Chapter 3
BCH Codes and RS Codes

3.1 Binary BCH Codes

- **BCH codes are a large class of multiple random error-correcting codes , first discovered by A. Hocquenghem in 1959 and independently by R.C. Bose and D. K. Ray-Chaudhuri in 1960.**
- **The first decoding algorithm for binary BCH codes was devised by Peterson in 1960. Since then Peterson's algorithm has been refined by Berlekamp, Massey , Chien , Forney and many others.**

- **For any integer $m \geqq 3$ and $t \leqq 2^{m-1}$, there exists a primitive BCH code with the following parameters :**

  $n = 2^m - 1$, $\quad n - k \leqq mt$

  $d_{min} \geqq 2t + 1$ **(3.1)**

  This code can correct t oe fewer random errors over a span of $2^m - 1$ bit positions .

3.2 Generation of BCH Codes

- **The generator polynomial of a *t*-error-correcting BCH codes of length $2^m - 1$ is given by**

  $g(x) = \mathrm{LCM}\ \{\psi_1(x), \psi_3(x), \ldots, \psi_{2t-1}(x)\}$ **(3.2)**

  where $\psi_i(x)$ is the minimum polynomial of the primitive element in GF(2^m) .

  Since the degree of $g(x)$ is mt or less , the number of parity-check bits , n-k , of the code is at most mt .

Example : *$m = 4$, $t = 3$*

Then $n = 2^4 - 1 = 15$, $n-k = m\,t = 12$ thus $k = 3$

The code is a (15 ,3) code.

The primitive polynomial $p(x) = 1 + x + x^4$

$\psi_1(x) = 1 + x + x^4$

$\psi_3(x) = 1 + x + x^2 + x^3 + x^4$

$\psi_5(x) = 1 + x + x^2$

Thus $g(x) = \text{LCM}\{\psi_1(x), \psi_3(x), \psi_5(x)\}$

$= \psi_1(x)\,\psi_3(x)\,\psi_5(x)$

$= 1 + x + x^2 + x^4 + x^5 + x^8 + x^{10}$

Table 3.1 **Minimal polynomials of the elements in GF(2^4)**

Elements	Conjugates	Minimal polynomials
α	$\alpha^2, \alpha^4, \alpha^8$	$m_1(x) = 1 + x + x^4$
α^2	$\alpha^4, \alpha^8, \alpha^{16} = \alpha$	$m_2(x) = 1 + x + x^4$
α^3	$\alpha^6, \alpha^{12}, \alpha^{24} = \alpha^9$	$m_3(x) = 1 + x + x^2 + x^3 + x^4$
α^4	$\alpha^8, \alpha^{16} = \alpha, \alpha^{32} = \alpha^2$	$m_4(x) = 1 + x + x^4$
α^5	α^{10}	$m_5(x) = 1 + x + x^2$
α^6	$\alpha^{12}, \alpha^{24} = \alpha^9, \alpha^{48} = \alpha^3$	$m_6(x) = 1 + x + x^2 + x^3 + x^4$
α^7	$\alpha^{14}, \alpha^{28} = \alpha^{13}, \alpha^{56} = \alpha^{11}$	$m_7(x) = 1 + x^3 + x^4$
α^8	$\alpha^{16} = \alpha, \alpha^{32} = \alpha^2, \alpha^{64} = \alpha^4$	$m_8(x) = 1 + x + x^4$
α^9	$\alpha^{18} = \alpha^3, \alpha^{36} = \alpha^6, \alpha^{72} = \alpha^{12}$	$m_9(x) = 1 + x + x^2 + x^3 + x^4$
α^{10}	$\alpha^{20} = \alpha^5$	$m_{10}(x) = 1 + x + x^2$
α^{11}	$\alpha^{22} = \alpha^7, \alpha^{44} = \alpha^{14}, \alpha^{88} = \alpha^{13}$	$m_{11}(x) = 1 + x^3 + x^4$
α^{12}	$\alpha^{24} = \alpha^9, \alpha^{48} = \alpha^3, \alpha^{96} = \alpha^6$	$m_{12}(x) = 1 + x + x^2 + x^3 + x^4$
α^{13}	$\alpha^{26} = \alpha^{11}, \alpha^{52} = \alpha^7, \alpha^{104} = \alpha^{14}$	$m_{13}(x) = 1 + x^3 + x^4$
α^{14}	$\alpha^{28} = \alpha^{13}, \alpha^{56} = \alpha^{11}, \alpha^{112} = \alpha^7$	$m_{14}(x) = 1 + x^3 + x^4$

Table 3.1 **Minimal polynomials of the elements in GF(2^4)**

Elements	Conjugates	Minimal polynomials
α	$\alpha^2, \alpha^4, \alpha^8$	$m_1(x) = 1 + x + x^4$
α^2	$\alpha^4, \alpha^8, \alpha^{16} = \alpha$	$m_2(x) = 1 + x + x^4$
α^3	$\alpha^6, \alpha^{12}, \alpha^{24} = \alpha^9$	$m_3(x) = 1 + x + x^2 + x^3 + x^4$
α^4	$\alpha^8, \alpha^{16} = \alpha, \alpha^{32} = \alpha^2$	$m_4(x) = 1 + x + x^4$
α^5	α^{10}	$m_5(x) = 1 + x + x^2$
α^6	$\alpha^{12}, \alpha^{24} = \alpha^9, \alpha^{48} = \alpha^3$	$m_6(x) = 1 + x + x^2 + x^3 + x^4$
α^7	$\alpha^{14}, \alpha^{28} = \alpha^{13}, \alpha^{56} = \alpha^{11}$	$m_7(x) = 1 + x^3 + x^4$
α^8	$\alpha^{16} = \alpha, \alpha^{32} = \alpha^2, \alpha^{64} = \alpha^4$	$m_8(x) = 1 + x + x^4$
α^9	$\alpha^{18} = \alpha^3, \alpha^{36} = \alpha^6, \alpha^{72} = \alpha^{12}$	$m_9(x) = 1 + x + x^2 + x^3 + x^4$
α^{10}	$\alpha^{20} = \alpha^5$	$m_{10}(x) = 1 + x + x^2$
α^{11}	$\alpha^{22} = \alpha^7, \alpha^{44} = \alpha^{14}, \alpha^{88} = \alpha^{13}$	$m_{11}(x) = 1 + x^3 + x^4$
α^{12}	$\alpha^{24} = \alpha^9, \alpha^{48} = \alpha^3, \alpha^{96} = \alpha^6$	$m_{12}(x) = 1 + x + x^2 + x^3 + x^4$
α^{13}	$\alpha^{26} = \alpha^{11}, \alpha^{52} = \alpha^7, \alpha^{104} = \alpha^{14}$	$m_{13}(x) = 1 + x^3 + x^4$
α^{14}	$\alpha^{28} = \alpha^{13}, \alpha^{56} = \alpha^{11}, \alpha^{112} = \alpha^7$	$m_{14}(x) = 1 + x^3 + x^4$

3.3 Reed-Solomon Codes

3.3.1 RS Codes over GF(2^m)

- **The Reed-Solomon codes (RS codes) are nonbinary cyclic codes with code symbols from a Galois field. They were discovered in 1960 by I.Reed and G. Solomon at MIT .**
- **In the decades since their discovery , RS codes have enjoyed countless applications from compact disc and digital TV in living room to spacecraft and satellite in outer space.**
- **The RS codes with symbols from GF(2^m) are the most important codes in application.**
- **Let be a primitive symbol in GF(2^m) .**

  For any positive integer $t \leqq 2^{m-1}$, there exists a *t*-symbol –error- correcting RS code with symbols from GF(2^m) and the following parameters :

$$n = 2^m - 1$$

$$n - k = 2t$$

$$k = 2^m - 1 - 2t$$

$$d_{min} = 2t + 1 = n - k + 1 \tag{3.3}$$

Example :

$m = 8$, $t = 16$

$n = 255$, $k = n - 2t = 223$

$d_{min} = 32$

It is a (255, 223) RS code . The code is NASA standard code for satellite and space application .

3.3.2 Generation and Encoding of RS Codes

- **The generator polynomial of RS codes are given by**

$$g(x) = (x+\alpha)(x+\alpha^2)\ldots(x+\alpha^{2t})$$

$$= g_0 + g_1 x + g_2 x^2 + \ldots + g_{2t-1} x^{2t-1} + x^{2t} \qquad (3.4)$$

where $g_i \varepsilon \mathrm{GF}(2^m)$.

It is noted that $g(x)$ has $\alpha, \alpha^2, \ldots, \alpha^{2t}$ as roots.

- **The encoding of RS codes can be done as follows.**

Let $m(x) = m_0 + m_1 x + m_2 x^2 + \ldots + m_{k-1} x^{k-1}$ be the message polynomial.

Dividing $x^{2t} m(x)$ by $g(x)$, we have

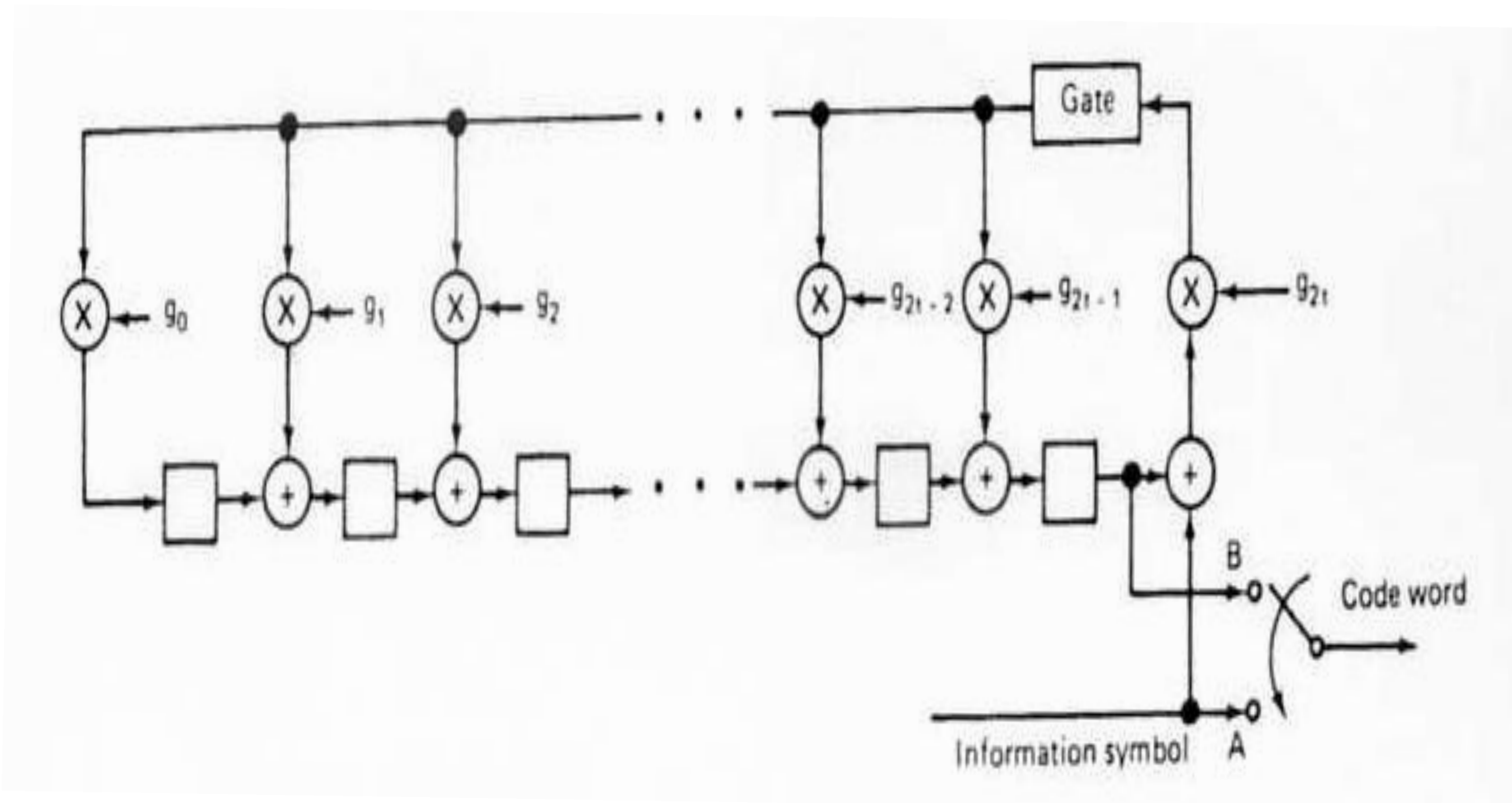
$$x^{2t} m(x) = a(x) g(x) + b(x) \qquad (3.5)$$

where $b(x) = b_0 + b_1 x + b_2 x^2 + \ldots + b_{2t-1} x^{2t-1}$ (3.6)

is the remainder.

- **The encoding circuit is shown in Fig. 3.1**

Fig.3.1 RS code encoder



3.3.3 RS Codes for Binary Data

- **Every symbol in GF(2^m) can be represented by a binary m-tuple , called m-bit byte..**
- **Suppose an (n,k) RS code is used for encoding mk bits of message sequence .This message sequence is first divided into k m-bit bytes. Each m-bit byte is regarded as a symbol in GF(2^m) .**

  The k-byte message is then encoded into n-byte codeword based on the RS encoding rule.

  By doing this , we actually expand a RS code with symbols from GF(2^m) into a binary (nm , km) linear , called a binary RS code .
- **Binary RS codes are very effective in correcting bursts of bit errors as long as no more than t bytes are affected.**

- **A popular RS code is the (255, 223) code over GF (2^8) . This code has a minimum distance of d_{min} = 255-223+1 =33 and is capable of correcting 16 symbol errors.**

- **Example # 1: RS(15,9) code**

Let $n = 2^4 - 1 = 15$, Construct a primitive three-error correcting RS code over the Galois field GF (2^4) using the primitive polynomial

$$p(x) = x^4 + x + 1.$$

The code generator has $\alpha, \alpha^2, \alpha^3, \alpha^4, \alpha^5, \alpha^6$ as roots.

The generator of the (15 ,9) code is

$$g(x) = (x+\alpha)(x+\alpha^2)(x+\alpha^3)(x+\alpha^4)(x+\alpha^5)(x+\alpha^6)$$

$$= \alpha^6 + \alpha^9 x + \alpha^6 x^2 + \alpha^4 x^3 + \alpha^{14} x^4 + \alpha^{10} x^5 + x^6$$

If the 4-bit data stream 5,2,1,6,8,3,10,15,4 are to be encoded. Find the systematically encoded code polynomial .

Sol. $t = 3$

$m(x) = 5 + 2x + x^2 + 6x^3 + 8x^4 + 3x^5 + 10x^6 + 15x^7 + 4x^8$

Using the vector –to –power conversion

$5 = 0101 \leftarrow\!\!\dashrightarrow \alpha^8$, $2 = 0010 \Leftrightarrow \alpha$, $1 = 0001 \Leftrightarrow 1$, ...

The message polynomial (expressed in power form) is the expressed as

$$m(x) = \alpha^8 + \alpha x + x^2 + \alpha^5 x^3 + \alpha^3 x^4 + \alpha^4 x^5 + \alpha^9 x^6 + \alpha^{12} x^7 + \alpha^2 x^8$$

Dividing $x^6 m(x)$ by $g(x)$ to obtain the remainder

$$b(x) = \alpha^8 + \alpha^2 x + \alpha^{14} x^2 + \alpha^3 x^3 + \alpha^5 x^4 + \alpha x^5$$

then we obtain

$$c(x) = \alpha^8 + \alpha^2 x + \alpha^{14} x^2 + \alpha^3 x^3 + \alpha^5 x^4 + \alpha x^5 + \alpha^8 x^6 + \alpha x^7 + x^8 + \alpha^5 x^9 + \alpha^3 x^{10} + \alpha^4 x^{11} + \alpha^9 x^{12} + \alpha^{12} x^{13} + \alpha^2 x^{14}$$

Example # : RS (255,223) RS code

$$p(x) = x^8 + x^4 + x^3 + x^2 + 1.$$

$$g(x) = \Pi_{j=1}^{32} (x - \alpha^j)$$

or $p(x) = x^8 + x^7 + x^2 + x + 1.$

$$g(x) = \Pi_{j=112}^{143} (x - (\alpha^{11})^j)$$

3.4 Decoding of BCH Codes and RS Codes

- **There are many algorithms which have been developed for decoding BCH codes. In general , the algebraic decoding binary BCH codes have the following steps :**

  (i) Computation of the syndrome

  (ii) Determination of an error- location polynomial whose roots provide an indication of the error- locations. The Berlekamp-Massey algorithm is an efficient algorithm for determining the error-locator polynomial .

  (iii) Finding the roots of the error-location polynomial . This is usually done using the Chien search , which is an exhaustive search over all the elements in the finite field.

3.4.1 Decoding of RS Codes

- **Decoding of a RS code is similar to the decoding of a BCH code except an additional step is needed.**

  The additional step is evaluating the error vales .

- **The Berlekamp-Massey algorithm is also an efficient algorithm for determining the error-locator polynomial for decoding RS codes.**

  A typical approach to find the error values is using Forney's Algorithm developed by J.D. Fornry in 1965.

- **In 1965, E. Berlekamp presented an extremely efficient algorithm for both BCH and RS codes.**

  Berlekamp's algorithm allowed for the first time the possibility of a quick and efficient decoding of dozens of symbol errors in some powerful RS codes. The algorithm was modified by J.L. Massey in 1969.

Chien,R.T.,” Cyclic Decoding Procedure for the Bose - Chaudhuri- Hocquenghem Codes ,” IEEE Trans. Inf. Theory , vol. IT-10, pp.357-363 ,October 1964

Forney , G.D., ‘ On Decoding BCH Codes ,” IEEE Trans. Inf. Theory , vol. IT-11, pp.549-557, October 1965,

Berlekamp , E.R., “On Decoding Bose - Chaudhuri- Hocquenghem Codes , “ IEEE Trans. Inf. Theory , vol. IT-11, pp.577-579 ,October 1965

Berlrkamp , E.R. Algebraic Coding Theory , McGraw0Hill, 1968

Massey, J.L.,” Shift Register Synthesis and BCH Decoding , “ IEEE Trans. Inf. Theory , vol. IT-15, pp. 122-127 Jan. 1969.

3.4.2 Computation of the Syndrome

- **Consider a code with codeword polynomial c(x) and generator polynomial g(x).**

Since $g(\alpha) = g(\alpha^2) = \ldots = g(\alpha^{2t}) = 0$

we have $c(\alpha) = c(\alpha^2) = \ldots = c(\alpha^{2t}) = 0$

If the received polynomial r(x) is expressed as

$$r(x) = c(x) + e(x) \qquad (3.7)$$

then the syndrome $S = (S_1, S_2, \ldots, S_{2t})$ **can be obtained by**

$$S_j = r(\alpha^j) = c(\alpha^j) + e(\alpha^j)$$
$$= e(\alpha^j) \quad j = 1, 2, \ldots, 2t \qquad (3.8)$$

This gives a relationship between the syndrome and the error pattern .

3.4.3 Syndrome and Error- Location Polynomial

- **Suppose $e(x)$ has ν errors , $\nu \leqq t$, at the locations specified by**

  $\mathbf{x}^{j_1}$, $\mathbf{x}^{j_2}$, …, $\mathbf{x}^{j_\nu}$.

  i.e. $e(x) = \mathbf{x}^{j_1} + \mathbf{x}^{j_2} + \ldots + \mathbf{x}^{j_\nu}$ **(3.9)**

  where $0 \leqq j_1 < j_2 < \ldots < j_\nu$

  From equations (3.8) & (3.9), we have the following relation between syndrome components and error location:

$$S_1 = e(\alpha) = \alpha^{j_1} + \alpha^{j_2} + \ldots + \alpha^{j_\nu}$$

$$S_2 = e(\alpha^2) = (\alpha^{j_1})^2 + (\alpha^{j_2})^2 + \ldots + (\alpha^{j_\nu})^2$$

$$\vdots$$

$$S_{2t} = e(\alpha^{2t}) = (\alpha^{j_1})^{2t} + (\alpha^{j_2})^{2t} + \ldots + (\alpha^{j_\nu})^{2t} \qquad (3.10)$$

If we can solve the $2t$ equations, we can determine

α^{j_1} , α^{j_2} , … , α^{j_ν}

- **The unknown parameter $\beta_\kappa = \alpha^{j_\kappa}$ for $\kappa = 1,2,\ldots, \nu$ are called the " error location numbers ".**
- **When , α^{j_κ} , $1 \leqq \kappa \leqq \nu$, are found, the powers , j_κ give us the error locations in $e(x)$.**

These $2t$ equations of (3.10) are known as power-sum symmetric function.

- **Eq.(3.8) can be written as**

$$
\begin{aligned}
S_1 &= \beta_1 + \beta_2 + \ldots + \beta_\nu \\
S_2 &= \beta_1^2 + \beta_2^2 + \ldots + \beta_\nu^2 \\
&\vdots \\
S_{2t} &= \beta_1^{2t} + \beta_2^{2t} + \ldots + \beta_\nu^{2t} \qquad (3.11)
\end{aligned}
$$

- **Suppose that $v \leqq t$ errors actually occur .Define the error-location polynomial σ(x) as**

$$\sigma(x) = (1 + \beta_1 x)(1 + \beta_2 x) \ldots (1 + \beta_v x)$$

$$= \sigma_0 + \sigma_1 x + \sigma_2 x^2 + \ldots + \sigma_v x^v \qquad (3.12)$$

σ(x) has $\beta_1^{-1}, \beta_2^{-1}, \ldots, \beta_v^{-1}$ as roots and $\sigma_0 = 1$

Note that $\beta_\kappa = \alpha^{j_\kappa}$.

If we can determine σ(x) from the syndrome $S = \{S_1, S_2, \ldots, S_{2t}\}$, then the roots of σ(x) give us the error-location numbers β_κ.

- **An efficient procedure, known as Chien search , to find these roots , and hence the error-locations, was given by R.T. Chien in 1964.**

- **Coefficients of the location polynomial Eq. (3.12) can be expressed as in the following manner :**

$$\sigma_0 = 1$$

$$\sigma_1 = \beta_1 + \beta_2 + \ldots + \beta_v$$

$$\sigma_2 = \beta_1 \beta_2 + \beta_2\beta_3 + \ldots + \beta_{v-1} \beta_v$$

$$.$$

$$\sigma_v = \beta_1 \beta_2 \beta_3 \ldots \beta_{v-1} \beta_v$$

This set of equations is known as the elementary symmetric functions and is related to the system of equations (3.11) as follows .

$$S_1 + \sigma_1 = 0$$

$$S_2 + \sigma_1 S_1 = 0$$

$$S_3 + \sigma_1 S_2 + \sigma_2 S_1 = 0$$

$$.$$

$$S_{2t} + \sigma_1 S_{2t-1} + \ldots + \sigma_{v-1} S_{2t-v+1} + \sigma_v S_{2t-v} = 0 \qquad (3.13)$$

These equations are called generalized Newton identities .

Special cases : Decoding BCH Codes with Small t

The σ_i can be solved directly as follows.

For $t = 1$, $\sigma_1 = S_1$

For $t = 2$, $\sigma_1 = S_1$ $\sigma_2 = (S_3 + S_1^3) / S_1$

For $t = 3$, $\sigma_1 = S_1$ $\sigma_2 = (S_1^2 S_3 + S_5) / (S_3 + S_1^3)$

$\sigma_3 = (S_1^3 + S_3) + S_1 \sigma_2)$

For $t = 4$, $\sigma_1 = S_1$

$\sigma_2 = \{ S_1 (S_1^7 + S_7) + S_3 (S_5 + S_1^5) \} / \{S_3(S_1^3 + S_3) + S_1 (S_5 + S_1^5) \}$

$\sigma_3 = (S_1^3 + S_3) + S_1 \sigma_2)$

$\sigma_4 = \{ (S_1^2 S_3 + S_5) + (S_1^3 + S_3) \sigma_2 \} / S_1$

3.4.4 Berlekamp-Massey Iterative Algorithm for Finding the Error-Location Polynomial

- **The B-M algorithm basically consists of finding the coefficient of the error-location polynomial, $\sigma_1, \sigma_2, \ldots, \sigma_v$.**
- **The algorithm proceeds as follows . The first step is to determine a minimum -degree polynomial $\sigma^{(1)}(x)$ that satisfies the first Newton identity described in Eq. (3.13) .**
- **Then the second Newton identity is tested.**

If the polynomial $\sigma^{(1)}(x)$ satisfies the second Newton identity in Eq. (3.13) ,

then $\sigma^{(2)}(x) = \sigma^{(1)}(x)$.

Otherwise the decoding procedure adds a correction term to $\sigma^{(1)}(x)$ in order to form the polynomial $\sigma^{(2)}(x)$, which is able to satisfies the first two Newton identities.

- **This procedure is subsequently applied to find $\sigma^{(3)}(x)$, and the following polynomials, until determination of the polynomial $\sigma^{(2t)}(x)$ is complete.**

- **This algorithm can be implemented in iterative form. Let the minimum-degree polynomial obtained in the μ- *th* iteration , denoted by $\sigma^{(\mu)}(x)$, be of the form**

$$\sigma^{(\mu)}(x) = 1 + \sigma_1^{(\mu)} x + \sigma_2^{(\mu)} x^2 + \ldots + \sigma_{L_\mu}^{(\mu)} x^{L_\mu} \qquad (3.14)$$

where L_μ is the degree of the polynomial $\sigma^{(\mu)}(x)$.

- **This minimum-degree polynomial $\sigma^{(\mu)}(x)$ satisfies the first *i* Newton identities in Eq.(3.13)**

- **To find $\sigma^{(\mu+1)}(x)$, we first check whether the coefficients of $\sigma^{(\mu)}(x)$ satisfy the next generalized Newton identity ; that is,**

$$S_{\mu+1} + \sum_{k=1}^{L_\mu} \sigma_k^{(\mu)} S_{\mu+1-k} = 0 \quad ? \qquad (3.15)$$

If yes , $\sigma^{(\mu+1)}(x) = \sigma^{(\mu)}(x)$ is the minimum-degree polynomial whose coefficients satisfy the generalized Newton identities.

If not ,a correction term is added to $\sigma^{(i)}(x)$ to obtain $\sigma^{(i+1)}(x)$.

- **To test the equality of Eq. (3.15) ,we calculate the discrepancy**

$$d_\mu = S_{\mu+1} + \sigma_1^{(\mu)} S_\mu + \sigma_{2,}^{(\mu)} S_{\mu-1} + \dots + \sigma_{L_\mu}^{(\mu)} S_{\mu+1-L_\mu} \qquad (3.16)$$

If $d_\mu = 0$, we set $\sigma^{(\mu+1)}(x) = \sigma^{(\mu)}(x)$

If $d_\mu \neq 0$, , we need to add a correction term to $\sigma^{(\mu)}(x)$ to obtain $\sigma^{(\mu+1)}(x)$

In the calculation of the correction term , the algorithm resorts to a previous step ρ such that $d_\rho \neq 0$ and $(\rho - L_\rho)$ is a maximum , where L_ρ is the degree of of $\sigma^{(\rho)}(x)$.

Massey demonstrated that ,when $d_\rho \neq 0$, one must have

$$L_{\mu+1} = max\,[L_\mu\, ,\; L_\rho + \mu - \rho\,]$$

Then

$$\sigma^{(\mu+1)}(x) = \sigma^{(\mu)}(x) + d_\mu\, d_\rho^{-1}\; x^{(\mu-\rho)}\; \sigma^{(\rho)}(x) \qquad (3.17)$$

The B-M algorithm can be implemented in the form of a table, as shown below.

Note that

$$\sigma^{(-1)}(x) = \sigma^{(0)}(x) = 1\;, \quad d_1 = S_1$$

$$\sigma^{(1)}(x) = 1 + S_1\, x$$

μ	$\sigma^{(\mu)}(x)$	d_μ	L_μ	$\mu - L_\mu$	ρ
-1	1	1	0	-1	
0	1	S_1	0	0	
1	$1 + S_1 x$	.	.	.	
2					
.					
.					
.					
2t					

Berlrkamp-Massey Algorithm

1. Set the initial conditions before taking the iterative step.

$$\sigma^{(-1)}(x) = 1 \qquad L_{-1} = 0 \qquad d_{-1} = 1$$

$$\sigma^{(0)}(x) = 1 \qquad L_0 = 0 \qquad d_0 = s_1$$

2. If $d_\mu = 0$, then set $\sigma^{(\mu+1)}(x) = \sigma^{(\mu)}(x)$ and $L_{\mu+1} = L_\mu$

3. If $d_\mu \neq 0$, , then find $\sigma^{(\rho)}(x)$ prior to $\sigma^{(\mu)}(x)$ such that $d_\rho \neq 0$, $\rho \leqq \mu$, and the number $(\rho - L_\mu)$ has the largest number. Then

$$\sigma^{(\mu+1)}(x) = \sigma^{(\mu)}(x) + d_\mu d_\rho^{-1} x^{(\mu-\rho)} \sigma^{(\rho)}(x)$$

$$L_{\mu+1} = max\,[\,L_\mu\,,\; L_\rho + \mu - \rho\,]$$

and $d_{\mu+1} = S_{\mu+2} + \sigma_1^{(\mu+1)} S_{\mu+1} + \sigma_2^{(\mu+1)} S_{\mu-1} + \ldots +$

$$\sigma_{L(\mu+1)}^{(\mu+1)} S_{\mu+2-L(\mu+1)}$$

where $\sigma_i^{(\mu+1)}$, $1 \leqq i \leqq L_{\mu+1}$, are the coefficients of $\sigma^{(\mu+1)}(x)$.

Example:

- **For the (15, 9) RS code over GF(2^4)**

  Use the Berlekamp-Massey algorithm to find the error-locator polynomial. The received polynomial is

  $r(x) = x^8 + \alpha^{11} x^7 + \alpha^8 x^5 + \alpha^{10} x^4 + \alpha^4 x^3 + \alpha^3 x^2 + \alpha^8 x + \alpha^{12}$

Solution: $\quad n - k = 6$

$$m = 4, \quad \alpha^{15} = 1$$

$$r(x) = x^8 + \alpha^{11} x^7 + \alpha^8 x^5 + \alpha^{10} x^4 + \alpha^4 x^3 + \alpha^3 x^2 + \alpha^8 x + \alpha^{12}$$

$$S_1 = 1$$

$$S_2 = 1$$

$$S_3 = \alpha^5$$

$$S_4 = 1$$

$$S_5 = 0$$

$$S_6 = \alpha^{10}$$

Solution

$$\sigma^{(\mu+1)}(x) = \sigma^{(\mu)}(x) + d_\mu d_\rho^{-1} x^{(\mu-\rho)} \sigma^{(\rho)}(x)$$

$$L_{\mu+1} = max [L_\mu , L_\rho + \mu - \rho]$$

$$d_{\mu+1} = S_{\mu+2} + \sigma_1^{(\mu+1)} S_{\mu+1} + \sigma_{2,}^{(\mu+1)} S_{\mu-1} + \dots + \sigma_{L(\mu+1)}^{(\mu+1)} S_{\mu+2-L(\mu+1)}$$

1. $\mu=0$, Choose $\rho = -1$

$$\sigma^{(1)}(x) = \sigma^{(0)}(x) + d_0 d_{-1}^{-1} x \sigma^{(-1)}(x) = 1\text{-}x = 1+x$$

$$d_1 = S_2 + \sigma_1^{(1)} S_1 = 1 + 1 = 0$$

$$L_1 = max [L_0 , L_{-1} + 0 + 1] = 1$$

2 . $\mu = 1$

Since $d_1 = 0$,

we have $\sigma^{(2)}(x) = \sigma^{(1)}(x) = 1+x$ and $L_2 = L_1 = 1$

$$d_2 = S_3 + \sigma_1^{(2)} S_2 + \sigma_{2,}^{(2)} S_1 = 1 + \alpha^5 = \alpha^{10}$$

3. $\mu = 2$, Since $d_2 \neq 0$, ρ must be chosen such that $(\rho - L_\mu)$ has the largest value. We choose $\rho = 0$

$$\sigma^{(3)}(x) = \sigma^{(2)}(x) - d_2 d_0^{-1} x^2 \sigma^{(0)}(x)$$

$$= 1 + x + \alpha^{10} x^2$$

$$d_3 = S_4 + \sigma_1^{(3)} S_3 + \sigma_{2,}^{(3)} S_2 + \sigma_{3,}^{(3)} S_1$$

$$= 1 + \alpha^5 + \alpha^{10} = 0$$

………………

Finally , we obtain

$$\sigma(x) = 1 + x + \alpha^{10} x^2$$

3.4.5 Chien Search

- **After the determination of the error-location polynomial, the roots of this polynomial are calculated by applying the Chien search. The roots of $\sigma(x)$ in GF(2^m) can be determined by substituting the elements of GF(2^m) in $\sigma(x)$.**

  If $\sigma(\alpha^i) = 0$, then α^i is the root of $\sigma(x)$.

  Thus, $\alpha^{-i} = \alpha^{n-I}$ is an error-location number.

- **To decode the first received digit r_{n-1} , we check whether α is a root of $\sigma(x)$.**

  If $\sigma(\alpha) = 0$, then is erroneous and must be corrected.

  If $\sigma(\alpha) \neq 0$, then r_{n-1} is error-free.

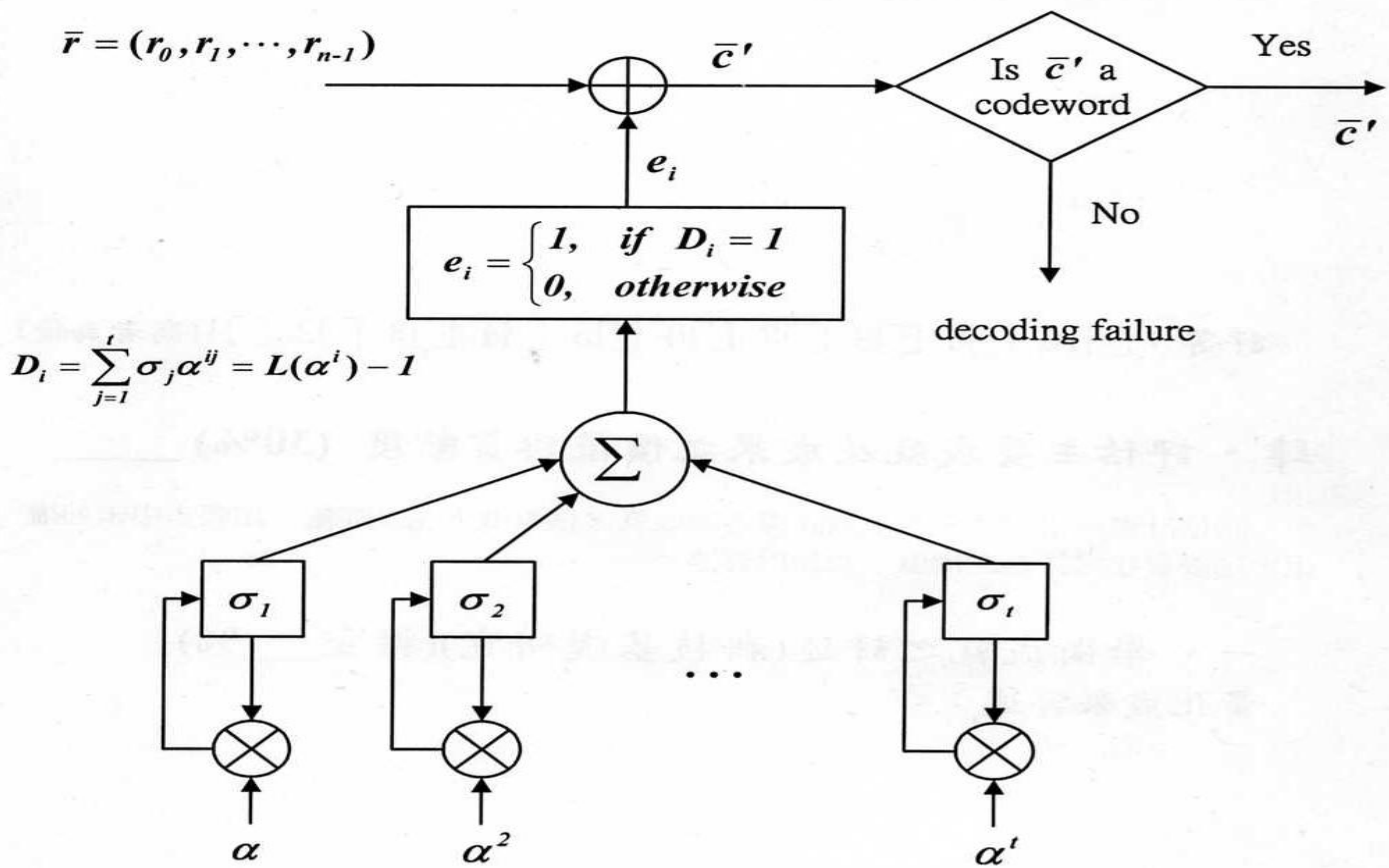
- **To decode r_{n-i} , we test whether $\sigma(\alpha^i) = 0$ or not.**

  If $\sigma(\alpha^i) = 0$, r_{n-i} is erroneous and must be corrected ,

  otherwise r_{n-i} is error-free .

- **A Chien-search circuit is shown in Fig.3. 2**

Fig.3.2 Chien-search circuit



3.4.6 Error-Value Calculation

- The generator polynomial of (n, k) RS codes can be expressed by

$$g(x) = (x+\alpha)(x+\alpha^2)\ldots(x+\alpha^{2t})$$

$$= g_0 + g_1 x + g_2 x^2 + \ldots + g_{2t-1} x^{2t-1} + x^{2t} \qquad (3.18)$$

where $g_i \varepsilon \mathrm{GF}(2^m)$.

If $c(x)$ is the transmitted codeword and $r(x)$ is the corresponding received word, then the error pattern caused by the channel impairments is given by

$$e(x) = r(x) + c(x) = \sum_{i=0}^{n-1} e_i x^i \qquad (3.19)$$

In order to determine e(x), we need to find the location x^{j_K} and the error values e_{j_K}.

- The error locator polynomial for a ν-error-correcting RS code is expressed as

$$\sigma(\mathrm{x}) = \prod_{\kappa=1}^{\nu} (1+\beta_\kappa \mathrm{x})$$
$$= (1+\beta_1 \mathrm{x})(1+\beta_2 \mathrm{x}) \ldots (1+\beta_\nu \mathrm{x})$$
$$= 1+\sigma_1 \mathrm{x}+\sigma_2 \mathrm{x}^2+\ldots+\sigma_\nu \mathrm{x}^\nu \qquad (3.20)$$

where $\beta_\kappa = \alpha^{j_k}$.

The error locations can be determined by the Berlekamp-Massey algorithm.

- Let the syndrome polynomial be

$$\mathrm{s(x)} = S_1 \mathrm{x} + S_2 \mathrm{x}^2 + \ldots + S_\nu \mathrm{x}^\nu = \sum_{i=1}^{\nu} S_i \mathrm{x}^i \qquad (3.21)$$

and define the error-evaluator polynomial as

$$\Omega(\mathrm{x}) = \sigma(x)\,\mathrm{s(x)}$$
$$= 1 + (S_1 + \sigma_1)\,\mathrm{x} + (S_2 + (\sigma_1 S_1 \div \sigma_2)\,\mathrm{x}^2 + \ldots +$$
$$(S_\nu + \sigma_1 s_{\nu-1} + \ldots + \sigma_\nu)\,\mathrm{x}^\nu \qquad (3.22)$$

- Suppose that ν errors have occurred in locations corresponding to the indices $j_1 < j_2 < \ldots < j_\nu$

Then , the syndrome components can be expressed as

$$S_q = \Sigma_{\kappa=1}^{\nu} \; Y_\kappa \; \beta_\kappa^{\,q} \qquad 1 \leqq q \leqq 2t \tag{3.23}$$

where $Y_\kappa = e_{j\kappa}$ is the error value at location j_κ

and $\beta_\kappa = \alpha^{j_\kappa}$

- For convenience sake, let us consider the syndrome polynomial of infinite degree such that

$$s(x) = \Sigma_{q=0}^{\infty} S_q x^q$$

Then , from , we obtain

$$s(x) = \Sigma_{q=1}^{\infty} \Sigma_{\kappa=1}^{\nu} \; Y_\kappa \; \beta_\kappa^{\,q} \, x^q$$

$$= \Sigma_{\kappa=1}^{\nu} \; Y_\kappa \; \Sigma_{q=1}^{\infty} \beta_\kappa^{\,q} \, x^q$$

Note that

$$\sum_{q=1}^{\infty} \beta_\kappa^q x^q = 1 + \beta_\kappa x + \beta_\kappa^2 x^2 + \ldots$$

$$= 1 / (1- \beta_\kappa x) = 1 / (1+\beta_\kappa x)$$

Then we have

$$s(x) = \sum_{\kappa=1}^{\nu} Y_\kappa / (1+\beta_\kappa x)$$

Using the above equations , the error-evaluator polynomial $Z(x)$ of degree less than ν can be written as

$$Z(x) = \sum_{\kappa=1}^{\nu} Y_\kappa \prod_{\substack{p=1 \\ p\neq\kappa}}^{\nu} (1+\beta_p x)$$

Thus , the error-value at location $x = \beta_m$ is easily obtained as

$$Y_m = Z(\beta_m^{-1}) \Big/ \prod_{\substack{p=1 \\ p\neq m}}^{\nu} (1+\beta_p \beta_m^{-1})$$

and then

$$e(x) = \sum Y_m x^m$$

Example :

Consider the triple-error-correcting (31,25) RS code. The received polynomial is

$r(x) = \alpha^8 x^2 + \alpha^2 x^5 + \alpha x^{10}$

$$s_1 = r(\alpha) = \alpha^{10} + \alpha^7 + \alpha^{11} = \alpha$$

$$s_2 = r(\alpha^2) = \alpha^{12} + \alpha^{12} + \alpha^{21} = \alpha^{21}$$

$$s_3 = r(\alpha^3) = \alpha^{14} + \alpha^{17} + \alpha^{31} = \alpha^{23}$$

$$s_4 = r(\alpha^4) = \alpha^{16} + \alpha^{22} + \alpha^{20} = \alpha^{15}$$

$$s_5 = r(\alpha^5) = \alpha^{18} + \alpha^{27} + \alpha^{20} = \alpha^2$$

$$s_6 = r(\alpha^6) = \alpha^{20} + \alpha + \alpha^{30} = \alpha^{13}$$

$r(x) = (0\ 0\alpha^8\ 0\ 0\alpha^2\ 0\ 0\ 0\ 0\alpha 000000000000000000000)$

$S = (S_1, S_2, \ldots, S_6) = (\alpha, \alpha^{21}, \alpha^{23}, \alpha^{15}\ \alpha^2, \alpha^{13})$

The error locator polynomial σ(x) can be found by applying the iterative algorithm as follows :

1. $\mu = 0$, Choose $\rho = -1$

$\sigma^{(1)}(x) = \sigma^{(0)}(x) + d_0 d_{-1}^{-1}\ x\, \sigma^{(-1)}(x) = 1 + \alpha x$

$d_1 = S_2 + \sigma_1^{(1)} S_1 = \alpha^{21} + \alpha^2 = \alpha^{13}$

$L_1 = \max [L_0, L_{-1} + 0 + 1]$

2. $\mu = 1 \quad \rho = 0$

$\sigma^{(2)}(x) = \sigma^{(1)}(x) + d_1 d_0^{-1}\ x\, \sigma^{(-1)}(x) = 1 + \alpha x + \alpha^{13}\alpha^{-1}\ x = 1 + \alpha^{20}\ x$

and $L_2 = L_1 = 1$

$d_2 = S_3 + \sigma_1^{(2)} S_2 = \alpha^{23} + \alpha^{10} = \alpha^{24}$

Note : $L_{\mu+1} = \max [L_\mu, L_\rho + \mu - \rho]$

The number $(\rho - L_\mu)$ has the largest number.

3. $\mu = 2 \quad \rho = 0$

$$\sigma^{(3)}(x) = \sigma^{(2)}(x) + d_2\, d_0^{-1}\, x^2\, \sigma^{(0)}(x)$$
$$= 1 + \alpha^{20} x + \alpha^{24} \alpha^{-1} x^2$$
$$= 1 + \alpha^{20} x + \alpha^{23} x^2$$

$$d_3 = S_4 + \sigma_1^{(3)} S_3 + \sigma_2^{(3)} S_2 = \alpha^{15} + \alpha^{12} + \alpha^{13} = \alpha^8$$

4. $\mu = 3 \quad \rho = 2$

$$\sigma^{(4)}(x) = \sigma^{(3)}(x) + d_3\, d_2^{-1}\, x^2\, \sigma^{(2)}(x)$$
$$= 1 + \alpha^{20} x + \alpha^{23} x^2 + \alpha^{15} x + \alpha^4 x^2$$
$$= 1 + \alpha^{17} x + \alpha^{15} x^2$$

$$d_4 = S_5 + \sigma_1^{(4)} S_4 + \sigma_2^{(4)} S_3 = \alpha^{15} + \alpha^{12} + \alpha^{13} = \alpha^8$$
$$= \alpha^2 + \alpha + \alpha^7 = \alpha^{30}$$

5. $\mu = 4 \quad \rho = 2$

$$\sigma^{(5)}(x) = \sigma^{(4)}(x) + d_4\, d_2^{-1}\, x^2\, \sigma^{(2)}(x)$$
$$= 1 + \alpha^{17} x + \alpha^{22} x^2 + \alpha^{26} x^3$$

$$d_5 = S_6 + \sigma_1^{(5)} S_5 + \sigma_2^{(5)} S_4 + \sigma_3^{(5)} S_3 = \alpha^{15} + \alpha^{12} + \alpha^{13} = \alpha^8$$
$$= \alpha^{13} + \alpha^{19} + \alpha^6 + \alpha^{18} = \alpha^{17}$$

6. $\mu = 5$ $\rho = 4$

$$\sigma^{(6)}(x) = \sigma^{(5)}(x) + d_5 d_4^{-1} \, x \, \sigma^{(4)}(x)$$

$$= 1 + \alpha^4 x + \alpha^5 x^2 + \alpha^{17} x^3$$

Since $\sigma(x) = \sigma^{(6}$, the error –locator polynomial is

$$\sigma(x) = 1 + \alpha^4 x + \alpha^5 x^2 + \alpha^{17} x^3$$

By the Chien search method , we can easily find that α^{21} , α^{26} and α^{29} roots of $\sigma(x)$. The reciprocals of these roots are to be the error-location number of $e(x)$. These numbers are calculated as α^{10} , α^5 and α^2 .

Thus, the triple errors occurs at positions x^{10} , x^5 and x^2 .

To find the error-values , we first calculate the error-evaluator polynomial $Z(x)$ by using eq. (3.22).

$$Z(x) = \Sigma_{\kappa=1}^{\nu} Y_\kappa \prod_{\substack{p=1 \\ p\neq\kappa}}^{\nu} (1+\beta_p x)$$

$$= 1 + (\alpha + \alpha^4) x + (\alpha^{21} + \alpha^4 \alpha + \alpha^5) x^2$$

$$+ (\alpha^{23} + \alpha^4 \alpha^{21} + \alpha^5 \alpha + \alpha^{17}) x^3$$

$$= 1 + \alpha^{30} x + \alpha^{21} x^2 + \alpha^{23} x^3$$

$$Y_2 = Z(\alpha^{-2}) \;/ (1 + \alpha^{5}\alpha^{-2})(1 + \alpha^{10}\alpha^{-2})$$

$$= \alpha^{26} / \alpha^{18} \;\; = \alpha^{8}$$

$$Y_5 = Z(\alpha^{-5}) \;/ (1 + \alpha^{2}\alpha^{-5})(1 + \alpha^{10}\alpha^{-10})$$

$$= \alpha^{30} / \alpha^{28} \;\; = \alpha^{2}$$

$$Y_{10} = Z(\alpha^{-10}) \;/ (1 + \alpha^{2}\alpha^{-10})(1 + \alpha_{5}\alpha^{-10})$$

$$= \alpha^{10} / \alpha^{9} \;\; = \alpha$$

Thus , the error-pattern polynomial is easily found as

$$e(x) = Y_2 x^2 + Y_5 x^5 + Y_{10} x^{10}$$

$$= \alpha^{8} x^2 + \alpha^{2} x^5 + \alpha x^{10}$$

3.5 Shortened RS Codes

- **In system design, if a code of natural length or suitable number of information digits can not be found , it may be desirable to shorten the code to meet the requirement.**
- **Given an (n,k) cyclic code C, consider the set of codewords for which the L leading high-order message digits are identical to zero .There are 2^{k-L} such codewords and they form a linear subcode of C. If we delete the L zero message digits from each of these codewords, we obtain a set of 2^{k-L} words of length $n-L$. These 2^{k-L} shortened words form an ($n-L$, $k-L$) linear code. This code is called a shortened cyclic code. The shortened code has the same error-correcting capability as the original code but is not cyclic is not cyclic in general.**

- **The (255, 251) RS code is designed over the Galois field GF (2^8) with error-correcting capability t = 2.**

Shortened RS codes $C_{RS}(32,28)$ and $C_{RS}(28, 24)$ are obtained from the original RS code $C_{RS}(255, 251)$ by deleting 227 digits and 223 digits, respectively , from the 255 codewords.

These two codes are the constituent codes of the compact disc (CD) error-control coding system.

Both shortened RS codes and the original RS code have the same generator polynomial.

The generator polynomial is given by

$$\begin{aligned} g(x) &= (x+\alpha)(x+\alpha^2)(x+\alpha^3)(x+\alpha^4) \\ &= x^4+\alpha^{76}x^3+\alpha^{251}x^2+\alpha^{81}x+\alpha^{10} \end{aligned}$$

All operations performed in te calculation of this generator polynomial are done in GF(2^8) .

Table 3.2 Minimal polynomials of the elements of GF(2^6)

Elements	Minimal polynomials
$\alpha, \alpha^2, \alpha^4, \alpha^{16}, \alpha^{32}$	$1 + X + X^6$
$\alpha^3, \alpha^6, \alpha^{12}\alpha^{24}, \alpha^{48}\alpha^{33}$	$1 + X + X^2 + X^4 + X^6$
$\alpha^5, \alpha^{10}, \alpha^{20}, \alpha^{40}, \alpha^{17}, \alpha^{34}$	$1 + X + X^2 + X^5 + X^6$
$\alpha^7, \alpha^{14}, \alpha^{28}, \alpha^{56}, \alpha^{49}, \alpha^{35}$	$1 + X^3 + X^6$
$\alpha^9, \alpha^{18}, \alpha^{36}$	$1 + X^2 + X^3$
$\alpha^{11}, \alpha^{22}, \alpha^{44}, \alpha^{25}, \alpha^{50}, \alpha^{37}$	$1 + X^2 + X^3 + X^5 + X^6$
$\alpha^{13}, \alpha^{26}, \alpha^{52}, \alpha^{41}, \alpha^{19}, \alpha^{38}$	$1 + X + X^3 + X^4 + X^6$
$\alpha^{15}, \alpha^{30}, \alpha^{60}, \alpha^{57}, \alpha^{51}, \alpha^{39}$	$1 + X^2 + X^4 + X^5 + X^6$
α^{21}, α^{42}	$1 + X + X^2$
$\alpha^{23}, \alpha^{46}, \alpha^{29}, \alpha^{58}, \alpha^{53}, \alpha^{43}$	$1 + X + X^4 + X^5 + X^6$
$\alpha^{27}, \alpha^{54}, \alpha^{45}$	$1 + X + X^3$
$\alpha^{31}, \alpha^{62}, \alpha^{61}, \alpha^{59}, \alpha^{55}, \alpha^{47}$	$1 + X^5 + X^6$

Table 3.3

Generator polynomials of all the BCH codes of length 63

n	k	t	$\mathbf{g}(X)$
63	57	1	$\mathbf{g}_1(X) = 1 + X + X^6$
	51	2	$\mathbf{g}_2(X) = (1 + X + X^6)(1 + X + X^2 + X^4 + X^6)$
	45	3	$\mathbf{g}_3(X) = (1 + X + X^2 + X^5 + X^6)\mathbf{g}_2(X)$
	39	4	$\mathbf{g}_4(X) = (1 + X^3 + X^6)\mathbf{g}_3(X)$
	36	5	$\mathbf{g}_5(X) = (1 + X^2 + X^3)\mathbf{g}_4(X)$
	30	6	$\mathbf{g}_6(X) = (1 + X^2 + X^3 + X^5 + X^6)\mathbf{g}_5(X)$
	24	7	$\mathbf{g}_7(X) = (1 + X + X^3 + X^4 + X^6)\mathbf{g}_6(X)$
	18	10	$\mathbf{g}_{10}(X) = (1 + X^2 + X^4 + X^5 + X^6)\mathbf{g}_7(X)$
	16	11	$\mathbf{g}_{11}(X) = (1 + X + X^2)\mathbf{g}_{10}(X)$
	10	13	$\mathbf{g}_{13}(X) = (1 + X + X^4 + X^5 + X^6)\mathbf{g}_{11}(X)$
	7	15	$\mathbf{g}_{15}(X) = (1 + X + X^3)\mathbf{g}_{13}(X)$

Appen.: Division circuit for dividing X(D) by G(D)

$X(D) = x_0 + x_1 D + x_2 D^2 + \ldots + x_{n-1} D^{n-1}$

$G(D) = g_0 + g_1 D + g_2 D^2 + \ldots + g_{n-k} D^{n-k}$

Note : 1. The high-order coefficients are input first .

2. First output is coefficient of D^{n-1} of quotient

3. Shift register contains coefficients of remainder

$r(D) = r_0 + r_1 D + r_2 D^2 + \ldots + r_{n-k-1} D^{n-k-1}$