Reed-Solomon (RS) Codes

RS Codes are a sub-class of non-bina
BCH Codes. In a non-binary code, codewords consist of symbols which are each $m \geq 2$ bits long.

In general, non-binary codes can be defined over any Galois field $GF(q)$ where q is either a prime or a power of a prime. However, for obvious reasons, people are most interested in codes defined over $GF(2^m)$.

For Reed-Solomon codes take some integer m. Then each symbol is m bits long. This means that symbols belong to $\{0, 1, \ldots, 2^m$

An (N, K) RS Code consists of N symbols each of which ~~is~~ m bits long and has K information symbols and $N-K$ parity symbols.

For an RS Code over $GF(2^m)$ we have

$N = 2^m - 1$.

K can be any value less than N.

An (N, K) RS Code has the minimum distance $d_{min} = N - K + 1$.

It can correct $t = \left\lfloor \frac{d_{min} - 1}{2} \right\rfloor = \frac{N-K}{2}$

The reason I used N and K instead of n and k was to differentiate between an (n, k) binary code that has codewords that are n bits long and have k information bits and non-binary codes with N and K symbols.

I hope we have so far have got used to the idea of symbols other than a single bit. So, from this point on, I will use n and k.

(n, k) RS Code over $GF(2^m)$ has codeword of length n symbols, i.e., $n*m$ bits out of which $k*m$ are information (or systematic) bits.

For example a (255, 239) RS Code

over $GF(2^8)$ has codewords each 255 bytes and each codeword has 239 bytes of information $(n-k)=16$ bytes of parity. Such a code can correct up to $\frac{16}{2}=8$ bytes of errors.

Note that here when we correct one symbol, we may have corrected $1, 2, \ldots, m$ bits. If we have a burst of errors, that is a lot of errors near one another, RS codes can be very useful. An RS code which can correct t error symbols can correct $(t-1)m$-bit long bursts.

The generating polynomial of t error correcting RS code is:

$$g(X) = (X+\alpha)(X+\alpha^2)\cdots(X+\alpha^{2t})$$
$$= g_0 + g_1 X + g_2 X^2 + \cdots + g_{2t-1} X^{2t-1} + X^{2t}$$

with $g_i \in GF(2^m)$ for $0 \leqslant i \leqslant 2t$.

$\alpha, \alpha^2, \ldots, \alpha^{2t}$ are roots of X^n+1.

$g(X)$ divides X^n+1. So, $g(X)$ generates a 2^m-ry cyclic code of length n with $2t$ parity symbols.

Encoding of RS Codes:

we can simply multiply the information polynomial $u(x)$ ~~by~~ by $g(x)$. However, this may not result in a systematic code to make the code systematic, we multiply $u(x)$ by X^{n-k} to get $X^{n-k}u(x)$ which we divide by $g(x)$ to get:

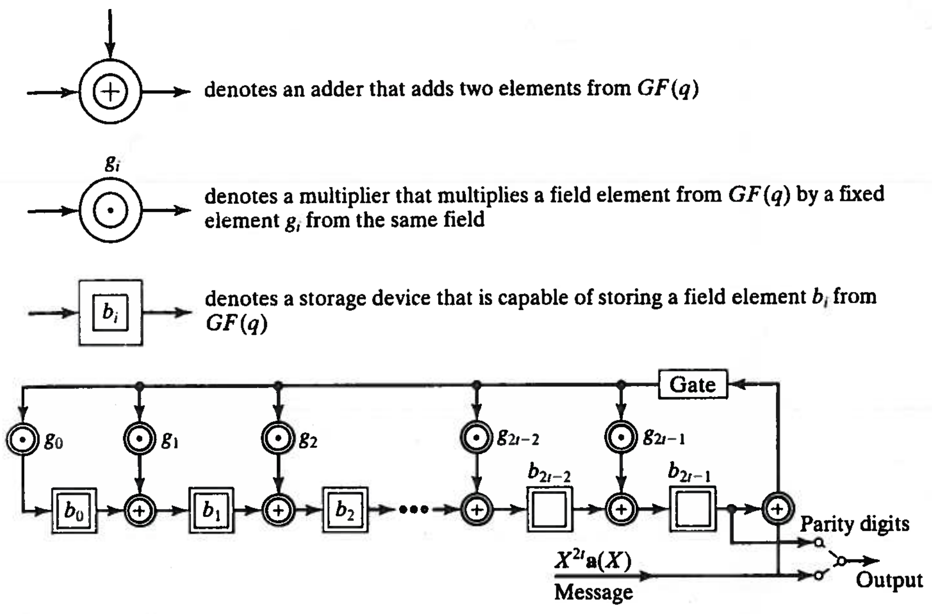
$$X^{n-k}u(x) = q(x)g(x)+b(x)$$

$q(x)g(x)$ is a code polynomial. Also, we have:

$$V(X) = q(X)g(X) = X^{n-k}u(X)+b(X).$$

This means that we have $u(X)$ as part of $V(X)$, i.e., the code is systematic and $b(X)$ is the parities' polynomial.

The following circuit shows the encoding procedure:


denotes an adder that adds two elements from $GF(q)$
g_i
denotes a multiplier that multiplies a field element from $GF(q)$ by a fixed element g_i from the same field
b_i
denotes a storage device that is capable of storing a field element b_i from $GF(q)$
Gate
g_0
g_1
g_2
g_{2t-2}
g_{2t-1}
b_0
b_1
b_2
b_{2t-2}
b_{2t-1}
Parity digits
$X^{2t}\mathbf{a}(X)$
Message
Output

Encoding circuit for a q-ary RS code with generator polynomial $\mathbf{g}(X) = g_0 + g_1X + g_2X^2 + \cdots + g_{2t-1}X^{2t-1} + X^{2t}$.

1) First we close the gate and feed the information symbols into the division circuit. At the same time these information symbols are put on the line (to be transmitted): switch in lower position.

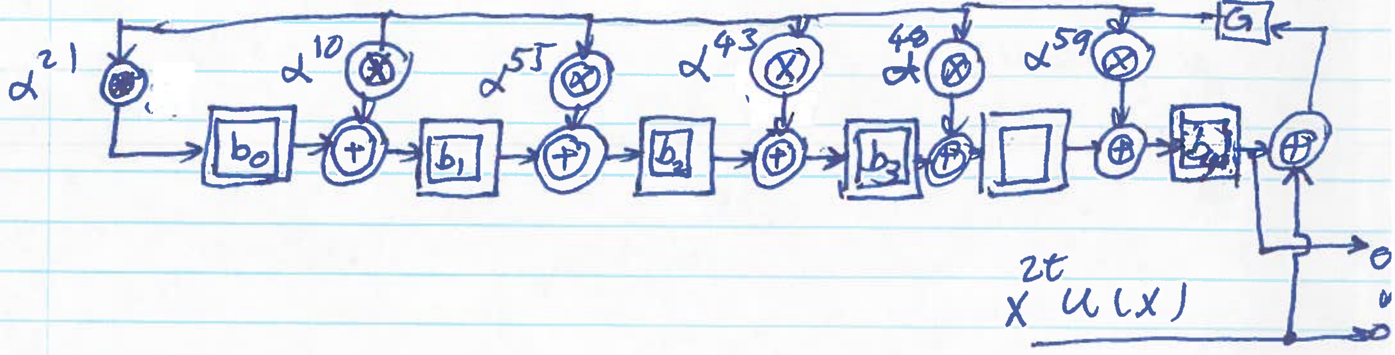
2) After feeding all k symbols, we open the gate (disconnect the feedback) and put switch in the up position, transmitting $2t$ parity symbols.

Example:

Find the generating polynomial of triple error correcting code over $GF(2^6)$.

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

$$= \alpha^{21} + \alpha^{10}X + \alpha^{55}X^2 + \alpha^{43}X^3 + \alpha^{48}X^4 + \alpha^{59}X^5 + X^6$$



The parity-check matrix of an RS Code is given as:

$$H = \begin{bmatrix} 1 & \alpha & \alpha^2 & \cdots & \alpha^{n-1} \\ 1 & \alpha^2 & (\alpha^2)^2 & \cdots & (\alpha^2)^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \alpha^{2t} & (\alpha^{2t})^2 & \cdots & (\alpha^{2t})^{n-1} \end{bmatrix}$$

Decoding of RS Codes

1) Find Syndrome.
2) Find error-location polynomial
3) Find error-value evaluator
4) Find the error locations and error values and correct.

Assume that the codeword $\underline{v} = (v_0, v_1, \cdots v_{n-1})$ is transmitted or equivalently

$$v(X) = v_0 + v_1 X + v_2 X^2 + \cdots + v_{n-1} X^{n-1}$$

Assume that $r(X)$ is received:

$$r(X) = r_0 + r_1 X + r_2 X^2 + \cdots + r_{n-1} X^{n-1}$$

$r(X) = v(X) + e(X)$ where $e(X)$ is the error polynomial $e(X) = r(X) + v(X) = e_0 + e_1 X + \cdots + e_{n-1} X^{n}$

Assume we have errors at locations

$$j_1, j_2, \cdots, j_\nu$$

Denote the values of error by $e_{j_1}, e_{j_2}, \cdots e_{j_\nu}$

Then $e_i = \begin{cases} 0 & i \neq j_1, \cdots j_\nu \\ e_{j_\ell} & \text{if } i = j_\ell \in \{j_1, \cdots j_\ell\} \end{cases}$

So, we can write:

$$e(X) = e_{j_1} X^{j_1} + e_{j_2} X^{j_2} + \cdots + e_{j_\nu} X^{j_\nu}$$

So, what we need to do is to find $j_1, \cdots j_\nu$ as well as $e_{j_1}, \cdots, e_{j_\nu}$.
That is we have 2ν unknowns.

Remember that

$$v(\alpha^i) = 0 \qquad i = 1, 2, \cdots, 2t$$

$$r(\alpha^i) = v(\alpha^i) + e(\alpha^i) = S_i$$

So,

$$S_i = r(\alpha^i) = e(\alpha^i).$$

That is we substitute α^i, $i = 1, \cdots, 2t$ in $r(X)$ to get $2t$ Syndromes. These provide $2t$ equations with j_i's and e_{j_i}'s as their Components. In order to be able to Solve for the 2ν unknowns, we need to have 2ν equations, i.e., $2t = 2\nu \Rightarrow t = \nu$. That is a proof that RS Code can correct t errors.

Now let's expand $S_i = e(\alpha^i)$'s:

$$S_1 = e_{j_1}\alpha^{j_1} + e_{j_2}\alpha^{j_2} + \cdots + e_{j_\nu}\alpha^{j_\nu}$$

$$S_2 = e_{j_1}\alpha^{2j_1} + e_{j_2}\alpha^{2j_2} + \cdots + e_{j_\nu}\alpha^{2j_\nu}$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$S_{2t} = e_{j_1}\alpha^{2tj_1} + e_{j_2}\alpha^{2tj_2} + \cdots + e_{j_\nu}\alpha^{2tj_\nu}$$

Let $\beta_i \triangleq \alpha^{j_i}$ and $\delta_i \triangleq e_{j_i}$

for $1 \leq i \leq \nu$.

Then:

$$S_1 = \delta_1\beta_1 + \delta_2\beta_2 + \cdots + \delta_\nu\beta_\nu$$

$$S_2 = \delta_1\beta_1^2 + \delta_2\beta_2^2 + \cdots + \delta_\nu\beta_\nu^2$$

$$\vdots$$

$$S_{2t} = \delta_1\beta_1^{2t} + \delta_2\beta_2^{2t} + \cdots + \delta_\nu\beta_\nu^{2t}$$

Define the error location polynomial:

$$\sigma(X) = (1+\beta_1 X)(1+\beta_2 X)\cdots(1+\beta_\nu X)$$

$$= \sigma_0 + \sigma_1 X + \sigma_2 X^2 + \cdots + \sigma_\nu X^\nu$$

We can see that

$$\sigma_0 = 1$$

$$\sigma_1 = \beta_1 + \beta_2 + \cdots + \beta_\nu = S_1$$

$$\sigma_2 = \beta_1\beta_2 + \cdots + \beta_{\nu-1}\beta_\nu = \sigma_1 S_1 + S_2$$

$$\vdots$$

Overall, we get the following equations named Newton equalities:

$$\begin{aligned}
&S_{\nu+1} + \sigma_1 S_\nu + \sigma_2 S_{\nu-1} + \cdots + \sigma_\nu S_1 = 0\\
&S_{\nu+2} + \sigma_1 S_{\nu+1} + \sigma_2 S_\nu + \cdots + \sigma_\nu S_2 = 0\\
&\vdots \qquad\qquad\qquad\qquad\qquad\qquad \vdots\\
&S_{2t} + \sigma_1 S_{2t-1} + \sigma_2 S_{2t-2} + \cdots + \sigma_\nu S_{2t-\nu} = 0
\end{aligned}$$

The same as BCH Codes, we start from $\sigma(X) = 1$ in stage 0, say we call it $\sigma^{(0)}(X)$ and try to increase the number of terms so that all equations are satisfied.

Assume that at stage μ we have

$$\sigma^{(\mu)}(X) = \sigma_0^{(\mu)} + \sigma_1^{(\mu)} X + \cdots + \sigma_{L_\mu}^{(\mu)} X^{L_\mu}.$$

This means that we have coefficients $\sigma_0^{(\mu)}, \sigma_1^{(\mu)}, \ldots \sigma_{L_\mu}^{(\mu)}$ of a polynomial that satisfy the first μ Newton equalities.

We try to apply these colfficents to $\mu+1$-st equality, i.e., form

$$S_{\mu+1} + \sigma_1^{(\mu)} S_\mu + \cdots + \sigma_{L_\mu}^{(\mu)} S_{\mu+1-L_\mu}$$

If this gives us a zero it means that $\sigma_0^{(\mu)}, \sigma_1^{(\mu)}, \ldots \sigma_{L_\mu}^{(\mu)}$ satisfy $\mu+1$-st equality.

otherwise we have to modify the polynomial So form:

$$d_\mu = S_{\mu+1} + \sigma_1^{(\mu)} S_\mu + \sigma_2^{(\mu)} S_{\mu-1} + \cdots + \sigma_{L_\mu}^{(\mu)} S_{\mu+1-L_\mu}$$

If the discrepancy $d_\mu = 0$ then

$$\sigma^{(\mu+1)}(X) = \sigma^{(\mu)}(X)$$

and Continue.

otherwise:

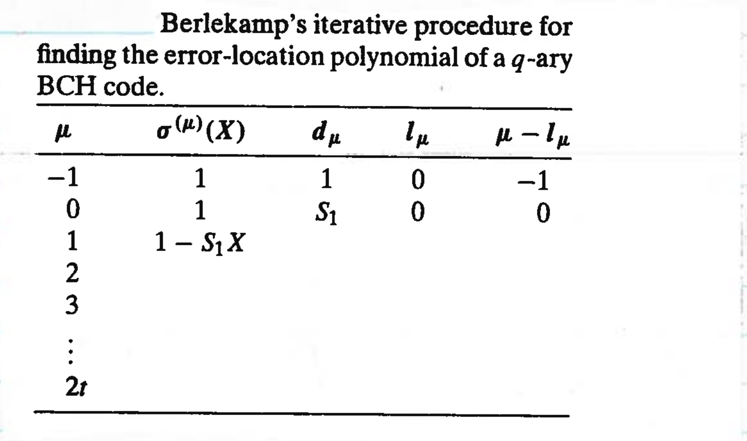
$$\sigma^{(\mu+1)}(X) = \sigma^{(\mu)}(X) + d_\mu d_\rho^{-1} X^{\mu-\rho} \sigma^{(\rho)}(X)$$

where ρ is the stage closest to μ such that $d_\rho \neq 0$

Continue this iteration until we get to stage 2t then

$$\sigma(X) = \sigma^{(2t)}(X).$$

Start by filling out the first two rows:



Berlekamp's iterative procedure for finding the error-location polynomial of a q-ary BCH code.

μ	$\sigma^{(\mu)}(X)$	d_μ	l_μ	$\mu - l_\mu$
-1	1	1	0	-1
0	1	S_1	0	0
1	$1 - S_1 X$			
2				
3				
$\vdots$				
$2t$				

Example:

Consider triple-error correcting code over $GF(2^4)$. Let $r(X) = \alpha^7 X^3 + \alpha^3 X^6 + \alpha^4 X^{12}$.

Then

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

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

$S_1 = r(\alpha) = \alpha^{10} + \alpha^9 + \alpha = \alpha^{12}$

$S_2 = r(\alpha^2) = \alpha^{13} + 1 + \alpha^{13} = 1$

$S_3 = r(\alpha^3) = \alpha + \alpha^6 + \alpha^{10} = \alpha^{14}$

$S_4 = r(\alpha^4) = \alpha^4 + \alpha^{12} + \alpha^7 = \alpha^{10}$

$S_5 = r(\alpha^5) = \alpha^7 + \alpha^3 + \alpha^4 = 0$

$S_6 = r(\alpha^6) = \alpha^{10} + \alpha^9 + \alpha = \alpha^{12}$.

TABLE 7.2: Steps for finding the error-location polynomial of the (15,9) RS code over $GF(2^4)$.

μ	$\sigma^{(\mu)}(X)$	d_μ	l_μ	$\mu - l_\mu$
-1	1	1	0	-1
0	1	α^{12}	0	0
1	$1+\alpha^{12}X$	α^7	1	0(take $\rho = -1$)
2	$1+\alpha^3X$	1	1	1(take $\rho = 0$)
3	$1+\alpha^3X+\alpha^3X^2$	α^7	2	1(take $\rho = 0$)
4	$1+\alpha^4X+\alpha^{12}X^2$	α^{10}	2	2(take $\rho = 2$)
5	$1+\alpha^7X+\alpha^4X^2+\alpha^6X^3$	0	3	2(take $\rho = 3$)
6	$1+\alpha^7X+\alpha^4X^2+\alpha^6X^3$	—	—	—

Step 2. To find the error-location polynomial $\sigma(X)$, we fill out Table 7.1 and obtain Table 7.2. Thus, $\sigma(X) = 1 + \alpha^7X + \alpha^4X^2 + \alpha^6X^3$.

Step 3. By substituting $1, \alpha, \alpha^2, \cdots, \alpha^{14}$ into $\sigma(X)$, we find that α^3, α^9, and α^{12} are roots of $\sigma(X)$. The reciprocals of these roots are α^{12}, α^6, and α^3, which are the error-location numbers of the error pattern $\mathbf{e}(X)$. Thus, errors occur at positions X^3, X^6, and X^{12}.

A more straightforward algorithm where the correction term is evolved as the iterations go ahead is given in Vicker's text.

The algorithm is as follows:

1) Compute Syndromes $S_1, \ldots, S_{2t}$.

2) Initialize the algorithm by letting

$\mu = 0$, $\sigma^{(0)}(X) = 1$, $L = 0$ and $T(X) = X$.

3) Set $\mu = \mu + 1$. Compute discrepancy d_μ as

$$d_\mu = S_\mu + \sum_{i=1}^{L} \sigma_i^{(\mu-1)} S_{\mu-i}$$

4) If $d_\mu = 0$ then go to 8.

5) Modify the polynomial as

$$\sigma^{(\mu)}(X) = \sigma^{(\mu-1)}(X) + d_\mu T(X)$$

6) If $2L \geq \mu$ then go to step 8.

7) Set $L = \mu - L$ and $T(X) = d_\mu^{-1} \sigma^{(\mu-1)}(X)$

8) Set $T(X) = X \cdot T(X)$.

9) If $\mu < 2t$ go to step 3.

10) Determine $\sigma(X) = \sigma^{(2t)}(X)$. If the roots are distinct and in the right field, then determine the error values, Correct the errors and STOP.

11) Declare a decoding failure and STOP.

Example: Consider (7,3) RS Code over GF(8) with $r(X) = \alpha^2 X^6 + \alpha^2 X^4 + X^3 + \alpha^5 X^2$.

Although we have done the generation of $g(X)$ and encoding, let's start from ground zero for doing some exercise in Galois field arithmatic. Let's start with $p(X) = X^3 + X + 1$. Take α to be a primitive element of this fied, i.e., a root of

$p(x)$. That is $\alpha^3+\alpha+1=0$ or $\alpha^3=\alpha+1$.

The field elements are:

0	0	0	0	0
1	1	0	0	1
α^1	0	1	0	α
$\alpha^2=\alpha\cdot\alpha$	0	0	1	α^2
$\alpha^3=\alpha^2\cdot\alpha$	1	1	0	$\alpha+1$
α^4	0	1	1	$\alpha^2+\alpha$
α^5	1	1	1	$\alpha^2+\alpha+1$
α^6	1	0	1	α^2+1
α^7	1	0	0	1

Note:

- $\alpha^3=\alpha^2\cdot\alpha=\alpha+1$
- $\alpha^4=\alpha\cdot\alpha^3=\alpha(\alpha+1)$
  $=\alpha^2+\alpha$
- $\alpha^5=\alpha^4\cdot\alpha=(\alpha^2+\alpha)\alpha$
  $=\alpha^3+\alpha^2=\alpha^2+\alpha+1$
- $\alpha^6=\alpha(\alpha^2+\alpha+1)$
  $=\alpha^2+1$
- $\alpha^7=\alpha(\alpha^2+1)$
  $=\alpha^3+\alpha$
  $=\alpha+1+\alpha=1$

Now, $g(x)$ is:

$$
\begin{aligned}
g(x) &= (x+\alpha)(x+\alpha^2)(x+\alpha^3)(x+\alpha^4) \\
&= [x^2+(\alpha+\alpha^2)x+\alpha^3][x^2+(\alpha^3+\alpha^4)x+\alpha^7] \\
&= [x^2+\alpha^4x+\alpha^3][x^2+\alpha^6x+1] \\
&= x^4+\alpha^3x^3+x^2+\alpha x+\alpha^3
\end{aligned}
$$

Computing Syndromes:

$S_i=r(\alpha^i)$. $\quad i=1,2,3,4$

In this case, since the number of parities are less th the number of information symbols, it is reasonabl

to use $r(\alpha^i) = S_i$. However, for high rate codes where $n-k \ll k$, it is better to divide $r(x)$ by $g(x)$ to get

$$r(x) = g(x)\,q(x) + b(x)$$

where $b(x)$ is a polynomial of degree less than or equal $n-k$.

$$S_i = r(\alpha^i) = g(\alpha^i)\,q(\alpha^i) + b(\alpha^i) \qquad i = 1, 2, \ldots, 2t.$$

since $g(\alpha^i) = 0 \qquad i = 1, \ldots, 2t$

$$S_i = b(\alpha^i)$$

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

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

So:

$$S_1 = b(\alpha) = \alpha^5 + \alpha^9 + \alpha^8 + \alpha^2 = \alpha^6$$

$$S_2 = b(\alpha^2) = \alpha^9 + \alpha^{12} + \alpha^{10} + \alpha^3 = \alpha^3$$

$$S_3 = b(\alpha^3) = \alpha^{13} + \alpha^{15} + \alpha^{12} + \alpha^4 = \alpha^4$$

$$S_4 = b(\alpha^4) = \alpha^{17} + \alpha^{18} + \alpha^{14} + \alpha^5 = \alpha^3$$

Now, we use the algorithm:

μ	S_μ	$\sigma^{(\mu)}(X)$	d_μ	L	$T(X)$
0	—	1	—	0	X
1	α^6	$1+\alpha^6 X$	α^6	1	αX *
2	α^3	$1+\alpha^4 X$	α^2	1	αX^2 **
3	α^4	$1+\alpha^4 X+\alpha^6 X^2$	α^5	2	$\alpha^2 X+\alpha^6 X^2$
4	α^3	$1+\alpha^2 X+\alpha X^2$	α^6	—	—

* Note:

For $\mu=1$ $\quad L=0 \Rightarrow 2L<\mu \Rightarrow L=\mu-L=1$

and $T(X)=\dfrac{\sigma^{(0)}(X)}{d_1}=\dfrac{X}{\alpha 6}=\alpha X.$

** Note:

For $\mu=2$

$$d_\mu = S_\mu+\sum_{i=1}^{L}\sigma_i^{(\mu-1)}S_{\mu-i} \Rightarrow \mu_2 = S_2+\sigma_1^{(1)}S_1$$

or

$$\mu_2=\alpha^3+\alpha^6\cdot\alpha^6=\alpha^3+\alpha^5=\alpha^2$$

$2L=2\geq\mu=2 \Rightarrow T(x)=x\,T(x) \Rightarrow T(x)=\alpha x^2$

So:

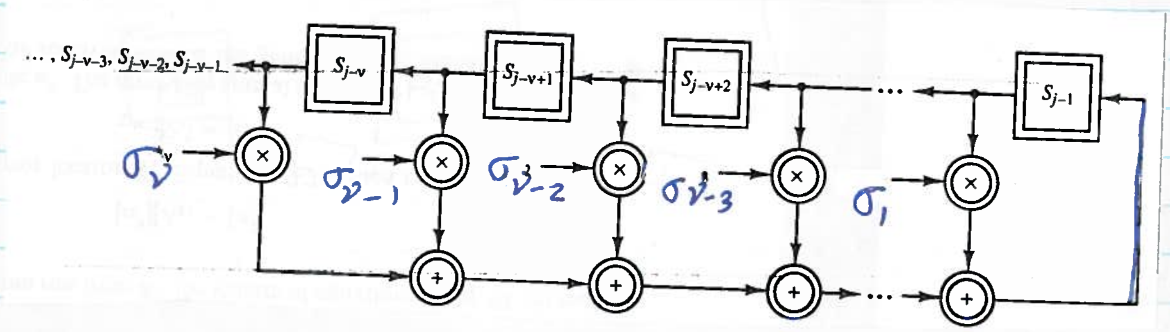
$$\sigma(X)=\alpha X^2+\alpha^2 X+1.$$

The above algorithm is based on Massey's Linear Feedback Shift Register (LFSR) Synthesis technique.

Note that for ν errors, we have the following Newton equalities

$$S_j = \sigma_1 S_{j-1} + \sigma_2 S_{j-2} + \cdots + \sigma_\nu S_{j-\nu}$$

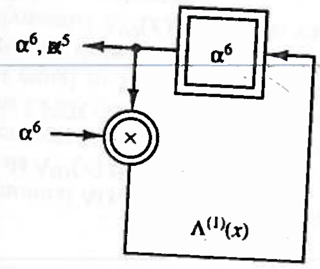
This relationship can be represented as LFSR circuit looking like:



The problem of finding error-locator polynomial is then to find an LFSR of minimal length such that the first $2t$ elements in the output sequence are $S_1, S_2, \ldots, S_2$.

The coefficients of the filter are then the coefficients of $\sigma(x)$.

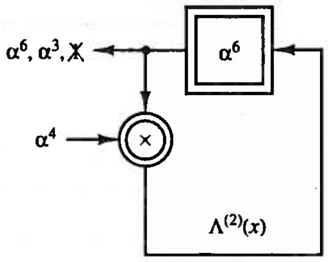
For the above (7,3) RS Code, we start with



This works for the $S_1 = \alpha^6$ as it outputs the content of the register, i.e., α^6.

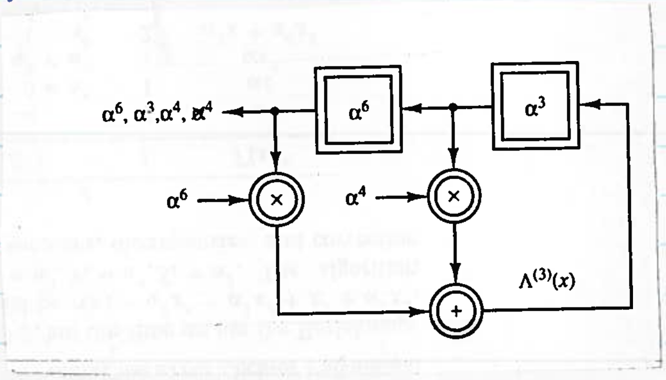
But after the application of the seconds clock, the output will be $\alpha^6 . \alpha^6 = \alpha^{12} = \alpha^5$ which is not $S_2 = \alpha^3$.

To correct the situation, we change the filter tap to α^4 which is $\frac{\alpha^6}{\alpha^2}$ and therefore, the output after the ~~c~~ clocking will be $\frac{\alpha^5}{\alpha^2} = \alpha^3 = S_2$.



After the next clock the output will be $\alpha^3 . \alpha^4 = 1$ which is not equal to $S_3 = \alpha^4$

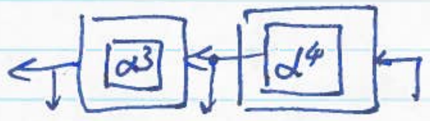
To correct this we need to add α^5 so that, we get $1+\alpha^5=\alpha^4=S_3$. We keep the above and add a stage with α^6 in the register and α^6 as the tap.



This circuit outputs α^6 first and then calculates

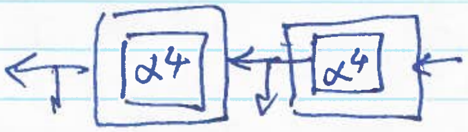
$$\alpha^{6}\cdot\alpha^6+\alpha^3\cdot\alpha^4=\alpha^5+1=\alpha^4$$

Content of the rightmost SR is moved to left and α^4 is loaded into it



So, the next output is $\alpha^3=S_2$.

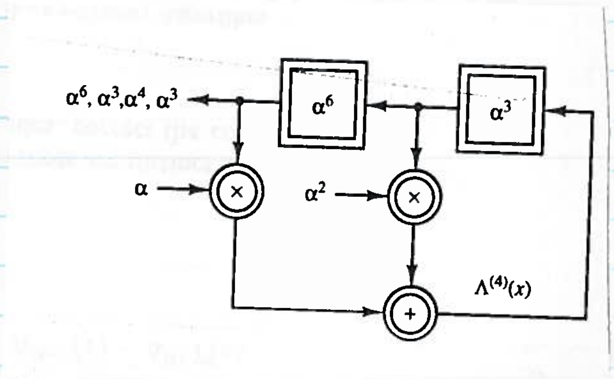
Next $\alpha^3\cdot\alpha^6+\alpha^4\cdot\alpha^4=\alpha^2+\alpha=\alpha^4$ is placed in right register and α^4 is moved left



Now α^4 is output which is S_3.

But the next output is $\alpha^4\neq S_4=\alpha^3$

To avoid this, we modify the taps of the LFSR to :



It is easy to see that this circuit outputs $\alpha^6, \alpha^3, \alpha^4, \alpha^3$, i.e., S_1, S_2, S_3, S_4.

Finding the error values

Now, we have found error-locator polynomial $\sigma(X)$. We can solve it to find the error locations $\beta_i = \alpha^{j_i}$ $i = 1, 2, \cdots, \nu$.

Now, we need to find $\delta_i = e_{j_i}$, i.e., error values at the error locations and correct them.

That is, the equations are :

$$S_1 = e_{j_1}\alpha^{j_2} + e_{j_2}\alpha^{j_2} + \cdots + e_{j_\nu}\alpha^{j_\nu}$$

$$\vdots$$

$$S_{2t} = e_{j_1}\alpha^{2tj_1} + e_{j_2}\alpha^{2tj_2} + \cdots + e_{j_\nu}\alpha^{2tj_\nu}.$$

with α^{j_i}'s and S_i's known. Or equivalently

$$S_1 = \delta_1\beta_1 + \delta_2\beta_2 + \cdots + \delta_\nu\beta_\nu$$
$$S_2 = \delta_1\beta_1^2 + \delta_2\beta_2^2 + \cdots + \delta_\nu\beta_\nu^2$$
$$\vdots \qquad\qquad \vdots$$
$$S_{2t} = \delta_1\beta_1^{2t} + \delta_2\beta_2^{2t} + \cdots + \delta_\nu\beta_\nu^{2t}$$

Let's define the Syndrome polynomial:

$$S(X) = S_1 + S_2X + \cdots + S_{2t}X^{2t} + S_{2t+1}X^{2t} + \cdots$$
$$= \sum_{j=1}^{\infty} S_j X^{j-1}$$

Note that this has an infinite number of terms whose first $2t$ terms are known:

$$S_j = \sum_{\ell=1}^{\nu} \delta_\ell \beta_\ell^j \qquad j = 1, 2, \ldots, 2t$$

Substituting this (but now for all terms), we get,

$$S(X) = \sum_{j=1}^{\infty} X^{j-1} \sum_{\ell=1}^{\nu} \delta_\ell \beta_\ell^j$$

$$= \sum_{\ell=1}^{\nu} \delta_\ell \beta_\ell \sum_{j=1}^{\infty} (\beta_\ell X)^{j-1}$$

But
$$\sum_{j=1}^{\infty} (\beta_\ell X)^{j-1} = \frac{1}{1+\beta_\ell X}$$

So:

$$S(X) = \sum_{\ell=1}^{\nu} \frac{\delta_\ell \beta_\ell}{1+\beta_\ell X}$$

$$\sigma(X) = \prod_{i=1}^{\nu} (1+\beta_\ell X)$$

So:

$$S(X)\sigma(X) = \sum_{\ell=1}^{\nu} \delta_\ell \beta_\ell \prod_{i=1, i\neq \ell}^{\nu} (1+\beta_i X) \triangleq Z_0(X)$$

Also,

$$\sigma(X)S(X) = [1+\sigma_1 X + \cdots + \sigma_\nu X^\nu][S_1 + S_2 X + S_3 X^2 + \cdots]$$

$$= S_1 + (S_2 + \sigma_1 S_1)X + (S_3 + \sigma_1 S_2 + \sigma_2 S_1)X^2 + \cdots$$

$$\cdots + (\sigma_{2t} + \sigma_1 S_{2t-1} + \cdots + \sigma_\nu S_{2t-\nu})X^{2t-1} + \cdots$$

So:

$$Z_0(X) = S_1 + (S_2 + \sigma_1 S_1)X + (S_3 + \sigma_1 S_2 + \sigma_2 S_1)X^2 + \cdots$$

$$+ \cdots + (S_\nu + \sigma_1 S_{\nu-1} + \cdots + \sigma_{\nu-1} S_1)X^{\nu-1}$$

Let's substitute β_k^{-1} in $Z_0(X)$:

$$Z_0(\beta_k^{-1}) = \sum_{\ell=1}^{\nu} \delta_\ell \beta_\ell \prod_{i=1, i\neq \ell}^{\nu} (1+\beta_i \beta_k^{-1})$$

$$= \delta_k \beta_k \prod_{i=1, i\neq k}^{\nu} (1+\beta_i \beta_k^{-1})$$

Taking derivative of $\sigma(X)$

$$\sigma'(X) = \frac{d}{dX} \prod_{i=1}^{\nu} (1+\beta_i X) = \sum \beta_\ell \prod_{i=1, i\neq \ell}^{\nu} (1+\beta_i X)$$

Then
$$\sigma'(\beta_k^{-1}) = \beta_k \prod_{i=1, i \neq k}^{\nu} (1 + \beta_i \beta_k^{-1})$$

So,
$$\delta_k = \frac{Z_0(\beta_k^{-1})}{\sigma'(\beta_k^{-1})}$$

Let's $[\sigma(X) S(X)]_{2t}$ represent the first $2t$ terms of $\sigma(X) S(X)$. Then
$$\sigma(X) S(X) - [\sigma(X) S(X)]_{2t}$$
is divisible by X^{2t}.

That is:
$$\sigma(X) S(X) \equiv [\sigma(X) S(X)]_{2t} \mod X^{2t}$$

But,
$$[\sigma(X) S(X)]_{2t} = Z_0(X)$$

and we have:
$$\sigma(X) S(X) \equiv Z_0(X) \mod X^{2t}.$$

This is called the key equation that has to be solved in decoding of RS Codes.

Example : Consider the (7,4) Code in the previous example:

We had $S_1 = \alpha^6$, $S_2 = \alpha^3$, $S_3 = \alpha^4$ and $S_4 = \alpha^3$

So:

$$S(X) = \alpha^6 + \alpha^3 X + \alpha^4 X^2 + \alpha^3 X^3$$

also, we found:

$$\sigma(X) = 1 + \alpha^2 X + \alpha X^2 \Rightarrow \sigma'(X) = \alpha^2 + 2\alpha X = \alpha^2$$

So:

$$Z_0(X) = \sigma(X) S(X) \mod X^4$$
$$= (1 + \alpha^2 X + \alpha X^2)(\alpha^6 + \alpha^3 X + \alpha^4 X^2 + \alpha^3 X^3)$$
$$= \alpha^6 + X$$

We can find the error locations by solving $\sigma(X) = 0$ to get $\beta_1 = \alpha^3$ and $\beta_2 = \alpha^5$

So,

$$e_3 = \delta_1 = \frac{Z_0(\alpha^{-3})}{\sigma'(\alpha^{-3})} = \frac{\alpha^6 + \alpha^{-3}}{\alpha^2} = \alpha^4 + \alpha^2 = \alpha$$

and

$$e_5 = \delta_2 = \frac{Z_0(\alpha^{-5})}{\sigma'(\alpha^{-5})} = \frac{\alpha^6 + \alpha^{-5}}{\alpha^2} = \alpha^4 + 1 = \alpha^5$$

So,

$$e(X) = \alpha X^3 + \alpha^5 X^5$$