MODULE 4

A Few Important Classes of Algebraic codes

- **Decoding of Cyclic codes**
- **Hamming codes**
- **BCH and Reed Solomon Codes**

Anila Kuriakose,
Assistant Professor, Department of ECE, RSET

Contents

- Decoding of Cyclic codes
- Hamming codes
- BCH codes
- Reed Solomon codes

Decoding of Cyclic Codes

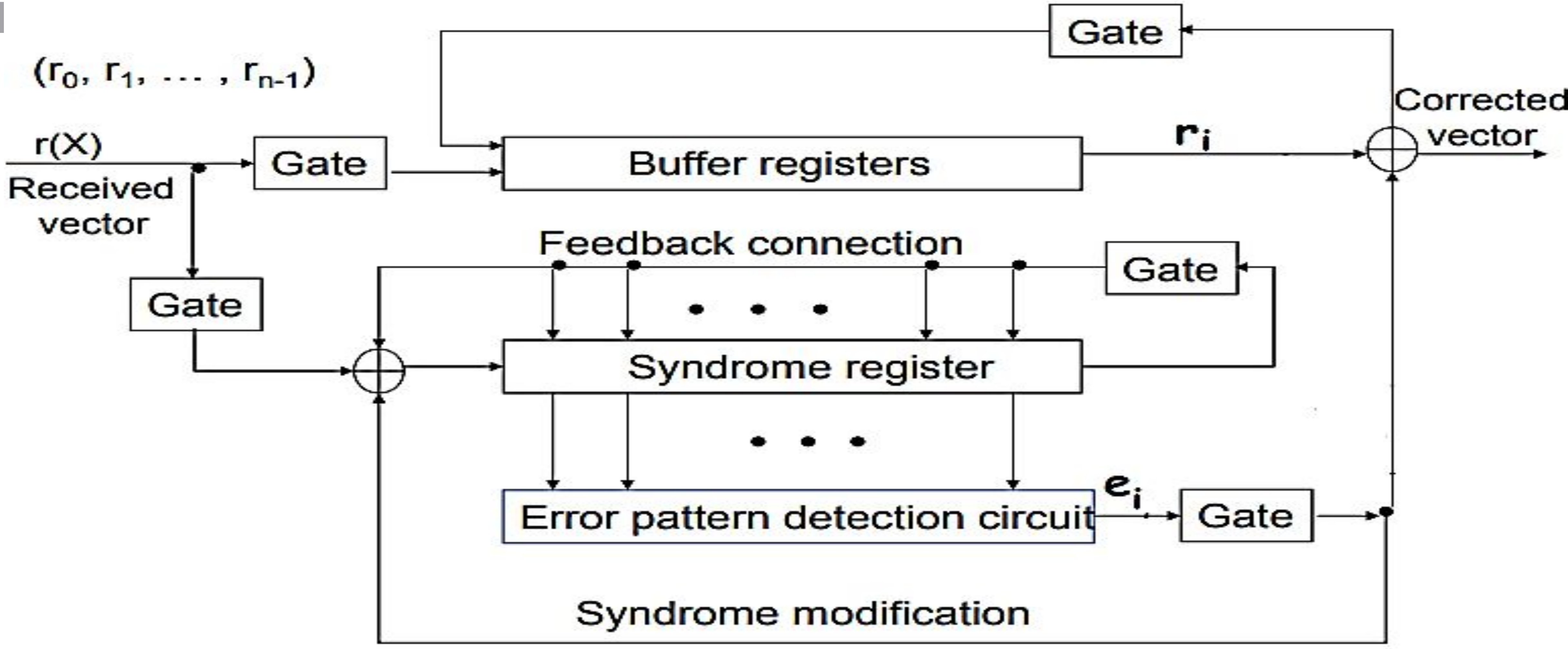
Decoding of Cyclic Codes

- Decoding of linear codes consists of three steps:

1) Syndrome computation
2) Association of the syndrome to an error pattern
3) Error correction.

- The decoder used is called Meggitt Decoder

Decoding of Cyclic Codes



Decoding of Cyclic Codes

Steps for Decoding:

- **Step 1 :** The syndrome is formed by shifting the entire received vector into the syndrome register. At the same time the received vector is stored into the buffer register.
- **Step 2 :** The syndrome is read into the detector and is tested for the corresponding error pattern.
  - The detector is a combinational circuit and its output is 1, iff the syndrome corresponds to a correctable error pattern with an error at the highest-order position $X^{(n-1)}$

Decoding of Cyclic Codes

Steps for Decoding:

- **Step 2 :** The syndrome is read into the detector and is tested for the corresponding error pattern.
  - If a "1" appears at the output of the detector, the received symbol in the rightmost stage of the buffer register is assumed to be erroneous and must be corrected
  - If a "0" appears at the output of the detector, the received symbol at the right most stage of the buffer register is assumed to be correct and no correction necessary

Decoding of Cyclic Codes

Steps for Decoding:

- **Step 3 :** The first received symbol is read out of the buffer
  - If the first received symbol is detected to be an erroneous symbol, it is corrected by the output of the detector
  - The output of the detector is fed back to the syndrome register to modify the syndrome
  - This results in a new syndrome, which corresponds to the altered received vector shifted one place to the right

Decoding of Cyclic Codes

Steps for Decoding:

- **Step 4:** The new syndrome formed in step 3 is used to detect whether or not the second received symbol is an erroneous symbol
  - The decoder repeats step 2 and 3
- **Step 5:** The decoder decodes the received vector symbol by symbol in the same manner until the entire received vector is read out of the buffer register

Problem 5.4

Q) Consider the decoding of the (7, 4) cyclic code generated by $g(X) = 1 + X + X^3$.

Solution:

- From the generator polynomial we have,
- $h(X) = \frac{X^n+1}{g(x)} = \frac{X^7+1}{1+X+X^3}$
- $h(X) = 1 + X + X^2 + X^4$
- $h(X^{-1}) = 1 + X^{-1} + X^{-2} + X^{-4}$
- $X^4 h(X^{-1}) = X^4 + X^3 + X^2 + 1$
- $X^5 h(X^{-1}) = X^5 + X^4 + X^3 + X$
- $X^6 h(X^{-1}) = X^6 + X^5 + X^4 + X^2$
- $H = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$

Problem 5.4

Q) Consider the decoding of the (7, 4) cyclic code generated by $g(X) = 1 + X + X^3$.

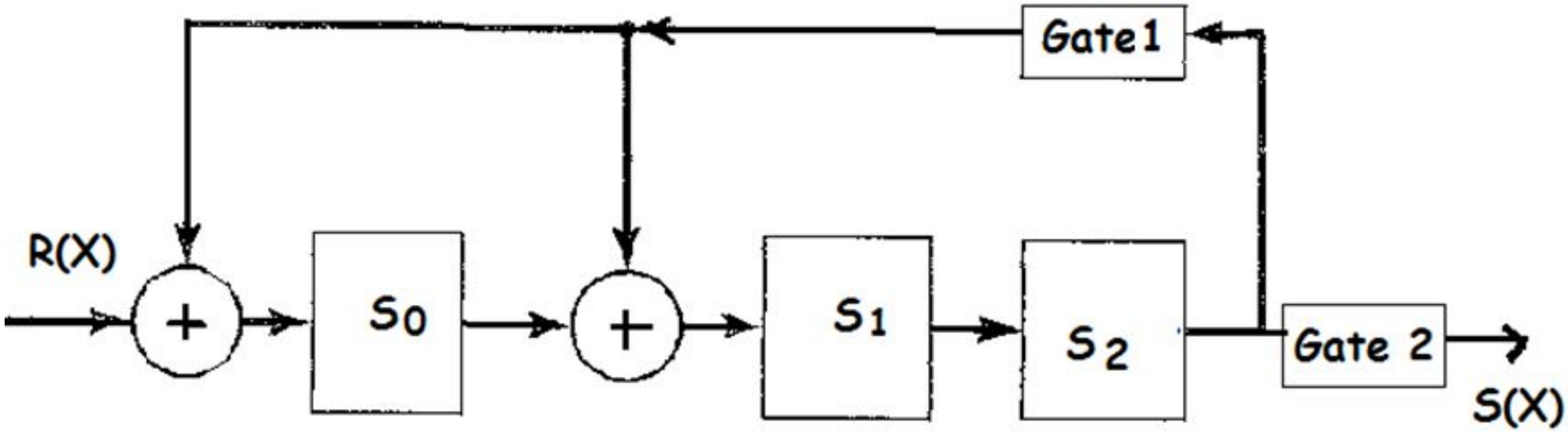
Solution:

- In systematic form
- 1^{st} row = 1^{st} + 3^{rd} row
- 2^{nd} & 3^{rd} row-no change
- $$\boldsymbol{H} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

$$\boldsymbol{H}^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

Problem 5.4

- Let R = (1 0 1 1 0 1 1)
- $g_1 = 1, g_2 = 0$; here shifted into syndrome register from left end.



Problem 5.4

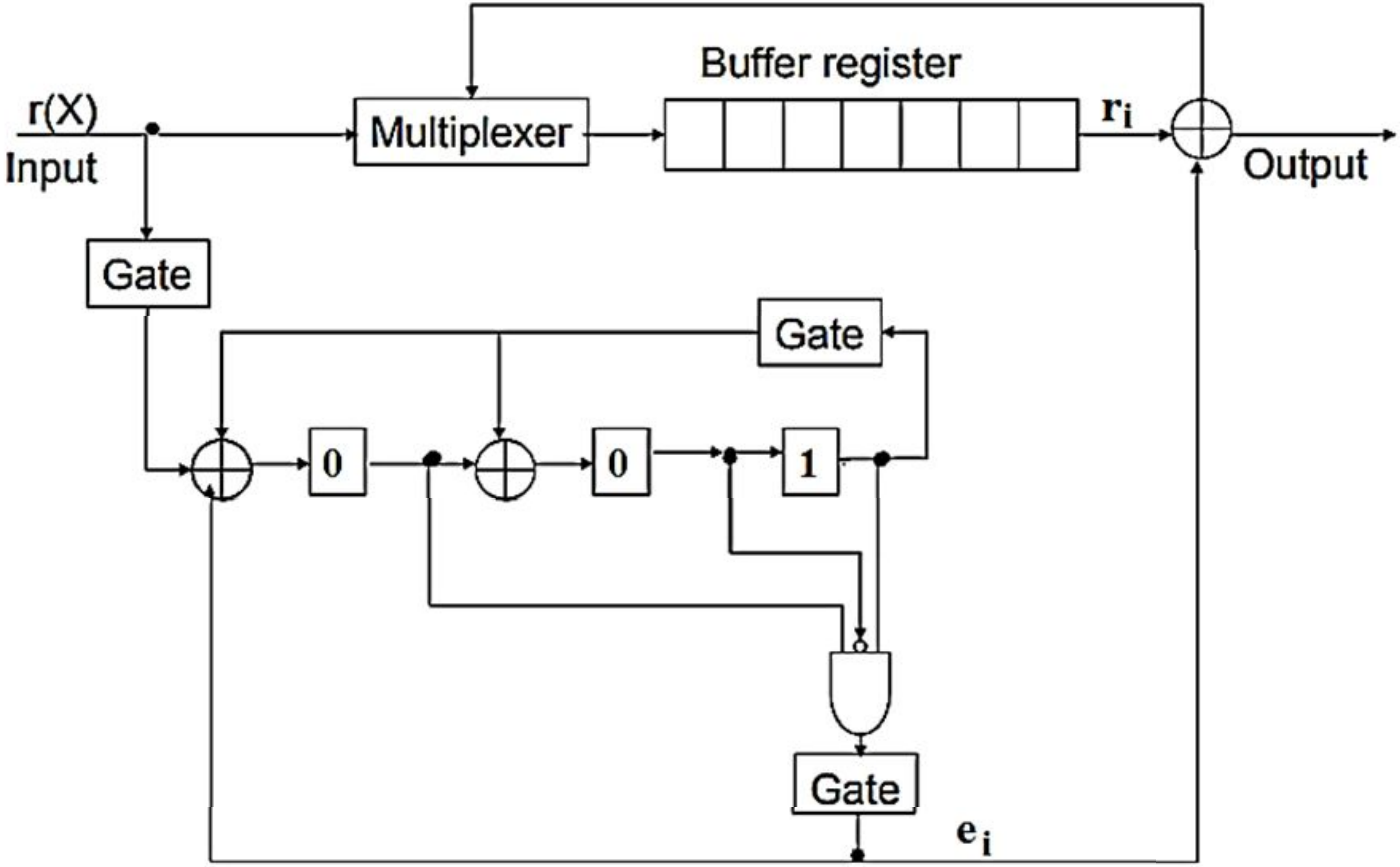
$$H = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

- This code has minimum distance 3 and is capable of correcting any single error. i.e. seven single-error patterns.

Error Vector	Error Pattern E(X)	Syndrome Vector	Syndrome S(X)
0000001		101	
0000010		111	
0000100		011	
0001000		110	
0010000		001	
0100000		010	
1000000		100	

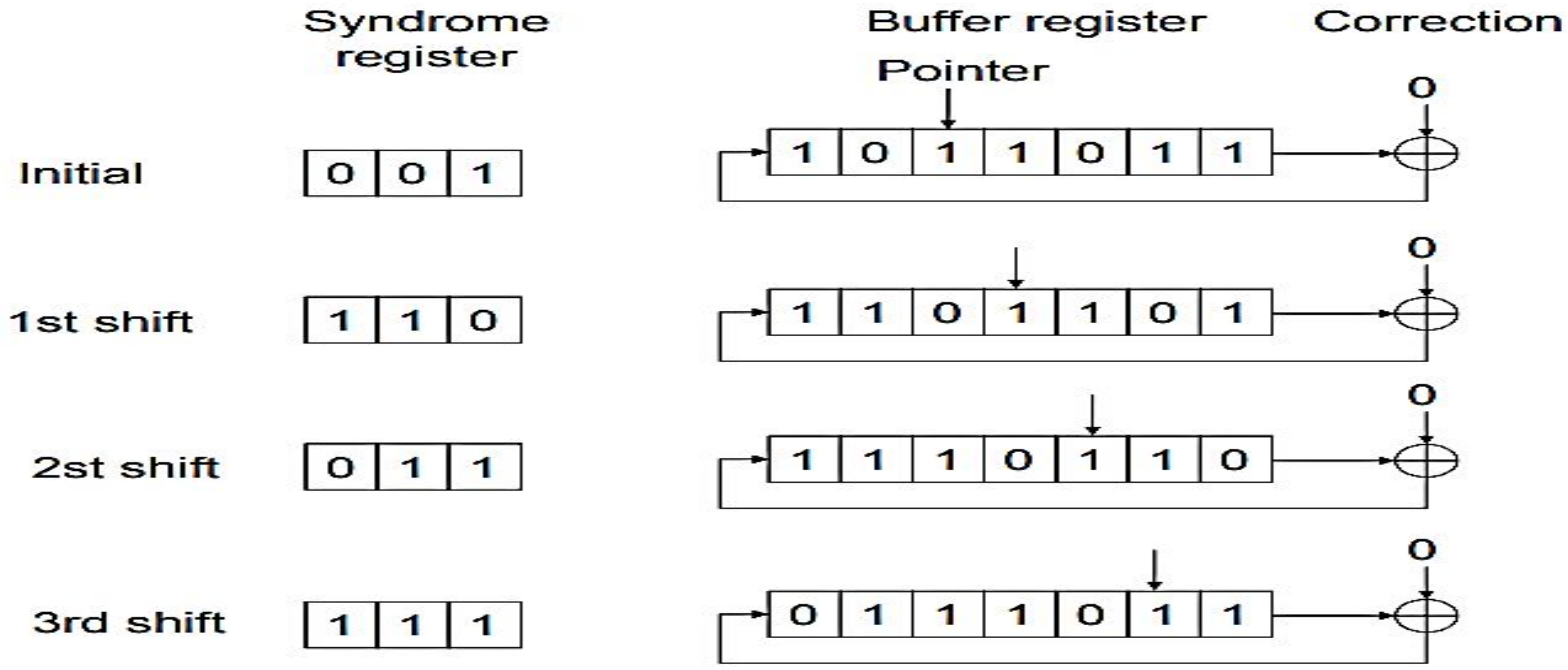
- R = (1 0 1 1 0 1 1) is received.
- S=001. Error is in the 3rd bit
- When e(X) = X^6 occurs, the syndrome is 101 and the detector should give an output 1.
- So the received code is shifted such that the error pattern appears at $X^{n-1} = X^6$

Received bits	Content of registers before shift			Content of registers after shift		
Y	s_0	s_1	s_2			
0	0	0	0	0	0	0
1	0	0	0	1	0	0
0	1	0	0	0	1	0
0	0	1	0	0	0	1
0	0	0	1	1	1	0
0	1	1	0	1	1	1
1	1	1	1	0	0	1

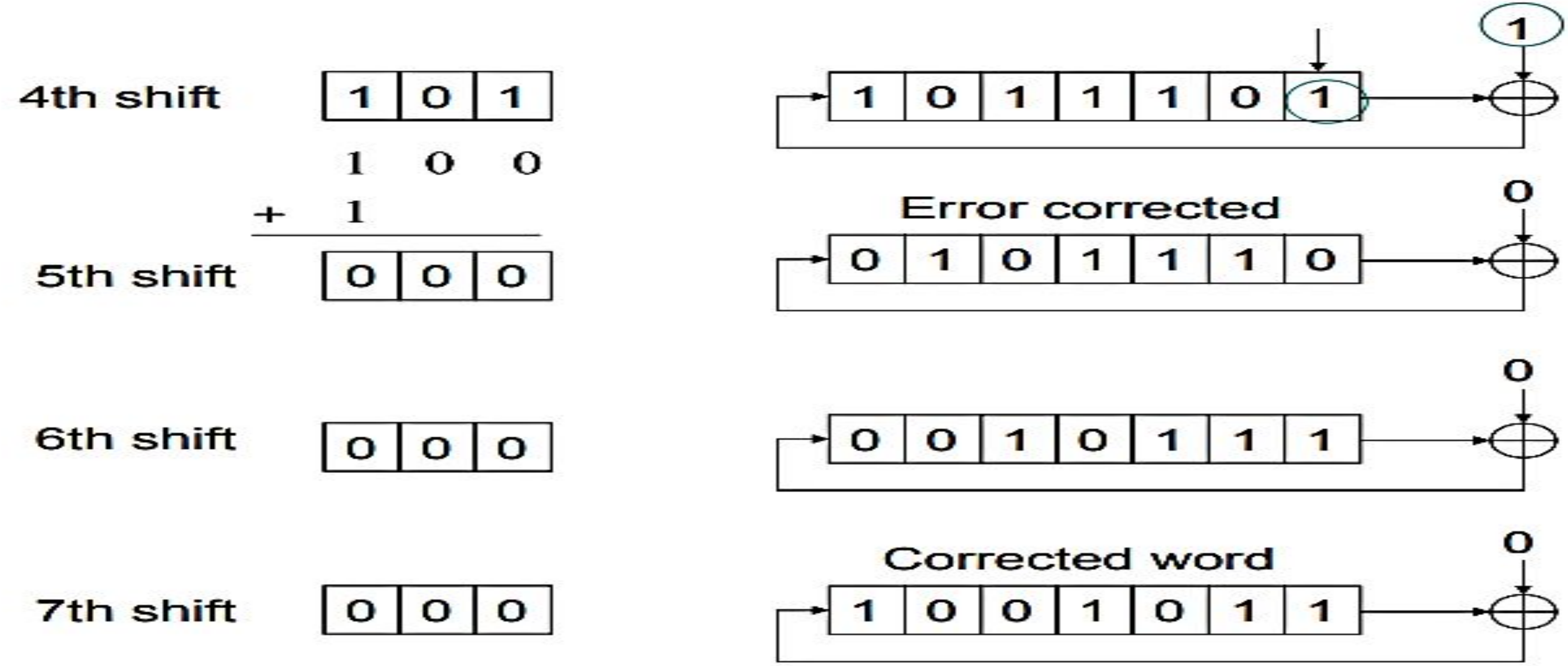


r(X)
Input
Multiplexer
Buffer register
r_i
Output
Gate
Gate
0
0
1
Gate
e_i

Problem 5.4


Syndrome register
Buffer register
Correction
Pointer
Initial
0 0 1
1 0 1 1 0 1 1
0
1st shift
1 1 0
1 1 0 1 1 0 1
0
2st shift
0 1 1
1 1 1 0 1 1 0
0
3rd shift
1 1 1
0 1 1 1 0 1 1
0

Problem 5.4


4th shift
1 0 1
1 0 0
+ 1
5th shift
0 0 0
6th shift
0 0 0
7th shift
0 0 0
1
1 0 1 1 1 0 1
Error corrected
0
0 1 0 1 1 1 0
0
0 0 1 0 1 1 1
Corrected word
0
1 0 0 1 0 1 1

Hamming codes

- First class of linear block codes devised for error correction.
- **Hamming codes** are perfect binary codes where $d_{min} = 3$.
- *Single error correcting (SEC)* Hamming codes are characterized by the following parameters.
  - Code length: $n = (2^m - 1)$
  - Number of Information symbols: $k = (2^m - m - 1)$
  - Number of parity check symbols: $(n - k) = m$
  - Error correcting capability: $t = 1, (d_{min} = 3)$

Hamming codes

- The parity check matrix ***H*** of this code consists of all the non-zero *m-tuples* as its columns. In systematic form:

$$H = [Q \vdots I_m]$$

  - Where I_m is an identity (unit) matrix of order $m \times m$ and
  - Q matrix consists of $(2^m\text{-}m\text{-}1)$ columns which are the m-tuples of weight 2 or more.

- Linear block code for which the error-correcting capability $t = 1$ is called a Hamming code.

Hamming codes - Illustration

- Consider m = 3 (parity check symbols)
- (n, k) Hamming Code →
  - Yielding the (7, 4) Hamming code with n = 7 and k = 4.
- For the (7, 4) linear systematic Hamming code is
- $H = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \vdots & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$
- The generator matrix of the code can be written in the form:

$$G = [I_{2^m - m - 1} \vdots Q^T]$$

Hamming codes - Illustration

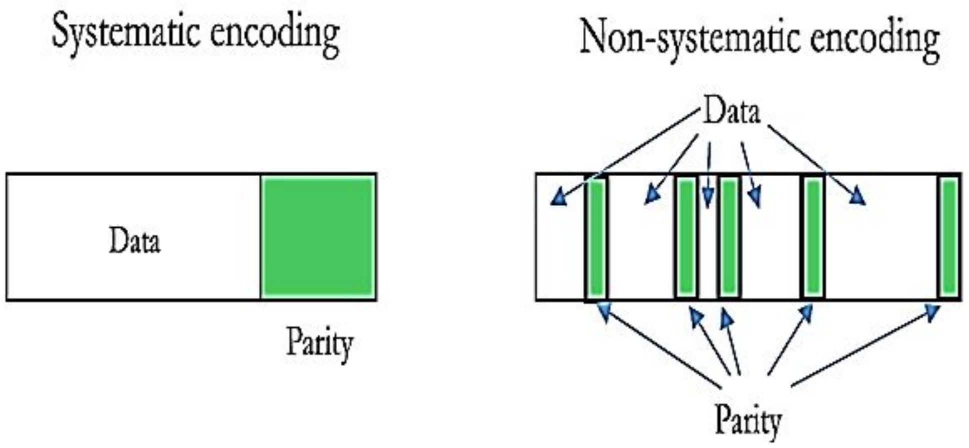
- For the (7, 4) systematic code, G is

$$G = \left[\begin{array}{cccc:ccc} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right]$$

- Note: *For (7, 4), (15, 11), (31, 26), (63, 57) are all single error correcting Hamming codes and are regarded quite useful*.

Systematic & Non-systematic encoding

- Block codes like Hamming codes are also classified into two categories that differ in terms of structure of the encoder output:
- Systematic encoding
  - Just by seeing the output of an encoder, we can separate the data and the redundant bits (also called parity bits).
- Non-systematic encoding
  - The redundant bits and data bits are interspersed.



Non Systematic Hamming Codes

- **Simple method:** A non systematic code can be constructed by placing check bits at $2^l, l=0,1,2\ldots$ locations of G *matrix*.
- Conventional method of construction in switching & computer applications.
- Procedure:
  1) Write BCD of length $(n-k)$ for decimals from *1 to n*.
  2) Arrange the sequence in the reverse order to form H^T.
  3) Transpose gives **H** matrix.
  4) Parity matrix ***P*** can be formed from ***H***.
  5) **G** matrix is formed by placing message bits at locations other than 2^l and parity bits at locations 2^l.

Non Systematic Hamming Codes

- The code word are in the form:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

  - Where $p_1, p_2, \dots$ are parity digits & $m_1, m_2, \dots$ are message digits.
- Parity check bit from H matrix position can be:
  - p_1 = 1, 3, 5, 7, 9, 11, 13, 15…
  - p_2 = 2, 3, 6, 7, 10, 11, 14, 15 …
  - p_3 = 4, 5, 6, 7, 12, 13, 14, 15…

Eg. – Non Systematic Hamming Codes (7, 4)

- Step 1: Write BCD of length $(n - k)$ for decimals from 1 to n

Number	BCD of length (n – k)
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Eg. – Non Systematic Hamming Codes (7, 4)

- Step 2: Arrange the sequence in the reverse order to form $\boldsymbol{H^T}$.

Number	BCD of length (n – k)
1	001
2	010
3	011
4	100
5	101
6	110
7	111

$$H^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Eg. – Non Systematic Hamming Codes (7, 4)

- Step 3: Transpose gives **H** matrix.

$$H^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

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

Eg. – Non Systematic Hamming Codes (7, 4)

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

- Q *sub-matrix* in the ***H matrix*** can be identified to contain those columns which have weights more than one.
- Transpose of this matrix then gives the columns to be filled in G-matrix.
- E.g. (7, 4) linear code, Q-sub matrix is

- $$Q = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \;\&\; hence\; Q^T = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

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

- First 2 columns of this matrix are in 2 columns of G-matrix & 3rd column in 4th column of G-matrix

Eg. – Non Systematic Hamming Codes (7, 4)

- Code construction from H-matrix which is unique and hence the codes are also unique.
- Consider the correctable error patterns and corresponding syndromes.

Messages				Codes						
m1	m2	m3	m4	p1	p2	m1	p3	m2	m3	m4
0	0	0	1	1	1	0	1	0	0	1
0	0	1	0	0	1	0	1	0	1	0
0	0	1	1	1	0	0	0	0	1	1
0	1	0	0	1	0	0	1	1	0	0
0	1	0	1	0	1	0	0	1	0	1
0	1	1	0	1	1	0	0	1	1	0
0	1	1	1	0	0	0	0	1	1	1
1	0	0	0	1	1	1	0	0	0	0

Eg. – Non Systematic Hamming Codes (7, 4)

- Code construction from H-matrix which is unique and hence the codes are also unique.
- Consider the correctable error patterns and corresponding syndromes.

Messages				Codes						
m1	m2	m3	m4	p1	p2	m1	p3	m2	m3	m4
1	0	0	1	0	0	1	1	0	0	1
1	0	1	0	1	0	1	1	0	1	0
1	0	1	1	0	1	1	0	0	1	1
1	1	0	0	0	1	1	1	1	0	0
1	1	0	1	1	0	1	0	1	0	1
1	1	1	0	0	0	1	0	1	1	0
1	1	1	1	1	1	1	1	1	1	1

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

...atic Hamming Codes (7, 4)

Table for Error patterns & Syndromes for (7, 4) linear non-systematic code.

- If the syndrome is read from right to left, it is observed that decimal equivalent of this binary sequence corresponds to the error location.

Error pattern							**Syndrome**		
e_1	e_2	e_3	e_4	e_5	e_6	e_7	s_1	s_2	s_3
1	0	0	0	0	0	0	1	0	0
0	1	0	0	0	0	0	0	1	0
0	0	1	0	0	0	0	1	1	0
0	0	0	1	0	0	0	0	0	1
0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	1	0	0	1	1
0	0	0	0	0	0	1	1	1	1

Bose- Chaudhury – Hocquenghem Codes (Binary BCH Codes)

- most important and powerful error-correcting cyclic codes known.
- This class of codes is a remarkable generalization of the Hamming code for multiple-error correction.
- For any positive integer $m \geq 3$, and $t < \frac{2^m - 1}{2}$, there exists a binary ***BCH*** code (called the '***primitive***' ***BCH code***) with the following parameters:
  - ***Block length :*** $n = 2^m - l$
  - ***Number of message bits :*** $k \leq n - mt$
  - ***Minimum distance :*** $d_{min} \geq 2t + 1$

BCH Codes

- ***BCH*** codes are "***t - error correcting codes***". They can detect and correct up to'***t***' random errors per code word.
- The parameters of some useful ***BCH*** codes are given below. Also indicated in the table are the generator polynomials for block lengths up to ***31***.

n	*k*	*t*	*Generator Polynomial*
7	4	1	1 011
15	11	1	10 011
15	7	2	111 010 001
15	5	3	10 100 110 111
31	26	1	100 101
31	21	2	11 101 101 001
31	16	3	1 000 111 110 101 111
31	11	5	101 100 010 011 011 010 101
31	6	7	11 001 011 011 110 101 000 100 111

$g(X) = 1 + X^4 + X^6 + X^7 + X^8$

BCH Codes

- The generator polynomial of the ***t***-error correcting ***BCH*** code is the least common multiple (***LCM***) of $\boldsymbol{M_1(x)}, \boldsymbol{M_2(x)}, \ldots\ \boldsymbol{M_{2t}(x)}$ where $\boldsymbol{M_i(x)}$ is the minimum polynomial of $\propto^i$, ***where i = 1, 2…2t.***
- There are several iterative procedures available for decoding of ***BCH*** codes.
- Majority of them can be programmed on a general purpose digital computer, which in many practical applications form an integral part of data communication networks.

Reed Solomon Codes (RS Codes)

- important sub class of ***BCH*** codes non binary BCH codes
- The encoder for an RS code differs from a binary encoder in that it operates on multiple bits rather than individual bits.
- A '***t***'-error correcting ***RS*** code has the following parameters.
  - ***Block length: n = (q - 1) symbols***
  - ***Number of parity Check symbols: r = (n - k) = 2t***
  - ***Minimum distance: dmin = (2t + 1)***
- The encoder for an ***RS*** (***n***, ***k***) code on ***m***-bit symbols groups the incoming binary data stream into blocks, each ***km*** bits long.

RS Codes

- Each block is treated as ***k*** symbols, with each symbol having ***m***-bits. The encoding algorithm expands a block of ***k*** symbols by adding (***n* - *k***) redundant symbols.
- Notice that no (***n*, *k***) linear block code can have ***dmin* > (*n* - *k* + *1*)**.
- For the RS code the block length is one less than the size of a code symbol and minimum distance is one greater than the number of parity symbols - - "*The **dmin** is always equal to the design distance of the code*“
- An (***n*, *k***) linear block code for which ***dmin* = (*n-k-l*)** is called 'Maximum - distance separable‘ code.

Advantages of RS codes

- Every ***RS*** code is '*maximum - distance separable*' code-They make highly efficient use of redundancy and can be adjusted to accommodate wide range of message sizes.
- They provide wide range of code rates (***k*** / ***n***) that can be chosen to optimize performance.
- Further, efficient decoding techniques are available for use with ***Reed Solomon*** codes.