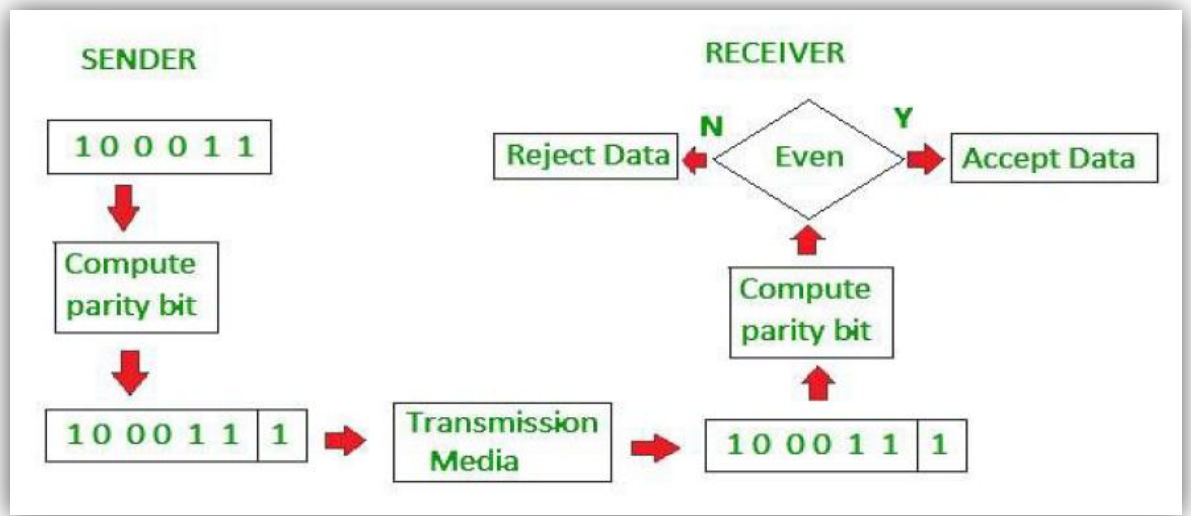
Course Name: A Level (2nd Sem) Subject: DCN

Topic: Error Detection and Correction contd. Date: 21-04-20

Simple Parity check:

The most familiar error-detecting code is the simple parity-check code. In this code, a k-bit data word is changed to an n-bit codeword where n = k + 1. The extra bit, called the parity bit (used as redundant bit), is selected to make the total number of 1s in the codeword even. Some implementations specify an odd number of 1s but even case is most common.

Blocks of data from the source are subjected to a check bit or parity bit generator form, where a parity of 1 is added to the block if it contains odd number of 1's, and 0 is added if it contains even number of 1's This scheme makes the total number of 1's even, that is why it is called even parity checking.



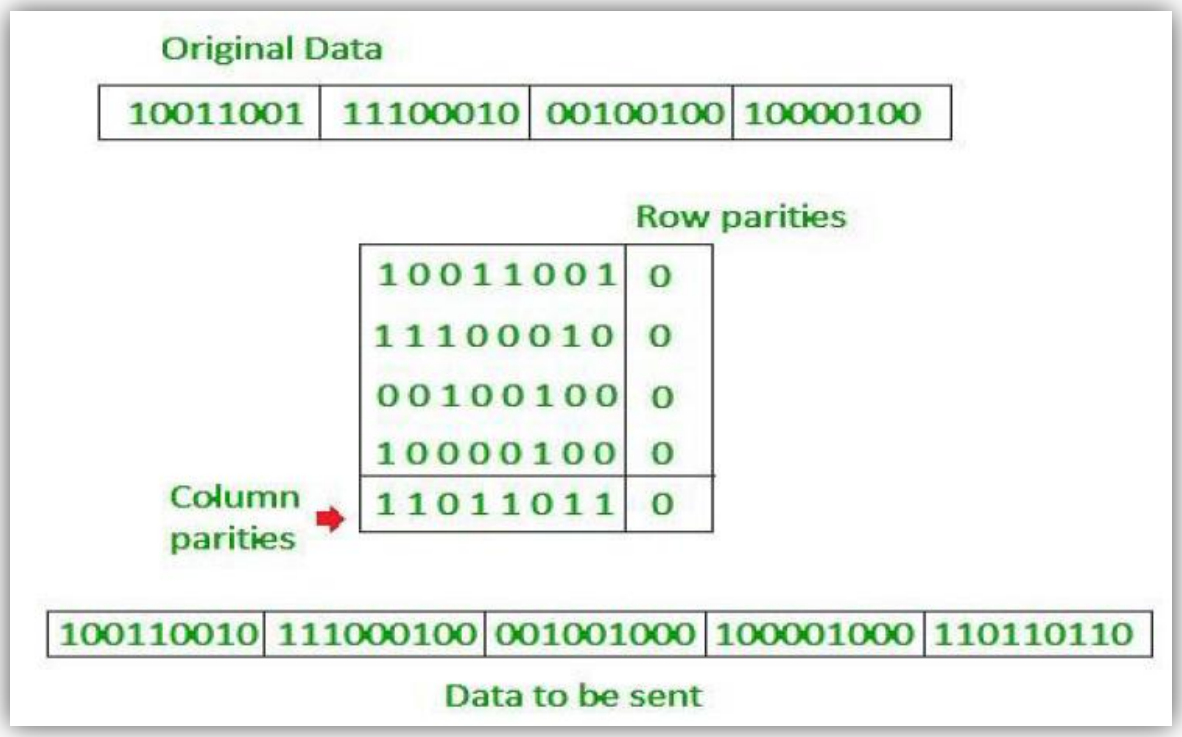
The code in the following table is also a parity-check code with k =4 and n =5.

Datawords	*Codewords*	*Datawords*	*Codewords*
0000	00000	1000	10001
0001	00011	1001	10010
0010	00101	1010	10100
0011	00110	1011	10111
0100	01001	1100	11000
0101	01010	1101	11011
0110	01100	1110	11101
0111	01111	1111	11110

Two-dimensional Parity check:

A better approach is the two-dimensional parity check in which the dataword is organized in a table (rows and columns). For each row and each column, 1 parity (simple parity) check bit is calculated.

In this process, parity check bits are calculated for each row, which is equivalent to a simple parity check bit. Parity check bits are also calculated for all columns, and then both are sent along with the data. At the receiving end these are compared with the parity bits calculated on the received data.



Exercises:

- **How is the simple parity check related to the two-dimensional parity check?**