ELEC 405
Error Control Coding and Sequences

Cyclic Codes

Definition

- A code C is cyclic if
  1) C is linear
  2) a cyclic shift of any codeword

$$\mathbf{c}_i = (c_0, c_1, \cdots, c_{n-1})$$

is another codeword

$$\mathbf{c}_j = (c_{n-1}, c_0, c_1, \cdots, c_{n-2})$$

- Examples:
  – C = {000, 101, 011, 110}
  – C = {000,111}

Another Example

- C = {0000,1001,0110,1111} is not cyclic

- Interchange positions 3 and 4

  (equivalent code)

- C = {0000,1010,0101,1111} is cyclic

- Code polynomials

$$c(x) = c_0 + c_1 x + \cdots + c_{n-1} x^{n-1}, \quad c_i \in \mathrm{GF}(q)$$

- GF(*q*)[*x*] is the set of polynomials with coefficients from GF(*q*)
- GF(*q*)[*x*] is a commutative ring with identity (not a field)

- Consider polynomials modulo $f(x)$ of degree n

  $GF(q)[x]/f(x)$

- This is the finite ring of polynomials modulo $f(x)$
- Example: $GF(2)[x]/x^2+x+1 \longrightarrow GF(4)$

$+$	1	x	$x+1$	0
1	0	$x+1$	x	1
x	$x+1$	0	1	x
$x+1$	x	1	0	$x+1$
0	1	x	$x+1$	0

$\cdot$	1	x	$x+1$
1	1	x	$x+1$
x	x	$x+1$	1
$x+1$	$x+1$	1	x

- Choose $f(x)=x^2+1$ in GF(2)

$+$	1	x	$1+x$	0
1	0	$1+x$	x	1
x	$1+x$	0	1	x
$1+x$	x	1	0	$1+x$
0	1	x	$1+x$	0

$\cdot$	1	x	$1+x$
1	1	x	$1+x$
x	x	1	$1+x$
$1+x$	$1+x$	$1+x$	0

Zero divisor

x^2+1 is not irreducible

- Over any field

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

- Let R_n denote GF(q)[x]/x^n-1
- Any polynomial of degree ≥ n can be reduced modulo x^n-1 to a polynomial of degree less than n

$$x^n \rightarrow 1$$

$$x^{n+1} \rightarrow x$$

$$x^{n+2} \rightarrow x^2$$

Ideals

- Let R be a ring. A nonempty subset $I \subseteq R$ is called an Ideal if it satisfies the following
  - I forms a group under addition
  - $a \cdot b \in I$ for all $a \in I$ and $b \in R$
    - superclosed under multiplication
- Examples
  - {0} and R are trivial Ideals in R
  - $\{0, x^4+x^3+x^2+x+1\}$ is an Ideal in $GF(2)[x]/x^5-1$
  - Even numbers in Z

Ideal Example

- GF(2)[x]/x^3-1

$$
\begin{array}{llll}
0 & \rightarrow 000 & 1 & \rightarrow 100 \\
x & \rightarrow 010 & 1+x & \rightarrow 110 \\
x^2 & \rightarrow 001 & 1+x^2 & \rightarrow 101 \\
x+x^2 & \rightarrow 011 & 1+x+x^2 & \rightarrow 111
\end{array}
$$

$I = \{0, 1+x, 1+x^2, x+x^2\}$ is an Ideal in R_3

{000, 110, 101, 011} is a cyclic code

Theorem 5-1

A code which is a vector subspace over a field GF(q) is a cyclic code iff it corresponds to an ideal in GF(q)[x]/x^n-1 (the ring of polynomials modulo x^n-1)

Generator Polynomial

- Let $f(x)$ be any polynomial in R_n and let $< f(x) >$ denote the subset of R_n consisting of all multiples of $f(x)$ modulo x^n-1

$$< f(x) >= \{r(x)f(x) \mid r(x) \in R_n\}$$

- $< f(x) >$ is the cyclic code generated by $f(x)$
- Example: $C = < 1+x^2 >$ in R_3 (GF(2)[x]/x^3-1)
  - Multiplying by all 8 elements in R_3 produces only 4 distinct codewords

$$C=\{0,1+x,1+x^2,x+x^2\}$$

Generator Polynomial

- Any cyclic code can be generated by a polynomial from R_n
- Let C be a cyclic code in R_n. Then we have the following facts:
  1. There exists a unique monic polynomial $g(x)$ of smallest degree in C
  2. $C = < g(x) >$
  3. $g(x) \mid x^n - 1$

  $g(x)$ is called the generator polynomial

- Any polynomial $c(x)$ of degree less than n is in C iff $g(x)|c(x)$
- If $g(x)$ has degree n-k, $|C|=q^k$, and every codeword is of the form

$$c(x) = m(x) \cdot g(x)$$

Codeword polynomial of degree n-1 or less

Message polynomial of degree k-1 or less

Generator polynomial of degree n-k

- To determine the possible $g(x)$, factor x^n-1
- Example:

$$x^3-1 = (x+1)(x^2+x+1) \text{ over } GF(2)$$

Generator polynomial	Code in R_3	Code in 3-tuples
1	R_3	V_3
$x+1$	$\{0,1+x,1+x^2,x+x^2\}$	{000,110,101,011}
x^2+x+1	$\{0,1+x+x^2\}$	{000,111}
x^3-1	$\{0\}$	{000}

Generator Matrix

- Since $c(x) = m(x)g(x) = (m_0 + m_1 x + \cdots + m_{k-1}x^{k-1})g(x)$

$$= m_0 g(x) + m_1 x g(x) + \cdots + m_{k-1}x^{k-1}g(x)$$

$$= [m_0 \;\; m_1 \;\; \cdots \;\; m_{k-1}] \begin{bmatrix} g(x) \\ xg(x) \\ \vdots \\ x^{k-1}g(x) \end{bmatrix} = \mathbf{mG}$$

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & \cdots & g_{n-k} & & & & \mathbf{0} \\ & g_0 & g_1 & \cdots & g_{n-k} & & & \\ & & \ddots & \ddots & & \ddots & & \\ & & & g_0 & g_1 & \cdots & g_{n-k} & \\ \mathbf{0} & & & & g_0 & g_1 & \cdots & g_{n-k} \end{bmatrix}$$

is a Generator matrix

Generator Matrix Example

- GF(2)[x]/ x^7-1
- $x^7-1 = (1+x+x^3)(1+x^2+x^3)(1+x)$
- $g(x) = 1+x+x^3$

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

- C is a (7,4,3) code – a binary cyclic code
- All binary cyclic codes with $g(x)$ a primitive polynomial are equivalent to Hamming codes

Example 5-1

- $g(x) = (1+x+x^3)(1+x) = 1+x^2+x^3+x^4$

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

- C is a (7,3,4) binary cyclic code

Parity Check Matrix

- The generator matrix is not in systematic form. How to find the parity check matrix?
- $g(x)$ is a factor of x^n-1, i.e., $g(x)h(x) = x^n-1$
- $h(x)$ is a monic polynomial with degree k, and is the generator polynomial of a cyclic code C', but not necessarily the dual code of C.
- (7,4,3) code example:

$$h(x) = (1+x^2+x^3)(1+x) = 1+x+x^2+x^4$$

- $g(x)h(x)=0 \bmod x^n-1$ in R_n is not the same as vectors in V_n being orthogonal.
- Let **H** be the matrix generated from

$h^*(x)=x^k h(x^{-1})=h_k+xh_{k-1}+\ldots+x^k h_0$ reciprocal poly. of $h(x)$

$$\mathbf{H} = \begin{bmatrix} h_k & \cdots & h_1 & h_0 & & & & \mathbf{0} \\ & h_k & \cdots & h_1 & h_0 & & & \\ & & \ddots & \ddots & \ddots & \ddots & & \\ & & & h_k & \cdots & h_1 & h_0 & \\ \mathbf{0} & & & & h_k & \cdots & h_1 & h_0 \end{bmatrix}$$

Parity Check Matrix **H**

- $c(x)h(x) = m(x)g(x)h(x) = m(x)(x^n\text{-}1) = m(x) + x^n m(x)$
- $m(x)$ has degree < k, thus the coefficients of x^k to x^{n-1} in $c(x)h(x)$ must be zero

$$
\begin{aligned}
&c_0 h_k + c_1 h_{k-1} + \cdots + c_k h_0 = 0 \\
&c_1 h_k + c_2 h_{k-1} + \cdots + c_{k+1} h_0 = 0 \qquad \Rightarrow \quad \mathbf{cH}^T = 0 \\
&\vdots \\
&c_{n-k-1} h_k + c_{n-k} h_{k-1} + \cdots + c_{n-1} h_0 = 0
\end{aligned}
$$

Hamming Code Example (Cont.)

- $h^*(x)$ =$1+x^2+x^3+x^4$ generates the parity check matrix of $g(x)$ and the dual cyclic code of $g(x)$

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

- **H** is the parity check matrix for the (7,4,3) Hamming code
- $h^*(x)=1+x^2+x^3+x^4$ is the generator polynomial for a (7,3,4) cyclic code since $h^*(x) \mid x^n-1$

Example 5.1 (Cont.)

- To construct the parity check matrix for the (7,3,4) code, use $h(\mathrm{x}) = 1+x^2+x^3$
- $h^*(x) = 1+x+x^3$ is the generator polynomial for a (7,4,3) code since $h^*(x)|x^n-1$
- $h^*(x)$ generates the parity check matrix **H** of $g(x)$ and the dual cyclic code of $g(x)$ with parameters (7,4,3)

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

Binary Cyclic Codes of Length 7

- x^7-1=(1+x+x^3)(1+x^2+x^3)(1+x)

- 1+x (7,6,2)

  dual code 1+x+x^2+x^3+x^4+x^5+x^6 (7,1,7)
- 1+x+x^3 (7,4,3)

  dual code 1+x^2+x^3+x^4 (7,3,4)
- 1+x^2+x^3 (7,4,3)

  dual code 1+x+x^2+x^4 (7,3,4)

Systematic Cyclic Codes

- GF(2)[x]/ x^7-1
- $x^7-1 = (1+x+x^3)(1+x^2+x^3)(1+x)$
- $g(x) = 1+x+x^3$

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

- C is a (7,4,3) code – not in systematic form
- To transform: permute columns 1 and 4, then add rows 2 and 4 to get a new row 4.

Systematic Generator Matrix

- Permute columns 1 and 4, then add rows 2 and 4 to get a new row 4. The resulting generator matrix has a systematic form, but is not cyclic.

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

- Check: divide the last row of **G**’ by g(x)
- $c'(x) = 1+x+x^2+x^6$ is not divisible by $g(x) = 1+x+x^3$

- We require an algebraic means of generating a systematic code while preserving divisibility by $g(x)$.
- Approach: divide x^{i} by $g(x)$, $i = n\text{-}k$ to n-1

  $x^{i} = g(x)q_{i}(x)+p_{i}(x)$ $\quad$ $p_{i}(x)$ has degree less than $n\text{-}k$

  rearranging $x^{i} - p_{i}(x) = g(x)q_{i}(x)$ $\quad$ divisible by $g(x)$
- $x^{i} - p_{i}(x)$ has only one non-zero coefficient for degrees $n\text{-}k$ to n-1
- Use $x^{i} - p_{i}(x)$ to form **G**

$$\mathbf{G} = [-\mathbf{P} \;\; \mathbf{I}_{k}] \qquad \mathbf{H} = [\mathbf{I}_{n-k} \;\; \mathbf{P}^{T}]$$

Example

- $g(x) = 1+x+x^3$

x^i	$g(x)q_i(x)$	$p_i(x)$	$x^i + p_i(x)$
x^3	$(1+x+x^3)\cdot 1$	$1+x$	$1+x+x^3$
x^4	$(1+x+x^3)\cdot x$	$x+x^2$	$x+x^2+x^4$
x^5	$(1+x+x^3)\cdot(1+x^2)$	$1+x+x^2$	$1+x+x^2+x^5$
x^6	$(1+x+x^3)\cdot(1+x+x^3)$	$1+x^2$	$1+x^2+x^6$

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

Systematic Encoding

- Encoding is now achieved by multiplying $m(x)$ by x^{n-k} and dividing the product by $g(x)$ to obtain $p(x)$

- $c(x) = m(x)x^{n-k} + m(x)x^{n-k}/g(x)$

- Example (7,4,3) code
  $m(x) = x^2+x+1$
  $m(x)x^{n-k} = x^5+x^4+x^3$ divide by $g(x) = x^3+x+1 \longrightarrow p(x) = x$
  $c(x) = x^5+x^4+x^3+x$

Implementation of Cyclic Codes

- Encoding
  - in non-systematic form $c(x) = m(x)g(x)$
  - in systematic form $c(x) = m(x)x^{n-k}+p(x)$

    $p(x) = m(x)x^{n-k} \bmod g(x)$

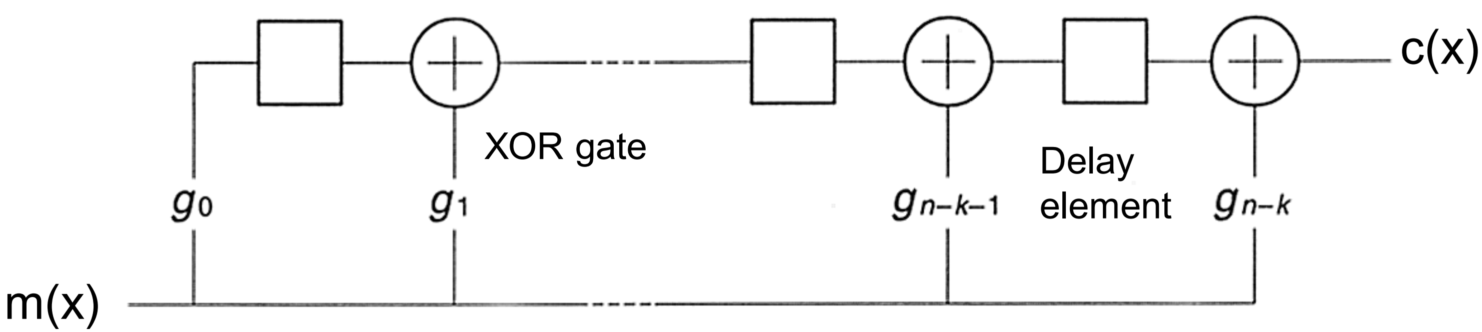
- Thus we require circuits for multiplying and dividing in R_n
- Solution: use shift registers

Nonsystematic Binary Cyclic Code Encoder

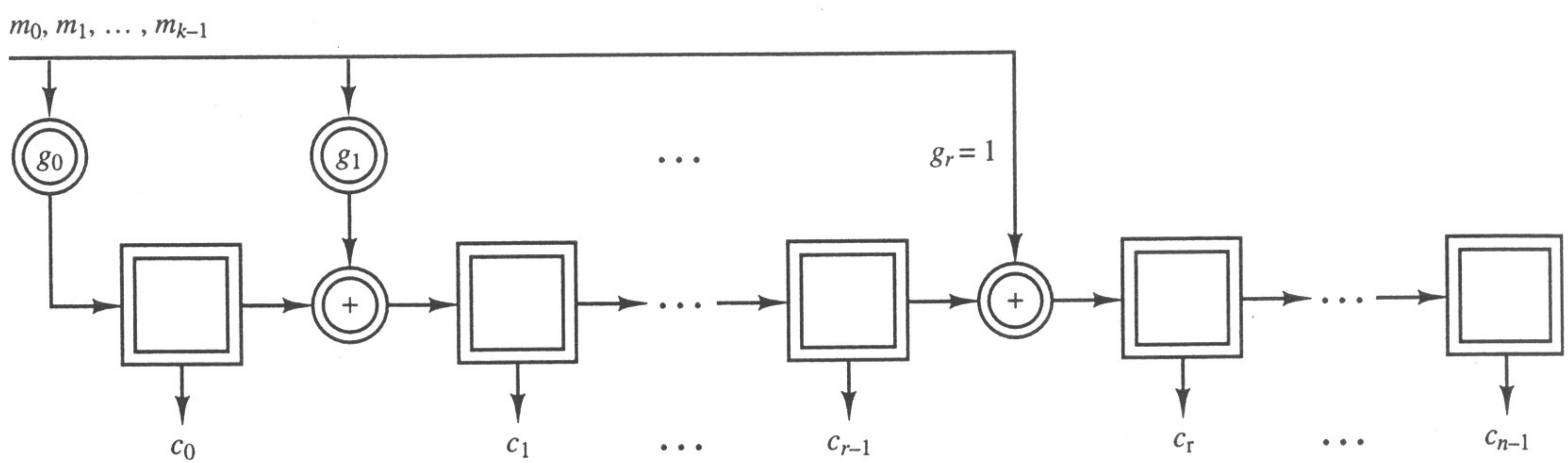
- Encoding can be done by multiplying two polynomials
  - a message polynomial m(x) and the generator polynomial g(x)
- The generator polynomial is

$$g(x) = g_0 + g_1 x + \dots + g_r x^r \qquad \text{of degree } r = n - k$$

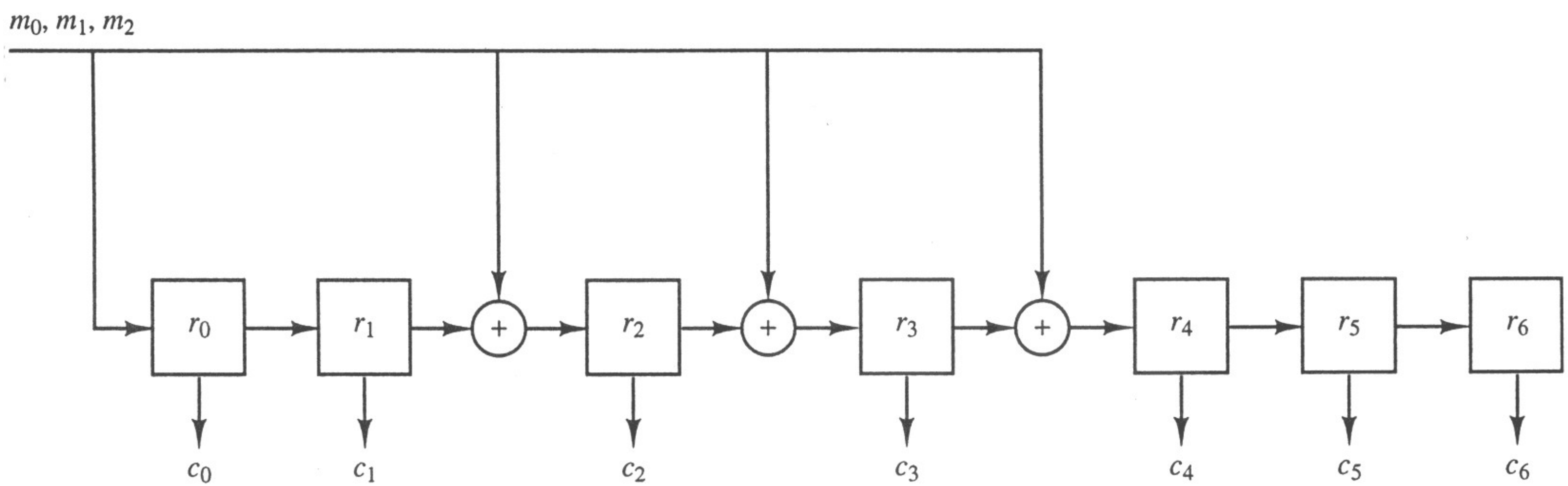
- If a message vector m is represented by a polynomial $m(x)$ of degree k-1, m(x) is encoded as $c(x) = m(x)g(x)$ using the following shift register circuit



Nonsystematic Shift Register Encoder



Encoder for the (7,3) Cyclic Code with $g(x) = 1+x^2+x^3+x^4$



SR cells	r_0	r_1	r_2	r_3	r_4	r_5	r_6
Initial state	0	0	0	0	0	0	0
Input $m_2 = 1$	1	0	1	1	1	0	0
Input $m_1 = 0$	0	1	0	1	1	1	0
Input $m_0 = 1$	1	0	0	1	0	1	1
Final state = $\mathbf{c}_4$	1	0	0	1	0	1	1

Figure 5-7. Shift-Register Cell Contents During Encoding of $m(x) = x^2 + 1$

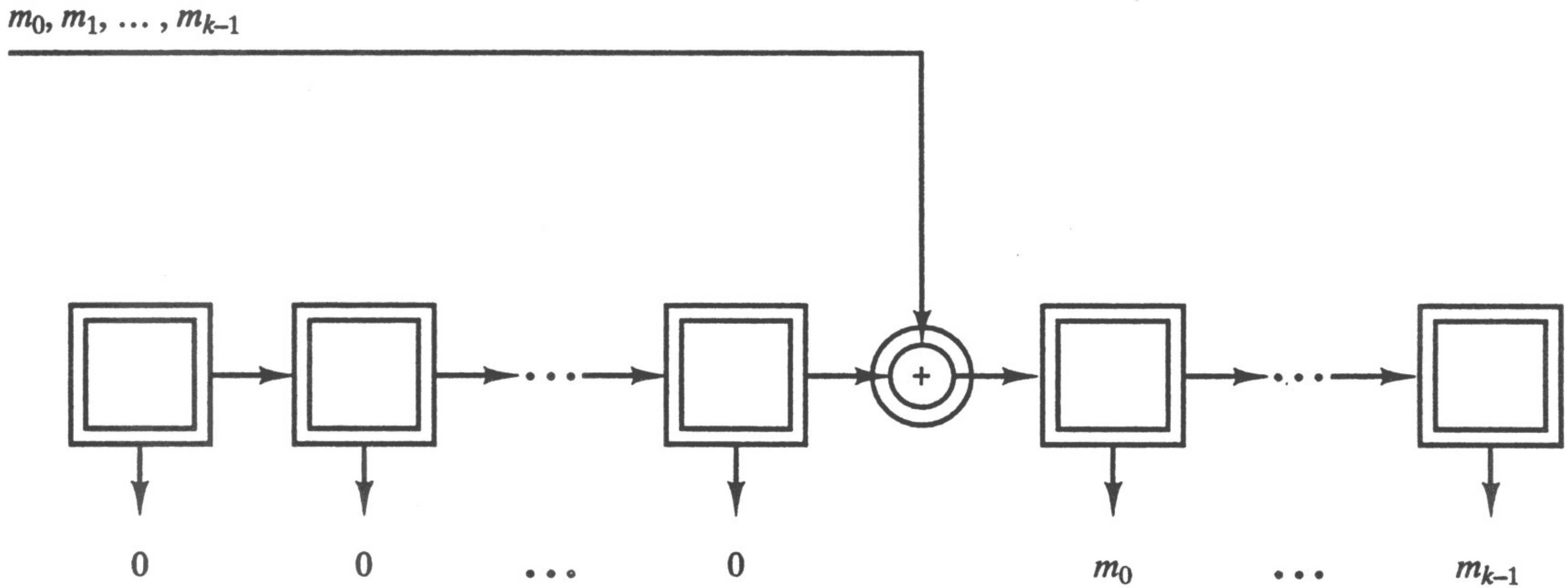


Figure 5-8. Shift-Register Multiplication of $m(x)$ by x^{n-k}

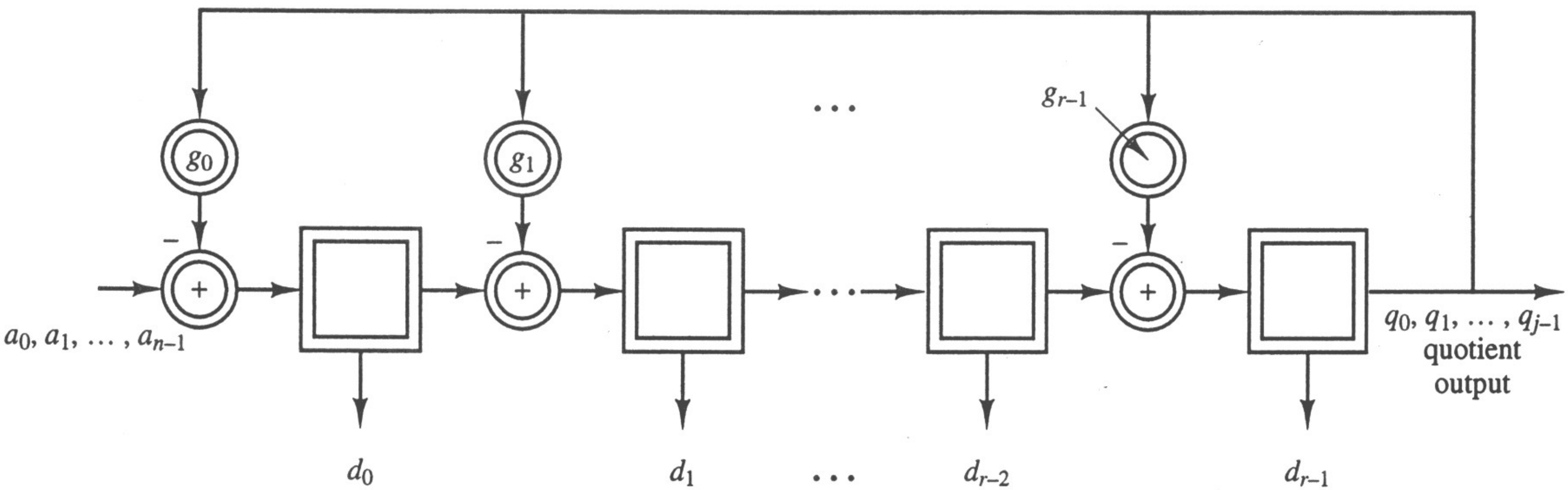


Figure 5-9. Shift-Register Division of $a(x)$ by $g(x)$

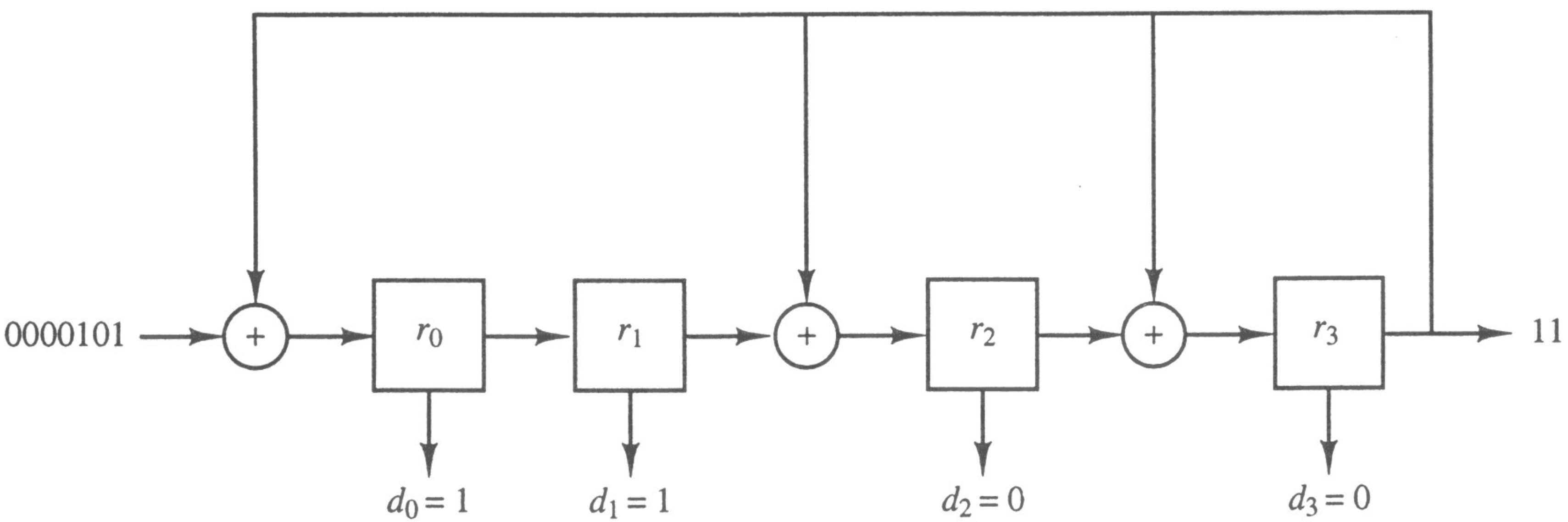
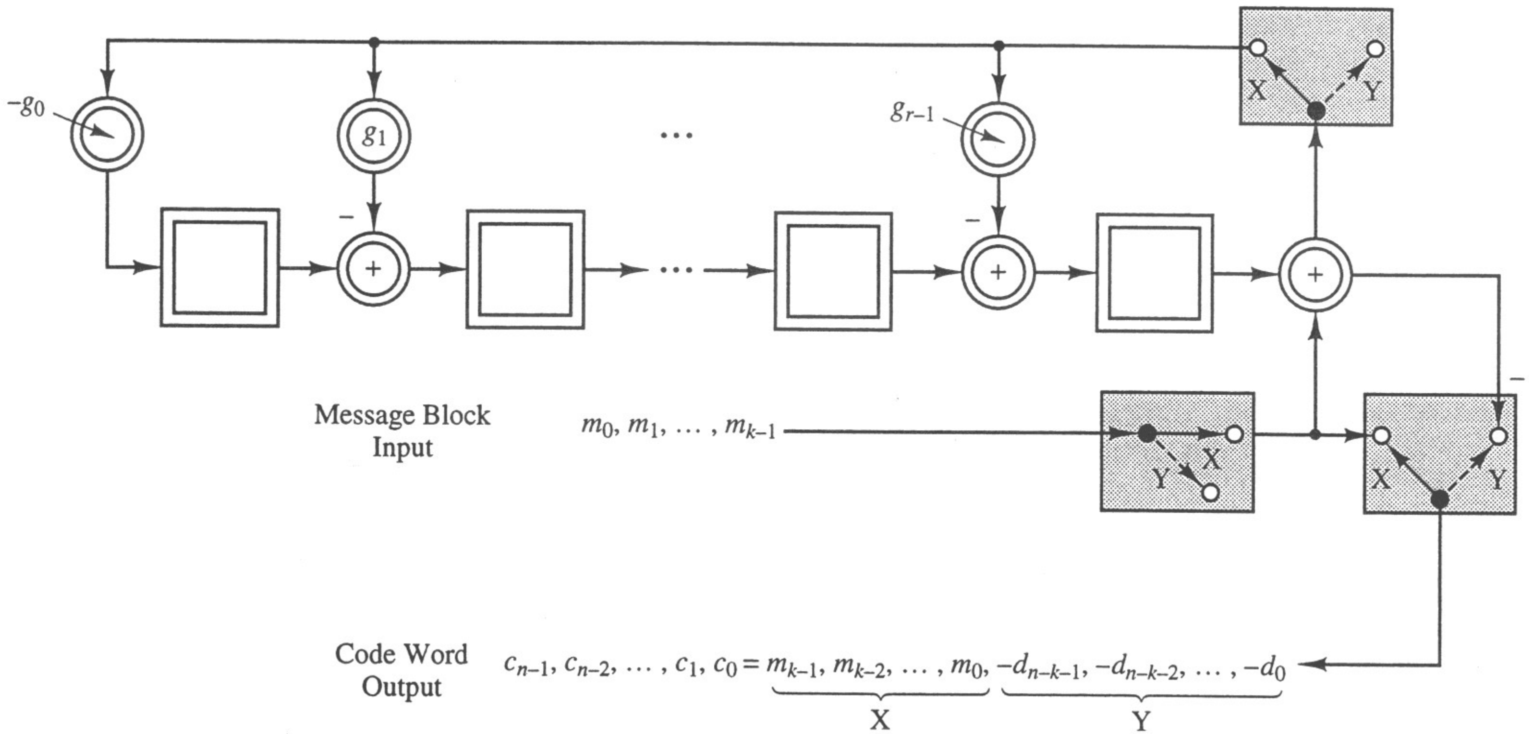


Figure 5-10. Shift-Register Division of $x^6 + x^4$ by $x^4 + x^3 + x^2 + 1$

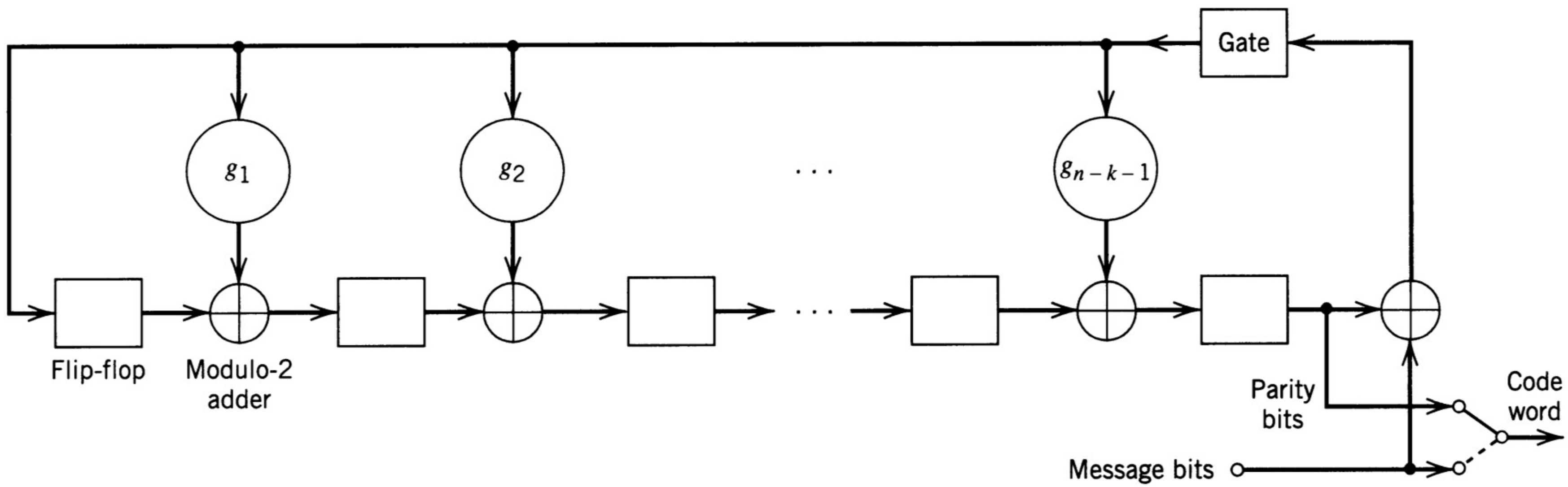
SR cells	r_0	r_1	r_2	r_3	
Initial state	0	0	0	0	
Input $a_6 = 1$	1	0	0	0	
Input $a_5 = 0$	0	1	0	0	
Input $a_4 = 1$	1	0	1	0	
Input $a_3 = 0$	0	1	0	1	
Input $a_2 = 0$	1	0	0	1	
Input $a_1 = 0$	1	1	1	1	
Input $a_0 = 0$	1	1	0	0	
Final state $= r$	1	1	0	0	$\Leftrightarrow d(x) = x + 1$

Figure 5-11. Shift-Register Cell Contents During Division of $x^6 + x^4$ by $x^4 + x^3 + x^2 + 1$

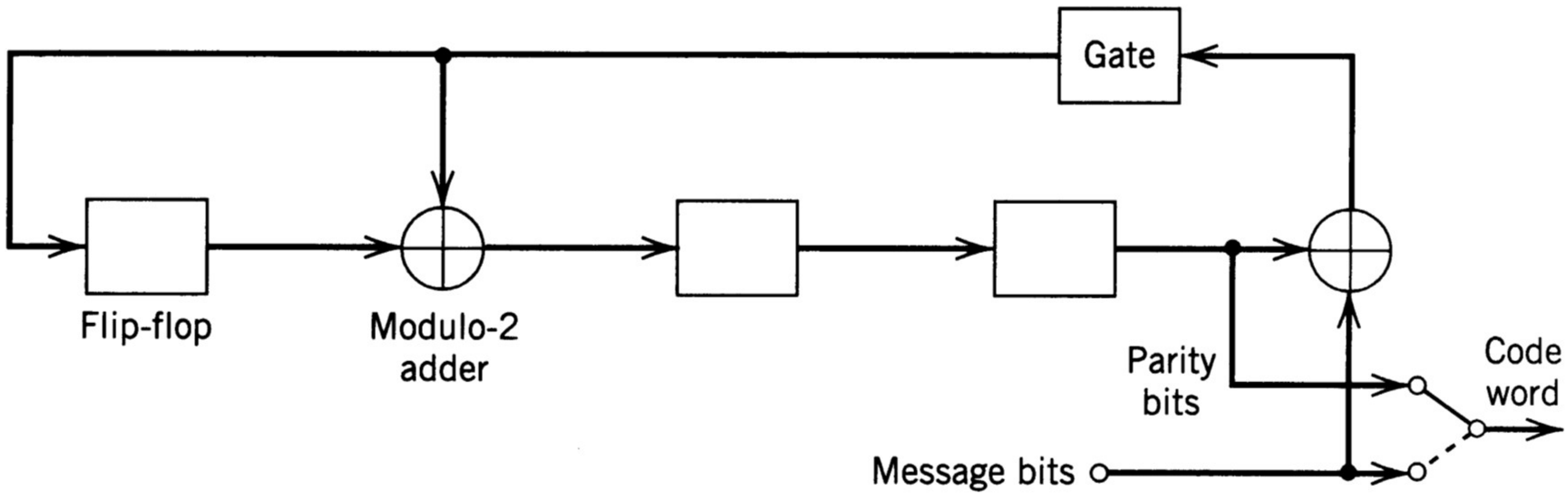
Encoder for an (n,k) Cyclic Code



Encoder for a Binary (n,k) Cyclic Code



Encoder for the (7,4) Cyclic Code Generated by $g(x) = 1+x+x^3$



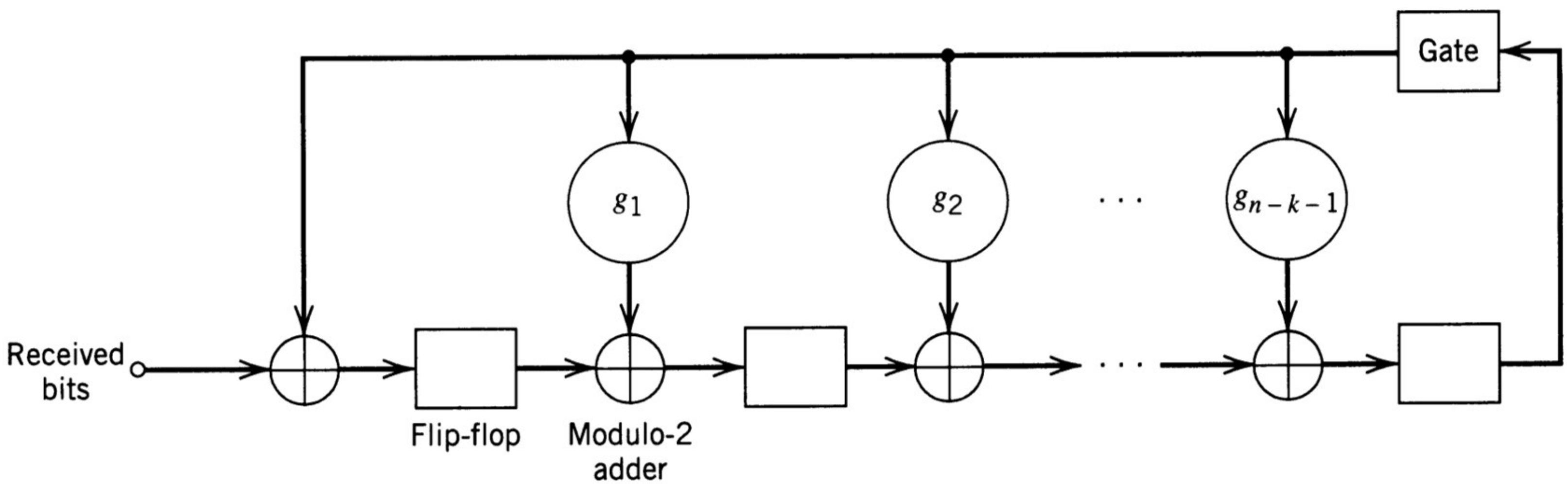
Encoding $1+x^2+x^3$

input	p_0	p_1	p_2	output
1	1	1	0	1
1	1	0	1	1
0	1	0	0	0
1	1	0	0	1
-		1	0	0
-			1	0
-				1

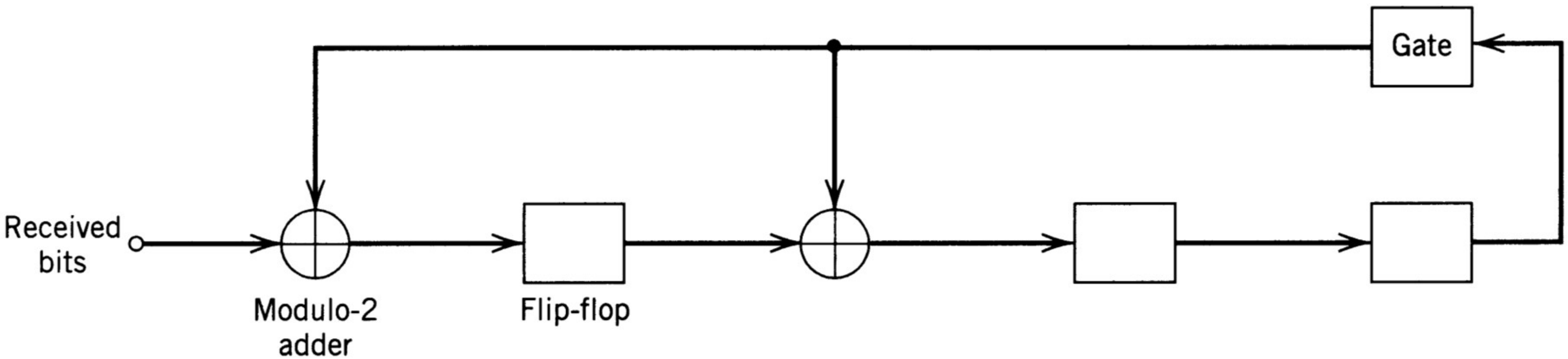
Encoding $1+x^2$ with $g(x) = 1+x^2+x^3+x^4$

input	p_0	p_1	p_2	p_3	output
1	1	0	1	1	1
0	1	1	1	0	0
1	1	1	0	0	1
-		1	1	0	0
-			1	1	0
-				1	1
-					1

Syndrome Computation Circuit



Syndrome Calculator for the (7,4) Cyclic Code Generated by $g(x) = 1+x+x^3$



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

Syndrome for $x^2+x^4+x^5$

input	s_0	s_1	s_2
0	0	0	0
1	1	0	0
1	1	1	0
0	0	1	1
1	0	1	1
0	1	1	1
0	1	0	1

Theorem 5-3

Let $s(x)$ be the syndrome polynomial for a received word $r(x)$. Then $s^{(1)}(x)$ resulting from dividing $xs(x)$ by $g(x)$ is the syndrome polynomial for $r^{(1)}(x)$, the cyclic shift of $r(x)$.

Hamming Code Example (Cont.)

- Example 5.8 – Shift register error correction for the (7,4,3) Hamming code with $g(x) = 1+x+x^3$
- Systematic form

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

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

Syndromes for $g(x) = 1+x+x^3$

error pattern	error polynomial	syndrome	syndrome polynomial
0000000	0	000	0
1000000	1	100	1
0100000	x	010	x
0010000	x^2	001	x^2
0001000	x^3	110	$1+x$
0000100	x^4	011	$x+x^2$
0000010	x^5	111	$1+x+x^2$
0000001	x^6	101	$1+x^2$

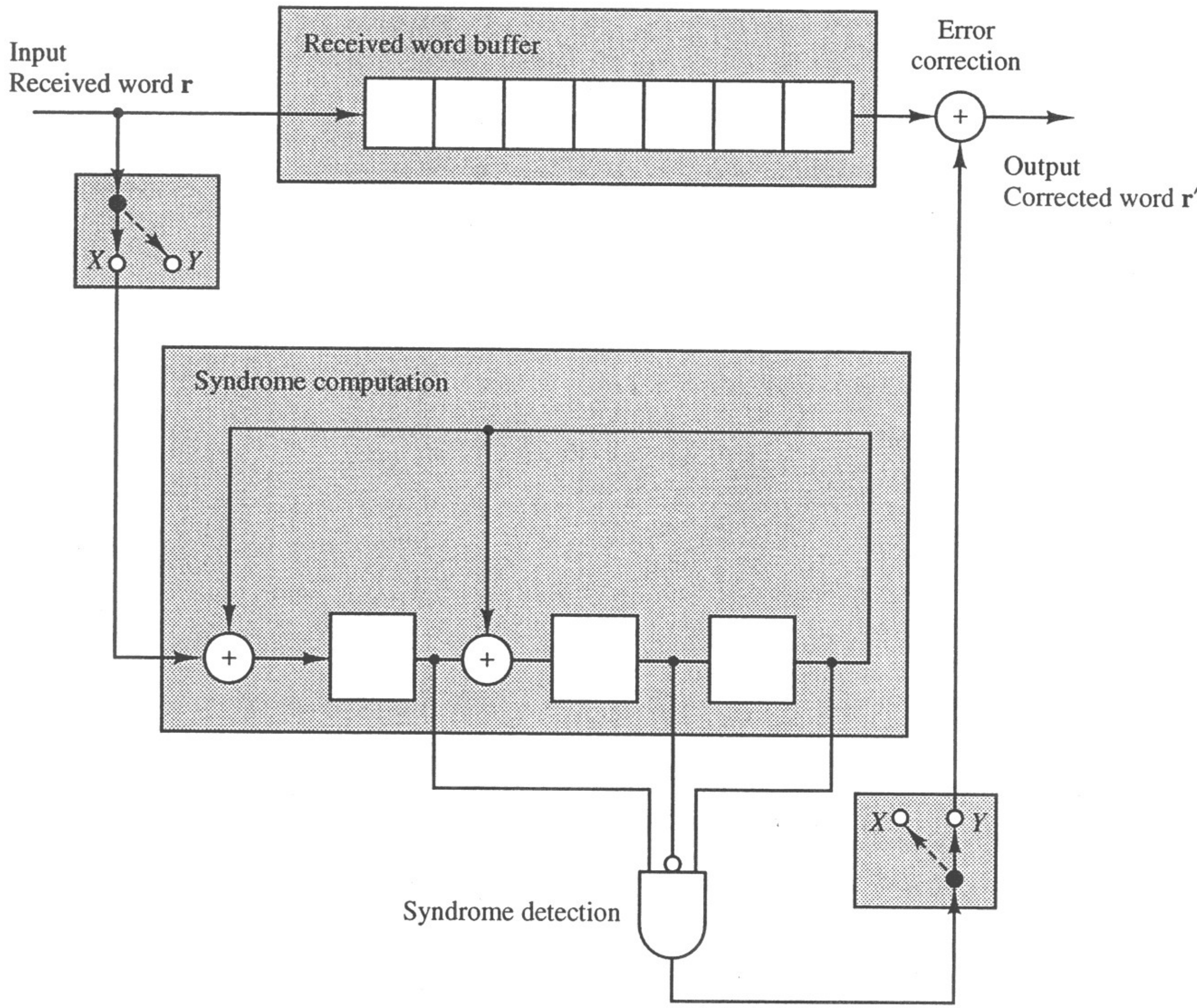


Figure 5-14. Shift-Register Decoder for (7, 4) Cyclic Code

Decoding r = 1101011

received word	syndrome	decoder output
1101011	010	1
-110101	001	11
--11010	110	011
---1101	011	1011
----110	111	01011
------10	**101**	**0**01011
------0	100	1001011

Shortened Cyclic Codes

- Systematic cyclic codes can be shortened by setting the u most significant bits of the codeword (message bits) to zero
- Length is only limited by the length of the original cyclic code n
- (n,k) code shortened to an $(n\text{-}u,\ k\text{-}u)$ code
- Since we are using a subset of the original codewords, the error correction and detection capability is at least as good as the original cyclic code

- Shortened cyclic codes are usually not cyclic, but we can still use the same shift registers for encoding and decoding as the original cyclic codes.
- Shortened cyclic codes are often called polynomial codes
- Widely used shortened cyclic codes:
  - Cyclic Redundancy Check (CRC) codes
- CRC codes are normally used for error detection

Cyclic Redundancy Check Codes

- Typical choice of generator polynomial is

  $g(x) = (x+1)p(x)$ (to detect all odd error patterns)

  where $p(x)$ is a primitive polynomial
- Example: CRC-12

  $g(x) = (x^{11}+x^2+1)(x+1)$

  This is a cyclic code of length $n = 2^{11}-1 = 2047$ and dimension $k = 2047-12= 2035$
- Only 12 bits of redundancy

CRC CODE	GENERATION POLYNOMIAL
CRC-4	$g_4(x) = x^4 + x^3 + x^2 + x + 1$
CRC-7	$g_7(x) = x^7 + x^6 + x^4 + 1 = (x^4 + x^3 + 1)(x^2 + x + 1)(x + 1)$
CRC-8	$g_8(x) = (x^5 + x^4 + x^3 + x^2 + 1)(x^2 + x + 1)(x + 1)$
CRC-12	$g_{12}(x) = x^{12} + x^{11} + x^3 + x^2 + x + 1 = (x^{11} + x^2 + 1)(x + 1)$
CRC-ANSI	$g_{ANSI}(x) = x^{16} + x^{15} + x^2 + 1 = (x^{15} + x + 1)(x + 1)$
CRC-CCITT	$g_{CCITT}(x) = x^{16} + x^{12} + x^5 + 1$ $= (x^{15} + x^{14} + x^{13} + x^{12} + x^4 + x^3 + x^2 + x + 1)(x + 1)$
CRC-SDLC	$g_{SDLC}(x) = x^{16} + x^{15} + x^{13} + x^7 + x^4 + x^2 + x + 1$ $= (x^{14} + x^{13} + x^{12} + x^{10} + x^8 + x^6 + x^5 + x^4 + x^3 + x + 1)$ $\cdot(x + 1)^2$
CRC24	$g_{24}(x) = x^{24} + x^{23} + x^{14} + x^{12} + x^8 + 1$ $= (x^{10} + x^8 + x^7 + x^6 + x^5 + x^4 + x^3 + x + 1)$ $\cdot(x^{10} + x^9 + x^6 + x^4 + 1)(x^3 + x^2 + 1)(x + 1)$
CRC32$_A$[Mer]	$x^{32} + x^{30} + x^{22} + x^{15} + x^{12} + x^{11} + x^7 + x^6 + x^5 + x$ $(x^{10} + x^9 + x^8 + x^6 + x^2 + x + 1)(x^{10} + x^7 + x^6 + x^3 + 1)$ $\cdot(x^{10} + x^8 + x^5 + x^4 + 1)(x + 1)(x)$
CRC-32$_B$[Ga12]	$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5$ $+x^4 + x^2 + x + 1$

- Coverage is the fraction of words that will be detected in error should the input be completely corrupted (worst case: a random bit stream)

$$\lambda = \frac{q^n - q^k}{q^n} = 1 - q^{-(n-k)} = 1 - q^{-r}$$

- For example, CRC-12

$$\lambda = 1 - 2^{-12} = 0.999756$$

- The larger *n*-*k*, the greater the coverage

Burst Errors

- Hardware faults and multipath fading environments cause burst errors
  - Error patterns of the form
    e = …00001XXX…XXX10000…
  - A burst error of length 6 is
    e = …0001XXXX100…
- One would like the CRC code to detect as many of these as possible

- It can be shown that a q-ary CRC code constructed from a cyclic code can detect
  - All burst error patterns of length n-k = r or less where r is the degree of $g(x)$
  - A fraction $1-q^{1-r}/(q-1)$ of all burst error patterns of length $r+1$
  - A fraction $1-q^{-r}$ of all burst error patterns of length $b > r+1$
- Example: CRC-12
  - detects 99.95% of all length 13 burst errors
  - detects 99.976% of all length > 13 burst errors

Encoder for a Binary (n,k) Cyclic Code