Chapter 7

Cyclic Codes

7.1 Introduction

The use of the linear space structure of $\mathbb{B}^n$ enabled us to define binary codes whose encoding and decoding operations depend on this structure. The development of more powerful codes requires the use of additional algebraic structure, in particular, a ring structure on $\mathbb{B}^n$. In this chapter, we will develop the algebraic concepts that enable us to define a ring structure on $\mathbb{B}^n$ and apply them to the design of *cyclic codes*. As in the previous chapter, the presentation of the algebraic concepts will be informal; a rigorous treatment can be found in textbooks such as [1].

7.2 Rings of Polynomials

In the previous sections on linear codes we have made use of the facts that $\mathbb{B}$ is a field and $\mathbb{B}^n$ can be made into a linear space over $\mathbb{B}$ by defining the operations of addition and multiplication by a scalar component by component.

There are other algebraic structures that can also be introduced on $\mathbb{B}^n$.

DEFINITION 7.1 Polynomial of degree n *A* polynomial of degree n *with coefficients in a ring* R *is an element of* R^{n+1}.

We may also refer to polynomials with coefficients in a ring R as polynomials over R.

This does not look like the more familiar definition of a polynomial function, say, $p(x) = ax^2 + bx + c$, but it is equivalent. The essential information about a polynomial is contained in its coefficients, and that is what the components of an element of R^{n+1} give us.

There is additional structure associated with polynomials, and to emphasize this we will not denote polynomials by $(n+1)$-tuples such as $(r_0, r_1, r_2, \ldots, r_n)$, but by the notation $r_0 + r_1 X + r_2 X^2 + \ldots + r_n X^n$.

Polynomials over R may be added and multiplied by elements of R. Addition is carried out by adding the coefficients of like powers of X, while multiplication is carried out by multiplying every coefficient by the multiplier.

EXAMPLE 7.1

The following are polynomials over the ring of integers, $\mathbb{Z}$:

$$p_1(X) = 1 + 2X + 3X^2 + 4X^3 + 5X^4$$

$$p_2(X) = 1 - 2X + 3X^2 - 4X^3 + 5X^4$$

$$p_3(X) = 1 - 3X^2 + 3X^4 - X^6$$

$$p_4(X) = 4X^7 - 8X^8 + 16X^9$$

$$p_5(X) = X^{10} - X^{100}.$$

Adding these polynomials means adding the coefficients:

$$\begin{aligned}(p_1 + p_2)(X) &= (1+1) + (2-2)X + (3+3)X^2 + (4-4)X^3 + (5+5)X^4 \\ &= 2 + 6X^2 + 10X^4\end{aligned}$$

$$\begin{aligned}(p_2 + p_3)(X) &= (1+1) + 2X + (3-3)X^2 - 4X^3 + (5+3)X^4 - X^6 \\ &= 2 + 2X - 4X^3 + 8X^4 - X^6\end{aligned}$$

$$(p_4 + p_5)(X) = 4X^7 - 8X^8 + 16X^9 + x^{10} - X^{100}.$$

Multiplying a polynomial by an integer means multiplying all the coefficients:

$$8p_1(X) = 8 + 16X + 24X^2 + 32X^3 + 40X^4$$

$$(-5)p_3(X) = -5 + 15X^2 - 15X^4 + 5X^6$$

$$100p_5(X) = 100X^{10} - 100X^{100}.$$

□

EXAMPLE 7.2

We also have polynomials over the binary field $\mathbb{B}$:

$$p_1(X) = 1 + X + X^2 + X^3 + X^4$$

$$p_2(X) = 1 + X^2 + X^4$$
$$p_3(X) = 1 + X^4 + X^6$$
$$p_4(X) = X^7 + X^8 + X^9$$
$$p_5(X) = X^{10} + X^{100}.$$

Adding these polynomials uses the rules for addition in $\mathbb{B}$:

$$\begin{aligned}(p_1 + p_2)(X) &= (1+1) + X + (X^2 + X^2) + X^3 + (X^4 + X^4)\\ &= X + X^3\end{aligned}$$

$$(p_2 + p_3)(X) = X^2 + X^6$$
$$(p_3 + p_4)(X) = 1 + X^4 + X^6 + X^7 + X^8 + X^9.$$

Multiplying polynomials by elements of $\mathbb{B}$ is trivial: we either multiply by 0 to get the zero polynomial, or multiply by 1, which gives the same polynomial. □

If R is a field, then these operations make the set of polynomials of degree n over R into a linear space. Since $\mathbb{B}$ is a field, the set of polynomials of degree n over $\mathbb{B}$ is a linear space; it is clearly the space $\mathbb{B}^{n+1}$. We identify elements of the linear space $\mathbb{B}^{n+1}$ with polynomials by matching the components of the vector with coefficients of the polynomial, matching the leftmost bit with the constant term. So, for $n+1 = 8$, 11000111 is matched with $1 + X + X^5 + X^6 + X^7$.

EXAMPLE 7.3

For $n = 1$, we identify 00 with (the polynomial) 0, 10 with (the polynomial) 1, 01 with X and 11 with $1 + X$.

For $n = 2$, we identify 010 with X, 011 with $X + X^2$, 111 with $1 + X + X^2$, and so on.

For $n = 9$, we identify 101010101 with $1 + X^3 + X^5 + X^7 + X^9$, 000111000 with $X^3 + X^4 + X^5$, 0110001111 with $X + X^2 + X^6 + X^7 + X^8 + X^9$, and so on. □

We can also define a multiplication operation on polynomials. To multiply polynomials, we proceed term by term, multiplying the coefficients and adding the exponents of X, so that the product of aX^m and bX^n is $(ab)X^{m+n}$.

EXAMPLE 7.4

Here are some examples of multiplying polynomials with coefficients in $\mathbb{Z}$:

$$\begin{aligned}(1 + 2X + 3X^2) \times (4 + 5X) &= 4 + 5X + 8X + 10X^2 + 12X^2 + 15X^3\\ &= 4 + 13X + 22X^2 + 15X^3\end{aligned}$$

$$(4-3X^2)\times(2X-X^3) = 8X-4X^3-6X^3+3X^5$$
$$= 8X-10X^3+3X^5$$

$$(X+X^3)\times(X^5-X^9) = X^6+X^8-X^{10}-X^{12}.$$

□

EXAMPLE 7.5

Here are some examples of multiplying polynomials over $\mathbb{B}$:

$$(1+X+X^2)\times(1+X) = 1+X+X+X^2+X^2+X^3$$
$$= 1+X^3$$

$$(1+X^2)\times(X+X^3) = X+X^3+X^3+X^5$$
$$= X+X^5$$

$$(X+X^3)\times(X^5+X^9) = X^6+X^8+X^{10}+X^{12}.$$

If we represent polynomials as elements of $\mathbb{B}^{n+1}$, we can also write these products as

$$111\times 11 = 1001$$

$$101\times 0101 = 010001$$

and

$$0101\times 0000010001 = 0000001010101.$$

□

We shall denote the ring of all polynomials with coefficients in $\mathbb{B}$ with operations of addition and multiplication defined above by $\mathbb{B}[X]$.

We can also define a division operation for polynomials. If the ring of coefficients is a field, we can use the *synthetic division* algorithm that is taught in high school algebra courses to carry out the division. In most cases, the division will not be exact, and there will be a remainder.

EXAMPLE 7.6

Consider $X+4$ and $X^3+2X^2-5X+15$ to be polynomials with coefficients in the field of real numbers. We start the synthetic division by setting them out as follows:

$$X+4\,\overline{)\,X^3+2X^2-5X+15}$$

We multiply $X + 4$ by X^2, and subtract the result from $X^3 + 2X^2 - 5X + 15$:

$$\begin{array}{r|l} & X^2 \\ \hline X+4 & X^3 + 2X^2 - 5X + 15 \\ & \underline{X^3 + 4X^2} \\ & \quad -2X^2 \end{array}$$

We bring down the $-5X$, multiply $X + 4$ by $-2X$ and subtract the result from $-2X^2 - 5X$:

$$\begin{array}{r|l} & X^2 - 2X \\ \hline X+4 & X^3 + 2X^2 - 5X + 15 \\ & \underline{X^3 + 4X^2} \\ & \quad -2X^2 - 5X \\ & \quad \underline{-2X^2 - 8X} \\ & \qquad\quad 3X \end{array}$$

We bring down the 15, multiply $X + 4$ by 3 and subtract the result from $3X + 15$:

$$\begin{array}{r|l} & X^2 - 2X + 3 \\ \hline X+4 & X^3 + 2X^2 - 5X + 15 \\ & \underline{X^3 + 4X^2} \\ & \quad -2X^2 - 5X \\ & \quad \underline{-2X^2 - 8X} \\ & \qquad\quad 3X + 15 \\ & \qquad\quad \underline{3X + 12} \\ & \qquad\qquad\quad 3 \end{array}$$

This shows that

$$X^3 + 2X^2 - 5X + 15 = (X + 4)(X^2 - 2X + 3) + 3.$$

□

EXAMPLE 7.7

We can perform synthetic division in the same way when the coefficients of the polynomials come from the binary field $\mathbb{B}$. In this case, addition and subtraction are the same operation, so the procedure uses addition.

To divide $X^3 + X^2 + X + 1$ by $X + 1$ we start by setting the polynomials out in the usual way:

$$X + 1 \,\big|\, X^3 + X^2 + X + 1$$

We multiply $X + 1$ by X^2, and add the result to $X^3 + X^2 + X + 1$:

$$\begin{array}{r|l} & X^2 \\ \hline X+1 & X^3 + X^2 + X + 1 \\ & \underline{X^3 + X^2} \\ & \qquad 0 \end{array}$$

We bring down the $X + 1$, multiply $X + 1$ by 1 and add the result to $X + 1$:

$$\begin{array}{r|l} & X^2 \quad + \quad 1 \\ \hline X+1 & X^3 + X^2 + X + 1 \\ & \underline{X^3 + X^2} \\ & \qquad\quad X + 1 \\ & \qquad\quad \underline{X + 1} \\ & \qquad\qquad 0 \end{array}$$

There is no remainder. This shows that

$$X^3 + X^2 + X + 1 = (X + 1)(X^2 + 1).$$

□

EXAMPLE 7.8

To divide $X^5 + 1$ by $X^2 + X$ in the ring of polynomials over $\mathbb{B}$, we set out the polynomials in the usual way:

$$X^2 + X \,\big|\, X^5 \qquad\quad + 1$$

leaving space for the terms that are not present in $X^5 + 1$.

We multiply $X^2 + X$ by X^3 and add the product to $X^5 + 1$:

$$\begin{array}{r|l} & X^3 \\ \hline X^2+X & X^5 \qquad\quad + 1 \\ & \underline{X^5 + X^4} \\ & \qquad X^4 \end{array}$$

We multiply $X^2 + X$ by X^2 and add the product to X^4:

$$\begin{array}{r|l} & X^3 + X^2 \\ \hline X^2+X & X^5 \qquad\qquad + 1 \\ & \underline{X^5 + X^4} \\ & \qquad X^4 \\ & \qquad \underline{X^4 + X^3} \\ & \qquad\quad X^3 \end{array}$$

We multiply $X^2 + X$ by X and add the product to X^3:

$$\begin{array}{r|l} & X^3 + X^2 + X \\ \hline X^2+X & X^5 \qquad\qquad + 1 \\ & \underline{X^5 + X^4} \\ & \qquad X^4 \\ & \qquad \underline{X^4 + X^3} \\ & \qquad\quad X^3 \\ & \qquad\quad \underline{X^3 + X^2} \\ & \qquad\qquad X^2 \end{array}$$

We bring down the 1 and add $X^2 + X$ to $X^2 + 1$:

$$\begin{array}{r|l} & X^3 + X^2 + X + 1 \\ X^2 + X & \overline{X^5 \qquad\qquad\qquad\qquad + 1} \\ & \underline{X^5 + X^4} \\ & \quad X^4 \\ & \quad \underline{X^4 + X^3} \\ & \qquad X^3 \\ & \qquad \underline{X^3 + X^2} \\ & \qquad\quad X^2 \qquad + 1 \\ & \qquad\quad \underline{X^2 + X} \\ & \qquad\qquad X + 1 \end{array}$$

The remainder is $X + 1$. This shows that

$$X^5 + 1 = (X^2 + X)(X^3 + X^2 + X^1 + X) + (X + 1).$$

□

If we multiply two polynomials of degree n together, the result is usually a polynomial of degree $2n$. This means that we cannot make the set of polynomials of degree n into a ring under the operations of polynomial addition and multiplication, as multiplying two polynomials in this set will give us a polynomial that is not in this set.

There is another way that we can define a multiplication operation on polynomials that will give a useful ring structure.

DEFINITION 7.2 Multiplication modulo a polynomial *Let p, q and r be polynomials with coefficients in some ring R. The* product of p and q modulo r *is the remainder when pq is divided by r.*

EXAMPLE 7.9

Let $p(X) = X^2 - 3$, $q(X) = X^2 + 5$ and $r(X) = X^3 - X^2$, where these polynomials have coefficients in the field of real numbers, $\mathbb{R}$.

The product of p and q is

$$p(X)q(X) = (X^2 - 3)(X^2 + 5) = X^4 + 2X^2 - 15.$$

Using synthetic division as in the examples above, we find that

$$X^4 + 2X^2 - 15 = (X + 1)(X^3 - X^2) + (-X^2 - 15).$$

So the product of $(X^2 - 3)$ and $(X^2 + 5)$ modulo $(X^3 - X^2)$ is $(-X^2 - 15)$.

□

EXAMPLE 7.10

Let $p(X) = X^3 + X$, $q(X) = X^2 + X$ and $r(X) = X^4 + X^2$ be polynomials with coefficients in $\mathbb{B}$.

The product of p and q is

$$p(X)q(X) = (X^3 + X)(X^2 + X) = X^5 + X^4 + X^3 + X^2.$$

Using synthetic division as in the examples above, we find that

$$X^5 + X^4 + X^3 + X^2 = (X + 1)(X^4 + X^2).$$

There is no remainder, so the product of $(X^3 + X)$ and $(X^2 + X)$ modulo $(X^4 + X^2)$ is 0.

□

EXAMPLE 7.11

Let $p(X) = X^4 + X^2$, $q(X) = X^4 + X$ and $r(X) = X^5 + X$ be polynomials with coefficients in $\mathbb{B}$.

The product of p and q is

$$p(X)q(X) = (X^4 + X^2)(X^4 + X) = X^8 + X^6 + X^5 + X^3.$$

Using synthetic division as in the examples above, we find that

$$X^8 + X^6 + X^5 + X^3 = (X^3 + X)(X^5 + X) + (X^4 + X^3 + X^2 + X).$$

So the product of $(X^4 + X^2)$ and $(X^4 + X)$ modulo $(X^5 + X)$ is $(X^4 + X^3 + X^2 + X)$.

□

The operation of multiplication modulo the polynomial $X^n + 1$ gives us products which are polynomials of degree $n - 1$ or less. The set of polynomials of degree less than n forms a ring with respect to addition and multiplication modulo $X^n + 1$; this ring will be denoted $\mathbb{B}_n[X]/(X^n + 1)$. It can be identified with $\mathbb{B}^n$ and the identification can be used to give $\mathbb{B}^n$ the structure of a ring.

The *Remainder Theorem* of high school algebra states that the remainder that is obtained when a polynomial $p(X)$ with real coefficients is divided by $(X - a)$ can be calculated by replacing a for X in $p(X)$. In particular we have the following:

RESULT 7.1

The remainder that is obtained when a polynomial in $\mathbb{B}[X]$ is divided by $X^n + 1$ can be calculated by replacing X^n with 1 in the polynomial. This operation "wraps" the powers of X around from X^n to $X^0 = 1$.

EXAMPLE 7.12

Using the rule above, we can quickly calculate some remainders.

The remainder of X^3+X^2+X when divided by X^3+1 is $1+X^2+X = X^2+X+1$.

The remainder of $X^4+X^3+X^2$ when divided by X^3+1 is $1X+1+X^2 = X^2+X+1$.

The remainder of $X^7 + X^6 + X^5$ when divided by $X^4 + 1$ is $1X^3 + 1X^2 + 1X = X^3 + X^2 + X$.

The product of $X^3 + X^2$ and $X^2 + X$ modulo $X^4 + 1$ is the remainder when $(X^3+X^2)(X^2+X) = X^5+X^3$ is divided by X^4+1, namely $1X+X^3 = X^3+X$.

∎

We can now compute the multiplication tables of the rings $\mathbb{B}_n[X]/(X^n + 1)$.

EXAMPLE 7.13

The elements of $\mathbb{B}_2[X]/(X^2 + 1)$ are the polynomials 0, 1, X and $1 + X$. Multiplication by 0 and 1 is trivial. The other products are

$(X)(X)$ modulo $(X^2 + 1) = X^2$ modulo $(X^2 + 1) = 1$,

$(X)(1 + X)$ modulo $(X^2 + 1) = X + X^2$ modulo $(X^2 + 1) = 1 + X$,

$(1 + X)(1 + X)$ modulo $(X^2 + 1) = 1 + X^2$ modulo $(X^2 + 1) = 0$.

So the multiplication table is

	0	1	X	$1+X$
0	0	0	0	0
1	0	1	X	$1+X$
X	0	X	1	$1+X$
$1+X$	0	$1+X$	$1+X$	0

Identifying the polynomial 0 with 00, 1 with 10, X with 01 and $1 + X$ with 11, the multiplication table becomes

	00	10	01	11
00	00	00	00	00
10	00	10	01	11
01	00	01	10	11
11	00	11	11	00

EXAMPLE 7.14

The elements of $\mathbb{B}_3[X]/(X^3+1)$ are the polynomials 0, 1, X, $1+X$, X^2, $1+X^2$, $X+X^2$, $1+X+X^2$. The multiplication table is too large to fit on the page, so we display it in two parts:

	0	1	X	$1+X$
0	0	0	0	0
1	0	1	X	$1+X$
X	0	X	X^2	$X+X^2$
$1+X$	0	$1+X$	$X+X^2$	$1+X^2$
X^2	0	X^2	1	$1+X^2$
$1+X^2$	0	$1+X^2$	$1+X$	$X+X^2$
$X+X^2$	0	$X+X^2$	$1+X^2$	$1+X$
$1+X+X^2$	0	$1+X+X^2$	$1+X+X^2$	0

	X^2	$1+X^2$	$X+X^2$	$1+X+X^2$
0	0	0	0	0
1	X^2	$1+X^2$	$X+X^2$	$1+X+X^2$
X	1	$1+X$	$1+X^2$	$1+X+X^2$
$1+X$	$1+X$	$X+X^2$	$1+X$	0
X^2	X	$X+X^2$	$1+X$	$1+X+X^2$
$1+X^2$	$1+X^2$	$1+X$	$1+X^2$	0
$X+X^2$	$1+X$	$1+X^2$	$X+X^2$	0
$1+X+X^2$	$1+X+X^2$	0	0	$1+X+X^2$

Identifying the polynomial 0 with 000, 1 with 100, X with 010, $1+X$ with 110, X^2 with 001, and so on, the multiplication table becomes

	000	100	010	110	001	101	011	111
000	000	000	000	000	000	000	000	000
100	000	100	010	110	001	101	011	111
010	000	010	001	011	100	010	101	111
110	000	110	011	101	110	011	110	000
001	000	001	100	101	010	011	110	111
101	000	101	110	011	101	110	101	000
011	000	011	101	110	110	101	011	000
111	000	111	111	000	111	000	000	111

The following result is very important.

RESULT 7.2
Multiplication by X in $\mathbb{B}_n[X]/(X^n+1)$ is equivalent to shifting the components of the corresponding vector in $\mathbb{B}^n$ cyclically one place to the right.

PROOF Let $p(X)$ be a polynomial in $\mathbb{B}_n[X]/(X^n+1)$,

$$p(X) = b_0 + b_1X + b_2X^2 + \ldots + b_{n-1}X^{n-1}. \tag{7.1}$$

The corresponding vector in $\mathbb{B}^n$ is $b_0b_1b_2\ldots b_{n-1}$.

$$\begin{aligned} Xp(X) &= b_0X + b_1X^2 + b_2X^3 + \ldots + b_{n-1}X^n \\ &= b_{n-1} + b_0X + b_1X^2 + \ldots + b_{n-1}(X^n+1) \end{aligned} \tag{7.2}$$

so

$$Xp(X) \bmod (X^n+1) = b_{n-1} + b_0X + \ldots b_{n-2}X^{n-1}. \tag{7.3}$$

The vector corresponding to $Xp(X) \bmod (X^n+1)$ is $b_{n-1}b_0b_1\ldots b_{n-2}$, which is the result of cyclically shifting $b_0b_1b_2\ldots b_{n-1}$ one place to the right.

□

7.3 Cyclic Codes

We now look at codes whose construction uses both the additive and multiplicative structures of the ring $\mathbb{B}_n[X]/(X^n+1)$. Since the elements of this ring can be represented both as code words in $\mathbb{B}^n$ and as polynomials, we will use the terms interchangeably as convenient.

DEFINITION 7.3 Cyclic Code *A* cyclic code *is a linear code with the property that any cyclic shift of a code word is also a code word.*

EXAMPLE 7.15

Consider the code whose generator matrix is

$$G = \begin{bmatrix} 1\,0\,1\,0 \\ 0\,1\,0\,1 \end{bmatrix}.$$

The linear code generated by G is $\{0000, 1010, 0101, 1111\}$. Note that shifting 0000 cyclically gives 0000, shifting 1010 one place cyclically gives 0101, shifting 0101 one place cyclically gives 1010 and shifting 1111 cyclically gives 1111. So this is a cyclic code.

In polynomial notation, the code is $\{0, 1 + X^2, X + X^3, 1 + X + X^2 + X^3\}$. A cyclic shift to the right can be accomplished by multiplying by X modulo $(X^4 + 1)$. Multiplying 0 by X gives 0, multiplying $1 + X^2$ by X gives $X + X^3$, multiplying $X + X^3$ by X (modulo $X^4 + 1$) gives $1 + X^2$ and multiplying $1 + X + X^2 + X^3$ by X (also modulo $X^4 + 1$) gives $1 + X + X^2 + X^3$.

□

RESULT 7.3

A cyclic code contains a unique non-zero polynomial of minimal degree.

PROOF There are only a finite number of non-zero polynomials in the code, so at least one of them is of minimal degree.

Suppose that there are two polynomials of minimum degree r, say

$$g(X) = g_0 + g_1 X + \ldots X^r \tag{7.4}$$

and

$$h(X) = h_0 + h_1 X + \ldots X^r. \tag{7.5}$$

(The X^r terms must be present if the degree of the polynomials is r.)

The sum of these polynomials is

$$(g + h)(X) = (g_0 + h_0) + (g_1 + h_1)X + \ldots (g_{r-1} + h_{r-1})X^{r-1} \tag{7.6}$$

$(g + h)$ must belong to the code, but it is a polynomial of degree $(r - 1)$, contradicting the assumption that the polynomials of minimum degree are of degree r. Hence the polynomial of minimum degree must be unique.

□

DEFINITION 7.4 Generator Polynomial *The unique non-zero polynomial of minimal degree in a cyclic code is the* generator polynomial *of the code.*

RESULT 7.4
If $g \in \mathbb{B}_n[X]/(X^n + 1)$ is the generator polynomial for some cyclic code, then every polynomial in the code can be generated by multiplying g by some polynomial in $\mathbb{B}_n[X]/(X^n + 1)$.

PROOF Since multiplication by X modulo $(X^n + 1)$ has the effect of shifting a code word cyclically one place to the right, multiplying by X^k modulo $(X^n + 1)$ has the effect of shifting a code word cyclically k places to the right. It follows that the product g by X^k modulo $(X^n + 1)$ will be a code word for any $k \geq 0$.

Multiplying g by any polynomial is equivalent to multiplying g by various powers of X modulo $(X^n + 1)$ and adding the products together. Since the products are all code words, so is their sum.

To show that every code word can be generated in this way, note that if the degree of g is r, then the polynomials $g, gX, gX^2 \ldots, gX^{n-r-1}$ form a basis for the code and hence that every code word can be generated by taking a linear combination of these basis elements.

∎

If the generator of a cyclic code is $g(X) = g_0 + g_1X + \ldots + g_rX^r$, the fact that the polynomials $g, gX, gX^2 \ldots, gX^{n-r-1}$ form a basis for the code means that the generator matrix of the code can be written in the form

$$G = \begin{bmatrix} g_0 & g_1 & \cdots & g_r & 0 & 0 & \cdots & 0 \\ 0 & g_0 & \cdots & g_{r-1} & g_r & 0 & \cdots & 0 \\ & & & \vdots & & & & \\ 0 & \cdots & 0 & g_0 & g_1 & \cdots & g_r & 0 \\ 0 & \cdots & 0 & 0 & g_0 & \cdots & g_{r-1} & g_r \end{bmatrix}. \tag{7.7}$$

G is a cyclic matrix (each row is obtained by shifting the previous row one column to the right).

EXAMPLE 7.16

$\{0000000, 1011100, 0101110, 0010111, 1001011, 1100101, 1110010, 0111001\}$ is a cyclic code. The code word 1011100 corresponds to the polynomial $1 + X^2 + X^3 + X^4$, which is the polynomial of minimal degree and hence the generator polynomial.

The generator matrix is

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

We interchange the third and seventh columns to reduce the generator matrix to canonical form:

$$G_c = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}.$$

The canonical form of the parity check matrix is:

$$H_c = \begin{bmatrix} 1\,1\,0\,1\,0\,0\,0 \\ 1\,1\,1\,0\,1\,0\,0 \\ 0\,1\,1\,0\,0\,1\,0 \\ 1\,0\,1\,0\,0\,0\,1 \end{bmatrix}.$$

Interchanging the third and seventh columns gives us the parity check matrix of G:

$$H = \begin{bmatrix} 1\,1\,0\,1\,0\,0\,0 \\ 1\,1\,0\,0\,1\,0\,1 \\ 0\,1\,0\,0\,0\,1\,1 \\ 1\,0\,1\,0\,0\,0\,1 \end{bmatrix}.$$

□

The next result gives us a way of finding generator polynomials for cyclic codes.

RESULT 7.5
g is the generator polynomial of a cyclic code in $\mathbb{B}_n[X]/(X^n+1)$ if and only if it is a factor of (X^n+1).

PROOF Let g by the generator polynomial of a cyclic code. If the degree of g is r, then $X^{n-r}g$ is a polynomial of degree n. Let h be the remainder when $X^{n-r}g$ is divided by (X^n+1). The quotient is 1, so we have

$$X^{n-r}g = (X^n+1) + h. \tag{7.8}$$

h is the result of multiplying g by X^{n-r} modulo (X^n+1), so it belongs to the code. Therefore, there is a polynomial $p \in \mathbb{B}_n[X]/(X^n+1)$ such that

$$h = pg. \tag{7.9}$$

Adding these equations, we get

$$(X^n+1) + h + h = X^{n-r}g + pg, \tag{7.10}$$

or

$$(X^n+1) = (X^{n-r}+p)g, \tag{7.11}$$

which shows that g is a factor of (X^n+1).

Conversely, if g divides (X^n+1), there is a polynomial h such that

$$g(X)h(X) = (X^n+1), \tag{7.12}$$

and the polynomials $g, Xg, X^2g, \ldots, X^{n-r-1}g$ represent linearly independent code words obtained by cyclically shifting the bits of g. These form the basis of the linear code generated by g.

Let $p(X) = p_0g(X) + p_1Xg(X) + \ldots + p_{n-r-1}X^{n-r-1}g(X)$ be an arbitrary element of the code generated by g. Then

$$Xp(X) = p_0Xg(X) + p_1X^2g(X) + \ldots + p_{n-r-1}X^{n-r}g(X). \tag{7.13}$$

If $p_{n-r-1} = 0$, then $Xp(X)$ is a linear combination of the basis vectors $Xg(X)$, $X^2g(X) \ldots X^{n-r-1}g(X)$, and so it belongs to the code.

If $p_{n-r-1} = 1$, $Xp(X)$ is a polynomial of degree n. The remainder when it is divided by $(X^n + 1)$ is

$$r(X) = p_{n-r-1} + p_0Xg(X) + \ldots + p_{n-r-2}X^{n-r-1}g(X), \tag{7.14}$$

so

$$r(X) = Xp(X) + p_{n-r-1}(X^n + 1). \tag{7.15}$$

By construction,

$$Xp(X) = (p_0X + p_1X^2 + \ldots + p_{n-r-1}X^{n-r})g(X), \tag{7.16}$$

and by assumption

$$g(X)h(X) = (X^n + 1), \tag{7.17}$$

so

$$r(X) = (p_0X + p_1X^2 + \ldots + p_{n-r-1}X^{n-r} + h(X))g(X). \tag{7.18}$$

This shows that $r(X)$ is a linear combination of the basis polynomials g, Xg, $X^2g, \ldots, X^{n-r-1}g$ and so $r(X)$ belongs to the code. Hence the code generated by $g(X)$ is cyclic. ∎

$(X^n + 1)$ always has at least two factors, because

$$(X^n + 1) = (X + 1)(X^{n-1} + X^{n-2} + \ldots + X^2 + X + 1). \tag{7.19}$$

The factorizations of $(X^n + 1)$ determine the cyclic codes of length n. There is a cyclic code for each factor.

EXAMPLE 7.17

The factorization of $(X^9 + 1)$ is

$$(X^9 + 1) = (1 + X)(1 + X + X^2)(1 + X^3 + X^6).$$

This gives six factors and six cyclic codes.

The cyclic code with generator polynomial $(1 + X)$ has a basis of eight code words:

$$\{110000000, 011000000, 001100000, 000110000, \\ 000011000, 000001100, 000000110, 000000011\}.$$

The cyclic code with generator polynomial $(1 + X + X^2)$ has a basis of seven code words:

$$\{111000000, 011100000, 001110000, \\ 000111000, 000011100, 000001110, 000000111\}.$$

The cyclic code with generator polynomial $(1 + X)(1 + X + X^2) = (1 + X^3)$ has a basis of six code words:

$$\{100100000, 010010000, 001001000, 000100100, 000010010, 000001001\}.$$

The cyclic code with generator polynomial $(1 + X^3 + X^6)$ has a basis of three code words:

$$\{100100100, 010010010, 001001001\}.$$

The cyclic code with generator polynomial $(1 + X)(1 + X^3 + X^6) = (1 + X + X^3 + X^4 + X^6 + X^7)$ has a basis of two code words:

$$\{110110110, 011011011\}.$$

The cyclic code with generator polynomial $(1 + X + X^2)(1 + X^3 + X^6) = (1 + X + X^2 + X^3 + X^4 + X^5 + X^6 + X^7 + X^8)$ has a basis with one code word: $\{111111111\}$.

∎

7.4 Encoding and Decoding of Cyclic Codes

Cyclic codes are linear codes; so the techniques described in the previous chapter can be used to encode and decode messages using them. However, the additional structure they possess allows us to devise other encoding and decoding techniques, some of which can be carried out efficiently in hardware.

We can define polynomial equivalents of the parity check matrix and the syndrome. We have seen that g is the generator polynomial of a cyclic code if and only if it is a factor of $(X^n + 1)$. This means that there is a polynomial h such that

$$(X^n + 1) = g(X)h(X). \tag{7.20}$$

h is the *parity check polynomial*; we shall see that it has properties similar to those of the parity check matrix.

The encoding process produces a code polynomial c from a message polynomial m by multiplication by the generator polynomial:

$$c(X) = m(X)g(X). \tag{7.21}$$

If we multiply c by the parity check polynomial, we get

$$c(X)h(X) = m(X)g(X)h(X) = m(X)(X^n + 1) = 0 \bmod (X^n + 1), \tag{7.22}$$

which is analogous to the result that the matrix product of a code word and the parity check matrix is the zero vector.

If the code polynomial is corrupted in some way, the result can be represented as the sum of the code polynomial and an error polynomial, e, that is,

$$r(X) = c(X) + e(X). \tag{7.23}$$

The *syndrome polynomial*, s, is the product of the result and the parity check polynomial,

$$s(X) = r(X)h(X) = c(x)h(X) + e(X)h(X) = e(X)h(X). \tag{7.24}$$

If the code polynomial is not corrupted, the error polynomial and its syndrome will be zero.

We have seen that we can reduce cyclic codes to systematic form using the canonical form of the generator matrix. We now consider how to describe the systematic form of the code in terms of polynomials. Suppose the code has k message bits and $n - k$ check bits. If it is in systematic form, the first k terms of the code polynomial will be the same as the message polynomial and there will be a polynomial d of degree $n - k - 1$ such that

$$c(X) = m(X) + d(X)X^k. \tag{7.25}$$

The code polynomials will still be multiples of the generator polynomial, but they will not be equal to the product of the message polynomial and the generator polynomial. Instead there will be some other polynomial q such that

$$c(X) = q(X)g(X) = m(X) + d(X)X^k. \tag{7.26}$$

We do not need to know what q is, but we do need to be able to compute d. If we take the remainder after dividing by g, we get

$$(m(X) + d(X)X^k) \bmod g(X) = 0. \tag{7.27}$$

If we multiply through by X^{n-k}, and use the fact that the multiplication is carried out in $\mathbb{B}_n[X]/(X^n + 1)$, we get

$$(m(X)X^{n-k} + d(X)) \bmod g(X) = 0, \tag{7.28}$$

or

$$d(X) \bmod g(X) = m(X)X^{n-k} \bmod g(X). \tag{7.29}$$

Since the degree of d is no greater than $n - k - 1$ and the degree of g is $n - k$, $d \bmod g = d$ and so

$$d(X) = m(X)X^{n-k} \bmod g(X). \tag{7.30}$$

EXAMPLE 7.18

Since

$$X^6 + 1 = (X^2 + X + 1)(X^4 + X^3 + X + 1)$$

there is a cyclic code whose code words are 6 bits long with generator polynomial

$$g(X) = 1 + X + X^2$$

and parity check polynomial

$$h(X) = 1 + X + X^3 + X^4.$$

The generator matrix of this code is

$$G = \begin{bmatrix} 1\,1\,1\,0\,0\,0 \\ 0\,1\,1\,1\,0\,0 \\ 0\,0\,1\,1\,1\,0 \\ 0\,0\,0\,1\,1\,1 \end{bmatrix};$$

and its canonical form is

$$G_c = \begin{bmatrix} 1\,0\,0\,0\,1\,1 \\ 0\,1\,0\,0\,1\,0 \\ 0\,0\,1\,0\,0\,1 \\ 0\,0\,0\,1\,1\,1 \end{bmatrix}.$$

We can verify that the same encoding is given when we use the procedure involving polynomials given above. We only need to verify it for the message polynomials 1, X, X^2 and X^3; all the others are linear combinations of these.

We have $n = 6$, and $k = 4$, so

$$d(X) = m(X)X^2 \bmod g(X)$$

and

$$c(X) = m(X) + d(X)X^4.$$

If $m(X) = 1$, we can use synthetic division to show that

$$d(X) = X^2 \bmod g(X) = 1 + X$$

and so

$$c(X) = 1 + (1 + X)X^4 = 1 + X^4 + X^5.$$

This corresponds to the code word 100011, which is the first row of the canonical form of the generator matrix.

If $m(X) = X$, we get

$$d(X) = X^3 \bmod g(X) = 1,$$

$$c(X) = X + X^4.$$

The corresponding code word is 010010, the second row of the canonical form of the generator matrix.

If $m(X) = X^2$,

$$d(X) = X^4 \bmod g(X) = X,$$

$$c(X) = X^2 + X^5.$$

The corresponding code word is 001001, the third row of the canonical form of the generator matrix.

Finally, if $m(X) = X^3$,

$$d(X) = X^5 \bmod g(X) = 1 + X,$$

$$c(X) = X^3 + (1 + X)X^4 = X^3 + X^4 + X^5.$$

The corresponding code word is 000111, the fourth row of the canonical form of the generator matrix.

∎

We define the syndrome polynomial to be the remainder when the received polynomial is divided by g. Since the code polynomials are precisely the multiples of g, the syndrome polynomial is zero if and only if the received polynomial is a code polynomial. Errors can be corrected by adding the syndrome polynomial to the first k terms of the received polynomial.

EXAMPLE 7.19

From the previous example, if we use the canonical form of the cyclic code of length 6 with generator polynomial $g(X) = 1 + X + X^2$, the message polynomial $1 + X^2$ is encoded as $1 + X^2 + X^4$, or 101010.

If we receive the code word 111010, which corresponds to $1 + X + X^2 + X^4$, the syndrome is X. Adding this to $1 + X + X^2$, we get $1 + X^2$, which is the polynomial that was encoded.

□

7.5 Encoding and Decoding Circuits for Cyclic Codes

A number of circuits for encoding and decoding cyclic codes efficiently have been devised. The simplest ones use only binary adders and registers that can store single bits.

We can carry out the encoding process for the canonical form of a cyclic code by finding remainder polynomials that represent the parity check bits. We can carry out the decoding process by finding remainder polynomials that represent the syndrome of the received code word. Both the encoding and decoding can be carried out by a circuit that computes the remainder polynomial, an example of which is shown in Figure 7.1

In Figure 7.1, the squares represent registers that store a single bit, the circles labelled with the $+$ sign represent binary adders and the circles labelled with the $\times$ sign represent either binary multipliers or switches that are open if the control input is 0 and closed if it is 1. The symbols g_0, $g_1, \ldots, g_{n-k-1}$ represent either one of the inputs to the binary multipliers or the control inputs of the switches. The g_i are the coefficients of the divisor polynomial.

The one-bit registers act as delays, producing an output which is equal to the value stored in during the previous cycle. The circuit is controlled by a clocking mechanism that is not shown in the figure. There are two switches that control the movement of bits into and out of the circuit. The first switch either connects the input to, or disconnects it from, the first binary adder. The second switch connects the output from the last shift register to either the output of the circuit or feeds it back to the first binary adder via the first multiplier.

Before the encoding begins, the values in the registers are set to zero and the switches are set so that the input to the circuit is connected to the first binary adder, and the output of the last register is fed back to the first multiplier.

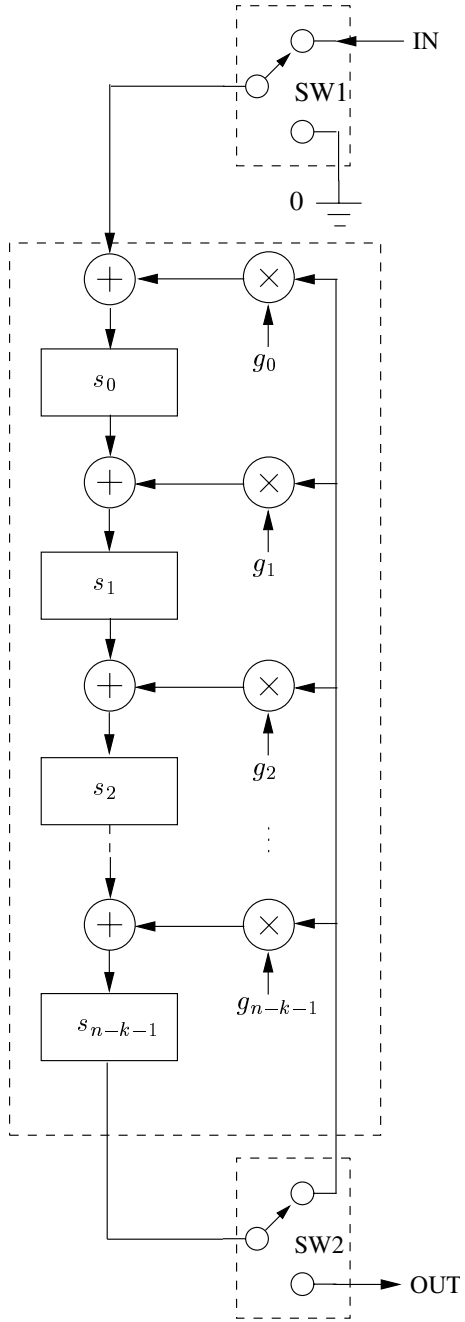

FIGURE 7.1
A circuit for computing remainder polynomials.

The synthetic division algorithm works by subtracting X^p times the divisor from the dividend for decreasing values of p. The feedback circuit accomplishes this in the following way.

The input to the circuit is the polynomial whose remainder is to be computed, represented as a string of bits with the coefficient of the highest power of X first. The bits are fed into the shift registers over the first $n - k$ clock cycles. While this is happening, the output of the last register will be 0.

At the next input step, the output of the last register will be 1, and this will be fed back to the adders after being multiplied by the appropriate values of the g_i. This has the effect of subtracting $g(X)X^p$ for some value of p from the input polynomial and shifting the polynomial coefficients up so that the coefficient of the second highest power of X is held in the last register. This process is repeated until there are no more bits in the input polynomial. The bits in the registers now represent the coefficients of the remainder. The input to the circuit is switched off and the output of the last register is now switched to the output of the circuit and the bits in the registers are shifted out.

EXAMPLE 7.20

Figure 7.2 shows the particular case of a circuit for computing remainders modulo $(1 + X^2 + X^4)$.

□

When this circuit is used for encoding, the g_i are the coefficients of the generator polynomial and the input polynomial is $m(X)X^{n-k}$. The output of the circuit is $d(X)$. Additional circuitry is required to concatenate the bits of the message word and the check bits of the code word.

When this circuit is used for decoding, the g_i are again the coefficients of the generator polynomial. The received word is the input and the output is the syndrome. Additional circuitry is required to add the syndrome to the received word to correct it if necessary.

Other circuits for encoding and decoding cyclic codes have been devised. Details of various algorithms and their hardware implementations can be found in the following references: [2], Sections 4.4, 4.5, 5.1, 5.2, 5.3 and 5.5; [3], Chapter 7, Section 8 and Chapter 9; [4], Sections 8.4 and 8.5; and [5], Sections 5.4 and 5.5.

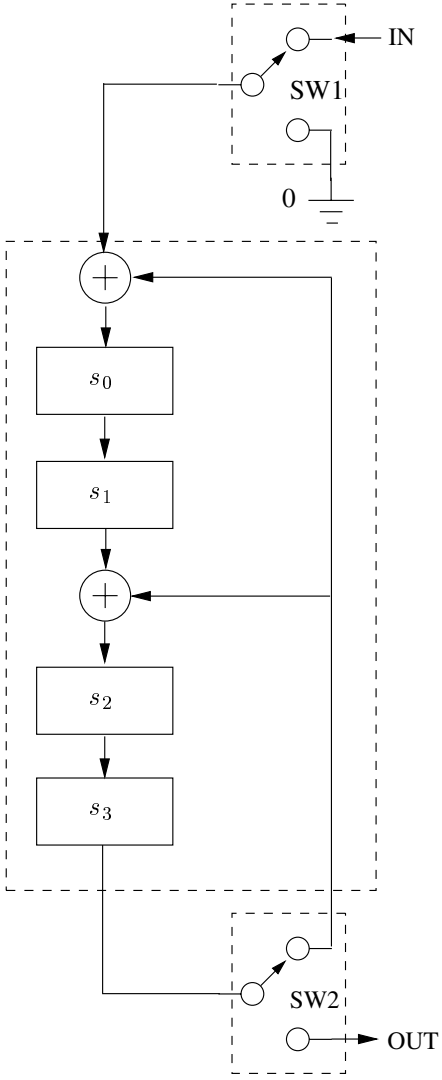

FIGURE 7.2
A circuit for computing remainder polynomials modulo $(1 + X^2 + X^4)$**.**

7.6 The Golay Code

An important example of a cyclic code that has had a number of practical applications is the *Golay code*.

The Golay code is a cyclic code with code words that are twenty-three bits long. It can be constructed from either of two generator polynomials, namely

$$g_1(X) = 1 + X^2 + X^4 + X^5 + X^6 + X^{10} + X^{11}, \tag{7.31}$$

for which the generator matrix is

$$G_1 = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & \dots & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & \dots & 0 \\ \vdots & & & & & & & \ddots & & & & & & & \vdots \\ 0 & \dots & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix},$$

and

$$g_2(X) = 1 + X + X^5 + X^6 + X^7 + X^9 + X^{11}, \tag{7.32}$$

for which the generator matrix is

$$G_2 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & \dots & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & \dots & 0 \\ \vdots & & & & & & & \ddots & & & & & & & \vdots \\ 0 & \dots & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

Note that both g_1 and g_2 divide $(X^{23} + 1)$, since

$$X^{23} + 1 = (1 + X)g_1(X)g_2(X). \tag{7.33}$$

A Golay code can be used to correct up to three errors in a 23-bit code word. It is an example of a perfect code.

7.7 Hamming Codes

DEFINITION 7.5 Hamming Code *Let $m \geq 3$. A* Hamming code *is a cyclic code with code words that are $2^m - 1$ bits long, having $2^m - m - 1$ information bits, m parity check bits and a minimum distance of* 3.

There is a simple way to construct the parity check matrix of a Hamming code. The columns of the parity check matrix are the binary representations of all the positive integers less than 2^m. If the columns are in the order of the numbers they represent, the syndrome will be the binary representation of the place where the error occurred.

EXAMPLE 7.21

For $m = 3$, the parity check matrix is

$$H = \begin{bmatrix} 0\,0\,0\,1\,1\,1\,1 \\ 0\,1\,1\,0\,0\,1\,1 \\ 1\,0\,1\,0\,1\,0\,1 \end{bmatrix}.$$

We interchange columns 1 and 7, columns 2 and 6 and columns 4 and 5 to reduce the parity check matrix to canonical form:

$$H_c = \begin{bmatrix} 1\,1\,0\,1\,1\,0\,0 \\ 1\,1\,1\,0\,0\,1\,0 \\ 1\,0\,1\,1\,0\,0\,1 \end{bmatrix}.$$

The canonical form of the generator matrix is

$$G_c = \begin{bmatrix} 1\,0\,0\,0\,1\,1\,1 \\ 0\,1\,0\,0\,1\,1\,0 \\ 0\,0\,1\,0\,0\,1\,1 \\ 0\,0\,0\,1\,1\,0\,1 \end{bmatrix}.$$

Interchanging columns 4 and 5, columns 2 and 6 and columns 1 and 7, we get

$$G = \begin{bmatrix} 1\,1\,0\,1\,0\,0\,1 \\ 0\,1\,0\,1\,0\,1\,0 \\ 1\,1\,1\,0\,0\,0\,0 \\ 1\,0\,0\,1\,1\,0\,0 \end{bmatrix}.$$

This gives us the code

$$\{0000000, 0001111, 0010110, 0011001, 0101010, 0101101, 0110011, 0111100,$$
$$1001011, 1001100, 1010101, 1011010, 1100110, 1101001, 1110000, 1111111\}.$$

To illustrate the error-correction procedure, suppose the third bit of the code word 1100110 gets corrupted, giving 1110110. The syndrome is $1110110H^T = 011$, indicating the third bit, as expected.

Similarly, the syndrome of 1110010 is $1110010H^T = 110$, indicating that the sixth bit has been corrupted, and that the correct code word is 1110000.

To show that the code generated by G is equivalent to a cyclic code, we replace the first row of G with the sum of the first and second rows to get

$$G_2 = \begin{bmatrix} 1\,0\,0\,0\,0\,1\,1 \\ 0\,1\,0\,1\,0\,1\,0 \\ 1\,1\,1\,0\,0\,0\,0 \\ 1\,0\,0\,1\,1\,0\,0 \end{bmatrix}.$$

We make the following columns interchanges: 1 and 7, 2 and 6, 3 and 6, 4 and 5, and 4 and 7 to get

$$G_c = \begin{bmatrix} 1\,1\,0\,1\,0\,0\,0 \\ 0\,1\,1\,0\,1\,0\,0 \\ 0\,0\,1\,1\,0\,1\,0 \\ 0\,0\,0\,1\,1\,0\,1 \end{bmatrix}.$$

This is the generator matrix of the cyclic code with generator polynomial $1 + X + X^3$.

□

DEFINITION 7.6 Primitive Polynomial *A* primitive polynomial *is a polynomial of degree p which divides $(X^{2^p-1} - 1)$ but does not divide $(X^k - 1)$ for any positive integer $k < p$.*

The generator polynomials of the Hamming codes are all primitive polynomials. The following table lists the generator polynomials for the Hamming codes for $3 \leq m \leq 10$.

Degree	Generator polynomial
3	$1 + X + X^3$
4	$1 + X + X^4$
5	$1 + X^2 + X^5$
6	$1 + X + X^6$
7	$1 + X^3 + X^7$
8	$1 + X^2 + X^3 + X^4 + X^8$
9	$1 + X^4 + X^9$
10	$1 + X^3 + X^{10}$

Hamming codes are perfect codes.

7.8 Cyclic Redundancy Check Codes

Cyclic Redundancy Check (CRC) codes are error-detecting codes that do not have any error-correcting capabilities. They are usually used in conjunction with an error-correcting code to improve performance of the system and to detect failures of the error-correction procedure.

When used for error detection, the syndrome of each received code word is computed. If it is non-zero, an error has been detected. The syndrome is not used to correct any error that may be detected.

CRC codes are commonly constructed from primitive polynomials, with a generator polynomial of the form

$$g(X) = (1 + X)p(X) \tag{7.34}$$

for some primitive polynomial p.

If the degree of the generator polynomial is r, the maximum block length is $n = 2^{r-1} - 1$ and the number of message bits is $n - r$. The usual practice is to select a large value of r and choose a block length slightly less than the maximum.

If $c(X)$ is a polynomial in the code generated by $g(X)$ then it is a multiple of $g(X)$, and so

$$c(X) \bmod G(X) = 0. \tag{7.35}$$

If the code word is corrupted, there is an error polynomial $e(X)$ such that the corrupted code word is $c(X) + e(X)$. Then

$$\begin{aligned}(c(X) + e(X)) \bmod g(X) &= c(X) \bmod g(X) + e(X) \bmod g(X) \\ &= e(X) \bmod g(X).\end{aligned} \tag{7.36}$$

The error will be undetectable if $e(X) \bmod g(X) = 0$. We can use this fact to determine which generator polynomials will detect various classes of errors.

If a single bit is corrupted, we will have $e(X) = X^k$ for some k. If $g(X)$ has two or more terms, it will not divide X^k and single bit errors will be detected.

If two bits are corrupted, then $e(X) = X^k + X^m = X^k(1 + X^{m-k})$ for some k and m with $k < m$. If $g(X)$ has two or more terms and does not divide $(1 + X^j)$ for $j = 1, 2, \ldots, n$, then it will not divide any such error polynomial and two bit errors will be detected.

If there is a burst of r errors in succession, then $e(X) = X^j(1 + X + \ldots + X^{r-1})$ for some j. If $g(X)$ is polynomial of degree r, it will not divide $e(X)$. In fact, if $g(X)$ is of degree r all bursts of not more than r errors will be detected.

EXAMPLE 7.22

$1 + X + X^4$ is a primitive polynomial. We construct a CRC by taking

$$g(X) = (1 + X)(1 + X + X^4) = 1 + X^2 + X^4 + X^5$$

as our generator polynomial. The degree of $g(X)$ is 5; so we can have up to 15 bits in our code words. If we choose to have code words that are 11 bits long, the generator matrix is

$$G = \begin{bmatrix} 1\,0\,1\,0\,1\,1\,0\,0\,0\,0\,0 \\ 0\,1\,0\,1\,0\,1\,1\,0\,0\,0\,0 \\ 0\,0\,1\,0\,1\,0\,1\,1\,0\,0\,0 \\ 0\,0\,0\,1\,0\,1\,0\,1\,1\,0\,0 \\ 0\,0\,0\,0\,1\,0\,1\,0\,1\,1\,0 \\ 0\,0\,0\,0\,0\,1\,0\,1\,0\,1\,1 \end{bmatrix}.$$

Replacing the first row with the sum of the first and third rows and replacing the second row with the sum of the second and fourth rows, we get

$$G_2 = \begin{bmatrix} 1\,0\,0\,0\,0\,1\,1\,1\,0\,0\,0 \\ 0\,1\,0\,0\,0\,0\,1\,1\,1\,0\,0 \\ 0\,0\,1\,0\,1\,0\,1\,1\,0\,0\,0 \\ 0\,0\,0\,1\,0\,1\,0\,1\,1\,0\,0 \\ 0\,0\,0\,0\,1\,0\,1\,0\,1\,1\,0 \\ 0\,0\,0\,0\,0\,1\,0\,1\,0\,1\,1 \end{bmatrix}.$$

Replacing the first row with the sum of the first and sixth rows, replacing the third row with the sum of the third and fifth rows, and replacing the fourth row with the sum of the fourth and sixth rows, we get the canonical form of the generator matrix:

$$G_c = \begin{bmatrix} 1\,0\,0\,0\,0\,0\,1\,0\,0\,1\,1 \\ 0\,1\,0\,0\,0\,0\,1\,1\,1\,0\,0 \\ 0\,0\,1\,0\,0\,0\,0\,1\,1\,1\,0 \\ 0\,0\,0\,1\,0\,0\,0\,0\,1\,1\,1 \\ 0\,0\,0\,0\,1\,0\,1\,0\,1\,1\,0 \\ 0\,0\,0\,0\,0\,1\,0\,1\,0\,1\,1 \end{bmatrix}.$$

The parity check matrix for the code is therefore

$$H = \begin{bmatrix} 1\,1\,0\,0\,1\,0\,1\,0\,0\,0\,0 \\ 0\,1\,1\,0\,0\,1\,0\,1\,0\,0\,0 \\ 0\,1\,1\,1\,1\,0\,0\,0\,1\,0\,0 \\ 1\,0\,1\,1\,1\,1\,0\,0\,0\,1\,0 \\ 1\,0\,0\,1\,0\,1\,0\,0\,0\,0\,0 \end{bmatrix}.$$

11001011001 is a code word. Its syndrome is

$$11001011001H^T = 00000,$$

as expected.

The CRC should be able to detect any one bit error. Suppose the fifth bit of the code word 11001011001 is corrupted, giving 11000011001. Its syndrome is

$$11000011001H^T = 10110,$$

indicating an error has occurred.

Suppose the third and eight bits of the code word 11001011001 are corrupted, giving 11101010001. Its syndrome is

$$11101010001H^T = 00110,$$

again indicating an error has occurred.

Since the degree of the generator polynomial is 5, the CRC should be able to detect a burst of five errors. Such a burst will change the code word 11001011001 to 11000100101. The syndrome of the corrupted code word is

$$11000100101H^T = 00001,$$

indicating that an error has occurred.

However, the CRC should not be able to detect a burst of six or more errors. A burst of six errors will change the code word 11001011001 to 11110101101. The syndrome of the corrupted code word is

$$11110101101H^T = 00000,$$

so this burst of errors will not be detected. □

A number of standard CRC codes have been defined. These include the CRC-12 code, with generator polynomial

$$g_{12}(X) = 1 + X^2 + X^9 + X^{10} + X^{11} + X^{12} \tag{7.37}$$

and the CRC-ANSI code with generator polynomial

$$g_{ANSI}(X) = X + X^2 + X^{15} + X^{17}. \tag{7.38}$$

CRC codes can detect bursts of errors, but cannot be used to correct them. Codes that can be used to correct bursts of errors will be discussed in the next chapter.

7.9 Reed-Muller Codes

Let $n = 2^m$ for some positive integer m, and let $v_0 \in \mathbb{B}^n$ be the word consisting of 2^m 1's. Let $v_1, v_2, \ldots, v_m$ be the rows of the matrix that has all possible m-tuples as columns.

EXAMPLE 7.23

If $m = 2$, $n = 4$ and the matrix whose columns are all possible 2-tuples is

$$V_2 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}.$$

So for $m = 2$, $v_0 = 1111$, $v_1 = 0011$, $v_2 = 0101$. ∎

EXAMPLE 7.24

If $m = 3$, $n = 8$ and the matrix whose columns are all possible 3-tuples is

$$V_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

So for $m = 3$, $v_0 = 11111111$, $v_1 = 00001111$, $v_2 = 00110011$ and $v_3 = 01010101$. ∎

Define $v \otimes w$ to be the component-wise product of elements in $\mathbb{B}^n$, that is, the kth component of $v \otimes w$ is the product of the kth component of the v and the kth component of w. Since the product in $\mathbb{B}$ is commutative, it follows that $v \otimes w = w \otimes v$.

EXAMPLE 7.25

For $m = 2$, we have $v_0 = 1111$, $v_1 = 0011$, $v_2 = 0101$. The products are

$$\begin{aligned} v_0 \otimes v_0 &= 1111 \\ v_0 \otimes v_1 &= 0011 \\ v_0 \otimes v_2 &= 0101 \\ v_1 \otimes v_1 &= 0011 \\ v_1 \otimes v_2 &= 0001 \\ v_2 \otimes v_2 &= 0101. \end{aligned}$$

∎

EXAMPLE 7.26

For $m = 3$, $v_0 = 11111111$, $v_1 = 00001111$, $v_2 = 00110011$ and $v_3 = 01010101$. The products are

$$v_0 \otimes v_0 = 11111111$$

$$
\begin{aligned}
v_0 \otimes v_1 &= 00001111 \\
v_0 \otimes v_2 &= 00110011 \\
v_0 \otimes v_3 &= 01010101 \\
v_1 \otimes v_1 &= 00001111 \\
v_1 \otimes v_2 &= 00000011 \\
v_1 \otimes v_3 &= 00000101 \\
v_2 \otimes v_2 &= 00110011 \\
v_2 \otimes v_3 &= 00010001 \\
v_3 \otimes v_3 &= 01010101.
\end{aligned}
$$

□

We always have $v_i \otimes v_i = v_i$ and $v_0 \otimes v_i = v_i$ for all i.

DEFINITION 7.7 rth-order Reed-Muller Code *The rth-order Reed-Muller code with word length $n = 2^m$ is the linear code in $\mathbb{B}^n$ whose basis is the set of vectors $v_0, v_1, \ldots, v_m$, together with all products of r or fewer of these vectors.*

The rth-order Reed-Muller code has

$$k = 1 + \binom{m}{1} \cdots + \binom{m}{r} \tag{7.39}$$

information bits and a distance of 2^{m-r}.

EXAMPLE 7.27

For $m = 3$ and $r = 2$, the basis is v_0, v_1, v_2, v_3, $v_1 \otimes v_2 = 00000011$, $v_1 \otimes v_3 = 00000101$, $v_2 \otimes v_3 = 00010001$.

The generator matrix is

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

□

The parity check matrix of the code is the matrix whose rows are the vectors v_0, $v_1, \ldots, v_m$, together with the vectors that are the products of no more than $m-r-1$ of these vectors.

EXAMPLE 7.28

For $m = 3$ and $r = 2$, the parity check matrix is

$$H = \begin{bmatrix} 1\,1\,1\,1\,1\,1\,1\,1 \\ 0\,0\,0\,0\,1\,1\,1\,1 \\ 0\,0\,1\,1\,0\,0\,1\,1 \\ 0\,1\,0\,1\,0\,1\,0\,1 \end{bmatrix}.$$

In this case $m - r - 1 = 0$, so there are no products in the parity check matrix.

Note that the parity check matrix for the 7-bit Hamming code is a sub-matrix of this matrix. ∎

Reed-Muller codes are equivalent to cyclic codes with an added overall check bit.

To detect and correct errors, it is possible to determine 2^{m-r} independent values for each information bit from the bits of the code word. If there are no errors, all these determinations will give the same value. If there are errors, some of these determinations will give 0 and some will give 1. The more frequently occurring value is taken as the correct value of the bit.

EXAMPLE 7.29

To illustrate the decoding process, consider the case $m = 3$ and $r = 1$, for which the generator matrix is

$$G = \begin{bmatrix} 1\,1\,1\,1\,1\,1\,1\,1 \\ 0\,0\,0\,0\,1\,1\,1\,1 \\ 0\,0\,1\,1\,0\,0\,1\,1 \\ 0\,1\,0\,1\,0\,1\,0\,1 \end{bmatrix}.$$

The decoding process depends on the fact that all arithmetic is done in base 2, so that $1 + 1 = 0$. If the word to be encoded is $a_0a_1a_2a_3$, and the resulting code word is $b_0b_1b_2b_3b_4b_5b_6b_7$, where each a_i and b_j denote a single bit, then the generator matrix gives us the following equations:

$$\begin{aligned} a_0 &= b_0 \\ a_0 + a_3 &= b_1 \\ a_0 + a_2 &= b_2 \\ a_0 + a_2 + a_3 &= b_3 \\ a_0 + a_1 &= b_4 \\ a_0 + a_1 + a_3 &= b_5 \\ a_0 + a_1 + a_2 &= b_6 \\ a_0 + a_1 + a_2 + a_3 &= b_7 \end{aligned}$$

Adding the equations in pairs gives us the following independent expressions for a_1, a_2 and a_3:

$$\begin{aligned} a_1 &= b_0 + b_4 \\ a_1 &= b_2 + b_6 \\ a_1 &= b_3 + b_7 \\ a_1 &= b_1 + b_5; \end{aligned}$$

$$\begin{aligned} a_2 &= b_0 + b_2 \\ a_2 &= b_1 + b_3 \\ a_2 &= b_4 + b_6 \\ a_2 &= b_5 + b_7; \end{aligned}$$

$$\begin{aligned} a_3 &= b_0 + b_1 \\ a_3 &= b_2 + b_3 \\ a_3 &= b_4 + b_5 \\ a_3 &= b_6 + b_7. \end{aligned}$$

We also have the following for a_0:

$$\begin{aligned} a_0 &= b_0 \\ a_0 &= b_1 + b_6 + b_7 \\ a_0 &= b_2 + b_5 + b_7 \\ a_0 &= b_3 + b_4 + b_7. \end{aligned}$$

To decode a code word, we simply compute these values from the b_j and choose the a_i from them.

Suppose 0101 is coded to give 01011010, but one bit is corrupted to give 00011010, so that $b_0 = 0$, $b_1 = 0$, $b_2 = 0$, $b_3 = 1$, $b_4 = 1$, $b_5 = 0$, $b_6 = 1$, and $b_7 = 0$.

We compute four values for each of the a_1 using the equations above. For a_0 we get

$$\begin{aligned} b_0 &= 0 \\ b_1 + b_6 + b_7 &= 1 \\ b_2 + b_5 + b_7 &= 0 \\ b_3 + b_4 + b_7 &= 0. \end{aligned}$$

Three of the four values are 0, so we take $a_0 = 0$.

For a_1 we get

$$\begin{aligned} b_0 + b_4 &= 1 \\ b_2 + b_6 &= 1 \\ b_3 + b_7 &= 1 \\ b_1 + b_5 &= 0; \end{aligned}$$

Three of the four values are 1, so we take $a_1 = 1$.

For a_2 we get

$$\begin{aligned} b_0 + b_2 &= 0 \\ b_1 + b_3 &= 1 \\ b_4 + b_6 &= 0 \\ b_5 + b_7 &= 0; \end{aligned}$$

Three of the four values are 0, so we take $a_2 = 0$.

For a_3 we get

$$\begin{aligned} b_0 + b_1 &= 0 \\ b_2 + b_3 &= 1 \\ b_4 + b_5 &= 1 \\ b_6 + b_7 &= 1. \end{aligned}$$

Three of the four values are 1, so we take $a_3 = 0$. This makes the decoded word 0101, which is correct. ∎

7.10 Exercises

1. Compute the following sums of polynomials with coefficients in the specified ring:

   (a) $(3 + 4X^2 + 9X^4) + (X - 2X^3 + X^5)$, coefficients in $\mathbb{Z}$;

   (b) $(3 - 4X^2 + 9X^4) + (X + 2X^2 - X^4)$, coefficients in $\mathbb{Z}$;

   (c) $(1 + X + X^2) + (X + X^3)$, coefficients in $\mathbb{B}$;

   (d) $(1 + X + X^4) + (X^2 + X^4 + X^6)$, coefficients in $\mathbb{B}$;

   (e) $(1 + X + X^4) + (X^2 + X^4 + X^6)$, coefficients in $\mathbb{Z}_3$;

   (f) $(1 + X + 2X^4) + (X^2 + 2X^4 + 3X^6)$, coefficients in $\mathbb{Z}_4$;

(g) $(4 + 3X + 2X^2) + (2X + X^3)$, coefficients in $\mathbb{Z}_5$;

(h) $(\pi + X) + (1 - \pi X)$, coefficients in $\mathbb{R}$.

2. Compute the following products of polynomials with coefficients in the specified ring:

(a) $(3 + 4X^2 + 9X^4) \times (X - 2X^3 + X^5)$, coefficients in $\mathbb{Z}$;

(b) $(3 - 4X^2 + 9X^4) \times (X + 2X^2 - X^4)$, coefficients in $\mathbb{Z}$;

(c) $(1 + X + X^2) \times (X + X^3)$, coefficients in $\mathbb{B}$;

(d) $(1 + X + X^4) \times (X^2 + X^4 + X^6)$, coefficients in $\mathbb{B}$;

(e) $(1 + X + X^4) \times (X^2 + X^4 + X^6)$, coefficients in $\mathbb{Z}_3$;

(f) $(1 + X + 2X^4) \times (X^2 + 2X^4 + 3X^6)$, coefficients in $\mathbb{Z}_4$;

(g) $(4 + 3X + 2X^2) \times (2X + X^3)$, coefficients in $\mathbb{Z}_5$;

(h) $(\pi + X) \times (1 - \pi X)$, coefficients in $\mathbb{R}$.

3. Show that if p is a prime number, $(1 + X)^p = (1 + X^p)$ if the addition and multiplication operations are performed in $\mathbb{Z}_p$.

4. Compute the following sums of polynomials with coefficients in $\mathbb{B}$, where the polynomials have been represented as elements of $\mathbb{B}^n$ for some n:

(a) $1001 + 1010$;

(b) $11001 + 10010$;

(c) $101001 + 101010$;

(d) $1010001 + 1010110$;

(e) $01010001 + 10101110$.

5. Compute the following products of polynomials with coefficients in $\mathbb{B}$, where the polynomials have been represented as elements of $\mathbb{B}^n$ for some n:

(a) 1001×1010;

(b) 11001×10010;

(c) 101001×101010;

(d) 1010001×1010110;

(e) 01010001×10101110.

6. Perform the following synthetic division operations for the polynomials below, with coefficients in the specified field:

(a) divide $(X^3 + 2X^2 + 3X + 4)$ by $(X^2 + X + 1)$, with coefficients in $\mathbb{R}$;

(b) divide $(X^5 + X^3 + X + 1)$ by $(X^3 + X^2 + 1)$, with coefficients in $\mathbb{B}$;

(c) divide $(X^7 + 1)$ by $(X^3 + X + 1)$, with coefficients in $\mathbb{B}$;

(d) divide $(2X^4 + X^2 + 2)$ by $(X^2 + 2)$, with coefficients in $\mathbb{Z}_3$;

(e) divide $(X^5 + 4X^3 + 3X^2 + 2X + 1)$ by $(X^2 + 3)$, with coefficients in $\mathbb{Z}_5$.

7. Compute the product of the polynomials p and q modulo r, when p, q and r are as given below, with coefficients from the specified field:

(a) $p(X) = (X^2 + 5X + 7)$, $q(x) = (X^3 + 11)$, $r(X) = (X^3 + 2X)$, with coefficients in $\mathbb{R}$;

(b) $p(X) = (X^2 + X + 1)$, $q(x) = (X^3 + 1)$, $r(X) = (X^3 + X)$, with coefficients in $\mathbb{B}$;

(c) $p(X) = (X^3 + X + 1)$, $q(x) = (X^5 + X)$, $r(X) = (X^4 + X^2 + 1)$, with coefficients in $\mathbb{B}$;

(d) $p(X) = (X^3+2X+1)$, $q(x) = (2X^4+X^3+2)$, $r(X) = (X^3+2X+1)$, with coefficients in $\mathbb{Z}_3$;

(e) $p(X) = (X^3 + 4X + 2)$, $q(x) = (3X^4 + 2X^3 + 4)$, $r(X) = (X^3 + 3X + 2)$, with coefficients in $\mathbb{Z}_5$.

8. Use the Remainder Theorem to compute the following:

(a) the remainder of $(X^5 + 7X^3 + 9X)$ when divided by $(X^2 + 13)$, with coefficients in $\mathbb{R}$;

(b) the remainder of $(X^5 + X^3 + X)$ when divided by $(X^2 + 1)$, with coefficients in $\mathbb{B}$;

(c) the remainder of $(X^9 + X^6 + X^3)$ when divided by $(X^4 + 1)$, with coefficients in $\mathbb{B}$;

(d) the remainder of $(X^9 + X^6 + X^3)$ when divided by $(X^4 + 1)$, with coefficients in $\mathbb{Z}_3$;

(e) the remainder of $(X^8 + 4X^5 + 2X^2)$ when divided by $(X^3 + 3)$, with coefficients in $\mathbb{Z}_5$.

9. Draw up the multiplication table of $\mathbb{B}_4[X]/(X^4 + 1)$.

10. Write down the generator polynomials of the cyclic codes whose generator matrices are given below:

(a)

$$G = \begin{bmatrix} 1\,0\,1\,1\,0\,0\,0 \\ 0\,1\,0\,1\,1\,0\,0 \\ 0\,0\,1\,0\,1\,1\,0 \\ 0\,0\,0\,1\,0\,1\,1 \end{bmatrix};$$

(b)

$$G = \begin{bmatrix} 1\,1\,0\,0\,1\,1\,0\,0 \\ 0\,1\,1\,0\,0\,1\,1\,0 \\ 0\,0\,1\,1\,0\,0\,1\,1 \end{bmatrix};$$

(c)

$$G = \begin{bmatrix} 1\,0\,0\,1\,0\,0\,1\,0\,0 \\ 0\,1\,0\,0\,1\,0\,0\,1\,0 \\ 0\,0\,1\,0\,0\,1\,0\,0\,1 \end{bmatrix};$$

(d)

$$G = \begin{bmatrix} 1\,1\,0\,0\,0\,1\,1\,0\,0\,0 \\ 0\,1\,1\,0\,0\,0\,1\,1\,0\,0 \\ 0\,0\,1\,1\,0\,0\,0\,1\,1\,0 \\ 0\,0\,0\,1\,1\,0\,0\,0\,1\,1 \end{bmatrix}.$$

11. Write down the generator matrices of the cyclic codes whose generator polynomials and lengths are given below:

    (a) $g(X) = (1 + X^3)$, $n = 6$;

    (b) $g(X) = (1 + X)$, $n = 7$;

    (c) $g(X) = (1 + X + X^4 + X^5)$, $n = 8$;

    (d) $g(X) = (1 + X^3 + X^6)$, $n = 9$;

    (e) $g(X) = (1 + X + X^5 + X^6)$, $n = 10$.

12. Find the parity check polynomials of the cyclic codes in the previous exercise.

13. Find all the cyclic codes of length 4.

14. Find all the cyclic codes of length 5.

15. Find all the cyclic codes of length 6.

16. Is

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

    the generator matrix of a cyclic code?

17. Write down the synthetic division of $X^5 + X^4 + 1$ by $X^2 + 1$. Draw the circuit in Figure 7.1 with $g_0 = 1$, $g_1 = 0$ and $g_2 = 1$. Trace the operation of the circuit when the input is 110001 and match the states of the registers with the stages of the synthetic division.

18. Construct the parity check matrix of the Hamming code for $m = 4$, and show that it is equivalent to the cyclic code with $n = 15$ and generator polynomial $(1 + X + X^4)$.

*19. Derive a decoding procedure for the Reed-Muller code with $m = 3$ and $r = 2$, and apply it to the received code words 01010101 and 10000011.

*20. Let $g(X)$ be a polynomial with coefficients in $\mathbb{B}$ and let n be the smallest integer such that $g(X)$ is a factor of $X^n + 1$. Show that the minimum distance of the cyclic code generated by $g(X)$ is at least 3. Construct two examples, one where the condition is satisfied and the minimum distance is 3, and one where the condition is not satisfied and there is a code word of weight 2.

7.11 References

[1] J. B. Fraleigh, *A First Course in Abstract Algebra,* 5th ed., Addison-Wesley, Reading, MA, 1994.

[2] S. Lin, *An Introduction of Error-Correcting Codes,* Prentice-Hall, Englewood Cliffs, NJ, 1970.

[3] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes Part I,* North-Holland, Amsterdam, 1977.

[4] W. W. Peterson, *Error-correcting Codes,* MIT Press, Cambridge, MA, 1961.

[5] R. B. Wells, *Applied Coding and Information Theory for Engineers,* Prentice-Hall, Upper Saddle River, NJ, 1999.