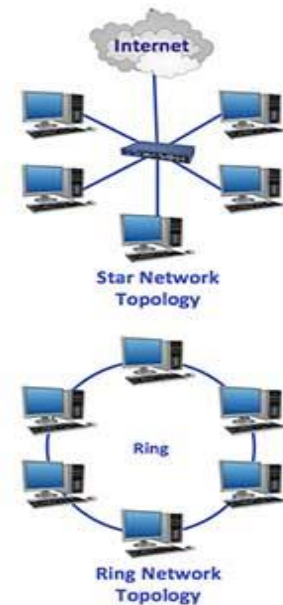
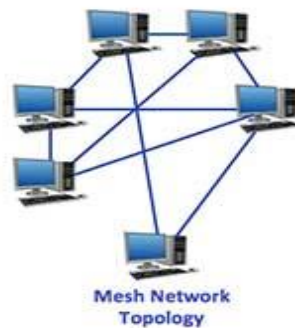


MẠNG MÁY TÍNH

CHƯƠNG 4. DATA LINK



NỘI DUNG

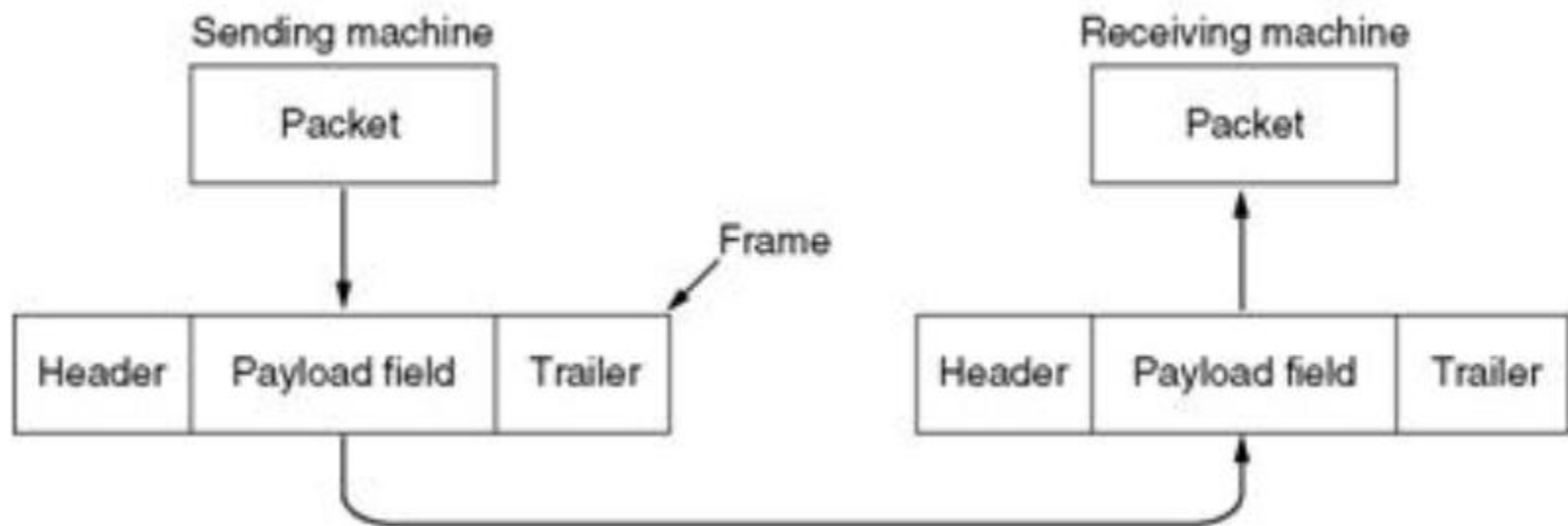
- Các dịch vụ cơ bản
- Điều khiển luồng
- Vấn đề xử lý lỗi
- Giao thức cửa sổ trượt

NỘI DUNG

- **Các dịch vụ cơ bản**
- Điều khiển luồng
- Vấn đề xử lý lỗi
- Giao thức cửa sổ trượt

Các dịch vụ cơ bản

- Tầng liên kết dữ liệu cung cấp các dịch vụ cho tầng mạng.
- Lấy các gói tin (Packet) nhận được từ tầng mạng và gói chúng vào trong các khung (frame) để truyền đi.



Các dịch vụ cơ bản

- Dịch vụ không nối kết không báo nhận (unacknowledged connectionless service), thường được sử dụng trong mạng LAN.
- Dịch vụ không nối kết có báo nhận (acknowledged connectionless service), thường dùng cho mạng không dây.
- Dịch vụ nối kết định hướng có báo nhận (acknowledged connection-oriented service), thường dùng trong mạng WANs.

NỘI DUNG

- Các dịch vụ cơ bản
- **Điều khiển luồng**
- Vấn đề xử lý lỗi
- Giao thức cửa sổ trượt

Điều khiển luồng

- Là kỹ thuật nhằm đảm bảo rằng bên phát không làm tràn dữ liệu bên nhận
- Hai phương pháp được sử dụng:
 - Phương pháp dừng và chờ (Stop and Wait)
 - Đơn giản nhất,
 - Kém hiệu quả, chỉ có một khung tin được truyền tại một thời điểm
 - Phương pháp cửa sổ trượt –(Sliding Window Flow Control)
 - Hiệu quả
 - Cho phép truyền nhiều khung tin cùng một lúc trên kênh truyền

Phương pháp dừng và chờ

- Truyền một gói tin và chờ báo nhận
 - Bên phát truyền một khung tin
 - Sau khi nhận được khung tin, bên nhận gửi lại xác nhận
 - Bên phát phải đợi đến khi nhận được xác nhận thì mới truyền khung tin tiếp theo
- Không hiệu quả
 - Bên nhận có thể dừng quá trình truyền bằng cách không gửi khung tin xác nhận
 - Tại một thời điểm chỉ có một khung tin trên đường truyền
→ chậm
 - Trường hợp độ rộng của kênh truyền lớn hơn độ rộng của khung tin thì nó tỏ ra cực kỳ kém hiệu quả.

Phương pháp cửa sổ trượt

- Cho phép nhiều khung tin được truyền tại một thời điểm -> Truyền thông hiệu quả hơn.
- A và B được kết nối trực tiếp song công (full-duplex).
- B có bộ đệm cho n khung tin -> B có thể chấp nhận n khung tin, A có thể truyền n khung tin mà không cần đợi xác nhận từ bên B
- Mỗi khung tin được gán nhãn bởi một số thứ tự.
- B xác nhận khung tin đã được nhận bằng cách gửi xác nhận cùng với số thứ tự của khung tin tiếp theo mà nó mong muốn nhận

NỘI DUNG

- Các dịch vụ cơ bản
- Điều khiển luồng
- **Vấn đề xử lý lỗi**
- Giao thức cửa sổ trượt

Phát hiện lỗi

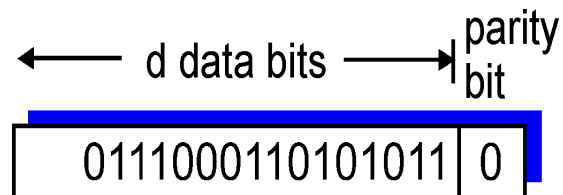
- Lý do một hay nhiều bit thay đổi trong khung tin được truyền:
 - Tín hiệu trên đường truyền bị suy yếu
 - Tốc độ truyền
 - Mất đồng bộ
- Việc phát hiện ra lỗi để khắc phục, yêu cầu phát lại là cần thiết và vô cùng quan trọng trong truyền dữ liệu.

Phát hiện lỗi Parity check

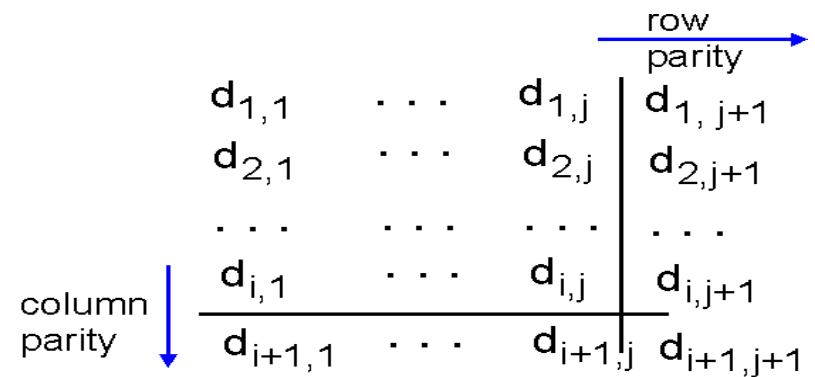
- Là kỹ thuật đơn giản nhất.
- Đưa một bit kiểm tra tính chẵn lẻ vào sau khối tin.
- Giá trị của bit này được xác định dựa trên số các số 1 là chẵn (even parity), hoặc số các số 1 là lẻ (odd parity).
- Lỗi sẽ không bị phát hiện nếu trong khung tin có 2 hoặc một số chẵn các bit bị đảo.
- Không hiệu quả khi xung nhiễu đủ mạnh.

Kiểm tra Parity

Bit Parity đơn:
phát hiện các lỗi bit



Bit Parity 2 chiều:
phát hiện & sửa các lỗi bit



1	0	1	0	1	1
1	1	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0

no errors

1	0	1	0	1	1
1	0	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0

parity error
parity error

*correctable
single bit error*

Phát hiện lỗi Cyclic redundancy Check (CRC) – kiểm tra phần dư tuần hoàn

Mô tả:

- Khối dữ liệu k bit
- Mẫu $n+1$ bit ($n < k$)
- Tạo ra dãy n bit gọi là dãy kiểm tra khung tin-FCS, Frame Check Sequence
- Tạo ra một khung tin $k+n$ bit
- Bên nhận khi nhận được khung tin sẽ chia cho mẫu, nếu kết quả là chia hết, việc truyền khung tin này là không có lỗi

CRC dưới dạng module của 2

M: Khối tin k bit

F: FCS n bit, n bit cuối của T

T: khung tin k+n bit

P: Mẫu n+1 bit, đây là một số chia được chọn trước.

Mục tiêu: xác định F để T chia hết cho P

$$T = 2^n M + F$$

Các bước tạo và kiểm tra CRC

- Các bước tạo CRC
 - Dịch trái M đi n bit
 - Chia kết quả cho P
 - Số dư tìm được là F
- Các bước kiểm tra CRC
 - Lấy khung nhận được $(n+k)$ bit
 - Chia cho P
 - Kiểm tra số dư, nếu số dư khác 0, khung bị lỗi, ngược lại là không lỗi



Q & A