

# 12. POLYNOMIAL CODES

## §12.1. Error Detection and Error Correction

Generally when people talk of *codes* they are thinking of cryptography where messages are coded for reasons of secrecy. But there's another reason why we might want to code a message. We might want to transmit it across a *noisy* channel, that is, a channel that could corrupt the text.

Of course we could simply repeat the message. Wherever the two copies differ, the receiver will know that an error has occurred. But the receiver won't know, for each error, which version is correct. We would need to transmit the message *three* times for the receiver to be able to correct the error – whichever alternative occurs twice will almost certainly be correct. However, having to transmit each message three times seems rather inefficient.

When information is transmitted electronically it is sent as strings of 0's and 1's. One very widespread system is the ASCII code where every letter, digit and punctuation symbol has a 7 bit code, that is, a string of 7 pulses each representing either a "0" or a "1".

Generally an 8'th bit (called a *check bit*) is sent after each seven based on how many 1's there are in the 7 bits. In one such system, if the number of 1's is odd (i.e. 1,3,5 or 7) the 8'th bit is "1" and if the number of 1's is even (i.e. 0,2,4 or 6) the 8'th bit would be "0". This rule ensures that the number of 1's in each block of 8 bits is always even. When the message is received the number of 1's in each block is counted. If there is a single error in a block (that is, a "0" being changed to a "1" or vice-versa) the number of 1's will be odd. The receiver will then know that an error has taken place in that block and, if possible, will ask the transmitter to send that block again. This system is widely used when a computer communicates with another device.

The system is very useful but it has two problems.

- (1) If there are 2 errors (or 4 or 6) the errors will go undetected. In practice, if the error rate is low and the blocks are small, the chance of getting 2 errors in a single block is small enough to be ignored.
- (2) There needs to be 2-way communication. In many applications this is not feasible.

In a compact disc player, the music is picked up digitally as pulses that represent 0's and 1's. If an error is detected there is no opportunity to say "hey run that by me again!" The laser head has had to move on.

## § 12.2. Polynomials

A **polynomial** is an expression of the form:

$$a(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0.$$

The symbol "x" is called an **indeterminate** and simply plays the role of a place marker. The role of the "x" is to provide positions in the expression which can be replaced (substituted) by a value. The numbers  $a_0, a_1, \dots$  are called the **coefficients** of the polynomial.

The coefficients can be rational numbers (from the set **Q**), real numbers (from the set **R**), or complex numbers (from the set **C**). Or they can be integers mod  $p$ , where  $p$  is prime, coming from the system  $\mathbf{Z}_p$ . All these number systems, called **fields**, have the property that  $1/x$  exists for every non-zero  $x$ . (This somewhat loose description will do for our present purposes. An exact definition comprises 11 separate properties, or axioms.)

The **degree** of a polynomial is the largest power of  $x$  that occurs with a non-zero coefficient. That is, if  $a(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$ , the degree of  $a(x)$  is  $n$  (provided  $a_n \neq 0$ ). We can write  $\deg a(x) = n$ .

Polynomials of degree 2 are **quadratics**, of the form  $ax^2 + bx + c$  (where  $a \neq 0$ ).

Polynomials of degree 1 are the **linear polynomials** such as  $2x + 3$  and  $\frac{1}{2}x - \frac{1}{4}$ .

Polynomials of degree 0 are the non-zero **constant polynomials** such as  $-3$  and  $\frac{3}{4}$ .

There is one polynomial for which the degree remains undefined. It is the **zero polynomial**, 0.

The coefficient of  $x^n$ , for a polynomial of degree  $n$ , is called its **leading coefficient**. For example the leading coefficient of  $3x^2 - x + 5$  is 3. But beware. The leading coefficient does not always come first – the leading coefficient of  $1 - x^2$  is  $-1$ , not 1.

A **monic** polynomial is one where the leading coefficient is 1. Clearly every non-zero polynomial can be made monic by dividing it by its leading coefficient.

If  $F$  is a field (for example  $F$  might be  $\mathbf{Q}$ , the system of rational numbers) we denote the set of all polynomials with coefficients coming from  $F$  by the symbol  $F[x]$ .

**Example 1:** The polynomials in  $\mathbf{Z}_2[x]$  are:

the constant polynomials 0, 1 ;

the linear polynomials  $x, x + 1$ ;

the quadratics  $x^2, x^2 + 1, x^2 + x, x^2 + x + 1$ ;

the cubics  $x^3, x^3 + 1, x^3 + x, x^3 + x + 1, x^3 + x^2, x^3 + x^2 + 1, x^3 + x^2 + x, x^3 + x^2 + x + 1$ ;

and so on for higher degree.

Polynomials are added, subtracted, and multiplied in the usual way. Just remember to use the appropriate field when doing your arithmetic.

**Example 2:** In  $\mathbf{R}[x]$ ,  $(x + 1)^2 = x^2 + 2x + 1$  but in  $\mathbf{Z}_2[x]$  we should write it as  $x^2 + 1$  since 2 mod 2 is zero.

With these operations the system  $F[x]$  behaves very much like a field itself, but with one important difference. In a field nearly every number has an inverse under multiplication (in fact 0 is the only exception). Most polynomials, on the other hand, do *not* have inverses.

For example, since  $\frac{1}{x}$  and  $\frac{1}{1-x}$  are not polynomials,  $x$  and  $1 - x$  do not have (polynomial) inverses. In fact the *only* polynomials with inverses are the non-zero constant polynomials such as  $-2$  (whose inverse is the constant polynomial  $-\frac{1}{2}$ ).

Now it *is* possible to write  $\frac{1}{1-x}$  as  $1 + x + x^2 + \dots$  but although this looks like a polynomial it has infinitely many terms while a polynomial, by definition, has only finitely many. An expression like  $1 + x + x^2 + \dots$  is called a **power series**.

The system  $F[x]$  of polynomials over a field behaves much more like the system of integers (where only  $\pm 1$  have integer inverses).

As mentioned earlier, one polynomial does not usually divide exactly into another. Like the system of integers we are usually left with a **remainder**. We get exact divisibility precisely when the remainder is zero. Furthermore this remainder, when it isn't zero, is in

some sense *smaller* than whatever we are dividing by. For polynomials, *smaller* means "of smaller degree".

The process of obtaining the remainder on dividing one polynomial by another is very similar to the familiar long-division algorithm.

**Example 3:**

$$\begin{array}{r}
 \phantom{x^2 - 2x + 7} \overline{2x + 4} \\
 x^2 - 2x + 7 \overline{) 2x^3 \phantom{+ 5x} - 3} \\
 \underline{2x^3 - 4x^2 + 14x} \phantom{- 3} \\
 4x^2 - 9x - 3 \\
 \underline{4x^2 - 8x + 28} \\
 -x - 31
 \end{array}$$

From this calculation we compute the remainder on dividing  $2x^3 + 5x - 3$  by  $x^2 - 2x + 7$  to be  $-x - 31$ . Note that the remainder has lower degree than that of  $x^2 - 2x + 7$ , the polynomial we are dividing by. Note also how we write the terms neatly underneath others of the same degree.

The result of the calculation can also be expressed as:

$$2x^3 + 5x - 3 = (x^2 - 2x + 7)(2x + 4) + (-x - 31).$$

**Theorem 1:(Division Algorithm)**

If  $a(x)$ ,  $b(x)$  are polynomials and  $b$  is non-zero then  $a(x) = b(x)q(x) + r(x)$  for some polynomials  $q(x)$  and  $r(x)$  where either  $r(x) = 0$  or  $\deg r(x) < \deg b(x)$ .

The polynomial  $q(x)$  is called the **quotient** and  $r(x)$  is called the **remainder**.

If the remainder on dividing  $a(x)$  by  $b(x)$  is zero we say that  $b(x)$  **divides**  $a(x)$ , or that  $a(x)$  is a **multiple** of  $b(x)$ . If we can't be bothered saying it in words we just write " $b(x) \mid a(x)$ " and read it as " $b(x)$  divides  $a(x)$ ".

If  $f(x) \in F[x]$ , in other words  $f(x)$  is a polynomial in  $x$  with coefficients coming from the field  $F$ , and  $\alpha$  is a number from  $F$ , or from some larger field that contains  $F$ , we define  $f(\alpha)$  to be the number that results from replacing, or **substituting**,  $\alpha$  for  $x$  in the polynomial  $f(x)$ .

For example if  $f(x) = x^2 + x - 2$  then  $f(2) = 4 + 2 - 2 = 4$ ,  $f(0) = -2$  and  $f(1)=0$ .

The following theorem connects the ideas of substitution and remainder.

**Theorem 2:(Remainder Theorem)** The remainder on dividing  $f(x)$  by  $x - k$  is  $f(k)$ .

**Proof:** By the Division Algorithm,  $f(x) = (x - a)q(x) + r(x)$  for some polynomials  $q(x)$ ,  $r(x)$  and the remainder  $r(x)$  is either zero or has degree less than 1. In other words  $r(x)$  must be a constant polynomial, so we can drop the " $(x)$ " and just call it  $r$ . Now substituting  $x = a$  into the equation  $f(x) = (x - a)q(x) + r$ , we get  $f(a) = r$ .

**Corollary:** The polynomial  $f(x)$  is divisible by  $x - \alpha$  if and only if  $f(\alpha) = 0$ .

A **root** of a polynomial  $f(x)$  is a number,  $\alpha$ , so that  $f(\alpha) = 0$ . Solving a polynomial equation  $f(x) = 0$  therefore means finding all its roots.

But where do we look for potential roots? From the coefficient field. But here we have to be a little careful. Does the polynomial  $f(x) = x^2 + 1$  have any roots? That depends

on the field of coefficients. If  $f(x) \in \mathbf{R}[x]$ , the answer is no, if  $f(x) \in \mathbf{C}[x]$  the answer is yes,  $\pm i$ , if  $f(x) \in \mathbf{Z}_2[x]$ , the answer is again yes, but this time we have a double root of 1, if  $f(x) \in \mathbf{Z}_3[x]$  the answer is no since putting  $x = 0, 1$  and  $2$  in  $x^2 + 1$  gives us the values  $1, 2, 2$  (never zero). The corollary to the Remainder Theorem can be expressed by saying that the number  $\alpha$  is a root of  $f(x)$  if and only if  $x - \alpha$  is one of its factors. Now the polynomial  $x - \alpha$  has degree 1 and it is a linear polynomial. So there is a connection between linear factors and roots of a polynomial.

**Theorem 3:** A polynomial has a root if and only if it has a linear factor.

**Proof:** Whenever we have a root,  $\alpha$ , we have a linear factor  $x - \alpha$ . Conversely having a linear factor  $bx + c$  for a polynomial means that we have a root  $\alpha = -c/b$

If we know one root of a polynomial we can use the remainder theorem and divide by the corresponding linear factor. The other roots will then be roots of the quotient.

## §12.3. Complete Polynomials

**Defn:** A polynomial  $f(x) \in \mathbf{Z}_2[x]$ , of degree  $n$ , is **complete** if  $f(t) = 0$  implies that every non-zero element of the form  $a_{n-1}t^{n-1} + a_{n-2}t^{n-2} + \dots + a_1t + a_0$  can be expressed as a power of  $t$ . (Here the number  $t$  comes from some larger field that contains  $\mathbf{Z}_2$ .)

**Example 4:**  $x^3 + x + 1$  is complete. To see this we put  $t^3 + t + 1 = 0$ .

This means that  $t^3 = -t - 1$ , but since the coefficients come from  $\mathbf{Z}_2$  where  $-1 = +1$ , we can write  $t^3 = t + 1$ .

$$\therefore t^4 = t^2 + t.$$

$$\therefore t^5 = t^3 + t^2 = (t + 1) + t^2 = t^2 + t + 1.$$

$$\therefore t^6 = t^3 + t^2 + t = (t + 1) + t^2 + t = t^2 + 1, \text{ remembering that } t + t = 0 \text{ mod } 2.$$

The seven powers of  $t$ :  $1, t, t^2, t^3, t^4, t^5, t^6$  reduce to:

$1, t, t + 1, t^2, t^2 + 1, t^2 + t, t^2 + t + 1$ , the seven polynomial expressions in  $t$ .

## §12.4. Polynomial Codes: How They Work

To begin with we need a convention as to how a string of length  $n+1$  can be represented by a polynomial with the bits representing the coefficients of a polynomial over  $\mathbf{Z}_2$ . We could take the first bit to represent the highest power of  $x$ , down to the last bit representing the constant term. Or we could consider the first bit to be the constant term and proceed up through the increasing powers of  $x$ . It doesn't matter which convention is adopted, so long as the transmitter and the receiver use the same one.

**Example 5:** Using one convention the string 10011 would be represented by  $x^4 + x + 1$ . Using the other it would become  $x^4 + x^3 + 1$ .

**SETUP**

- \* Choose a complete polynomial of degree  $n$ :  $p(x) \in \mathbf{Z}_2[x]$ .
- \* Define  $t$  to be a root of  $p(x)$ .
- \* Construct a table of values of  $t^k$  for  $k$  up to  $t^{2^n-2}$ , expressing each one in terms of powers of  $t$  up to  $t^{n-1}$ , that is, as an element of  $\mathbf{Z}_2[t \mid p(t) = 0]$ .

**ENCODE**

- \* Given a message with  $2^n - n - 1$  bits convert to a polynomial,  $m(x)$ , of degree  $\leq 2^n - n - 2$ .
- \* Calculate  $s(x) = m(x)p(x)$ . (This has degree  $\leq 2^n - 2$ .)

**SEND**

- \* Send the  $2^n - 1$  bit string that corresponds to  $s(x)$ .

**RECEIVE**

- \* Convert the received string to a polynomial  $r(x)$ .

**DECODE**

- \* Substitute  $x = t$  in  $r(x)$ .
- \* Use the table to write  $r(t)$  in terms of powers of  $t$  up to  $t^{n-1}$ .
- \* If  $r(t) = 0$  there was no error in transmission and  $s(x) = r(x)$ .
- \* If  $r(t) \neq 0$  use the table in reverse to express it as  $t^k$  where  $0 \leq k \leq 2^n - 2$ .
- \* This indicates that the bit corresponding to  $x^k$  was corrupt.
- \* Correct the error by adding  $x^k$  to  $r(x)$ . That is,  $s(x) = r(x) + x^k$ .
- \* Divide  $s(x)$  by  $p(x)$  to get  $m(x)$ .
- \* Convert this polynomial to a string of length  $2^n - n - 1$ .

**Example 6:****SETUP**

$n = 3$ ,  $p(x) = x^3 + x + 1$ , a complete polynomial.

Let  $t$  be such that  $t^3 + t + 1 = 0$ .

The table of powers is:

$1 =$	$1$
$t =$	$t$
$t^2 =$	$t^2$
$t^3 =$	$t + 1$
$t^4 =$	$t^2 + t$
$t^5 =$	$t^2 + t + 1$
$t^6 =$	$t^2 + 1$

**ENCODING**

Suppose the message is 1 1 0 0. Then  $m(x) = x^3 + x^2$ .

The sent polynomial is thus  $s(x) = (x^3 + x^2)(x^3 + x + 1) = x^6 + x^5 + x^4 + x^2$ .

### TRANSMISSION

This is transmitted as a string  
1 1 1 0 1 0 0.

Suppose during transmission an error occurs in the 3rd bit and that it is received as  
1 1 0 0 1 0 0.

### DECODING

As a polynomial, the received string is  $r(x) = x^6 + x^5 + x^2$ .

Substituting, we get  $r(t) = t^6 + t^5 + t^2$ , which from the table can be written as

$$(t^2 + 1) + (t^2 + t + 1) + t^2 = t^2 + t.$$

Using the table in reverse we see that  $r(t) = t^2 + t = t^4$ .

So an error occurred in the bit corresponding to  $x^4$ .

Correcting this we get  $s(x) = x^6 + x^5 + x^4 + x^2$ .

Hence, using polynomial "long division" the original message was

$$m(x) = \frac{x^6 + x^5 + x^4 + x^2}{x^3 + x + 1} = x^3 + x^2$$

Thus, as a string, the original message was 1 1 0 0.

## §12.5. Polynomial Codes: Why They Work

If the original message is represented by the polynomial  $m(x)$  the string corresponding to  $s(x) = m(x)p(x)$  is what is sent. This is a polynomial of degree  $\leq 2^n - 2$ . Assuming at most bit is altered during transmission, what is received,  $r(x)$ , differs from  $t(x)$  in at most one term. Thus  $r(x) = t(x) + e(x)$  where the error polynomial,  $e(x)$ , is given by  $e(x) = 0$  (if there are no errors) or  $e(x) = x^k$  for some  $k \leq 2^n - 2$  if there is an error.

When we substitute  $x = t$  we get  $r(t) = s(t) + e(t) = m(t)p(t) + e(t) = e(t)$ . This is because  $t$  is defined as a root of the equation  $p(x) = 0$ .

The error-correcting capabilities of this code rely on us being able to recover  $e(x)$  from the value of  $e(t)$ . If  $e(t) = 0$  we can infer that there was no error. In all other cases we need to be sure that the  $2^n - 1$  powers of  $t$ :  $1, t, t^2, \dots, t^{2^n-2}$  are distinct. They all look different, but when reduced to powers of  $t$  up to  $t^{n-1}$  it could be that there is a repetition.

Now there are precisely  $2^n$  distinct expressions in  $t$  up to  $t^{n-1}$  and so  $2^n - 1$  non-zero ones. So it is just possible for the powers  $1, t, t^2, \dots, t^{2^n-2}$  to be distinct. Every non-zero expression in powers of  $t$  up to  $t^{n-1}$  would have to be hit exactly once as we go through the powers  $1, t, t^2, \dots, t^{2^n-2}$ .

If you examine the table of powers of  $t$  in the above example you will see that they are all distinct. If we had used the non-prime polynomial  $x^3 + x^2 + x + 1$  the table would be:

$1 =$	$1$
$t =$	$t$
$t^2 =$	$t^2$
$t^3 =$	$t^2 + t + 1$
$t^4 =$	$1$
$t^5 =$	$t$
$t^6 =$	$t^2$

This would be of no use in error correction, because if an error occurred in the bit corresponding to the  $x^2$  term the value of  $r(t)$  would be the same as if the error had occurred in the bit corresponding to the  $x^6$  term.

## §12.6. Analysis of Probabilities

If two or more errors occur, this procedure will either incorrectly say "no errors" or it will detect one "error" and in "correcting" it will introduce yet another error. We need to ensure that the message string is broken into packets, small enough so that the probability of this happening is acceptably low. On the other hand the smaller the packet size the less efficient is the code (the more bits that need to be sent for the same message). We therefore do a trade-off.

**Defn:** The **efficiency** of the above process is defined to be the ratio of bits in the message to the total bits sent.

**Example 7:** For the above code, the efficiency =  $4/7$  or  $57\%$ .

**Defn:** The **reliability** is defined as the probability of at most one error in transmission.

**Example 8:** If the probability of an error in a single bit is  $1/100$ , the reliability is  $0.99^7 + 0.07(0.99)^6 \approx 0.998$  or  $99.8\%$ .

**Theorem 4:** For the polynomial code using a complete polynomial of degree  $n$ :

$$\text{Efficiency} = \frac{\text{\# bits in message}}{\text{\#bits sent}} = \frac{N - n}{N},$$

$$\text{Reliability} \approx 1 - \frac{1}{2}N(N - 1)p^2.$$

(Here  $N = 2^n - 1$ .)

As  $n \rightarrow \infty$ ,  $N \rightarrow \infty$ , and so the efficiency  $\rightarrow 1$  and the reliability  $\rightarrow 0$ .

n	N	Efficiency	Reliability if p = 0.01	Reliability if p = 0.001
2	3	33%	100%	100%
3	7	57%	100%	100%
4	15	60%	99%	100%
5	31	84%	95%	100%
6	63	90%	80%	100%
7	127	94%	20%	99%
8	255	97%	0%	97%

## EXERCISES FOR CHAPTER 12

### EXERCISES 12A (Polynomials Mod $p(x)$ )

**Ex 12A1:** Construct the addition and multiplication tables for  $\mathbf{Z}_2[t \mid 1 + t^2 = 0]$ , the system of the remainders of polynomials in  $t$  over  $\mathbf{Z}_2$  on division by  $1 + t^2$ .

(ii) Which of the four elements of this system have an inverse under addition?

(iii) Which of the four elements of this system have an inverse under multiplication?

**Ex 12A2:** Construct the addition and multiplication tables for  $\mathbf{Z}_2[t \mid 1 + t + t^2 = 0]$ , the system of remainders of polynomials in  $t$  over  $\mathbf{Z}_2$  on division by  $1 + t + t^2$ .

(ii) Which of the four elements of this system have an inverse under addition?

(iii) Which of the four elements of this system have an inverse under multiplication?

## EXERCISES 12B (Polynomial Codes)

**Ex 12B1:** A binary string  $b_0 b_1 b_2 b_3$  is converted to the polynomial  $b_0 + b_1 t + b_2 t^2 + b_3 t^3$  and is then coded by multiplying by the polynomial  $1 + t + t^3$ . The resulting polynomial is then converted back to a binary string of length 7 and transmitted. This string is received as 1011001. Assume that at most one of these 7 bits has been corrupted.

(i) Show that indeed an error has occurred in one of the bits.

(ii) Which bit was wrong?

(iii) Correct the error and recover the original message  $b_0 b_1 b_2 b_3$ .

**Ex 12B2:** A binary string  $b_0 b_1 b_2 b_3$  is converted to the polynomial  $b_0 + b_1 t + b_2 t^2 + b_3 t^3$  and is then coded by multiplying by the polynomial  $1 + x + x^3$ . The resulting polynomial is then converted back to a binary string of length 7 and transmitted. This string is received as 0010011. Assume that at most one of these 7 bits has been corrupted.

(i) Show that indeed an error has occurred in one of the bits.

(ii) Which bit was wrong?

(iii) Correct the error and recover the original message  $b_0 b_1 b_2 b_3$ .

**Ex 12B3:** Find an irreducible polynomial of degree 4 over  $\mathbf{Z}_2$  and use it to encode the binary string 11011.

(The polynomial  $a_0 + a_1 t + a_2 t^2 + \dots$  corresponds to the string  $a_0 a_1 a_2 \dots$ )

### Ex 12B4:

(a) Show that  $p(x) = x^3 + x^2 + 1 \in \mathbf{Z}_2[x]$  is a complete polynomial.

(b) Suppose binary strings of length 4 are converted to polynomials by the convention that  $a_3 x^3 + a_2 x^2 + a_1 x + a_0$  represents the string  $a_3 a_2 a_1 a_0$ .

(i) Use  $p(x)$  to encode the message string 1001.

(ii) Suppose the string 1011010 is received. Assuming that  $p(x)$  was used to transmit it, using Polynomial Coding, and that at most one error occurred, show that indeed there was an error occurred. Correct the error and work out the original 4 bit message.

**Ex 12B5:** Suppose binary strings of length 11 are converted to binary polynomials by regarding the bits as the coefficients of increasing powers of  $x$  (so that the first bit becomes the constant term). A binary 11 bit message is converted to a polynomial in this way and then multiplied by the complete polynomial  $x^4 + x^3 + 1$ . The coefficients are transmitted as a 15 bit string. Suppose it is received as 01110111011001. Assuming that exactly one of these bits is wrong, correct the error and recover the original 11 bit message.

# SOLUTIONS FOR CHAPTER 12

## Exercise 12A1:

+	0	1	t	1+t
0	0	1	t	1+t
1	1	0	1+t	t
t	t	1+t	0	1
1+t	1+t	t	1	0

×	0	1	t	1+t
0	0	0	0	0
1	0	1	t	1+t
t	0	t	1	1+t
1+t	0	1+t	1+t	0

(ii) all four.; (iii) 1 and t only.

## Exercise 12A2:

+	0	1	t	1+t
0	0	1	t	1+t
1	1	0	1+t	t
t	t	1+t	0	1
1+t	1+t	t	1	0

×	0	1	t	1+t
0	0	0	0	0
1	0	1	t	1+t
t	0	t	1+t	1
1+t	0	1+t	1	t

(ii) **all four;** (iii) **all except 0.**

## Ex 12B1:

Converting the received string 1011001 to a polynomial we get  $1 + t^2 + t^6$ .

Now if  $1 + t + t^3 = 0$  then

$$t^3 = 1 + t;$$

$$t^4 = t + t^2;$$

$$t^5 = t^2 + t^3 = 1 + t + t^2;$$

$$t^6 = t + t^2 + t^3 = 1 + t^2;$$

$$t^7 = t + t^3 = 1.$$

Hence mod  $1 + t + t^3$ , the received polynomial is  $1 + t^2 + 1 + t + 1 + t^2 = 1 + t = t^3$ .

If no errors occurred this would have been 0. Hence an error occurred.

(ii) Since the coefficient of  $t^3$  was incorrect, the 4'th bit was wrongly received.

(iii) The received polynomial should have been  $1 + t^2 + t^6$ . Dividing this by  $1 + t + t^3$  we get a remainder of zero and a quotient of  $1 + t + t^3$ . The transmitted message was thus 1101.

**Exercise 12B2:**

Converting the received string 0010011 to a polynomial we get  $t^2 + t^5 + t^6$ .

Now if  $1 + t + t^3 = 0$  then

$$t^3 = 1 + t;$$

$$t^4 = t + t^2;$$

$$t^5 = t^2 + t^3 = 1 + t + t^2;$$

$$t^6 = t + t^2 + t^3 = 1 + t^2;$$

$$t^7 = t + t^3 = 1.$$

Hence mod  $1 + t + t^3$ , the received polynomial is  $t^2 + 1 + t + t^2 + 1 + t^2 = t + t^2 = t^4$ .

If no errors occurred this would have been 0. Hence an error occurred.

(ii) Since the coefficient of  $t^4$  was incorrect, the **5'th bit** was wrongly received.

(iii) The received polynomial should have been  $t^2 + t^4 + t^5 + t^6$ . Dividing this by  $1 + t + t^3$  we get a remainder of zero and a quotient of  $t^2 + t^3$ . The corrected message is thus **0011**.

**Ex 12B3:**  $t^4 + t^3 + 1$  (or  $t^4 + t + 1$ ).

11011 becomes  $1 + t + t^3 + t^4$  which is coded as  $(t^4 + t^3 + 1)(1 + t + t^3 + t^4)$   
 $= 1 + t + t^4 + t^5 + t^6 + t^8$ . This is transmitted as 110011101.

(If  $t^4 + t + 1$  had been used the transmitted message would be 101110011)

**Ex 12B4:**

(a) Suppose  $t^3 + t^2 + 1 = 0$ .

$$\therefore t^3 = t^2 + 1$$

$$\therefore t^4 = t^3 + t = (t^2 + 1) + t = t^2 + t + 1$$

$$\therefore t^5 = t^3 + t^2 + t = (t^2 + 1) + t^2 + t = t + 1$$

$$\therefore t^6 = t^2 + t$$

Since all 7 of the non-zero polynomials of degree less than 3 occur as a power of  $t$  this polynomial is complete.

(b) (i) The message 1001 corresponds to the message polynomial  $m(x) = x^3 + 1$ .

The transmitted message is  $t(x) = m(x)p(x) = (x^3 + 1)(x^3 + x^2 + 1)$

$$= x^6 + x^5 + x^3 + x^3 + x^2 + 1 = x^6 + x^5 + x^2 + 1.$$

This is transmitted as the string 1100101.

(ii) From (a) the table of equivalents is:

1	1
t	t
t <sup>2</sup>	t <sup>2</sup>
t <sup>3</sup>	t <sup>2</sup> + 1
t <sup>4</sup>	t <sup>2</sup> + t + 1
t <sup>5</sup>	t + 1
t <sup>6</sup>	t <sup>2</sup> + t

The received polynomial is  $r(x) = x^6 + x^4 + x^3 + x$ .

$$\begin{aligned} r(t) &= t^6 + t^4 + t^3 + t \\ &= (t^2 + t) + (t^2 + t + 1) + (t^2 + 1) + t \\ &= t^2 + 1 \\ &= t^6. \end{aligned}$$

So an error has occurred and  $e(x) = x^6$ .

$$\therefore m(x)p(x) = t(x) = r(x) + e(x) = x^4 + x^3 + x.$$

$$\therefore m(x) = (x^4 + x^3 + x)/(x^3 + x^2 + 1) = x.$$

So the original message was 0010.

**Ex 12B5:** Suppose that  $t^4 + t^3 + 1 = 0$ . The table of powers is:

$t^4$	$t^3 + 1$
$t^5$	$t^3 + t + 1$
$t^6$	$t^3 + t^2 + t + 1$
$t^7$	$t^2 + t + 1$
$t^8$	$t^3 + t^2 + t$
$t^9$	$t^2 + 1$
$t^{10}$	$t^3 + t$
$t^{11}$	$t^3 + t^2 + 1$
$t^{12}$	$t + 1$
$t^{13}$	$t^2 + t$
$t^{14}$	$t^3 + t^2$

The received polynomial is  $r(x) = x + x^2 + x^3 + x^5 + x^6 + x^7 + x^9 + x^{10} + x^{11} + x^{14}$ .

$$\text{Thus } r(t) = t + t^2 + t^3 + t^5 + t^6 + t^7 + t^9 + t^{10} + t^{11} + t^{14}$$

$$= t + t^2 + t^3 + (t^3 + t + 1) + (t^3 + t^2 + t + 1) + (t^2 + t + 1) + (t^2 + 1) + (t^3 + t) + (t^3 + t^2 + 1) + (t^3 + t^2)$$

$$= 1 + t = t^{12}.$$

Therefore the transmitted polynomial must have been:

$$x + x^2 + x^3 + x^5 + x^6 + x^7 + x^9 + x^{10} + x^{11} + x^{12} + x^{14}.$$

Dividing by  $x^4 + x^3 + 1$  we get a quotient of  $x^{10} + x^9 + x^7 + x^6 + x^5 + x^4 + x^3$  and so the original message must have been 0111111011.

