Academic Session 2025-26

ODD Semester Jul-Dec 2025

UNIVERSITY INSTITUTE OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Bachelor of Computer Science

5th Semester

Computer Networks(23CST-302/ 23ITT-302)

Unit No. 2 **Chapter No. 4** **Lecture No.16**

Topic : DLL- Error Detection & Correction

Faculty Name: Monika(E9911) **Designation: Assistant Professor**

Learning Objectives & Outcomes

Learning Objective

To understand the concepts of Error Detection & Correction in Data link layer.

Learning Outcome

Students will be able to learn various Error Detection methods (parity, checksum, CRC) & Error correction methods(Hamming code etc).

Error

Error

A condition when the receiver's information does not matches with the sender's information.

Error-detecting codes

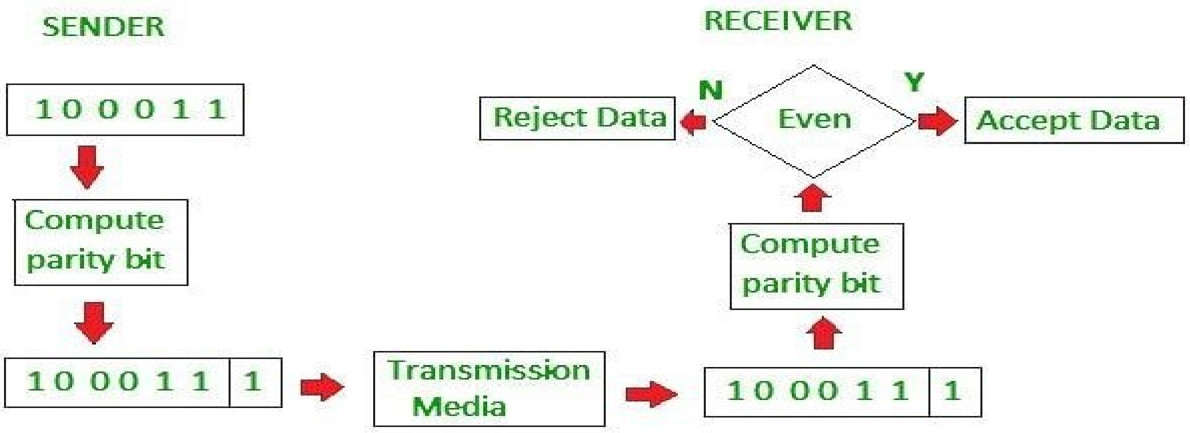
Additional data added to a given digital message to help us detect if any error has occurred during transmission of the message.

Various types of Error-detecting codes are:

1. Simple Parity check
2. Two-dimensional Parity check
3. Checksum
4. Cyclic redundancy check

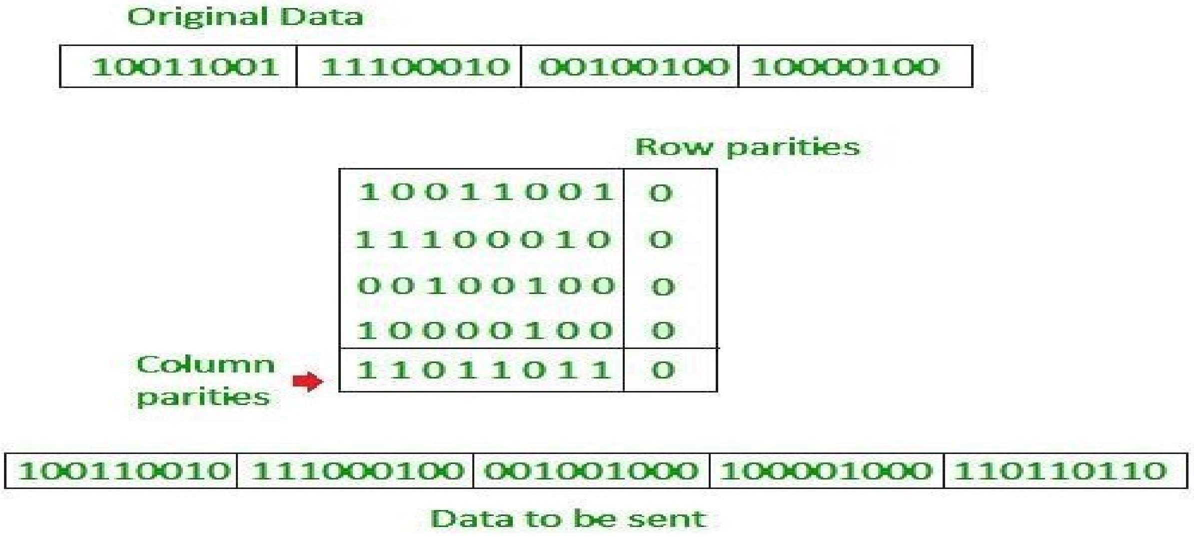
Error-detecting codes

1. **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, and0 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.



Error-detecting codes

2. 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.



Error-detecting codes(CRC)

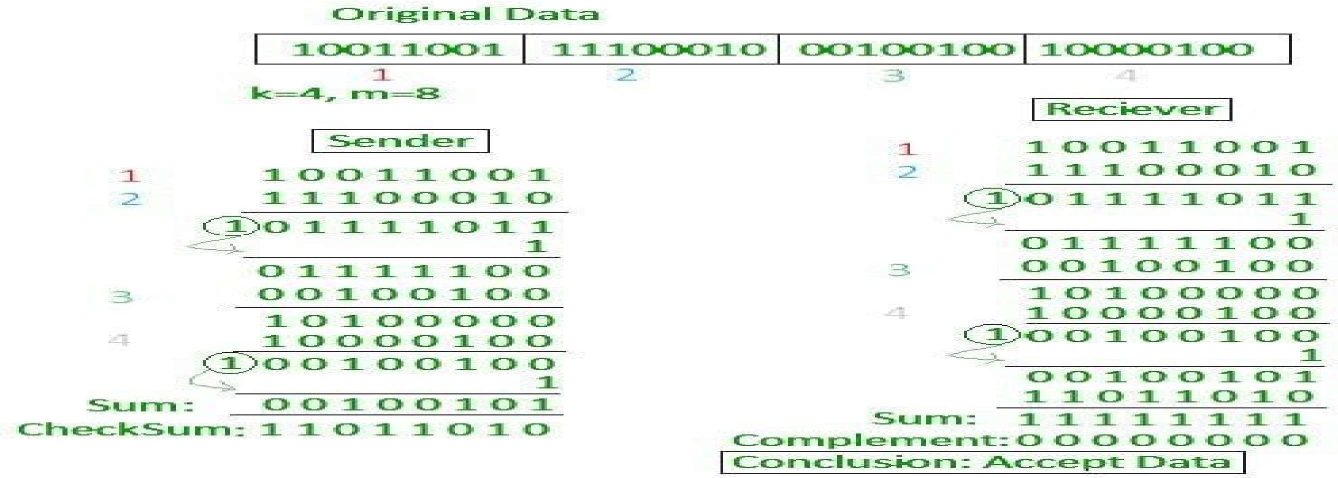
3. CRC-

- 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.

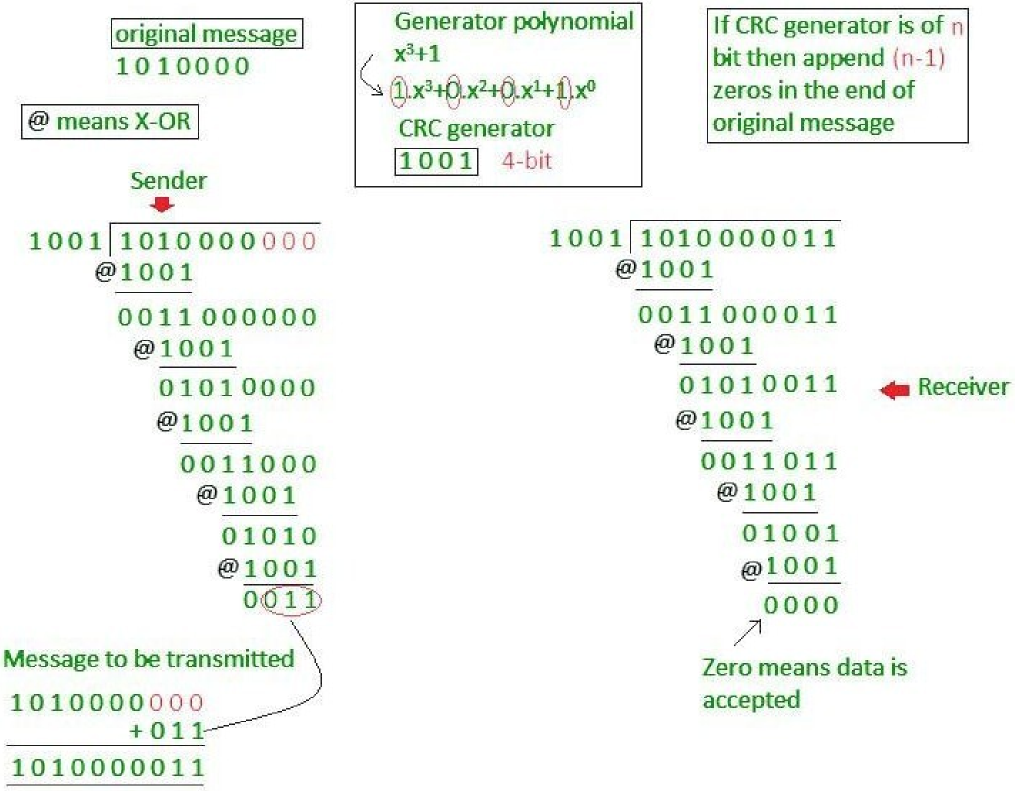
Error-detecting codes

3. 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.
- 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



3. CRC(Example)



Error Correction

Error Correction

Error Correction codes are used to detect and correct the errors when data is transmitted from the sender to the receiver.Error Correction can be handled in two ways:

- **Backward error correction**: Once the error is discovered, the receiver requests the sender to retransmit the entire data unit.
- **Forward error correction:** In this case, the receiver uses the error-correctingcode which automatically corrects the errors.

-A single additional bit can detect the error, but cannot correct it.
-For correcting the errors, one has to know the exact position of the error.

Error Correction

For example If we want to calculate a single-bit error, the error correction code will determine which one of seven bits is in error. To achieve this, we have to add some additional redundant bits.

Suppose r is the number of redundant bits and d is the total number of the data bits. The number of redundant bits r can be calculated by using the formula:

$$2r >= d+r+1$$

The value of r is calculated by using the above formula. For example, if the value of d is 4, then the possible smallest value that satisfies the above relation would be 3.

Error Correction(Hamming Code)

To determine the position of the bit which is in error, a technique developed by R.W Hamming is Hamming code.

Hamming Code

Parity bits: The bit which is appended to the original data of binary bits so that the total number of 1s is even or odd.

Even parity: To check for even parity, if the total number of 1s is even, then the value of the parity bit is 0. If the total number of 1s occurrences is odd, then the value of the parity bit is 1.

Odd Parity: To check for odd parity, if the total number of 1s is even, then the value of parity bit is 1. If the total number of 1s is odd, then the value of parity bit is 0.

Error Correction(Hamming Code)

Algorithm of Hamming code:

1. An information of 'd' bits are added to the redundant bits 'r' to form d+r. The location of each of the (d+r) digits is assigned a decimal value.
2. The 'r' bits are placed in the positions 1,2,…2^{k-1}
3. At the receiving end, the parity bits are recalculated. The decimal value of the parity bits determines the position of an error.

Relationship b/w Error position & binary number:

Error Position	Binary Number
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Error Correction(Hamming Code)

Example

Suppose the original data is 1010 which is to be sent.

Total number of data bits 'd' = 4

Number of redundant bits r : 2r >= d+r+1

2r >= 4+r+1

Therefore, the value of r is 3 that satisfies the above relation. Total number of bits = d+r = 4+3 = 7

Error Correction(Hamming Code)

Determining the position of the redundant bits

The number of redundant bits is 3. The three bits are represented by r1, r2, r4. The position of the redundant bits is calculated with corresponds to the raised power of 2. Therefore, their corresponding positions are 1, 2^1, 2^2.

The position of r1 = 1, The position of r2 = 2 , The position of r4 = 4

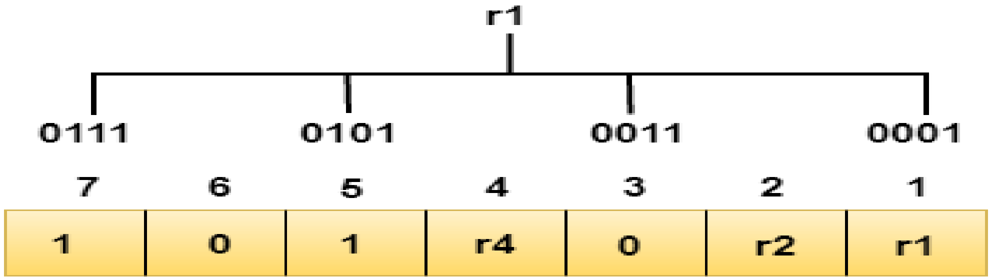
Representation of Data on the addition of parity bits:

7	6	5	4	3	2	1
1	0	1	r4	0	r2	r1

Error Correction(Hamming Code)

Determining the Parity bits

Determining the r1 bit: The r1 bit is calculated by performing a parity check on the bit positions whose binary representation includes:
We observe from the above figure that the bit position that includes 1 in the first position are 1, 3, 5, 7. Now, we perform the even-parity check at these bit positions. The total number of 1 at these bit positions corresponding to r1 is even, therefore, the value of the r1 bit is 0.



Summary of the Lecture

Error detection and correction are essential techniques in data communication and storage systems to ensure data integrity. Methods are:

a) Parity bits
b) Checksums,
c) Cyclic redundancy checks (CRC).

Once an error is detected, error correction techniques, such as

d) Hamming code or Reed-Solomon codes, are used to identify and fix the erroneous bits without needing retransmission.

These mechanisms help maintain accuracy and reliability, especially in noisy environments like wireless networks or optical storage devices.

Next Lecture

- Sliding Window Protocols

FAQ's

Q1:What is the difference between error detection and error correction?

Q2: What are common methods of error detection?

Q3: How does error correction work?

Q4: What is CRC used for?

Q5: What types of errors can Hamming Code detect and correct?

References/ Articles/ Videos

References

- Computer Networks, Tanenbaum, Andrew, Fifth Edition, PHI.
- Data Communication and Networking, Behrouz A. Forouzan, Fourth Edition.Larry

- **Videos**
  - https://www.youtube.com/watch?v=oQ96ceboZuo
  - https://www.youtube.com/watch?v=2U6kPu0dfqI

Faculty-curated videos, NPTEL, Coursera, LinkedIn, or other relevant learning resources

- NPTEL CN COURSE
- https://www.coursera.org/learn/introduction-to-networking-nvidia

Class-Wise Feedback

Thank You

For queries
Email: monika.e9911@cumail.in