Data Communications and Networking Fourth Edition

Forouzan

Chapter 10

Error Detection and Correction

Note

Data can be corrupted during transmission.

Some applications require that errors be detected and corrected.

10-1 INTRODUCTION

Let us first discuss some issues related, directly or indirectly, to error detection and correction.

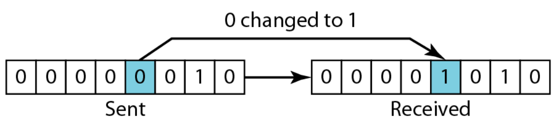
Topics discussed in this section:

Types of Errors
Redundancy
Detection Versus Correction

Note

In a single-bit error, only 1 bit in the data unit has changed.

Figure 10.1 *Single-bit error*



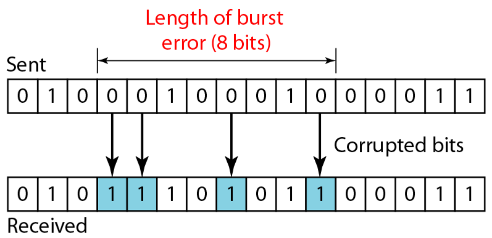
10.5

Note

A burst error means that 2 or more bits in the data unit have changed.

10.6

Figure 10.2 *Burst error of length 8*



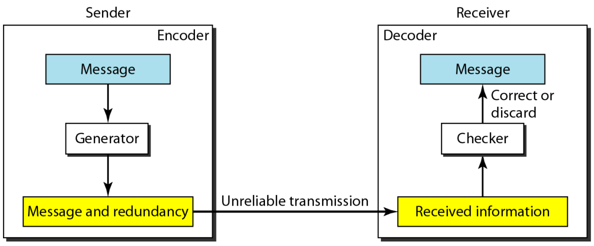
10.7

Note

To detect or correct errors, we need to send extra (redundant) bits with data.

10.8

Figure 10.3 *The structure of encoder and decoder*



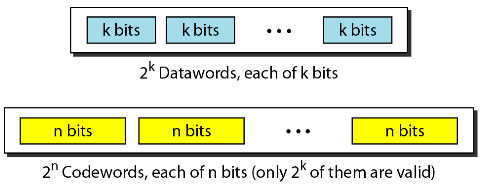
10-2 BLOCK CODING

In block coding, we divide our message into blocks, each of k bits, called datawords. We add r redundant bits to each block to make the length n = k + r. The resulting n-bit blocks are called codewords.

Topics discussed in this section:

Error Detection
- Parity Check
Error Correction

Figure 10.5 *Datawords and codewords in block coding*



Error Detection

- Enough redundancy is added to detect an error.
- The receiver knows an error occurred but does not know which bit(s) is(are) in error.

Parity Check

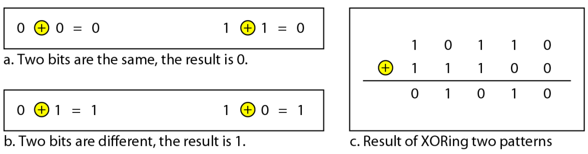
- Very simple technique used to detect errors
- In Parity check, a parity bit is added to the data block.
- Assume a data block of size k bits
- Adding a parity bit will result in a block of size k+1bits
- The value of the parity bit depends on the number of "1"s in the k bits data block

Parity Bit

- If we have a message = 1010111
- k = 7 bits
- Adding a parity check so that the number of 1's is even
- The message would be : 11010111
- k+1 = 8 bits
- At the receiver ,if one bitchanges its values, then an error can be detected

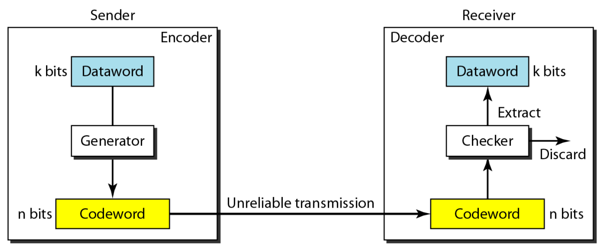
Figure 10.4 *XORing of two single bits or two words*

- The circuit that generates the parity bit in the transmitter is called a parity generator.
- XOR (even parity generator)



10.15

Figure 10.6 *Process of error detection in block coding*



10.16

Example 10.2

Let us assume that k = 2 and n = 3. Table 10.1 shows the list of datawords and codewords. Later, we will see how to derive a codeword from a dataword.

Assume the sender encodes the dataword 01 as 011 and sends it to the receiver. Consider the following cases:

1. *The receiver receives 011. It is a valid codeword. The receiver extracts the dataword 01 from it.*

Example 10.2 (continued)

2. *The codeword is corrupted during transmission, and 111 is received. This is not a valid codeword and is discarded.*

Table 10.1 *A code for error detection (Example 10.2)*

Datawords	Codewords
00	000
01	011
10	101
11	110

Note

An error-detecting code can detect only the types of errors for which it is designed; other types of errors may remain undetected.

Example 10.3

Let us add more redundant bits to Example 10.2 to see if the receiver can correct an error without knowing what was actually sent. We add 3 redundant bits to the 2-bit dataword to make 5-bit codewords. Table 10.2 shows the datawords and codewords. Assume the dataword is 01. The sender creates the codeword 01011. The codeword is corrupted during transmission, and 01001 is received. First, the receiver finds that the received codeword is not in the table. This means an error has occurred. The receiver, assuming that there is only 1 bit corrupted.

Example 10.3 (continued)

1. *Comparing the received codeword with the first codeword in the table (01001 versus 00000), the receiver decides that the first codeword is not the one that was sent because there are two different bits.*

2. *By the same reasoning, the original codeword cannot be the third or fourth one in the table.*

3. *The original codeword must be the second one in the table because this is the only one that differs from the received codeword by 1 bit. The receiver replaces 01001 with 01011 and consults the table to find the dataword 01.*

10.21

Table 10.2 *A code for error correction (Example 10.3)*

Dataword	Codeword
00	00000
01	01011
10	10101
11	11110

10.22

Error Correction method

No Time

10.23