

2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.1. Các vấn đề thiết kế tầng Liên kết dữ liệu.

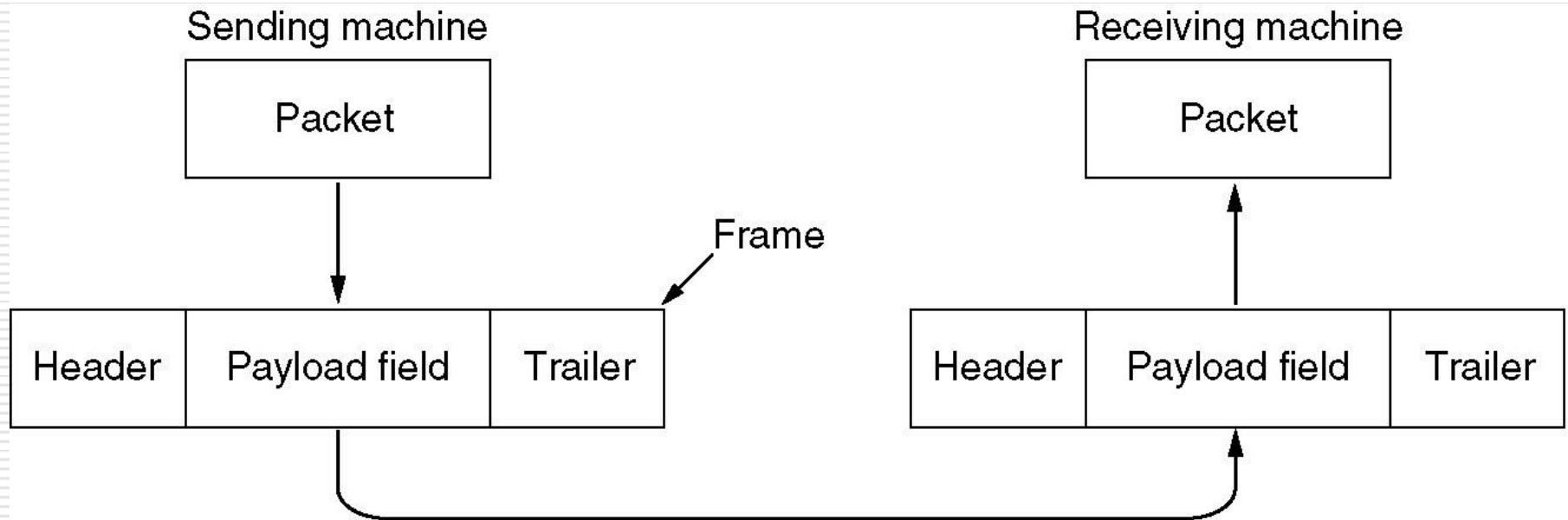
a Vai trò chức năng của tầng Liên kết dữ liệu:

- Chức năng chủ yếu của tầng liên kết dữ liệu là thực hiện thiết lập các liên kết, duy trì và huỷ bỏ các liên kết dữ liệu.
- Kiểm soát lỗi và kiểm soát lưu lượng.
- Chia thông tin thành các khung thông tin (Frame), truyền các khung tuần tự và xử lý các thông điệp xác nhận (Acknowledgement Frame) từ bên máy thu gửi về.
- Tháo gỡ các khung thành chuỗi bit không cấu trúc chuyển xuống tầng vật lý.
- Tầng 2 bên thu, tái tạo chuỗi bit thành các khung thông tin.
- Đường truyền vật lý có thể gây lỗi, nên tầng liên kết dữ liệu phải giải quyết vấn đề kiểm soát lỗi, kiểm soát luồng, kiểm soát lưu lượng, ngăn không để nút nguồn gây “ngập lụt” dữ liệu cho bên thu có tốc độ thấp hơn.
- Tầng Data link được chia làm 2 tầng con:
 - ✓ Trong các mạng quảng bá, tầng con MAC (Medium Access Sublayer) điều khiển việc truy nhập đường truyền.
 - ✓ Logical Link Control (LLC): Có nhiệm vụ tạo lập và duy trì liên kết giữa các thiết bị giao tiếp.

2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.1. Các vấn đề thiết kế tầng Liên kết dữ liệu.

a Vai trò chức năng của tầng Liên kết dữ liệu:

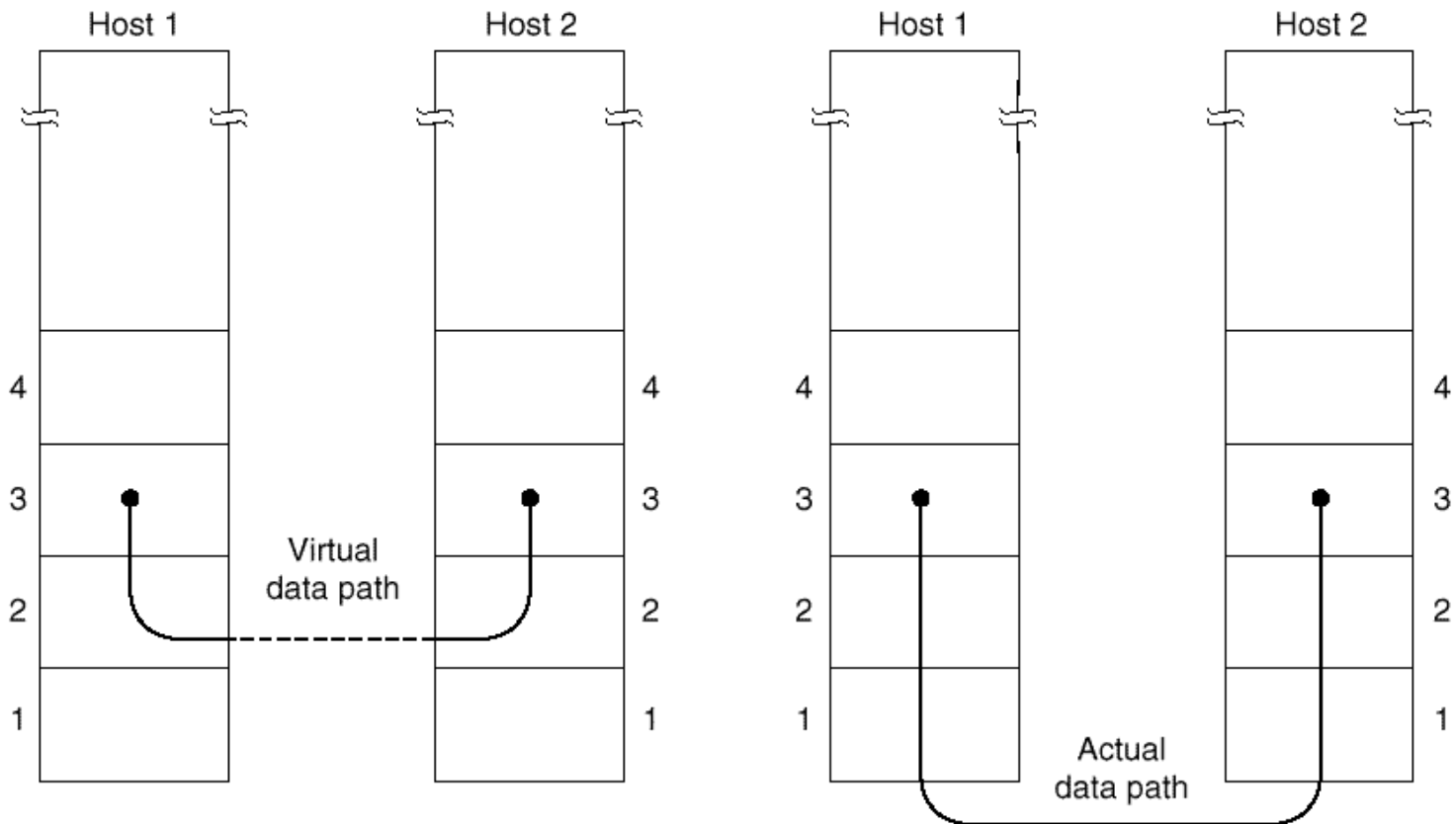


2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.1. Các vấn đề thiết kế tầng Liên kết dữ liệu.

b. Các dịch vụ được cung cấp cho lớp Mạng (Network)

Dịch vụ chính chuyển dữ liệu từ lớp Mạng trên máy nguồn sang lớp Mạng của máy đích:



2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.1. Các vấn đề thiết kế tầng Liên kết dữ liệu.

b. Các dịch vụ được cung cấp cho lớp Mạng (Network)

Các dữ liệu thật sự được cung cấp có thể thay đổi từ hệ thống này sang hệ thống khác. Ba khả năng hợp lý thường được cung cấp:

✓ Dịch vụ không nối kết không được báo nhận:

Máy nguồn gửi các frame độc lập tới máy đích mà không yêu cầu máy đích báo nhận chúng. Hầu hết các LAN sử dụng dịch vụ này.

✓ Dịch vụ không nối kết được báo nhận:

Không có các kết nối logic, nhưng các frame được báo nhận riêng lẻ. Được sử dụng trên các kênh không đáng tin cậy như các hệ thống không dây.

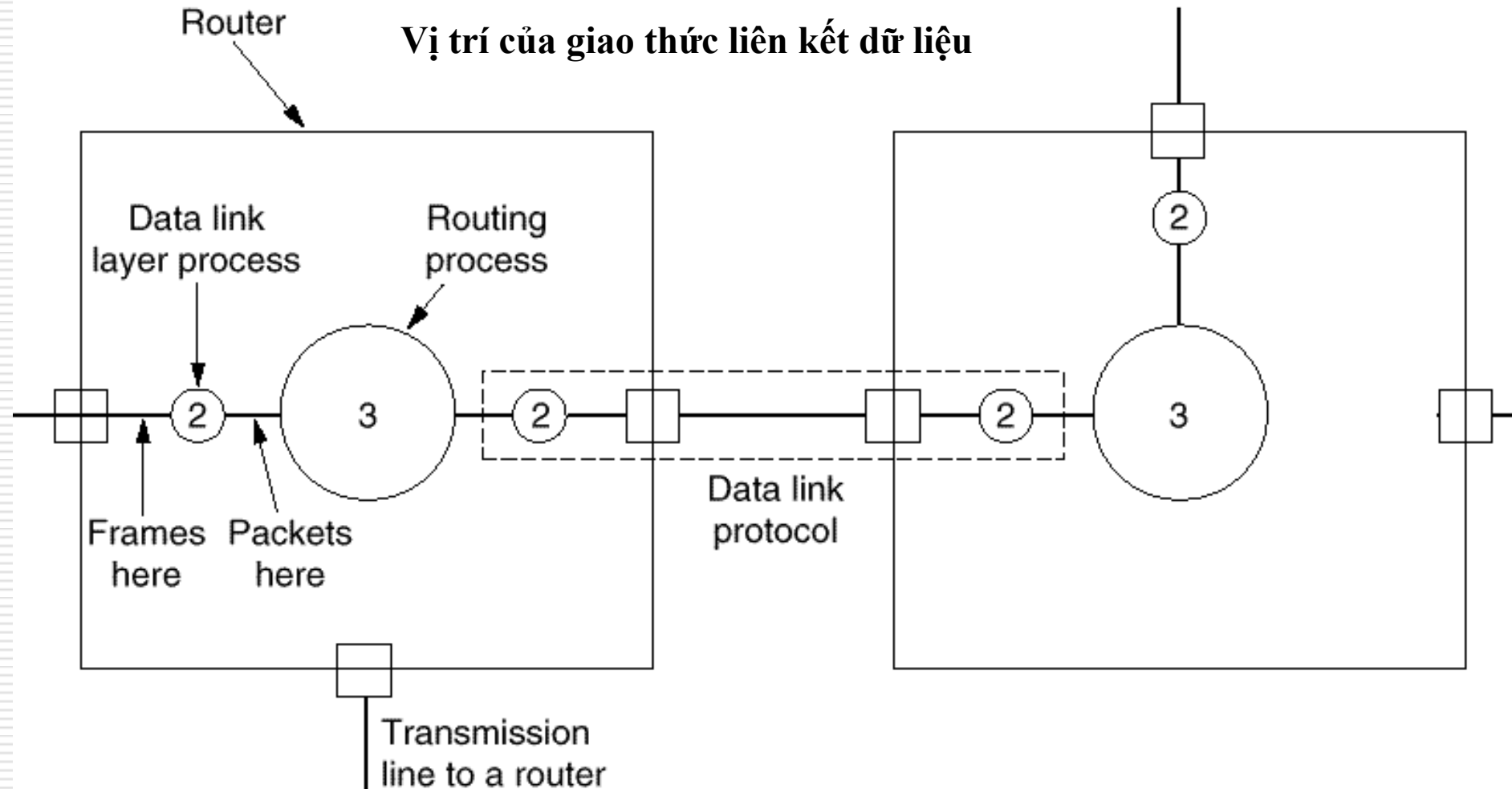
✓ Dịch vụ hướng nối kết được báo nhận

Máy gửi và máy nhận thiết lập liên kết trước khi dữ liệu được truyền. Mỗi frame được đánh số. Mỗi frame được nhận 1 lần, đúng thứ tự.

2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.1. Các vấn đề thiết kế tầng Liên kết dữ liệu.

b. Các dịch vụ được cung cấp cho lớp Mạng (Network)



2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.1. Các vấn đề thiết kế tầng Liên kết dữ liệu.

c. Tạo frame

- **Nhiệm vụ của lớp Data link là phát hiện và sửa lỗi nếu cần, do vậy nó chia luồng bit thành các frame riêng biệt và tính Checksum cho mỗi frame.**
- **Tách luồng bit thành các frame có thể thực hiện bằng chèn các khoảng trống thời gian – để xảy ra rủi ro.**

Có 4 phương pháp tách frame:

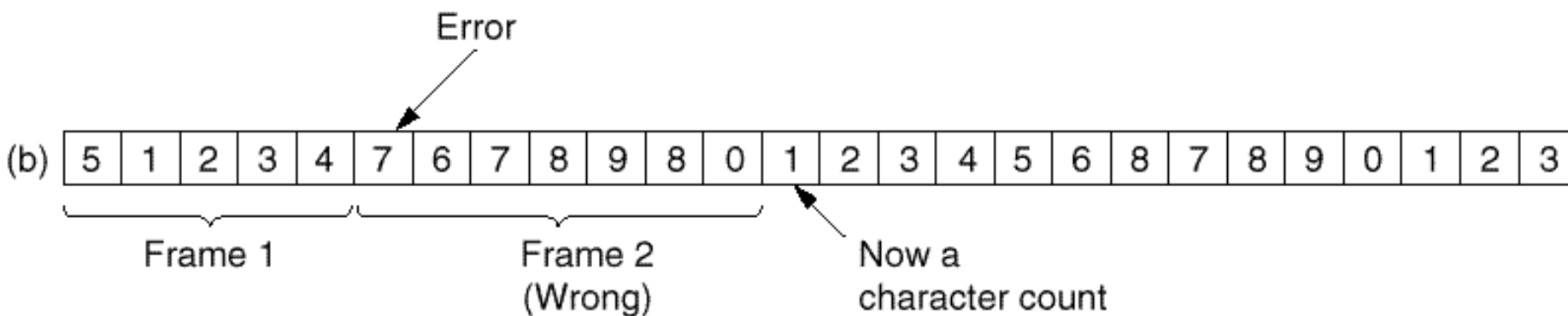
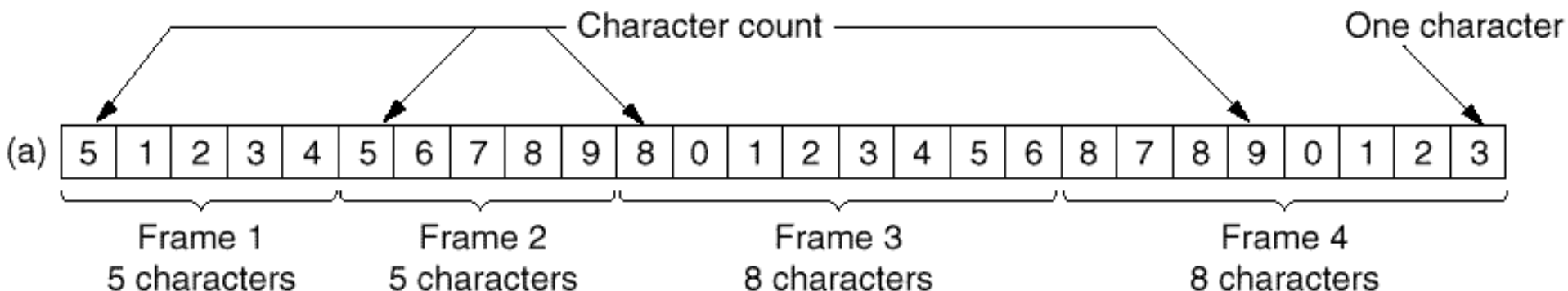
- ✓ **Số đếm ký tự**
 - ✓ **Các byte cờ với việc nhồi byte**
 - ✓ **Cờ bắt đầu và cờ kết thúc với việc nhồi bit**
 - ✓ **Các vi phạm mã hoá lớp vật lý**
-

2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.1. Các vấn đề thiết kế tầng Liên kết dữ liệu.

c. Tạo frame

✓ Số đếm ký tự - khó khắc phục lỗi nên dần bị loại bỏ



2.2.3. Tầng Liên kết dữ liệu (Data link)

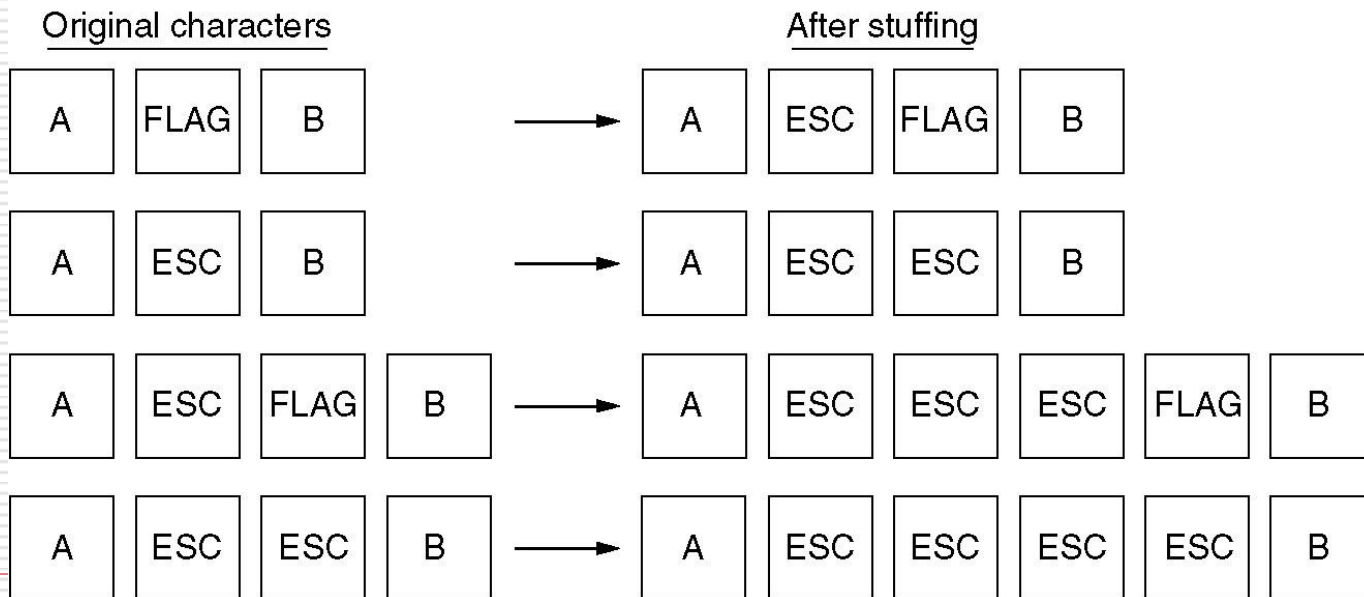
2.2.3.1. Các vấn đề thiết kế tầng Liên kết dữ liệu.

c. Tạo frame

✓ Các byte cờ với việc nhồi byte: Mọi frame được bắt đầu và kết thúc bằng những byte đặc biệt – flag byte. Hai flag byte liên tục báo hiệu kết thúc 1 frame và bắt đầu 1 frame mới. Khi có các flage của dữ liệu thì cần bổ sung thêm một byte ESC



(a)



(b)

2.2.3. Tầng Liên kết dữ liệu (Data link)


2.2.3.1. Các vấn đề thiết kế tầng Liên kết dữ liệu.

c. Tạo frame

✓Cờ bắt đầu và cờ kết thúc với việc nhồi bit: Cho phép frame dữ liệu chứa số bit tùy ý. Mỗi frame bắt đầu và kết thúc bằng mẫu bit 0111110, lớp data link gặp 5 bit 1 liên nhau thì bổ sung thêm 1 bit 0.

(a) 0 1 1 0 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



Stuffed bits

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

2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.1. Các vấn đề thiết kế tầng Liên kết dữ liệu.

c. Tạo frame

✓ Các vi phạm mã hoá lớp vật lý: Áp dụng cho các mạng, việc mã hoá trên phương tiện vật lý chứa một sự dư thừa nào đó.

Ví dụ như trong mạng LAN, mã hoá 1 bit dữ liệu bằng cách sử dụng 2 bit vật lý. Một bit thường là cặp cao-thấp và một 0 bit là một cặp thấp-cao. Như vậy mọi bit dữ liệu đều có sự chuyển tiếp ở giữa. Tổ hợp cao-cao và thấp-thấp không sử dụng cho dữ liệu, nhưng được dùng trong việc tách frame

1001001100010111

10010110010110100101011001101010

2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.1. Các vấn đề thiết kế tầng Liên kết dữ liệu.

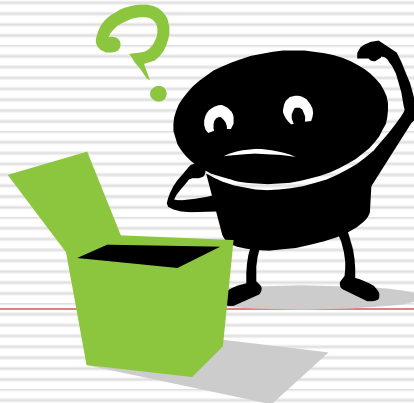
d. Kiểm soát lỗi

Một giao thức yêu cầu máy nhận gửi một frame đặc biệt chứa thông báo nhận dương hoặc âm để xác định frame đến đích an toàn.

Trường hợp frame gửi bị mất, sẽ dẫn tới tình trạng treo mãi.

Giải quyết vấn đề này bằng cách đưa thêm các timer vào lớp Data link:

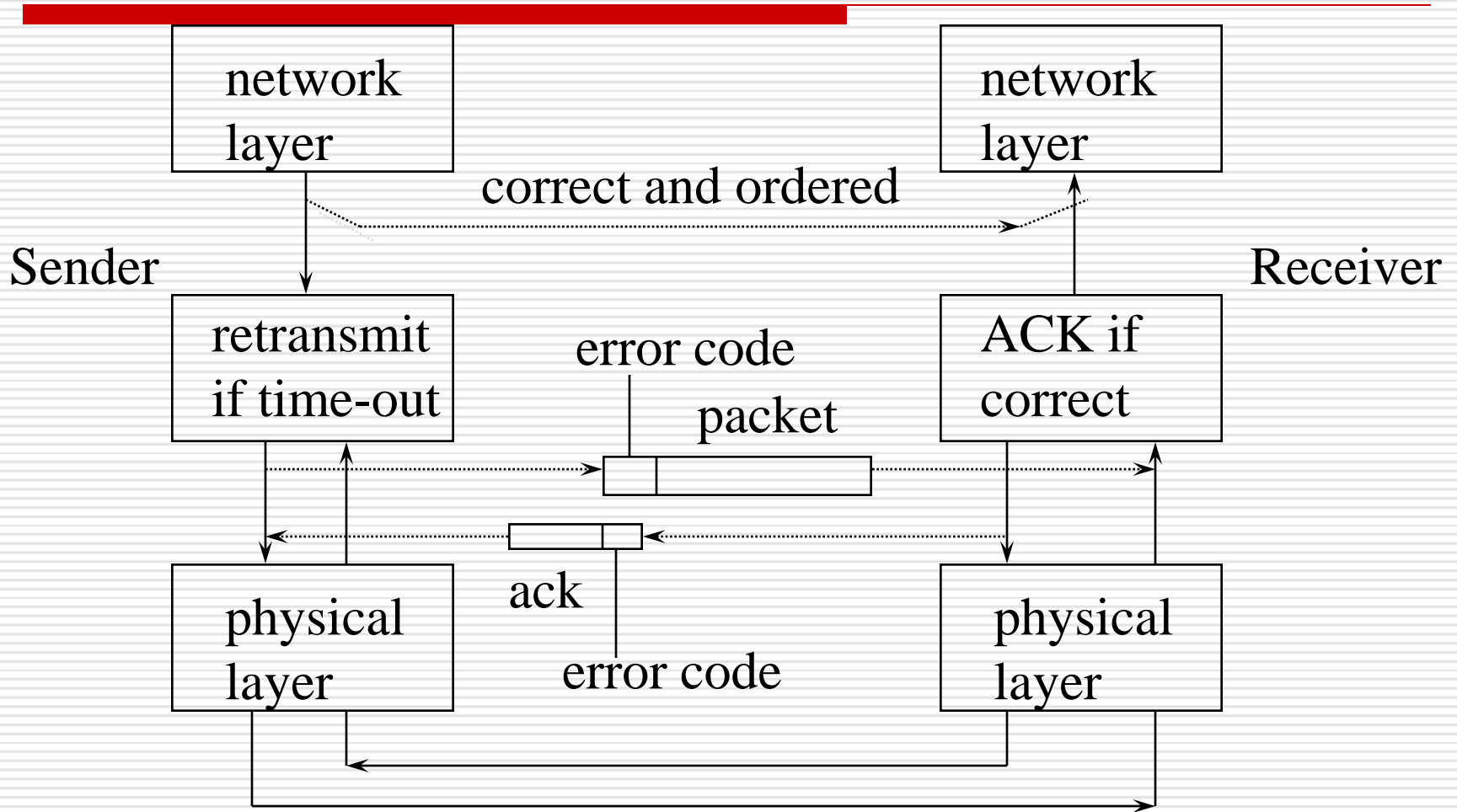
- Khi máy gửi một frame, nó cũng khởi động một timer.**
- Khi máy gửi nhận được thông báo nhận thì timer tự động hủy.**
- Khi bị mất gói tin, timer sẽ ngừng hoạt động và máy gửi sẽ gửi lại gói tin đó**



2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.1. Các vấn đề thiết kế tầng Liên kết dữ liệu.

d. Kiểm soát lỗi



2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.1. Các vấn đề thiết kế tầng Liên kết dữ liệu.

d. Điều khiển lưu lượng

- **Lý do cần điều khiển lưu lượng:**

- Máy gửi chạy trên một máy nhanh (hoặc có tải nhẹ) còn máy nhận chạy trên một máy chậm (hoặc có tải nặng) – dẫn tới máy nhận gặp khó khăn trong việc xử lý gói tin.
- Máy nhận không thể xử lý được các frame khi chúng đến, dẫn tới mất gói tin.

- **Cách khắc phục:**

- Điều khiển lưu lượng dựa vào hồi tiếp: Máy nhận gửi lại thông tin cho phép máy gửi gửi thêm dữ liệu hoặc cho máy gửi biết mình nhận như thế nào. (Không thuộc lớp data link)
 - Điều khiển lưu lượng dựa vào tốc độ: Giao thức có sẵn cơ chế giới hạn tốc độ mà máy gửi có thể truyền mà không có hồi tiếp từ máy nhận.
-

2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.2. Phát hiện và sửa lỗi

-
- **Phát hiện lỗi:**
 - ✓ Các lỗi gây ra bởi sự suy giảm tín hiệu, nhiễu.
 - ✓ Bên nhận phát hiện sự xuất hiện của các lỗi: thông báo bên gửi truyền lại hoặc bỏ frame đó
 - **Sửa lỗi:**
 - Bên nhận xác định và sửa bit bị lỗi không cần phải truyền lại
-

2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.2. Phát hiện và sửa lỗi

a. Các mã sửa lỗi

Cấu tạo codeword n bit

m bit dữ liệu	r bit kiểm tra
-----------------	------------------

$$n=m+r$$

There are 2^n possible codewords and 2^m possible data messages.

Hamming distance between codewords:

$\min d(C_1, C_2) = \text{number of (same bit position) bits which differ}$

$$d(10010010, 00010001) = 3$$

2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.2. Phát hiện và sửa lỗi

a. Các mã sửa lỗi

Mã Hamming

- Tất cả các bit ở vị trí là các số mũ của 2 (*powers of two*) được dùng làm bit chẵn lẻ. (các vị trí như 1, 2, 4, 8, 16, 32, 64 v.v. hay nói cách khác $2^0, 2^1, 2^2, 2^3, 2^4, 2^5, 2^6$ v.v.)
 - Tất cả các vị trí bit khác được dùng cho dữ liệu sẽ được mã hóa. (các vị trí 3, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 17, etc.)
 - Mỗi bit chẵn lẻ tính giá trị chẵn lẻ cho một số bit trong từ mã (*code word*). Vị trí của bit chẵn lẻ quyết định chuỗi các bit mà nó luân phiên kiểm tra và bỏ qua (*skips*).
-

2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.2. Phát hiện và sửa lỗi

a. Các mã sửa lỗi

Mã Hamming

- ✓ Vị trí 1 ($n=1$): bỏ qua 0 bit($n-1$), kiểm 1 bit(n), bỏ qua 1 bit(n), kiểm 1 bit(n), bỏ qua 1 bit(n), v.v.
 - ✓ Vị trí 2 ($n=2$): bỏ qua 1 bit($n-1$), kiểm 2 bit(n), bỏ qua 2 bit(n), kiểm 2 bit(n), bỏ qua 2 bit(n), v.v.
 - ✓ Vị trí 4 ($n=4$): bỏ qua 3 bit($n-1$), kiểm 4 bit(n), bỏ qua 4 bit(n), kiểm 4 bit(n), bỏ qua 4 bit(n), v.v.
 - ✓ Vị trí 8 ($n=8$): bỏ qua 7 bit($n-1$), kiểm 8 bit(n), bỏ qua 8 bit(n), kiểm 8 bit(n), bỏ qua 8 bit(n), v.v.
 - ✓ Vị trí 16 ($n=16$): bỏ qua 15 bit($n-1$), kiểm 16 bit(n), bỏ qua 16 bit(n), kiểm 16 bit(n), bỏ qua 16 bit(n), v.v.
 - ✓ Vị trí 32 ($n=32$): bỏ qua 31 bit($n-1$), kiểm 32 bit(n), bỏ qua 32 bit(n), kiểm 32 bit(n), bỏ qua 32 bit(n), v.v.
 - ✓ và tiếp tục như trên.
-

Cách tính các bit chẵn lẻ trong mã Hamming (từ trái sang phải)

2.2.3. Tầng Liên kết dữ liệu (Data link)

2.2.3.2. Phát hiện và sửa lỗi

a. Các mã sửa lỗi

Mã Hamming – Ví dụ:

Nhóm dữ liệu mới - bao gồm các bit chẵn lẻ - bây giờ là: **10001100101**

Giả sử bit cuối cùng bị truyền lỗi, nhận được là: **10001100100**

Thứ tự bit	1	2	3	4	5	6	7	8	9	10	11		
Vị trí bit chẵn lẻ và các bit dữ liệu	p_1	p_2	d_1	p_3	d_2	d_3	d_4	p_4	d_5	d_6	d_7	Kiểm chẵn lẻ	Bit chẵn lẻ
Nhóm dữ liệu nhận được:	1	0	0	0	1	1	0	0	1	0	0	1	
p_1	1		0		1		0		1		0	Sai	1
p_2		0	0			1	0			0	0	Sai	1
p_3				0	1	1	0					Đúng	0
p_4								0	1	0	0	Sai	1

Kiểm tra các bit chẵn lẻ (bit bị đảo lộn có nền xám)

Xác định giá trị của các bit chẵn lẻ (nên nhớ bit nằm dưới cùng được viết về bên phải - viết ngược lại từ dưới lên trên). Giá trị số nguyên của các bit chẵn lẻ là $11_{(10)}$, và như vậy có nghĩa là bit thứ 11 trong nhóm dữ liệu (*data word*) - bao gồm cả các bit chẵn lẻ - là bit có giá trị không đúng, và bit này cần phải đổi ngược lại.

