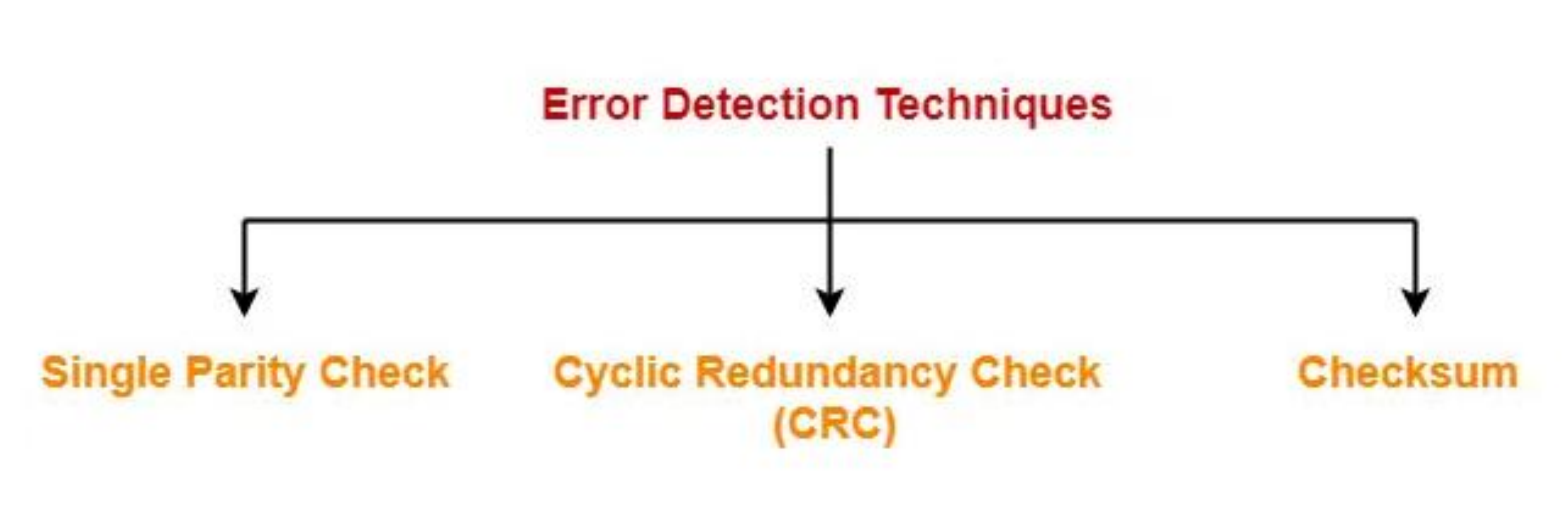
Error detection and correction

Error

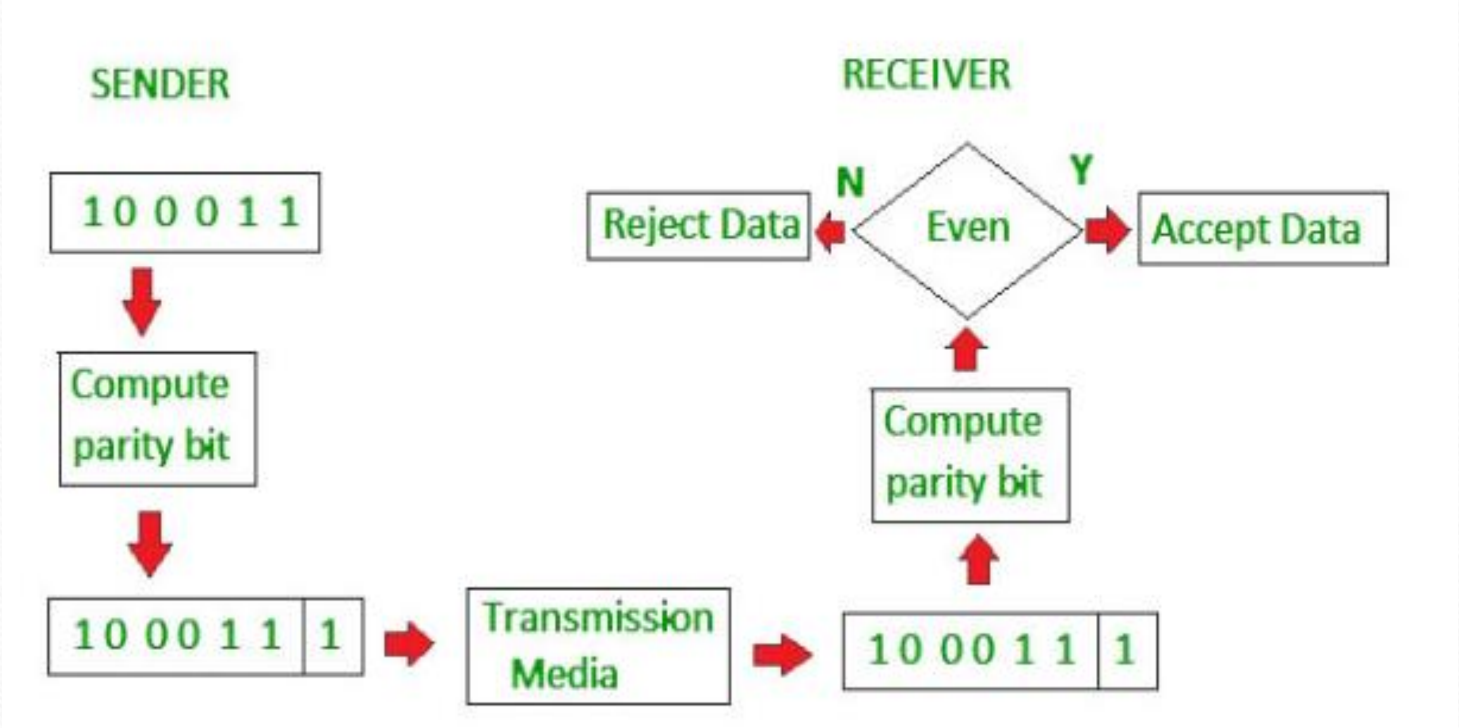
- A condition when the receiver's information does not match with the sender's information. During transmission, digital signals suffer from noise that can introduce errors in the binary bits travelling from sender to receiver. That means a 0 bit may change to 1 or a 1 bit may change to 0.
- Basic approach used for error detection is the use of redundancy bits, where additional bits are added to facilitate detection of errors.
- Some popular techniques for error detection are:
  1. Simple Parity check
  2. Two-dimensional Parity check
  3. Checksum
  4. Cyclic redundancy check


Error Detection Techniques
Single Parity Check
Cyclic Redundancy Check (CRC)
Checksum

Simple Parity check

- 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.

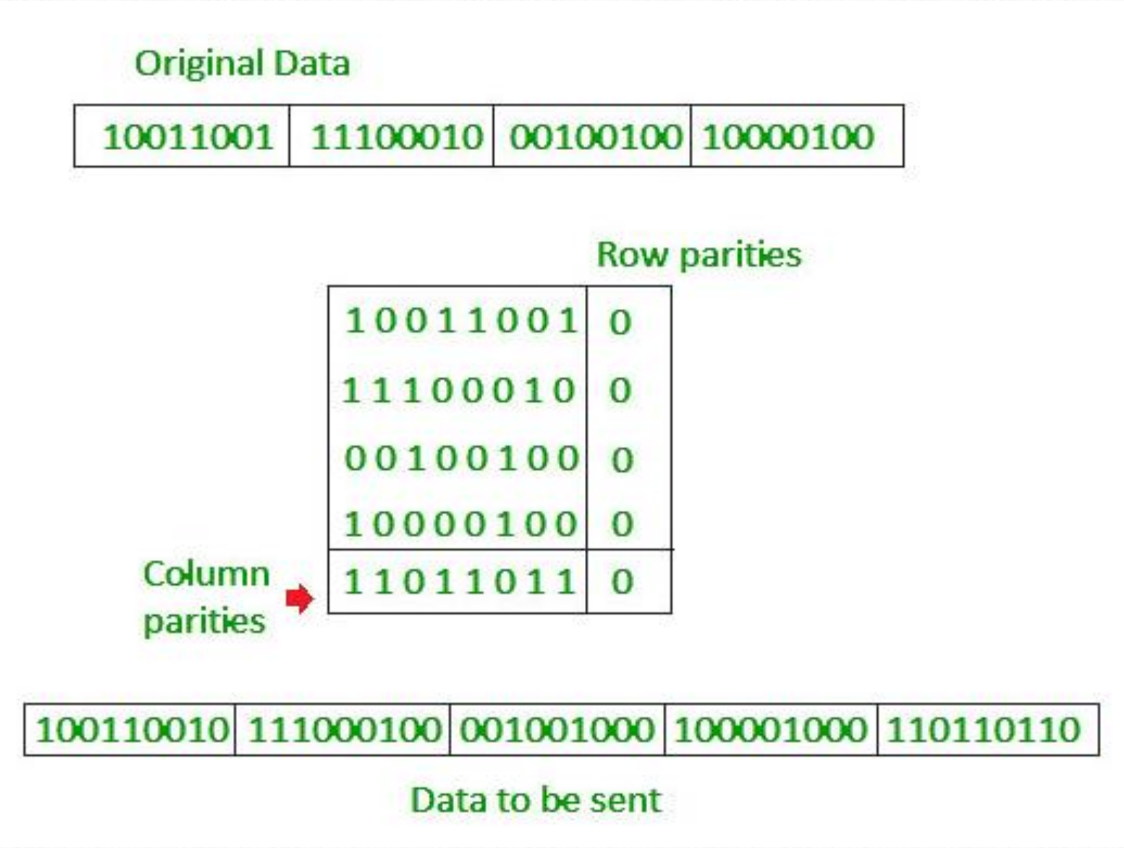
Simple Parity check



Two-dimensional Parity check

- 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, then both are sent along with the data. At the receiving end these are compared with the parity bits calculated on the received data.

Two-dimensional Parity check



Longitudinal Redundancy Check (LRC)

- **Longitudinal Redundancy Check (LRC)** is also known as 2-D parity check. In this method, data which the user want to send is organised into tables of rows and columns. A block of bit is divided into table or matrix of rows and columns. In order to detect an error, a redundant bit is added to the whole block and this block is transmitted to receiver. The receiver uses this redundant row to detect error. After checking the data for errors, receiver accepts the data and discards the redundant row of bits.

Longitudinal Redundancy Check (LRC)

Example :

If a block of 32 bits is to be transmitted, it is divided into matrix of four rows and eight columns which as shown in the following figure :

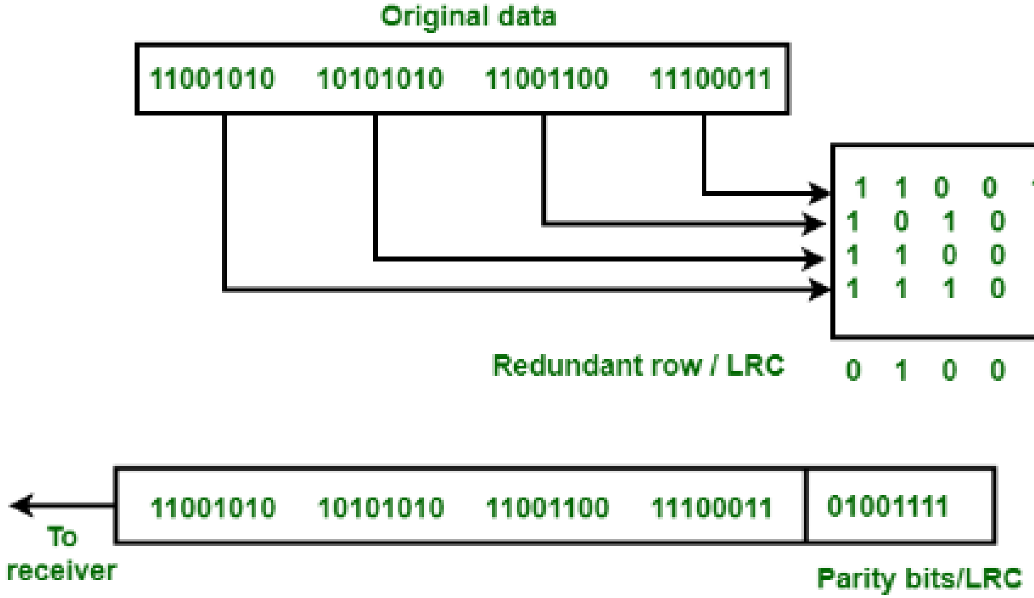


Figure: LRC

In this matrix of bits, a parity bit (odd or even) is calculated for each column. It means 32 bits data plus 8 redundant bits are transmitted to receiver. Whenever data reaches at the destination, receiver uses LRC to detect error in data.

Advantage :
LRC is used to detect burst errors.

Example : Suppose 32 bit data plus LRC that was being transmitted is hit by a burst error of length 5 and some bits are corrupted as shown in the following figure :

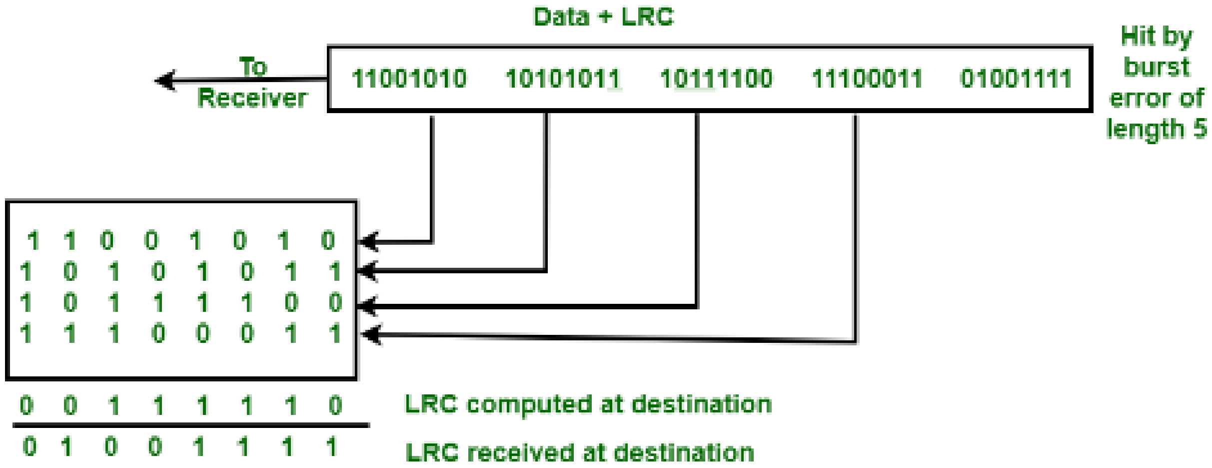


Figure : Burst error & LRC

The LRC received by the destination does not match with newly corrupted LRC. The destination comes to know that the data is erroneous, so it discards the data.

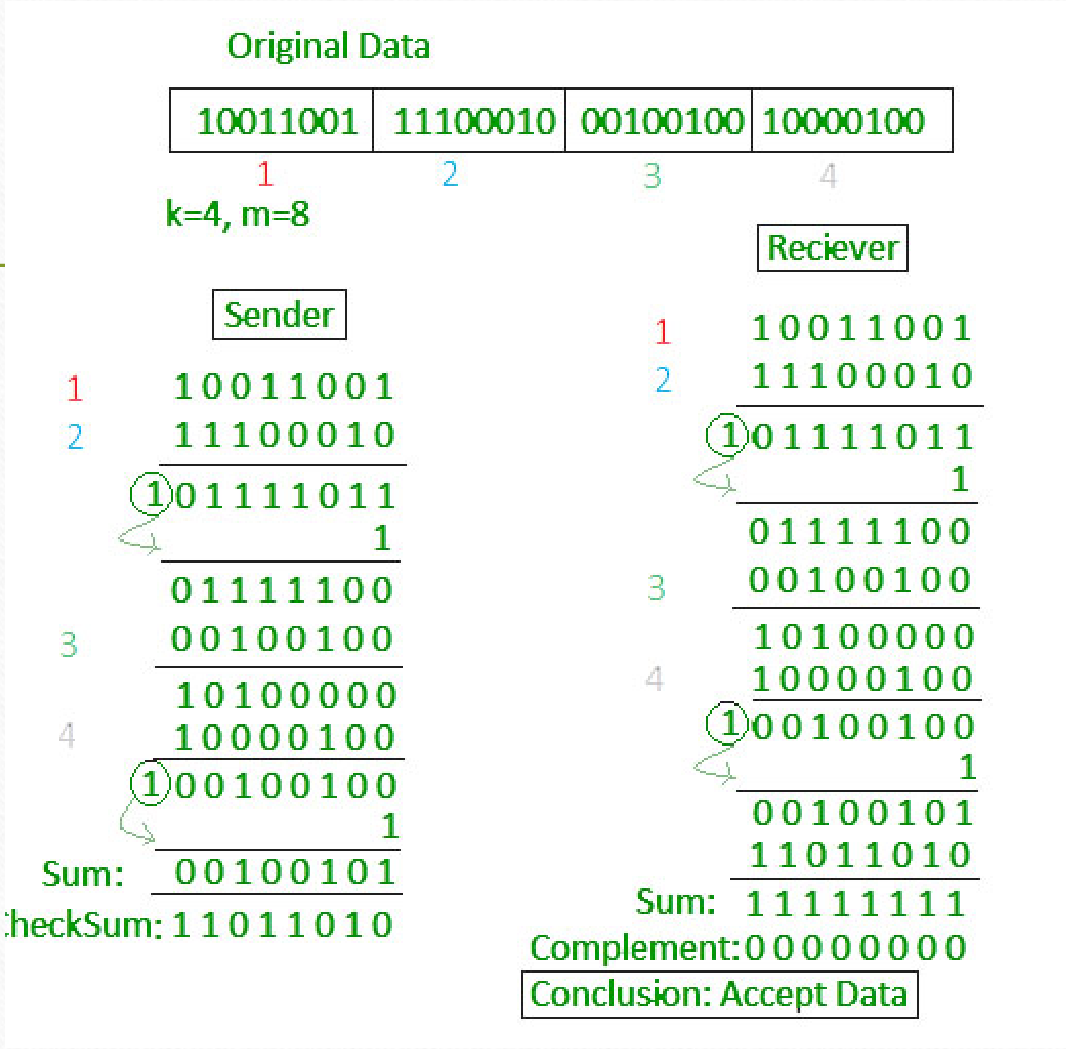
Disadvantage :

The main problem with LRC is that, it is not able to detect error if two bits in a data unit are damaged and two bits in exactly the same position in other data unit are also damaged.

Checksum

- In checksum error detection scheme, the data is divided into k segments each of m bits.
- In the sender's end the segments are added using 1's complement arithmetic to get the sum. The sum is complemented to get the checksum.
- The checksum segment is sent along with the data segments.
- At the receiver's end, all received segments are added using 1's complement arithmetic to get the sum. The sum is complemented.
- If the result is zero, the received data is accepted; otherwise discarded.

Checksum



Original Data
10011001
11100010
00100100
10000100
1
2
3
4
k=4, m=8
Sender
1 10011001
2 11100010
1 01111011
1
01111100
3 00100100
10100000
4 10000100
1 00100100
1
Sum: 00100101
CheckSum: 11011010
Reciever
1 10011001
2 11100010
1 01111011
1
01111100
3 00100100
10100000
4 10000100
1 00100100
1
00100101
11011010
Sum: 11111111
Complement: 00000000
Conclusion: Accept Data

Cyclic Redundancy Check-

- Cyclic Redundancy Check (CRC) is an error detection method.
- It is based on binary division.

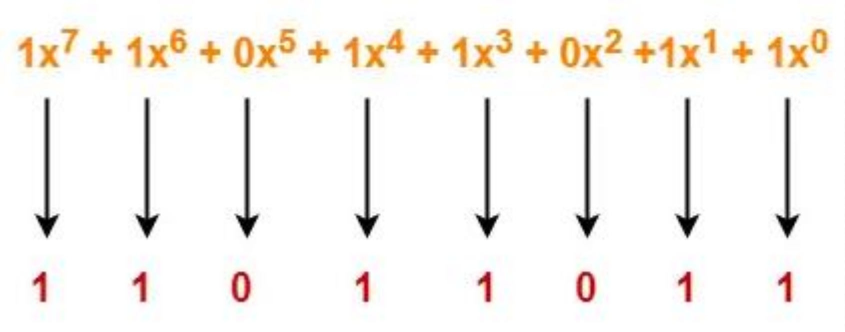
CRC Generator

- CRC generator is an algebraic polynomial represented as a bit pattern.
- Bit pattern is obtained from the CRC generator using the following rule-

The power of each term gives the position of the bit and the coefficient gives the value of the bit.

Example

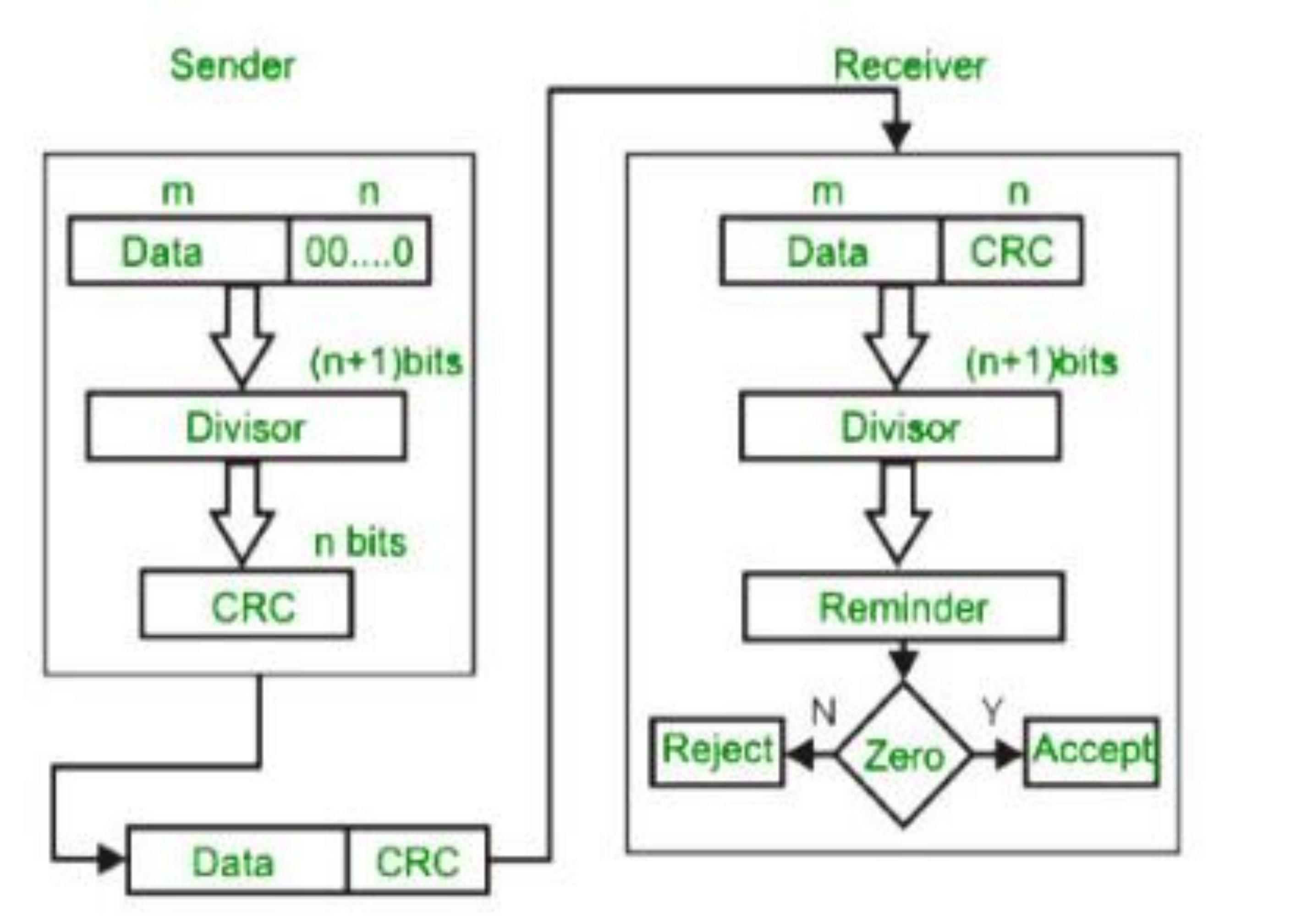
- Consider the CRC generator is $x^7 + x^6 + x^4 + x^3 + x + 1$.
- The corresponding binary pattern is obtained as-



- Thus, for the given CRC generator, the corresponding binary pattern is 11011011.

Cyclic redundancy check (CRC)

- Unlike checksum scheme, which is based on addition, CRC is based on binary division.
- In CRC, a sequence of redundant bits, called cyclic redundancy check bits, are appended to the end of data unit so that the resulting data unit becomes exactly divisible by a second, predetermined binary number.
- At the destination, the incoming data unit is divided by the same number. If at this step there is no remainder, the data unit is assumed to be correct and is therefore accepted.
- A remainder indicates that the data unit has been damaged in transit and therefore must be rejected.



Sender
Receiver
m
n
Data
00....0
(n+1)bits
Divisor
n bits
CRC
Data
CRC
m
n
Data
CRC
(n+1)bits
Divisor
Reminder
N
Y
Reject
Zero
Accept

Example of CRC :