



GLOBAL JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY: C  
SOFTWARE & DATA ENGINEERING

Volume 14 Issue 8 Version 1.0 Year 2014

Type: Double Blind Peer Reviewed International Research Journal

Publisher: Global Journals Inc. (USA)

Online ISSN: 0975-4172 & Print ISSN: 0975-4350

## Data Link Layer Designing Issues: Error Control—A Roadmap

By Monika Singh & Ruhi Saxena

*Mody University, India*

**Abstract-** Different networks are used to transfer data from one device to another with acceptable accuracy. For most applications, a system must guarantee that the data received are identical to the data transmitted. Transmission media are most error-prone link. In a network, the capacity of nodes is different and the rate at which the sender is sending data might not be the same rate at which the receiver accepts it. In this paper, we discuss on designing issues of data link layer. The primary focus is on various error detecting and controlling mechanisms.

**Keywords:** error detection, LRC, VRC, CRC, checksum.

**GJCST-C Classification :** C.2.5 C.2.6



*Strictly as per the compliance and regulations of:*



© 2014. Monika Singh & Ruhi Saxena. This is a research/review paper, distributed under the terms of the Creative Commons Attribution-Noncommercial 3.0 Unported License (<http://creativecommons.org/licenses/by-nc/3.0/>), permitting all non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

# Data Link Layer Designing Issues: Error Control—A Roadmap

Monika Singh <sup>α</sup> & Ruhi Saxena <sup>σ</sup>

**Abstract** Different networks are used to transfer data from one device to another with acceptable accuracy. For most applications, a system must guarantee that the data received are identical to the data transmitted. Transmission media are most error-prone link. In a network, the capacity of nodes is different and the rate at which the sender is sending data might not be the same rate at which the receiver accepts it. In this paper, we discuss on designing issues of data link layer. The primary focus is on various error detecting and controlling mechanisms.

**Keywords:** error detection, LRC, VRC, CRC, checksum.

## I. INTRODUCTION

**D**ata Link Layer is the second layer of OSI Layered Model. Data link layer is responsible for converting data stream to signals bit by bit and to send that over the underlying hardware. At the receiving end, Data link layer picks up data from hardware which are in the form of electrical signals assembles them in a recognizable frame format, and hands over to upper layer [2][6].

Data link layer has two sub-layers:

- **Logical Link Control** : Deals with protocols, flow-control and error control.
- **Media Access Control** : Deals with actual control of media.

Data link layer does many tasks on behalf of upper layer. The main functionalities are:

### a) Framing

Data-link layer takes packets from Network Layer and encapsulates them into Frames. Then, sends each Frame bit-by-bit on the hardware. At receiver's end Data link layer picks up signals from hardware and assembles them into frames.

### b) Addressing

Data-link layer provides layer-2 hardware addressing mechanism. Hardware address is assumed to be unique on the link. It is encoded into hardware at the time of manufacturing.

### c) Synchronization

When data frames are sent on the link, both machines must be synchronized in order to transfer to take place.

### d) Error Control

Sometimes signals may have encountered problem in transition and bits are flipped. These errors are detected and attempted to recover actual data bits. It also provides error reporting mechanism to the sender.

### e) Flow Control

Stations on same link may have different speed or capacity. Data-link layer ensures flow control that enables both machines to exchange data on same speed.

### f) Multi-Access

Hosts on shared link when tries to transfer data, has great probability of collision. Data-link layer provides mechanism like CSMA/CD to equip capability of accessing a shared media among multiple Systems.

## II. DESIGNING ISSUES IN DATA LINK LAYER

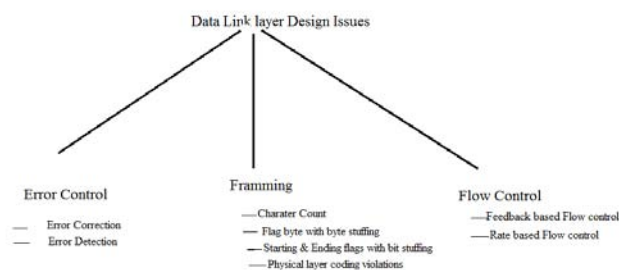


Figure 1 : Designing issues in data link layer

The designing issues of data link layer are following:

### a) Error Control

- Error control includes both error detection and error correction.
- It allows the receiver to inform the sender if a frame is lost or damaged during transmission and coordinates the retransmission of those frames by the sender.
- Error control in the data link layer is based on automatic repeat request (ARQ). Whenever an error is detected, specified frames are retransmitted [4][7].

### b) Framing

- Break down a stream of bits into smaller, digestible chunks called frames
- Allows the physical media to be shared

Author <sup>α</sup> <sup>σ</sup> : FET, Mody University of Science and Technology, India.  
e-mails: Dhariwal.monika@gmail.com, ruhi.saxena2011@gmail.com

- Multiple senders and/or receivers can time multiplex the link
- Each frame can be separately addressed
- Provides manageable unit for error handling
  - Easy to determine whether something went wrong
  - And perhaps even to fix it if desire

- Receiver must inform the sender before the limits are reached and request that the transmitter to send fewer frames or stop temporarily.
- Since the rate of processing is often slower than the rate of transmission, receiver has a block of memory (buffer) for storing incoming data until they are processed

c) Flow Control

- Flow control coordinates the amount of data that can be sent before receiving acknowledgement
- It is one of the most important functions of data link layer.
- Flow control is a set of procedures that tells the sender how much data it can transmit before it must wait for an acknowledgement from the receiver.
- Receiver has a limited speed at which it can process incoming data and a limited amount of memory in which to store incoming data.

III. ERROR CONTROL

Error controls are the techniques that enable reliable delivery of digital data over unreliable communication channels. Many communication channels are subject to channel noise, and thus errors may be introduced during transmission from the source to a receiver.

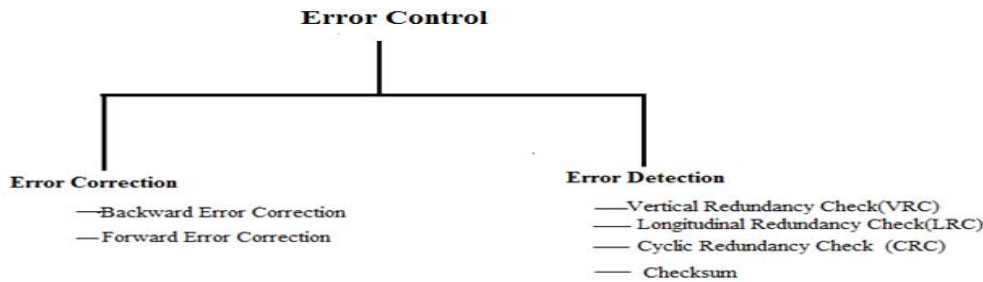


Figure 2 : Error Control Methodologies

Generally, there are two types of Error:

*Single bit error:* Single-bit error means that only one bit of data been change through transportation of data [7]. It changes either from 0 to 1 or 1 to 0. This one

bit changed cannot be ignored since one bit change can change the whole meaning of the data that is transmitted. Figure-3 shows an example of this type of error.

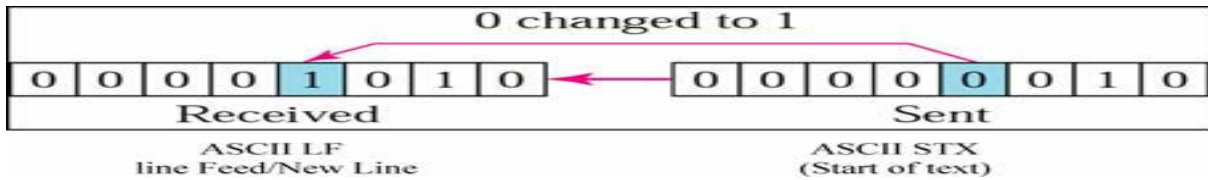


Figure 3 : Single-bit error

*Burst error:* Burst error means that two or more bits are changed when the transmitting data from the sender to the receiver the data units have change from 0 to 1 or 1 to 0 because of the channel interference [8]. Figure-4 shows the burst error of 8-bits. Burst errors are

likely to occur rather that the single-bit error. The duration of the error was longer that duration of 1 bit, which means the data is affected by the noise usually affect a set of bits. The number of bits that corrupted always depends on the data rate and duration of noise.

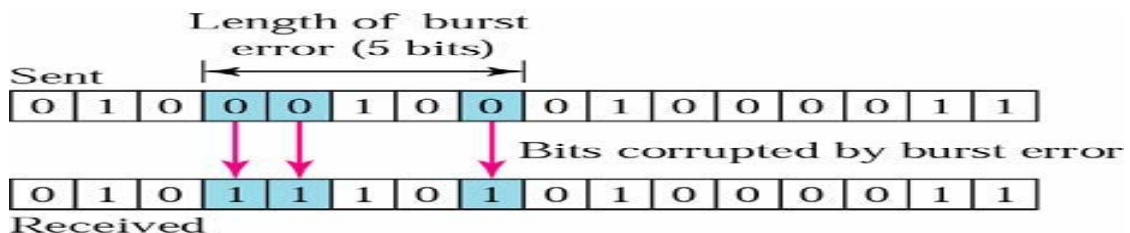


Figure 4 : Burst error of length 8

a) Error Correction

In digital world, error correction can be done in two ways:

- **Backward Error Correction:** When the receiver detects an error in the data received, it requests back the sender to retransmit the data unit.
- **Forward Error Correction:** When the receiver detects some error in the data received, it uses an error-correcting code, which helps it to auto-recover and corrects some kinds of errors.

The first one, Backward Error Correction, is simple and can only be efficiently used where retransmitting is not expensive, for example fiber optics. But in case of wireless transmission retransmitting may cost too much. In the latter case, Forward Error Correction is used [5].

To correct the error in data frame, the receiver must know which bit (location of the bit in the frame) is corrupted. To locate the bit in error, redundant bits are used as parity bits for error detection. If for example, we take ASCII words (7 bits data), then there could be 8 kind of information we need. Up to seven information to tell us which bit is in error and one more to tell that there is no error.

b) Error Detection

Error detection means to decide whether the received data is correct or not without having a copy of the original message. Error detection uses the concept of redundancy, which means adding extra bits for detecting errors at the destination.

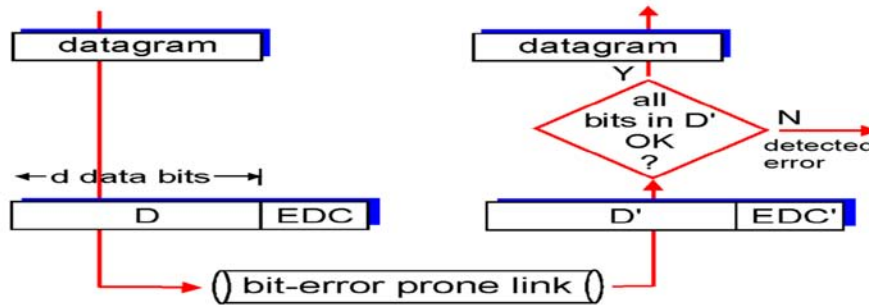


Figure 5 : Error Detection Mechanism

Where D = Data, EDC = Error Detection Code (redundancy)

There are four types of redundancy checks:

- VRC (Vertical Redundancy Check).
- LRC (Longitudinal Redundancy Check).
- CRC (Cyclical Redundancy Check).
- Checksum

i. Vertical Redundancy Check (VRC)

It is also known as parity check. There are two types of parity check schemes: even and odd parityChecks [8]. In an even parity check scheme, the

sender simply includes one additional bit and choose its value such that the total number of 1's in the d+1 bits (the original information plus a parity bit) is even. For odd parity check scheme, the parity bit value is chosen such that there is odd number of 1's. Table-1 shows a table contains 3 bits string. The transmitter will add 0 or 1 to the bits string according to the parity check mechanism (even or odd). When the receiver receives the bits string, the receiver will use the same mechanism to count the 1's in the bit string to determine whether it matches the counted parity from the transmitter or not [1].

Table 1 : Parity bits that are compute for bits string

3 bits string	Odd parity	Even parity
000	1	0
001	0	1
010	0	1
100	0	1
110	1	0
111	1	0

This scheme can detect only single bits. So if two or more bits are changed then that cannot be detected.

ii. Longitudinal Redundancy Checksum (LRC)

Longitudinal Redundancy Checksum is an error detecting scheme which overcomes the problem of two erroneous bits. In this concept of parity bit is used but with slightly more intelligence. With each byte we send

one parity bit then send one additional byte which has the parity corresponding to the each bit position of the sent bytes. So the parity bit is set in both horizontal and vertical direction. If one bit gets flipped we can tell which row and column have error then we find the intersection of the two and determine the erroneous bit. If 2 bits are in error and they are in the different column and row then they can be detected. If the errors are in the same column then the row will differentiate and vice versa. Parity can detect the only odd number of errors. If they are even and distributed in a fashion that in all direction then LRC may not be able to find the error [2][5].

iii. *Cyclic Redundancy Checksum (CRC)*

In this method, a sequence of redundant bits, called the CRC or the CRC remainder, is appended to the end of the data unit so that the resulting data unit becomes exactly divisible by a second, predetermined binary number. At its destination, the incoming data unit is divided by the same number. If at this step there is no remainder, the data unit assumes to be correct and is accepted, otherwise it indicates that data unit has been damaged in transmission and therefore must be rejected. The redundancy bits used by CRC are derived by dividing the data unit by a predetermined divisor. The remainder is the CRC.

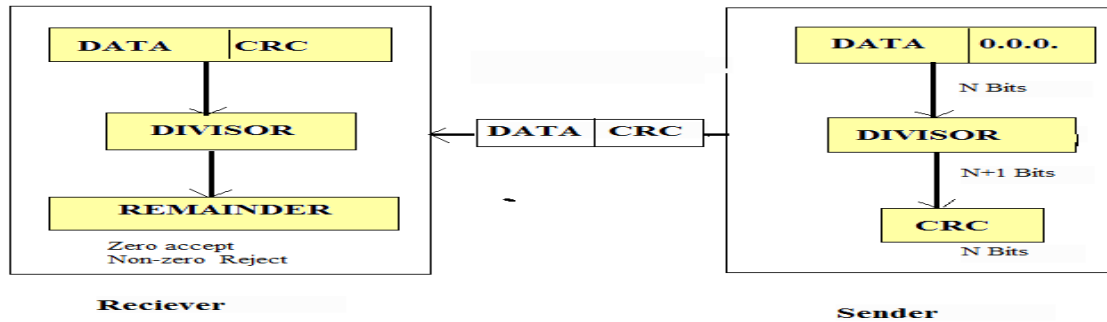


Figure 6 : Mechanism of CRC at sender and Receiver side

For example, the CRC generator at the sender side:

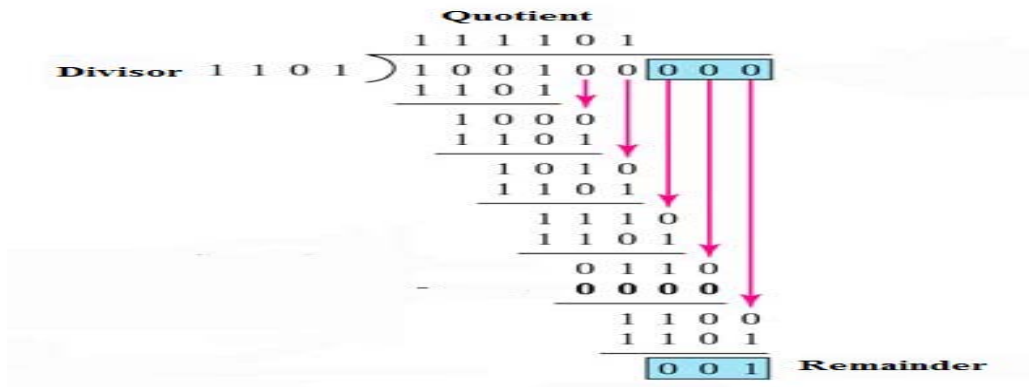


Figure 7 : CRC in sender side

The CRC checker at receiver side:

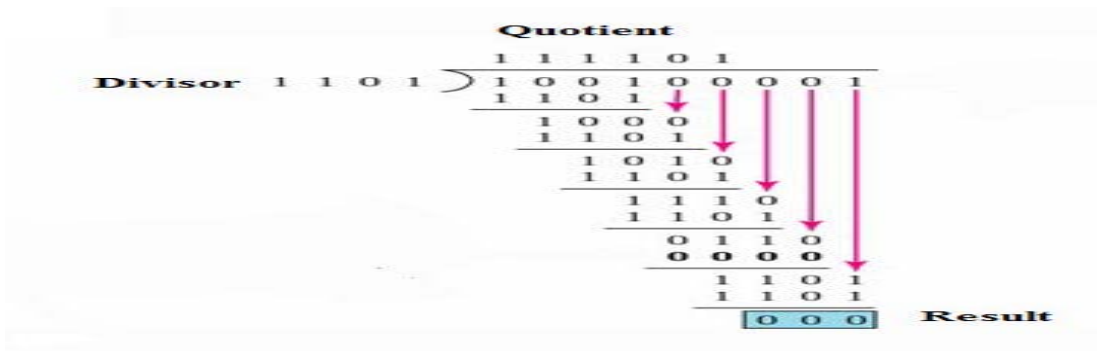


Figure 8 : CRC in receiver side

iv. Checksum

The checksum is based on the concept of redundancy. As shown in Figure-9, in the sender, the checksum generator subdivides the data unit into equal segments of  $n$  bits (usually 16). These segments are added using ones complement arithmetic in such a way that the total is also  $n$  bits long. That total (sum) is then complemented and appended to the end of the original data unit as redundancy bits, called the checksum field.

The extended data unit is transmitted across the network. So if the sum of the data segments is  $T$ , the checksum will be  $\sim T$ . The receiver performs the same calculation on the received data and compares the result with the received checksum. If the result is 0, the receiver keeps the transmitted data; otherwise, the receiver knows that an error occurred discards the transmitted data [9].

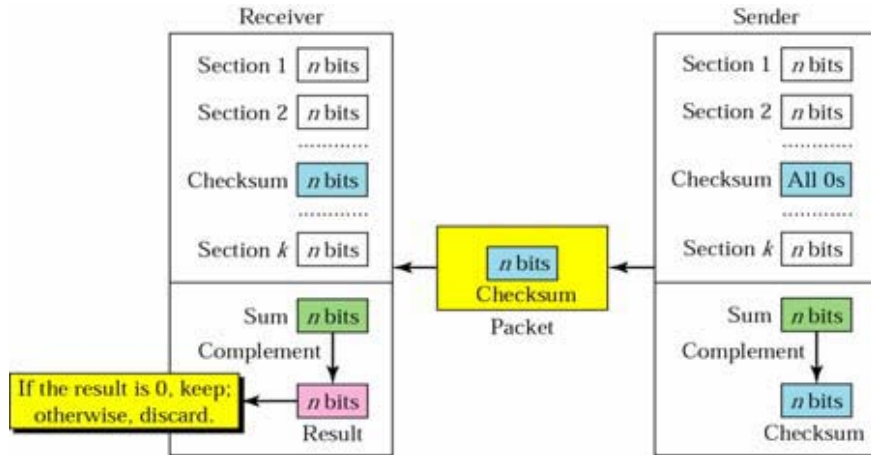


Figure 9 : Working of Checksum at sender and Receiver side

For example, the following block of 16 bits is to be sent using a checksum of 8 bits

10101001 00111001

The numbers are added using one's complement

```

00111001
          Sum  11100010
Checksum00011101
    
```

The pattern sent is

10101001 00111001 00011101

At the receiver end, same calculation has been done on the received data and compares the result with the received checksum in the following way:

```

00111001
00011101
          Sum  11111111
Checksum 00000000
    
```

When the receiver adds the three sections, it will get all 1s, which, after complementing, is all 0s and shows that there is no error.

IV. CONCLUSION

There are various methods of detecting error in the data link layer. Every method of error detection can detect error accurately and effectively. Every method has its own advantage and its own mechanism to detect error. VRC is simple and can detect all single-bit error. CRC has a very good performance in detecting single-

bit errors, double errors, an odd number of errors, and burst errors while checksum is not as efficient as the CRC.

REFERENCES RÉFÉRENCES REFERENCIAS

1. Dimitri Bertsekas and Robert Gallager, "Data Networks".
2. Larry Peterson and Bruce Davie, "Computer Networks: A Systems Approach".
3. <http://www.cse.iitk.ac.in/users/dheeraj/cs425/lec09.html/>
4. <http://ecomputernotes.com/computernetworkingnotes/communication-networks/what-is-error-correction-and-detection>.
5. Forouzan, "Data Communications and Networking".
6. J. Chellis, C. Perkins and M. Strebe, "MCSE: Networking Essentials Study Guide with CDROM", 2nd Ed. 1999.
7. F. Behrouz and M. Firouz, "Computer Networks: A Top Down Approach", 1st International Ed. McGraw-Hill., 2012.
8. J. Fletcher, "An arithmetic checksum for serial transmissions, Communications". IEEE Transactions on. 30: 247-252, 1982.
9. L. L. Peterson and B. S. Davie, "Computer networks: a systems approach", 2007, 3rd Ed. Elsevier.



This page is intentionally left blank