Data Link Layer

DLL Design issues

- Services Provided to the Network Layer
    - Unacknowledged connectionless service
    - Acknowledged connectionless service
    - Acknowledged connection-oriented service

Design issues: protocols

- Framing
  - Break stream of bits up into discrete frames
- Error control
  - How does a sender know that all packets are correctly received
- Flow control
  - How to prevent a sender to overload the receiver with packets

DATA LINK LAYER DESIGN ISSUES

3- Acknowledged connection-oriented service.

- The source and destination machines establish a connection before any data are transferred.

- Each frame sent over the connection is number.
  - ✓ Guarantees that each frame sent is indeed received.
  - ✓ Guarantees that each frame is received exactly once.
  - ✓ All frames are received in the right order.

- It is appropriate over long, unreliable links

DATA LINK LAYER DESIGN ISSUES

2- Framing

- Data-link layer takes packets from Network Layer and encapsulates them into Frames.

- Then, it sends each frame bit-by-bit on the hardware. At receiver end, data link layer picks up signals from hardware and assembles them into frames.

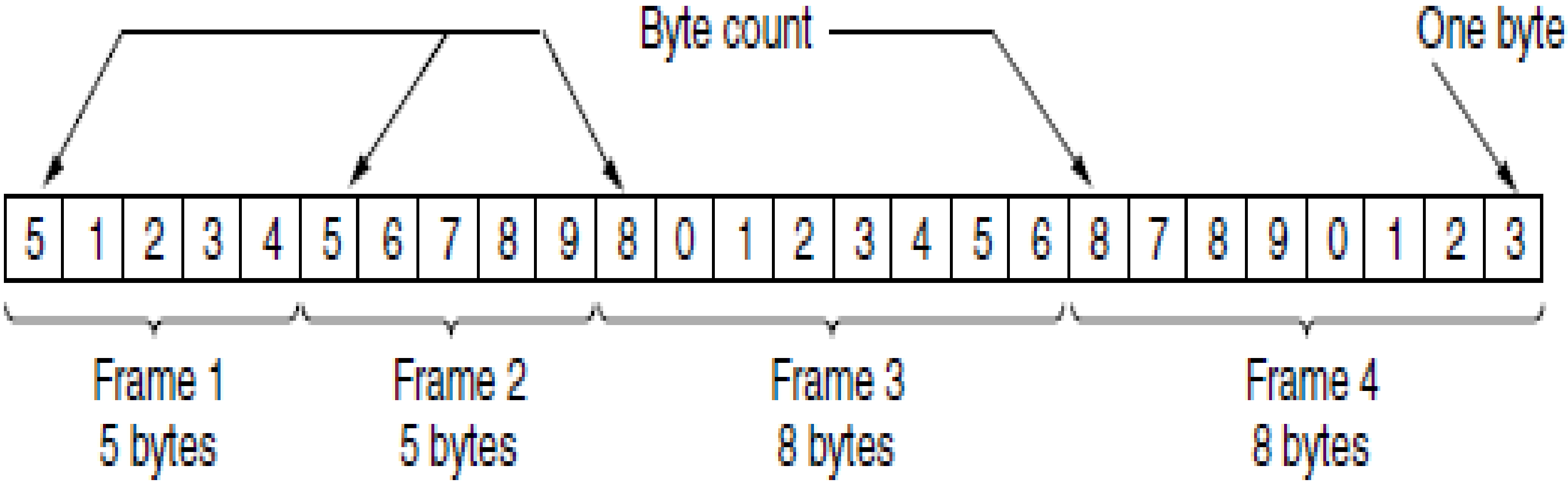
- A good design must make it easy for a receiver to find the start of new frames while using little of the channel bandwidth.

DATA LINK LAYER DESIGN ISSUES

- There are four methods:

A- Byte count

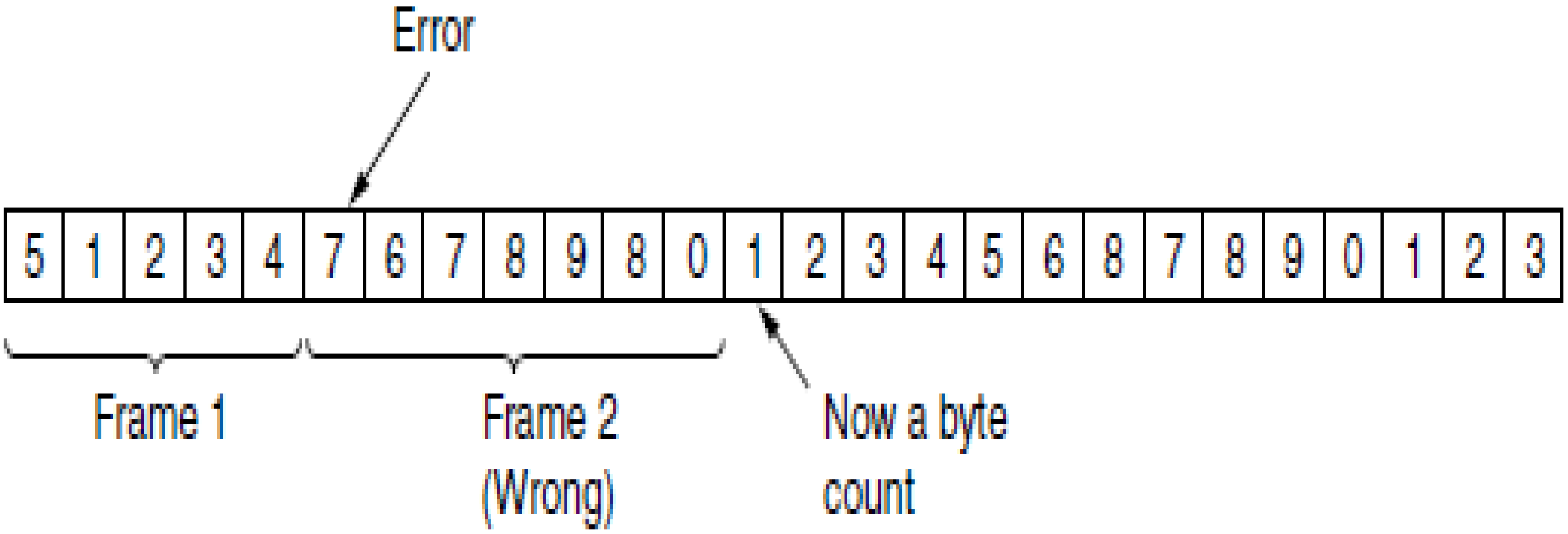
- The first framing method uses a field in the header to specify the number of bytes in the frame.



5	1	2	3	4	5	6	7	8	9	8	0	1	2	3	4	5	6	8	7	8	9	0	1	2	3

DATA LINK LAYER DESIGN ISSUES

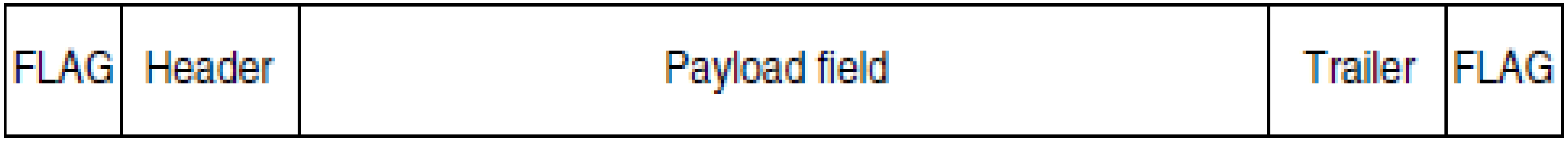
- The trouble with this algorithm is that the count can be garbled by a transmission error.



DATA LINK LAYER DESIGN ISSUES

B- Flag bytes with byte stuffing

- Each frame starts with special start and end bytes. Often the same byte, called a flag byte
- Two consecutive flag bytes indicate the end of one frame and the start of the next.



FLAG	Header	Payload field	Trailer	FLAG

DATA LINK LAYER DESIGN ISSUES

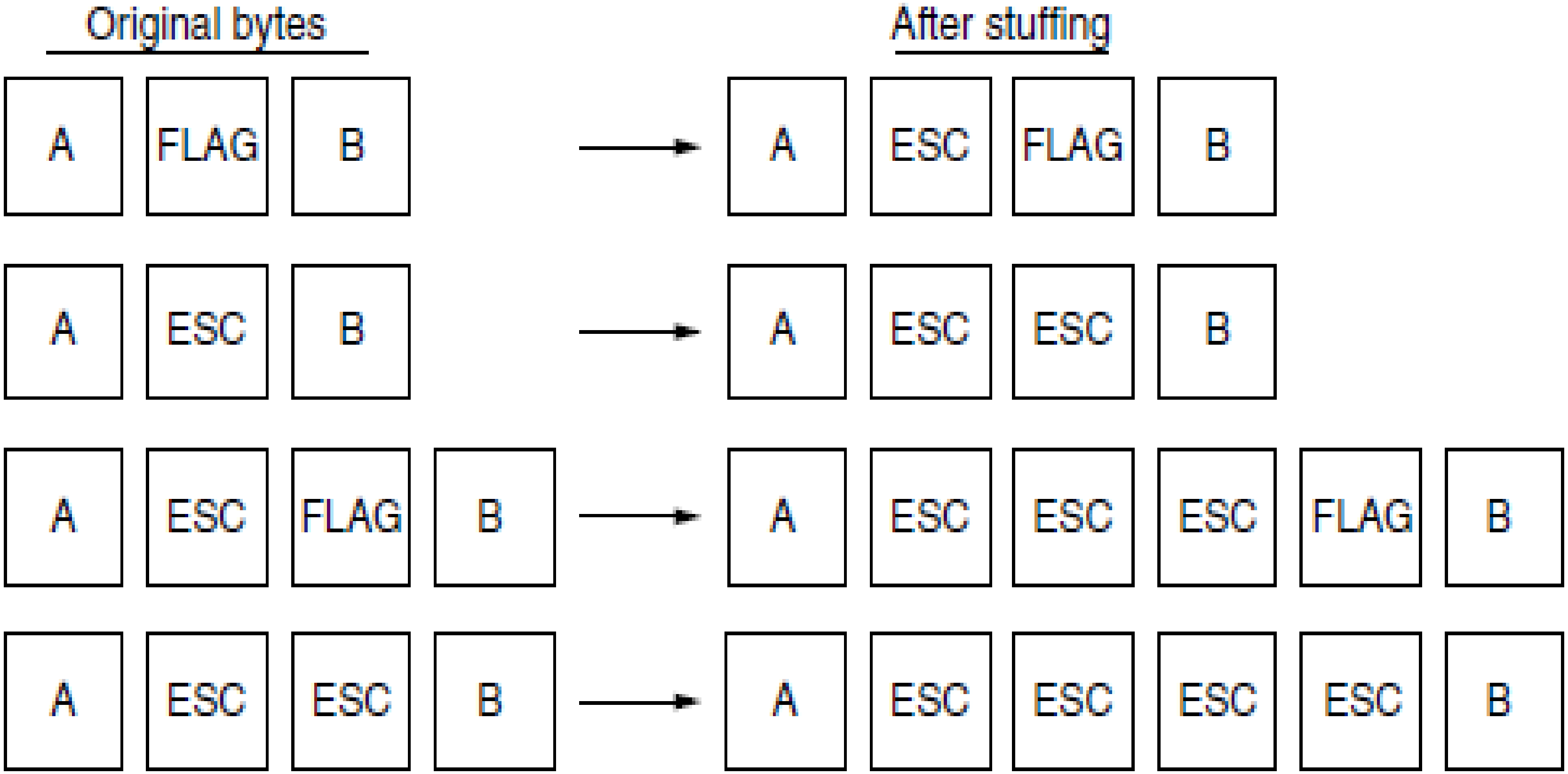
- However, there is a still a problem we have to solve. It may happen that the flag byte occurs in the data.

  - This problem is to have the sender's data link layer insert a special escape byte (ESC) just before each "accidental" flag byte in the data

  - This technique is called byte stuffing.

DATA LINK LAYER DESIGN ISSUES

- what happens if an escape byte occurs in the middle of the data?

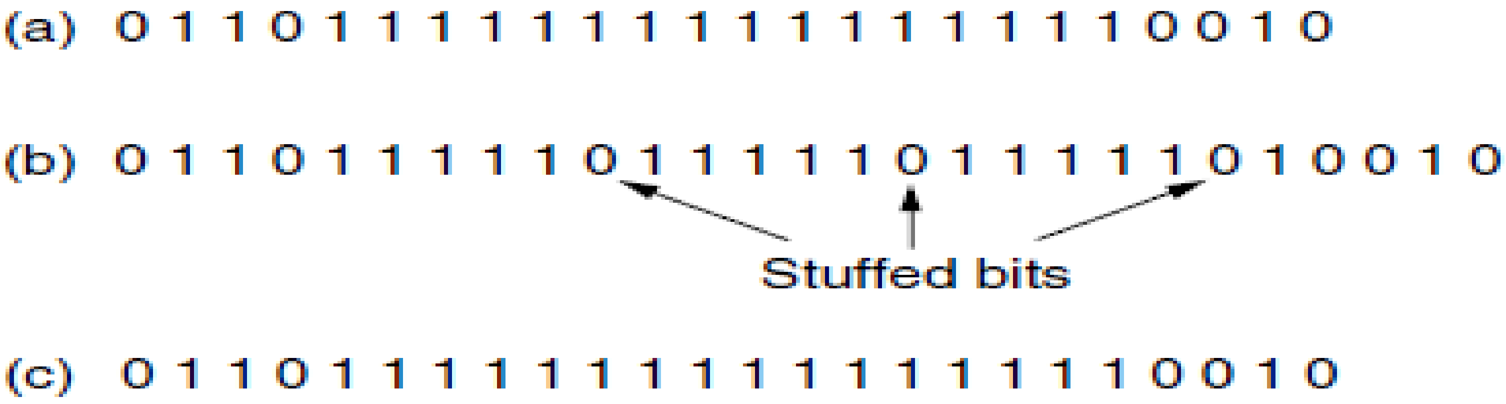


Original bytes		After stuffing
A FLAG B	→	A ESC FLAG B
A ESC B	→	A ESC ESC B
A ESC FLAG B	→	A ESC ESC ESC FLAG B
A ESC ESC B	→	A ESC ESC ESC ESC B

DATA LINK LAYER DESIGN ISSUES

C- Flag bits with bit stuffing

- Each frame begins and ends with a special bit pattern, 01111110 or 0x7E in hexadecimal. This pattern is a flag byte.
- Whenever the sender's data link layer encounters five consecutive 1s in the data, it aut ; bit



(a) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

(b) 0 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 0 0 1 0

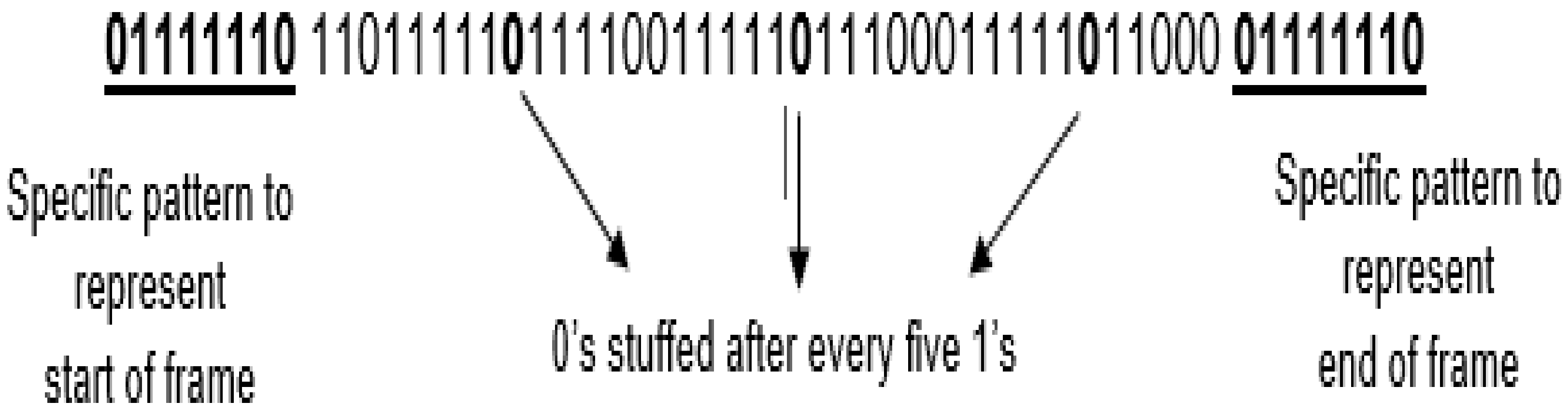
(c) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

DATA LINK LAYER DESIGN ISSUES

Example: Use the **flag bits with bit stuffing** method to encapsulate the data below into frames:

11011111111100111111110001111111000

Solution:



01111110 11011111**0**111100111111**0**11100011111**0**11000 **01111110**

Specific pattern to represent start of frame

0's stuffed after every five 1's

Specific pattern to represent end of frame

DATA LINK LAYER DESIGN ISSUES

Notes:

- With both bit and byte stuffing, a side effect is that the length of a frame now depends on the contents of the data it carries.

- Many data link protocols use a combination of these methods for safety. A common pattern used for Ethernet and 802.11 is to have a frame begin with a well-defined pattern called a preamble.
  - long (72 bits is typical for 802.11)
  - The preamble is then followed by a length (i.e., count) field in the header

DATA LINK LAYER DESIGN ISSUES

3- Error Control

How to make sure all frames are eventually delivered to the network layer at the destination and in the proper order.

- For the connection oriented service, the data link layer must guarantee that all the frames sent arrive at the receiver in the proper order, and only one correct copy of each frame is expected at the receiver.

- The receiver usually sends an acknowledgement (ACK) or a negative acknowledgement (NACK) back to the transmitter.
- These ACKs and NACKs are typically very short in length and do not consume the channel capacity.

DATA LINK LAYER DESIGN ISSUES

- To stop the loss of a frame or an ACK/NACK due to channel errors causing the transmitter to wait forever to transmit the next frame, a time out procedure is used. If after certain time, an ACK or NACK is not reached the transmitter, the frame is sent again.

- However, if only the ACK is lost, the sender transmits a copy of the same frame causing duplication and waste of channel capacity.

- By introducing sequence numbers to all frames, passing of a duplicate frame to the network layer can be prevented.

DATA LINK LAYER DESIGN ISSUES

4- Flow Control

- What to do with a sender that systematically wants to transmit frames faster than the receiver can accept them.

- This situation can occur when the sender is running on a fast, powerful computer and the receiver is running on a slow, low-end machine.

- A common situation is when a smart phone requests a Web page from a far more powerful server.

DATA LINK LAYER DESIGN ISSUES

- Clearly, something has to be done to prevent this situation. Two approaches are commonly used:

  - Feedback-based flow control, the receiver sends back information to the sender giving it permission to send more data, or at least telling the sender how the receiver is doing. (The link layer and higher layers)

  - Rate-based flow control, the protocol has a built-in mechanism that limits the rate at which senders may transmit data, without using feedback from the receiver. (Transport layer)