Implementation of systematic cyclic encoder:

$$c_3^+ = a_i + c_1^-$$

$$c_2^+ = c_3^-$$

$$c_1^+ = c_2^- + c_3^+$$

For [D]=[0101] with always zero initial states, then

a_i
0
1
0
1

	c_3	c_2	c_1
initial	0	0	0
Z=1 for K clock pulses	0	0	0
	1	0	1
	1	1	1
	0	1	1
Z=0 for r clock pulses	0	0	1
	0	0	0
	0	0	0

Then $c_1c_2c_3$=110 and [C]=[0101110]

K bits r bits

For [D]=[0010]

a_i
0
0
1
0

	c_3	c_2	C_1
initial	0	0	0
Z=1 for K clock pulses	0	0	0
	0	0	0
	1	0	1
	1	1	1
Z=0 for r clock pulses	0	1	1
	0	0	1
	0	0	0

Then $c_1c_2c_3$=111 and [C]=[0010111]

K bits r bits

At the receiver [R]=[C]+[E] where:
[R] is the received codeword
[C] is the transmitted codeword;
[E] is the error word
writing above in polynomial form:

$$R(x)=C(x)+E(x)$$

Dividing both sides by g(x) taking the remainder, then:

$$\mathrm{Rem}\frac{R(x)}{g(x)} = \mathrm{Rem}\frac{C(x)}{g(x)} + \mathrm{Rem}\frac{E(x)}{g(x)}$$ and since $\mathrm{Rem}\frac{C(x)}{g(x)} = 0$ from transmitter side,

then: $\boxed{\mathrm{Rem}\frac{R(x)}{g(x)} = \mathrm{Rem}\frac{E(x)}{g(x)} = s(x)}$ =syndrome polynomial of order (r-1).

1- if ***s(x)=0,*** then the receiver decides on no error.
2-***if s(x)≠0,*** then errors occur.

To find the location(s) of errors, the receiver may prepare a syndrome table and store it in its memory as a look up table, use it to find [E] from [s]. This look up syndrome table starts with most probable errors(less no of errors).

<u>Example:</u> Prepare the syndrome table for the (7,4) systematic cyclic code with $g(x)=x^3+x^2+1$ for single error

<u>Solution:</u> [G]=[1101]. Each [s] is found from [E] by using the equation

$$Rem\frac{E(x)}{g(x)} = s(x)$$

For example if [E]=[0100000] which corresponds to a single error at the 2nd position from the left, then:

```
1101 | 0100000
        1101
        01010
         1101
         01110
          1101
          0011
```

Hence [s]=[011], and so on.

Error word [E]							[s]		
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	1
0	0	0	0	0	1	0	0	1	0
0	0	0	0	1	0	0	1	0	0
0	0	0	1	0	0	0	1	0	1
0	0	1	0	0	0	0	1	1	1
0	1	0	0	0	0	0	0	1	1
1	0	0	0	0	0	0	1	1	0
0	0	0	0	0	1	1	0	1	1
0	0	0	0	1	0	1	1	0	1

No error (first row); One error (rows 2–8); 2 errors (rows 9–10)

<u>Note</u> 1)no repeated syndromes are observed for single error. 2)when you start to find the syndromes for double error the syndrome will be similar to the syndrome for single error case(because t=1)

<u>Example</u>: Using previous syndrome table, find the corrected word for the received word **[R]=[1011001].**

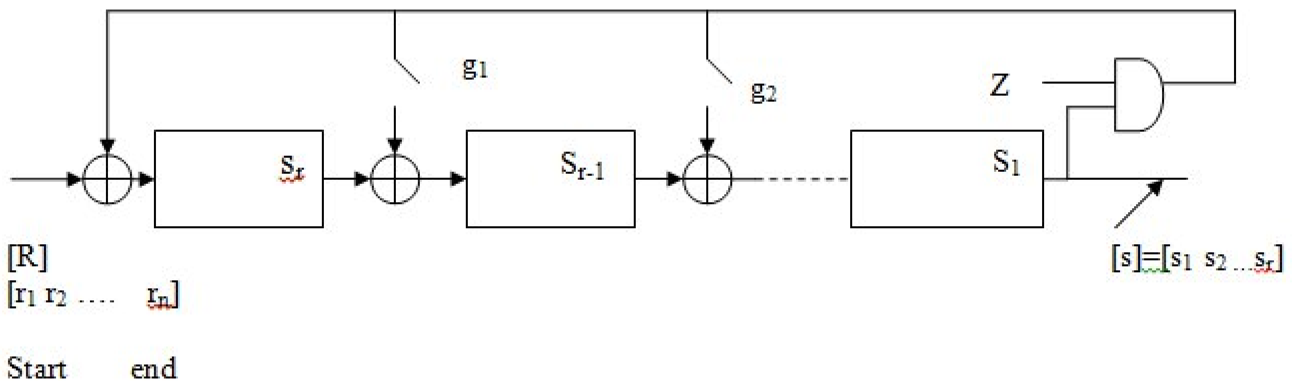
<u>Solution:</u> we find the syndrome from the equation below

$$s(x) = Rem\frac{R(x)}{g(x)}$$

```
1101 | 1011001
        1101
        01100
         1101
         000101
```

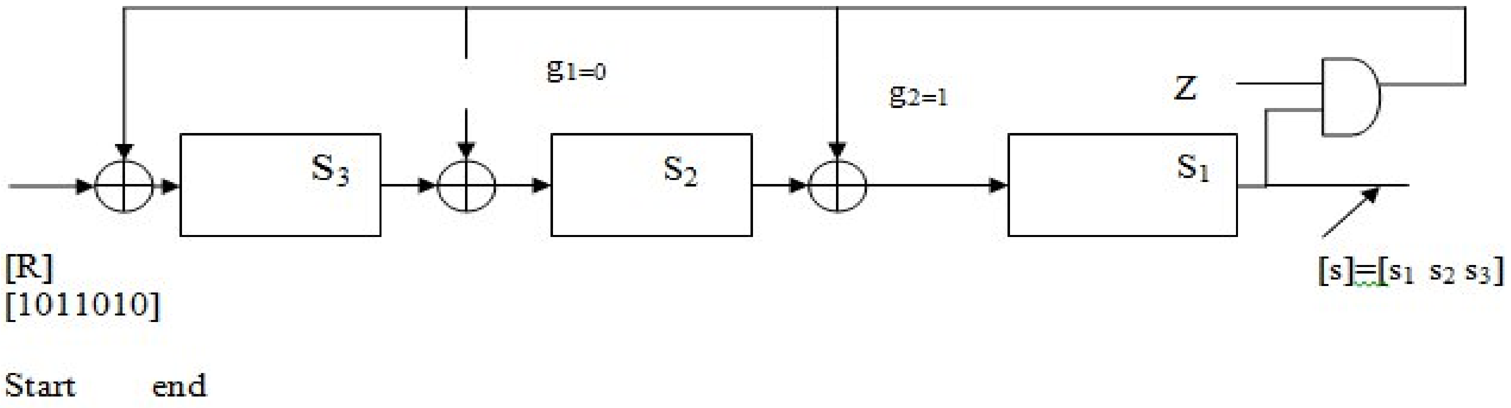
[s]=[101], using previous syndrome table for [s]=[101], then for single error [E]=[0001000], i.e. a single error at the 4th position from the left. Hence corrected word=[R]+[E] =[1011001]+[0001000]= [1010001].

The long division of R(x) by g(x) to obtain the remainder is implemented using a modular feedback shift register as shown.
The control Z is set (Z=1) for n clock pulses and reset (Z=0) for r clock pulses.



Example: Use the decoder circuit to find the syndrome and hence correct the received word [R]=[1011010] for generator polynomial $g(x)=x^3+x^2+1$.

Solution: Above circuit will be as shown for $g(x)=x^3+x^2+1$.



First we need to write the transition equations for s_1 s_2 s_3 when Z=1 :-

Implementation of systematic cyclic decoder:

When Z=1, the transition eqs for s will be:

$$s_3^+ = r_i + s_1^-$$

$$s_2^+ = s_3^-$$

$$s_1^+ = s_2^- + s_1^-$$ for zero initial states, then:

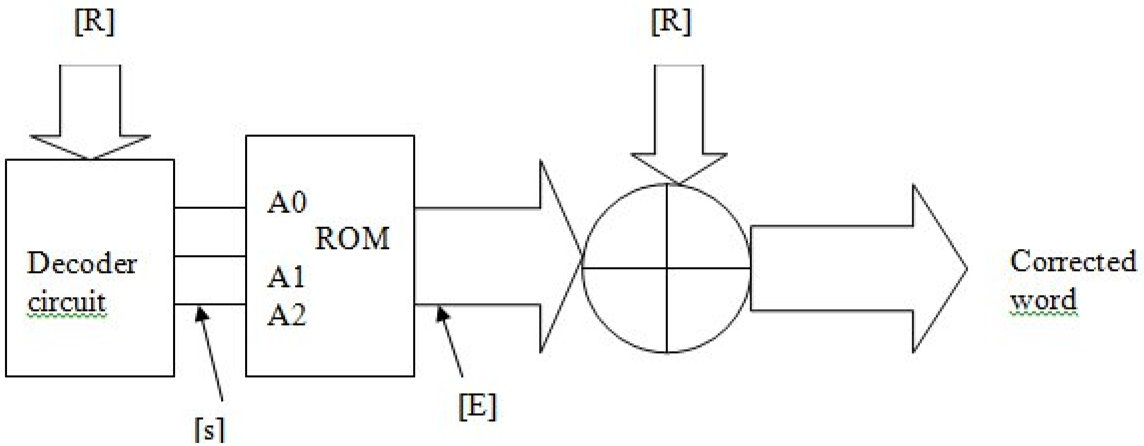
[R]
1
0
1
1
0
1
0

Z=1 for n clock pulses

s_3	s_2	s_1	
0	0	0	← Initial
1	0	0	
0	1	0	
1	0	1	
0	1	1	
1	0	0	
1	1	0	
0	1	1	

Then **[s]=[s_1 s_2 s_3]=[**110] and using previous syndrome table then:
[E]=[1000000] single error at the 1st position from the left, i.e. corrected word will be
[C]=[R]+[E]=[1011010]+[1000000]= [0011010].
Exercise : repeat previous example for [R]=[1110110].

The complete circuit diagram of the systematic cyclic decoder that includes the syndrome generator logic circuit and the look up table that stores the syndrome table will be as shown:



Questions?

Thank you for
Listening & Attending

EE426 INFORMATION THEORY