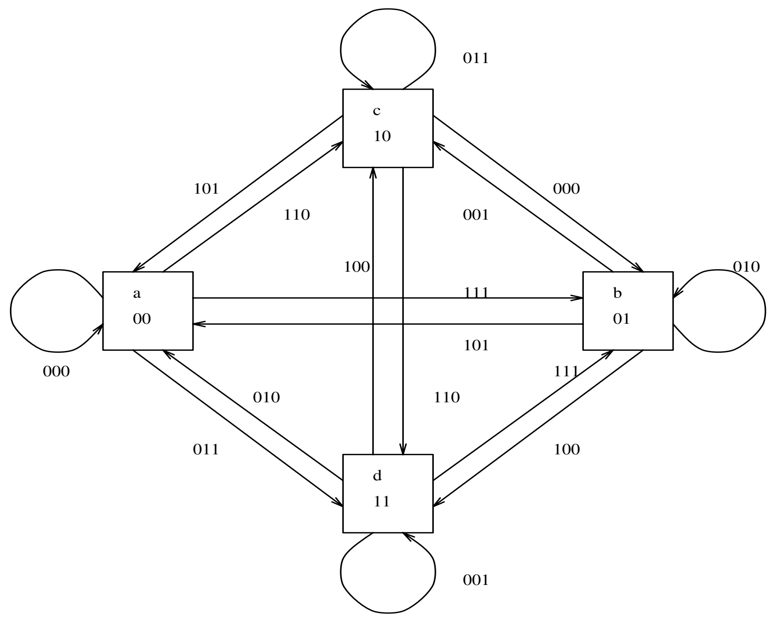
(c)

$$g_1 = [17] = [1111],\ g_2 = [06] = [0110],\ g_3 = [15] = [1101]$$



Problem 8.32 :

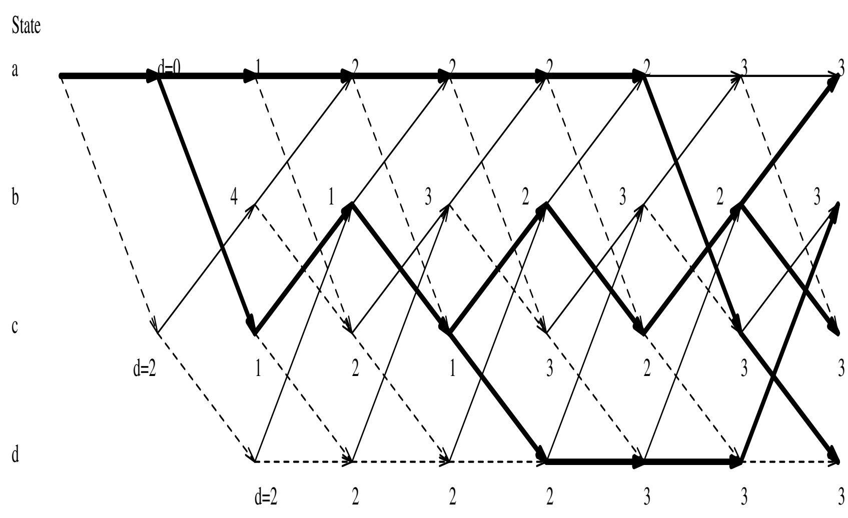
For the encoder of Probl. 8.31(c), the state diagram is as follows :



The 2-bit input that forces the transition from one state to another is the 2-bits that characterize the terminal state.

Problem 8.33 :

The encoder is shown in Probl. 8.30. The channel is binary symmetric and the metric for Viterbi decoding is the Hamming distance. The trellis and the surviving paths are illustrated in the following figure :



Problem 8.34 :

In Probl. 8.30 we found :

$$T(D,N,J) = \frac{J^3ND^5}{1-JND(1+J)}$$

Setting $J=1$:

$$T(D,N) = \frac{ND^5}{1-2ND} \Rightarrow \frac{dT(D,N)}{dN} = \frac{D^5}{(1-2ND)^2}$$

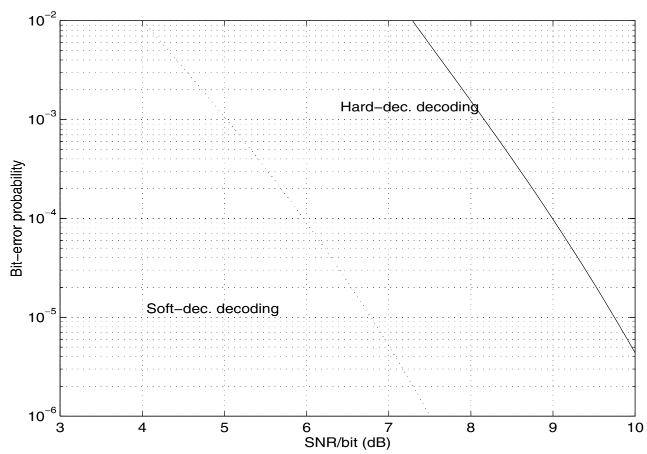
For soft-decision decoding the bit-error probability can be upper-bounded by :

$$P_{bs} \leq \frac{1}{2}\frac{dT(D,N)}{dN}|_{N=1,D=\exp(-\gamma_b R_c)} = \frac{1}{2}\frac{D^5}{(1-2ND)^2}|_{N=1,D=\exp(-\gamma_b/2)} = \frac{1}{2}\frac{\exp(-5\gamma_b/2)}{(1-\exp(-\gamma_b/2))^2}$$

For hard-decision decoding, the Chernoff bound is :

$$P_{bh} \leq \frac{dT(D,N)}{dN}|_{N=1,D=\sqrt{4p(1-p)}} = \frac{\left[\sqrt{4p(1-p)}\right]^{5/2}}{\left[1-2\sqrt{4p(1-p)}\right]^2}$$

where $p = Q\left(\sqrt{\gamma_b R_c}\right) = Q\left(\sqrt{\gamma_b/2}\right)$ (assuming binary PSK). A comparative plot of the bit-error probabilities is given in the following figure :

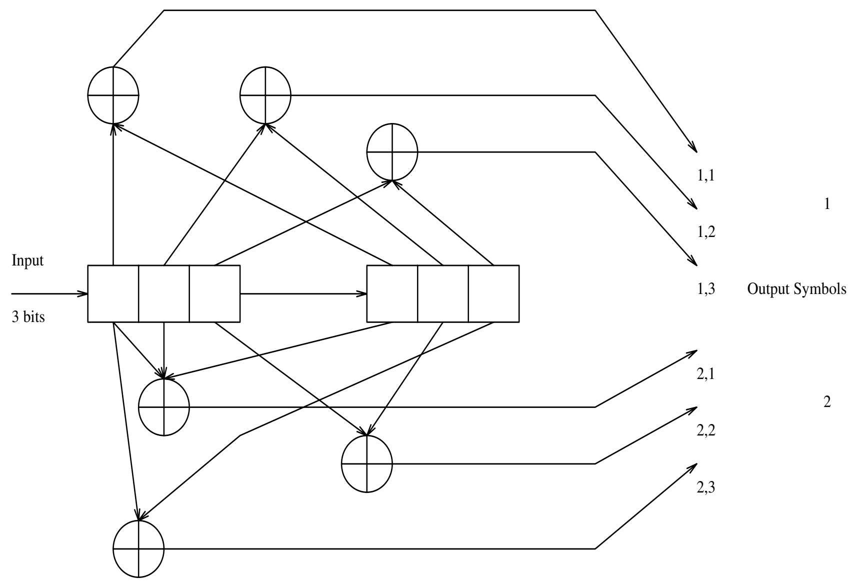


Problem 8.35 :

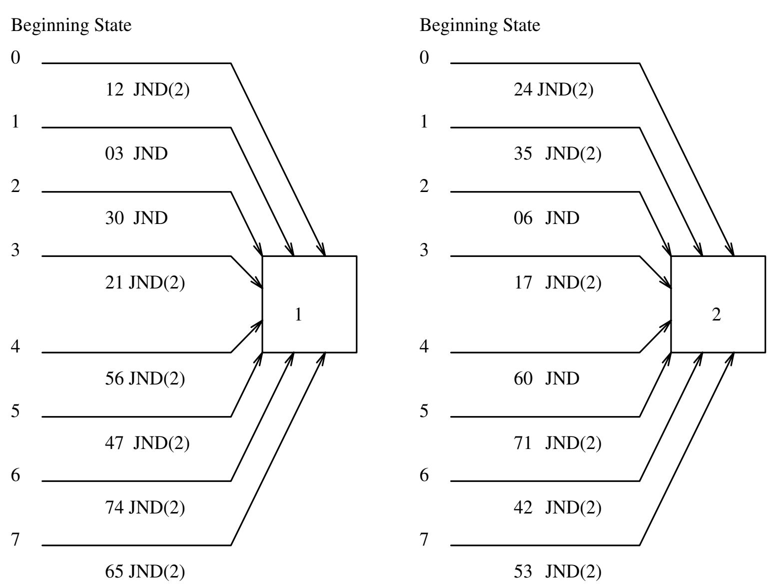
For the dual-3 (k=3), rate 1/2 code, we have from Table 8-2-36 :

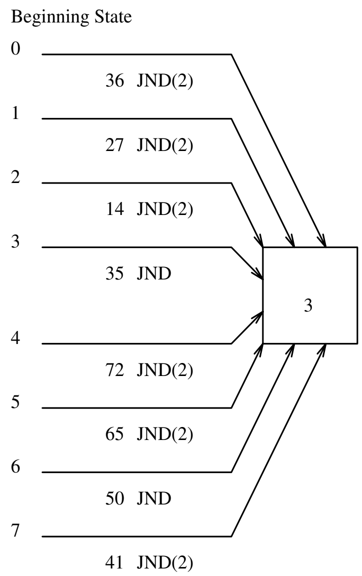
$$\begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix} = \begin{bmatrix} 100100 \\ 010010 \\ 001001 \end{bmatrix}, \qquad \begin{bmatrix} g_4 \\ g_5 \\ g_6 \end{bmatrix} = \begin{bmatrix} 110100 \\ 001010 \\ 100001 \end{bmatrix}$$

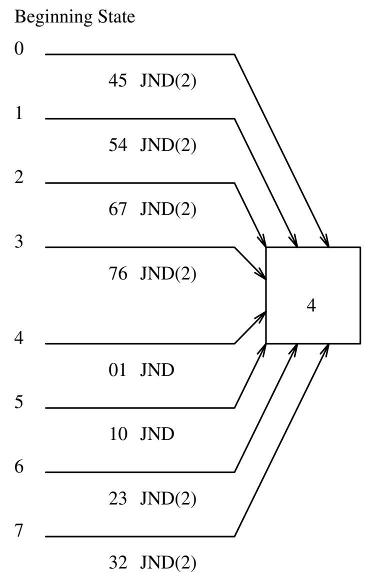
Hence, the encoder will be :

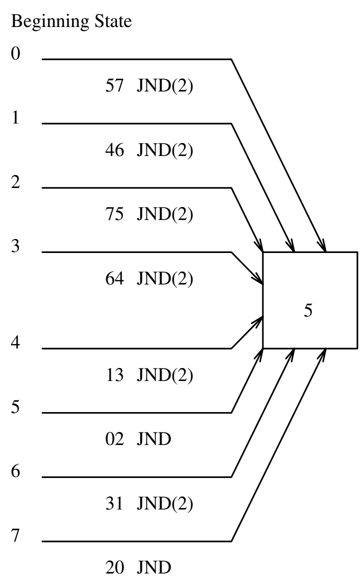


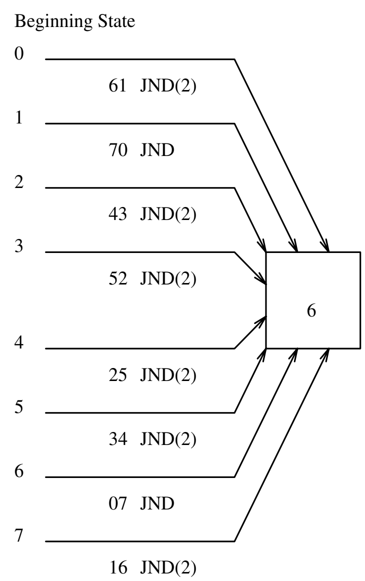
The state transitions are given in the following figures :

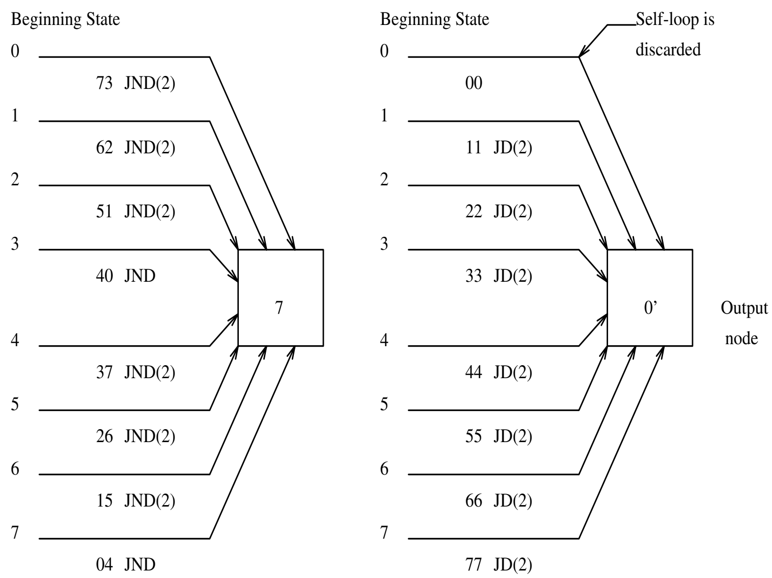



Beginning State
0
36 JND(2)
1
27 JND(2)
2
14 JND(2)
3
35 JND
3
4
72 JND(2)
5
65 JND(2)
6
50 JND
7
41 JND(2)


Beginning State
0
45 JND(2)
1
54 JND(2)
2
67 JND(2)
3
76 JND(2)
4
4
01 JND
5
10 JND
6
23 JND(2)
7
32 JND(2)


Beginning State
0
57 JND(2)
1
46 JND(2)
2
75 JND(2)
3
64 JND(2)
5
4
13 JND(2)
5
02 JND
6
31 JND(2)
7
20 JND


Beginning State
0
61 JND(2)
1
70 JND
2
43 JND(2)
3
52 JND(2)
6
4
25 JND(2)
5
34 JND(2)
6
07 JND
7
16 JND(2)



The states are : $(1) = 000,\ (2) = 001,\ (3) = 010,\ (4) = 011,\ (5) = 100,\ (6) = 101,\ (7) = 110,\ (8) = 111$. The state equations are :

$$\begin{aligned}
X_1 &= D^2NJ\left(X_0 + X_3 + X_4 + X_5 + X_6 + X_7\right) + DNJ\left(X_1 + X_2\right)\\
X_2 &= D^2NJ\left(X_0 + X_1 + X_3 + X_5 + X_6 + X_7\right) + DNJ\left(X_2 + X_4\right)\\
X_3 &= D^2NJ\left(X_0 + X_1 + X_2 + X_4 + X_5 + X_7\right) + DNJ\left(X_3 + X_6\right)\\
X_4 &= D^2NJ\left(X_0 + X_1 + X_2 + X_3 + X_6 + X_7\right) + DNJ\left(X_4 + X_5\right)\\
X_5 &= D^2NJ\left(X_0 + X_1 + X_2 + X_3 + X_4 + X_6\right) + DNJ\left(X_5 + X_7\right)\\
X_6 &= D^2NJ\left(X_0 + X_2 + X_3 + X_4 + X_5 + X_7\right) + DNJ\left(X_1 + X_6\right)\\
X_7 &= D^2NJ\left(X_0 + X_1 + X_2 + X_4 + X_5 + X_6\right) + DNJ\left(X_3 + X_7\right)\\
X_0' &= D^2J\left(X_1 + X_2 + X_3 + X_4 + X_5 + X_6 + X_7\right)
\end{aligned}$$

where, note that D, N correspond to symbols and not bits. If we add the first seven equations, we obtain :

$$\sum_{i=1}^{7} = 7D^2NJX_0 + 2DNJ\sum_{i=1}^{7} X_i + 5D^2NJ\sum_{i=1}^{7} X_i$$

Hence :

$$\sum_{i=1}^{7} X_i = \frac{7D^2NJ}{1 - 2DNJ - 5D^2NJ}$$

Substituting the result into the last equation we obtain :

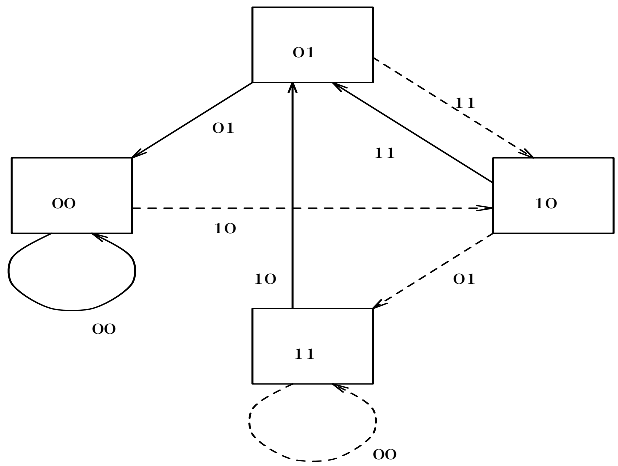
$$\frac{X_0'}{X_0} = T(D, N, J) = \frac{7D^4NJ^2}{1 - 2DNJ - 5D^2NJ} = \frac{7D^4NJ^2}{1 - DNJ(2 + 5D)}$$

which agrees with the result (8-2-37) in the book.

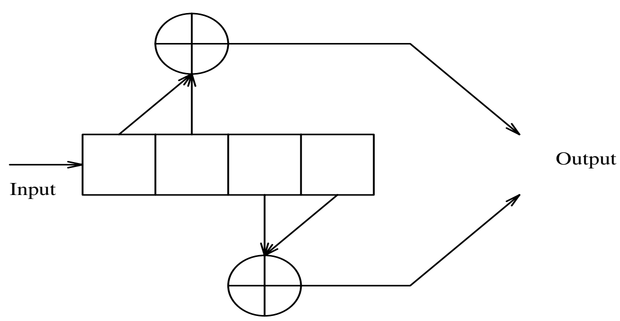
Problem 8.36 :

$$g_1 = [110],\ g_2 = [011], \quad \text{states} \ : \ (\text{a}) = [00],\ (\text{b}) = [01],\ (\text{c}) = [10],\ (\text{d}) = [11]$$

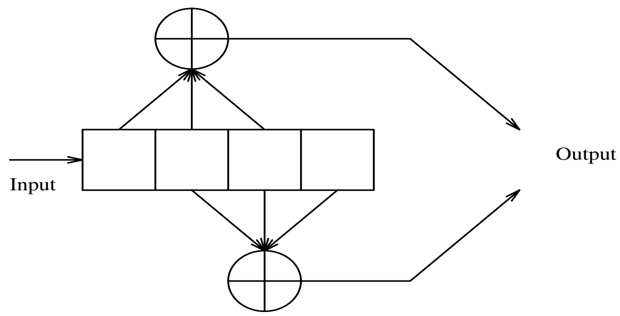
The state diagram is given in the following figure :



We note that this is a catastrophic code, since there is a zero-distance path from a non-zero state back to itself, and this path corresponds to input 1.
A simple example of an $K = 4$, rate $1/2$ encoder that exhibits error propagation is the following :



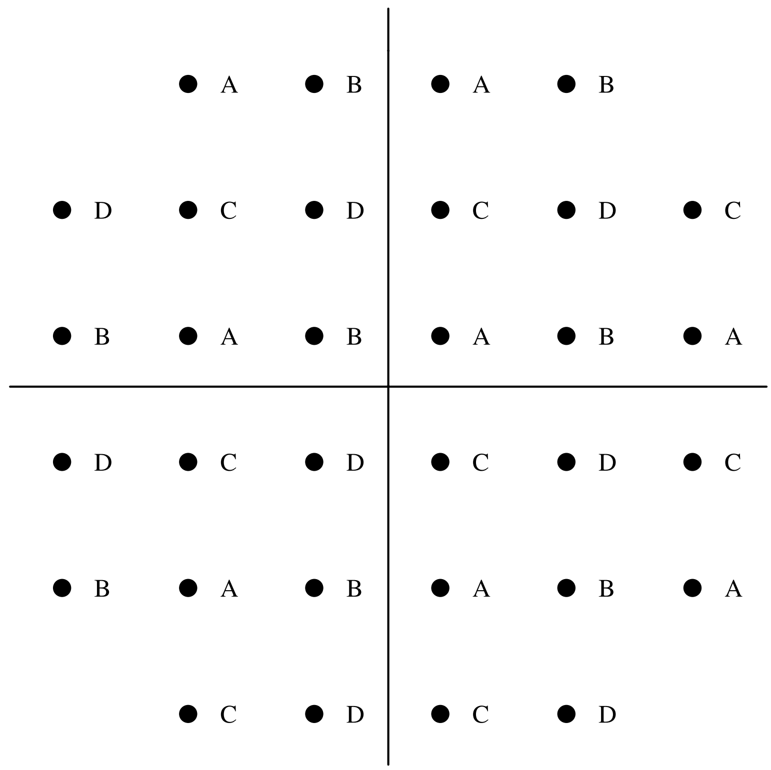
The state diagram for this code has a self-loop in the state 111 with input 1, and output 00.
A more subtle example of a catastrophic code is the following :



In this case there is a zero-distance path generated by the sequence 0110110110..., which encompasses the states 011,101, and 110. that is, if the encoder is in state 011 and the input is 1, the output is 00 and the new state is 101. If the next bit is 1, the output is again 00 and the new state is 110. Then if the next bit is a zero, the output is again 00 and the new state is 011, which is the same state that we started with. Hence, we have a closed path in the state diagram which yields an output that is identical to the output of the all-zero path, but which results from the input sequence 110110110...
For an alternative method for identifying rate 1/n catastrophic codes based on observation of the code generators, please refer to the paper by Massey and Sain (1968).

Problem 8.37 :

There are 4 subsets corresponding to the four possible outputs from the rate 1/2 convolutional encoder. Each subset has eight signal points, one for each of the 3-tuples from the uncoded bits. If we denote the sets as A,B,C,D, the set partitioning is as follows :



The minimum distance between adjacent points in the same subset is doubled.

Problem 8.38 :

(a) Let the decoding rule be that the first codeword is decoded when $\mathbf{y}_i$ is received if

$$p(\mathbf{y}_i|\mathbf{x}_1) > p(\mathbf{y}_i|\mathbf{x}_2)$$

The set of $\mathbf{y}_i$ that decode into $\mathbf{x}_1$ is

$$Y_1 = \{\mathbf{y}_i : p(\mathbf{y}_i|\mathbf{x}_1) > p(\mathbf{y}_i|\mathbf{x}_2)\}$$

The characteristic function of this set $\chi_1(\mathbf{y}_i)$ is by definition equal to 0 if $\mathbf{y}_i \notin Y_1$ and equal to 1 if $\mathbf{y}_i \in Y_1$. The characteristic function can be bounded as

$$1 - \chi_1(\mathbf{y}_i) \leq \left(\frac{p(\mathbf{y}_i|\mathbf{x}_2)}{p(\mathbf{y}_i|\mathbf{x}_1)}\right)^{\frac{1}{2}}$$

This inequality is true if $\chi(\mathbf{y}_i) = 1$ because the right side is nonnegative. It is also true if $\chi(\mathbf{y}_i) = 0$ because in this case $p(\mathbf{y}_i|\mathbf{x}_2) > p(\mathbf{y}_i|\mathbf{x}_1)$ and therefore,

$$1 \leq \frac{p(\mathbf{y}_i|\mathbf{x}_2)}{p(\mathbf{y}_i|\mathbf{x}_1)} \Longrightarrow 1 \leq \left(\frac{p(\mathbf{y}_i|\mathbf{x}_2)}{p(\mathbf{y}_i|\mathbf{x}_1)}\right)^{\frac{1}{2}}$$

Given that the first codeword is sent, then the probability of error is

$$
\begin{aligned}
P(\text{error}|\mathbf{x}_1) &= \sum_{\mathbf{y}_i \in Y - Y_1} p(\mathbf{y}_i|\mathbf{x}_1) = \sum_{\mathbf{y}_i \in Y} p(\mathbf{y}_i|\mathbf{x}_1)[1-\chi_1(\mathbf{y}_i)] \\
&\leq \sum_{\mathbf{y}_i \in Y} p(\mathbf{y}_i|\mathbf{x}_1) \left(\frac{p(\mathbf{y}_i|\mathbf{x}_2)}{p(\mathbf{y}_i|\mathbf{x}_1)}\right)^{\frac{1}{2}} = \sum_{\mathbf{y}_i \in Y} \sqrt{p(\mathbf{y}_i|\mathbf{x}_1)p(\mathbf{y}_i|\mathbf{x}_2)} \\
&= \sum_{i=1}^{2^n} \sqrt{p(\mathbf{y}_i|\mathbf{x}_1)p(\mathbf{y}_i|\mathbf{x}_2)}
\end{aligned}
$$

where Y denotes the set of all possible sequences $\mathbf{y}_i$. Since, each element of the vector $\mathbf{y}_i$ can take two values, the cardinality of the set Y is 2^n.

(b) Using the results of the previous part we have

$$
\begin{aligned}
P(\text{error}) &\leq \sum_{i=1}^{2^n} \sqrt{p(\mathbf{y}_i|\mathbf{x}_1)p(\mathbf{y}_i|\mathbf{x}_2)} = \sum_{i=1}^{2^n} p(\mathbf{y}_i) \sqrt{\frac{p(\mathbf{y}_i|\mathbf{x}_1)}{p(\mathbf{y}_i)}} \sqrt{\frac{p(\mathbf{y}_i|\mathbf{x}_2)}{p(\mathbf{y}_i)}} \\
&= \sum_{i=1}^{2^n} p(\mathbf{y}_i) \sqrt{\frac{p(\mathbf{x}_1|\mathbf{y}_i)}{p(\mathbf{x}_1)}} \sqrt{\frac{p(\mathbf{x}_2|\mathbf{y}_i)}{p(\mathbf{x}_2)}} = \sum_{i=1}^{2^n} 2p(\mathbf{y}_i)\sqrt{p(\mathbf{x}_1|\mathbf{y}_i)p(\mathbf{x}_2|\mathbf{y}_i)}
\end{aligned}
$$

However, given the vector $\mathbf{y}_i$, the probability of error depends only on those values that $\mathbf{x}_1$ and $\mathbf{x}_2$ are different. In other words, if $x_{1,k} = x_{2,k}$, then no matter what value is the k^{th} element of $\mathbf{y}_i$, it will not produce an error. Thus, if by d we denote the Hamming distance between $\mathbf{x}_1$ and $\mathbf{x}_2$, then

$$p(\mathbf{x}_1|\mathbf{y}_i)p(\mathbf{x}_2|\mathbf{y}_i) = p^d(1-p)^d$$

and since $p(\mathbf{y}_i) = \frac{1}{2^n}$, we obtain

$$P(\text{error}) = P(d) = 2p^{\frac{d}{2}}(1-p)^{\frac{d}{2}} = [4p(1-p)]^{\frac{d}{2}}$$

Problem 8.39 :

Over P frames, the number of information bits that are being encoded is

$$k_P = P\sum_{j=1}^{J} N_J$$

The number of bits that are being transmitted is determined as follows: For a particular group of bits j, $j = 1, ..., J$, we may delete, with the corresponding puncturing matrix, x_j out of nP bits, on the average, where x may take the values $x = 0, 1, ...(n-1)P - 1$. Remembering that each frame contains N_j bits of the particular group, we arrive at the total average number of bits for each group

$$n(j) = N_j(nP - x_j) \Rightarrow n(j) = N_j(P + M_j), \quad M_j = 1, 2, ..., (n-1)P$$

In the last group $j = J$ we should also add the $K - 1$ overhead information bits, that will add up another $(K-1)(P + M_J)$ transmitted bits to the total average number of bits for the J^{th} group.
Hence, the total number of bits transmitted over P frames be

$$n_P = (K-1)(P + M_J) + \sum_{j=1} JN_j(P + M_j)$$

and the average effective rate of this scheme will be

$$R_{av} = \frac{k_P}{n_P} = \frac{\sum_{j=1}^{J} N_J P}{\sum_{j=1} JN_j(P + M_j) + (K-1)(P + M_J)}$$

Problem 8.39 :

Over P frames, the number of information bits that are being encoded is

$$k_P = P \sum_{j=1}^{J} N_J$$

The number of bits that are being transmitted is determined as follows: For a particular group of bits j, $j = 1, ..., J$, we may delete, with the corresponding puncturing matrix, x_j out of nP bits, on the average, where x may take the values $x = 0, 1, ...(n-1)P - 1$. Remembering that each frame contains N_j bits of the particular group, we arrive at the total average number of bits for each group

$$n(j) = N_j(nP - x_j) \Rightarrow n(j) = N_j(P + M_j), \quad M_j = 1, 2, ..., (n-1)P$$

In the last group $j = J$ we should also add the $K - 1$ overhead information bits, that will add up another $(K-1)(P + M_J)$ transmitted bits to the total average number of bits for the J^{th} group.
Hence, the total number of bits transmitted over P frames be

$$n_P = (K-1)(P + M_J) + \sum_{j=1} JN_j(P + M_j)$$

and the average effective rate of this scheme will be

$$R_{av} = \frac{k_P}{n_P} = \frac{\sum_{j=1}^{J} N_J P}{\sum_{j=1} JN_j(P + M_j) + (K-1)(P + M_J)}$$