

Bài giảng Truyền số liệu

Chương 5: Các nghi thức cơ sở

GV: Nguyen Tam Hien

9/2012





Nội dung

5.1

Kiểm soát lỗi

5.2

Idle ARQ

5.3

Continuous ARQ



Nội dung

5.1

Kiểm soát lỗi

5.2

Idle ARQ

5.3

Continuous ARQ



Kiểm soát lỗi

- Khi truyền dữ liệu từ phía phát tới phía thu thì thông thường phía thu sẽ kiểm tra các khung nhận được và trả về phía phát một thông điệp để xác nhận là đã nhận đúng hoặc là yêu cầu gửi một bản sao khác
- Loại kiểm soát lỗi như vậy gọi là ARQ (Automatic Repeat Request)
- ARQ có hai loại
 - Idle RQ
 - Continuous RQ (RQ liên tục)



Nội dung

5.1

Kiểm soát lỗi

5.2

Idle ARQ

5.3

Continuous ARQ



Idle RQ

- Phía gửi (nguồn) : Primary hay Sender
- Phía nhận (đích): Secondary hay Receiver
- Khung dữ liệu: I-frame
- Khung giám sát: ACK, NACK frame
- Có 2 cách thực hiện nguyên lý này:
 - Stop and wait ARQ hiểu ngầm
 - Stop and wait ARQ tường minh



Stop and Wait ARQ

- Việc truyền lại được hiểu ngầm: phía thu chỉ xác nhận khung truyền nào nhận đúng và nếu phía thu không xác nhận thì phía phát phải tự hiểu ngầm là có một khung truyền sai hay mất
- Phía phát (Primary- P) phát một frame tại một thời điểm
- Nếu phía thu (Secondary - S) nhận đúng thì trả về một ACK- frame



Stop and Wait ARQ

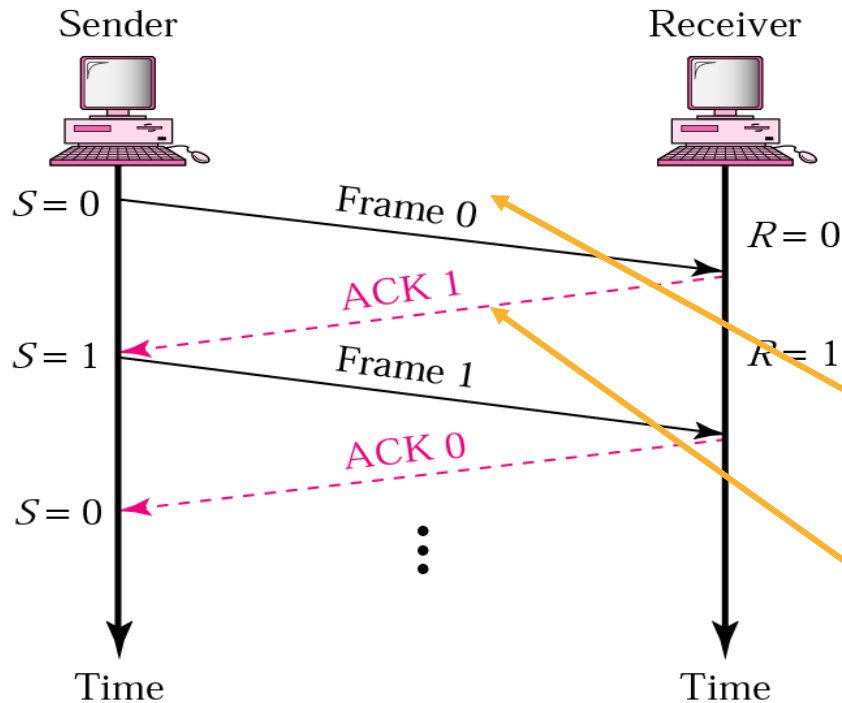
- Nếu P nhận đúng một ACK- frame thì sẽ phát một khung kế tiếp
- Khi P truyền một frame nó sẽ bắt đầu đếm thời gian và chờ
- Nếu bộ đếm thời gian kết thúc trước khi P nhận được ACK- frame thì P sẽ phát lại khung cũ
- Nếu một khung ACK bị lỗi hay mất thì S sẽ nhận một bản sao khác và S tự động loại bỏ



Stop and Wait ARQ

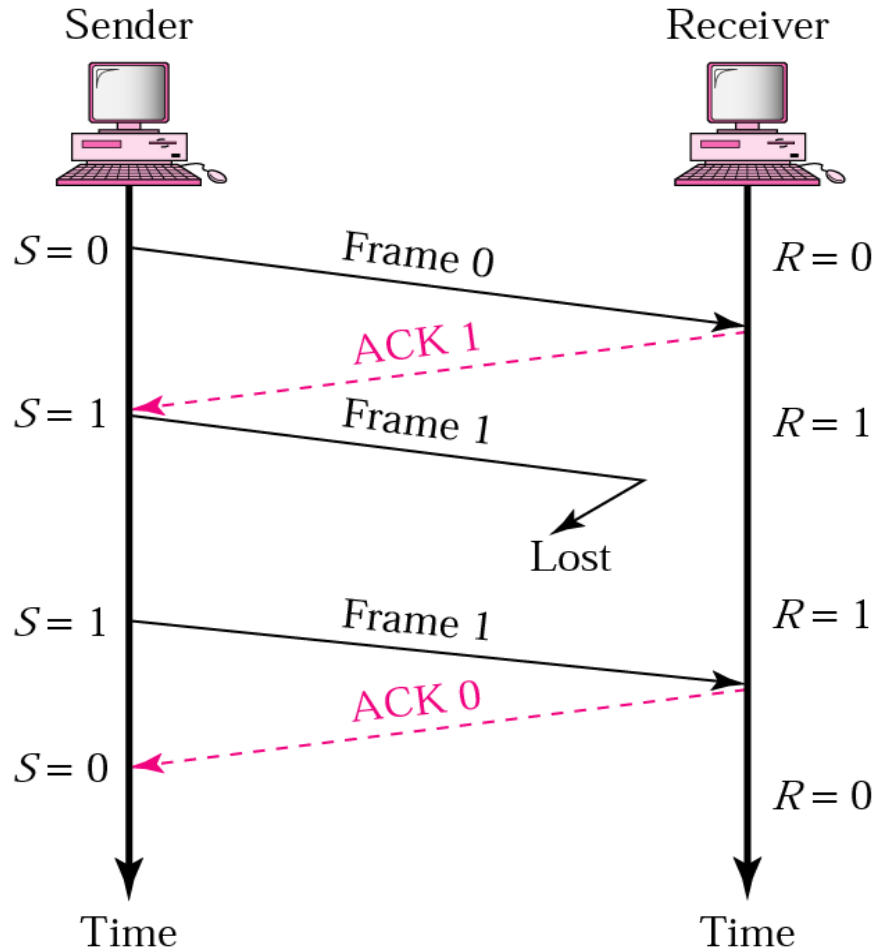
- Các trường hợp có thể xảy ra
 - Hoạt động bình thường
 - Mất khung dữ liệu (I-frame)
 - Mất khung ACK (ACK- frame)
 - Khung ACK tới trễ

Trường hợp bình thường

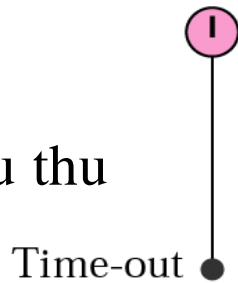


- Sender sẽ không gửi khung tiếp theo nếu không chắc chắn khung trước đó nhận được đúng.
- Số tuần tự cần thiết để kiểm tra khung nhận được là mới hay cũ.
- **ACK** – khi khung đúng và **NACK** - khi khung hỏng.

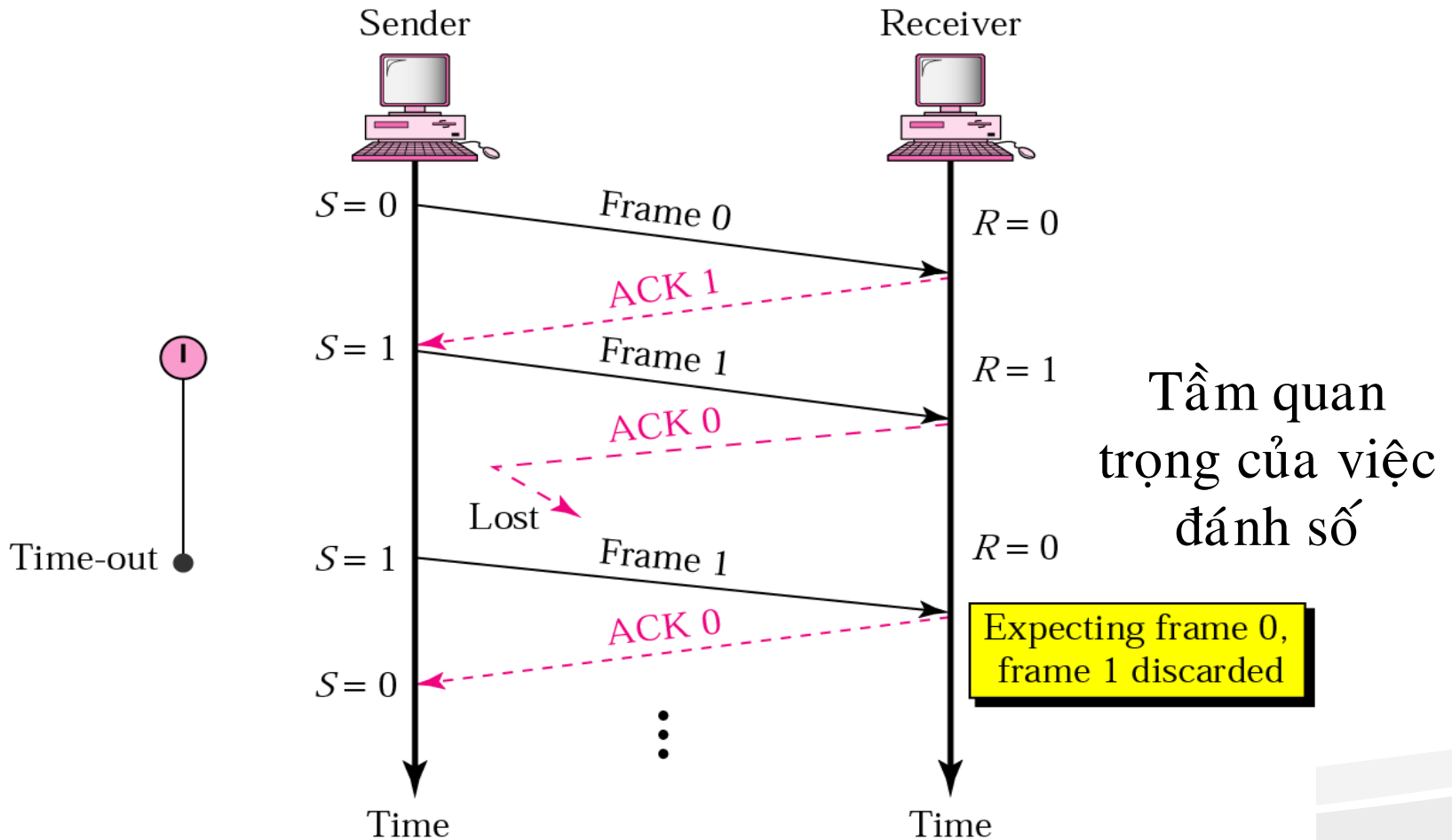
Mất khung dữ liệu hay khung dữ liệu hỏng



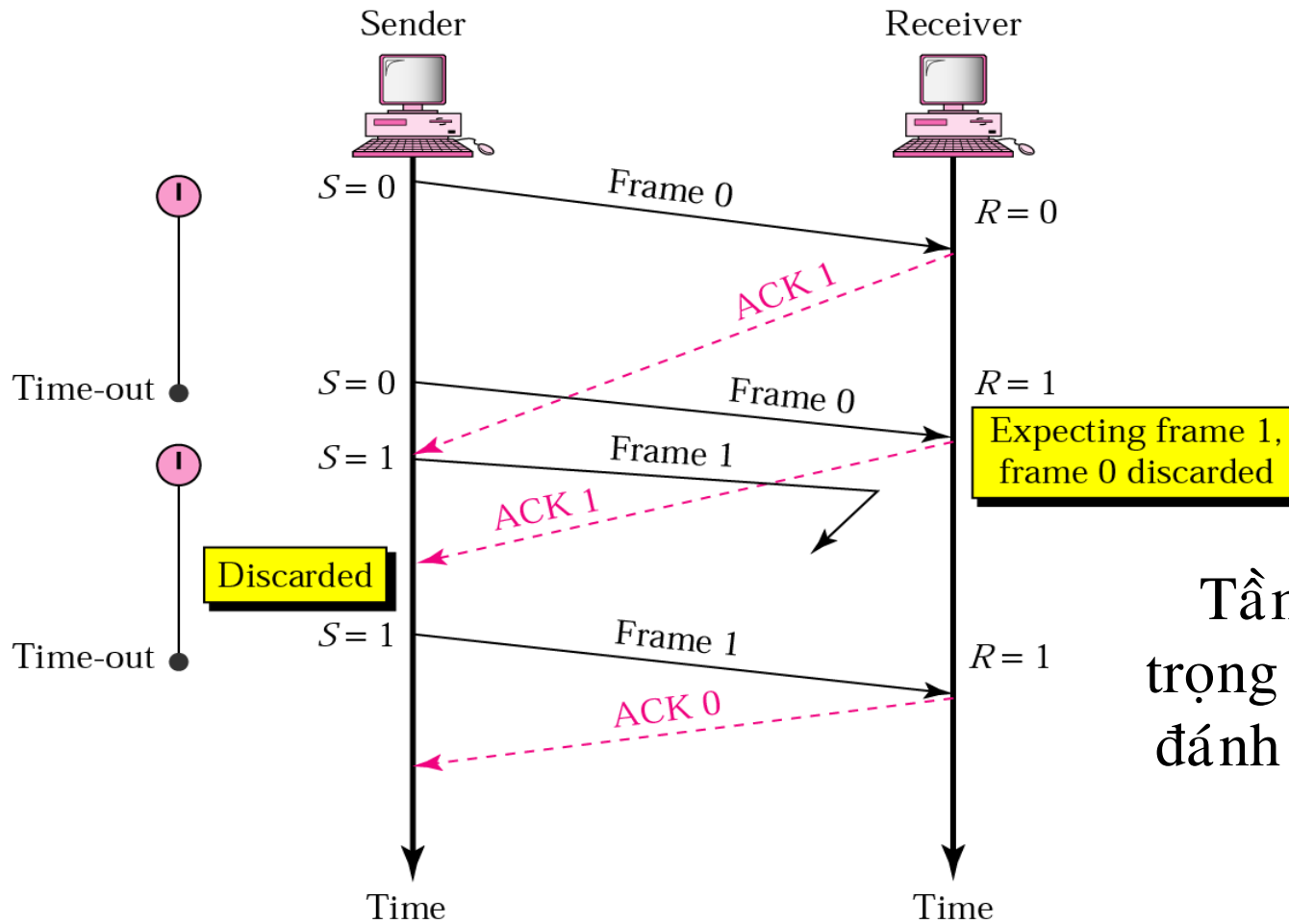
Trễ khứ hồi
Xử lý tại đầu thu



Mất ACK-frame



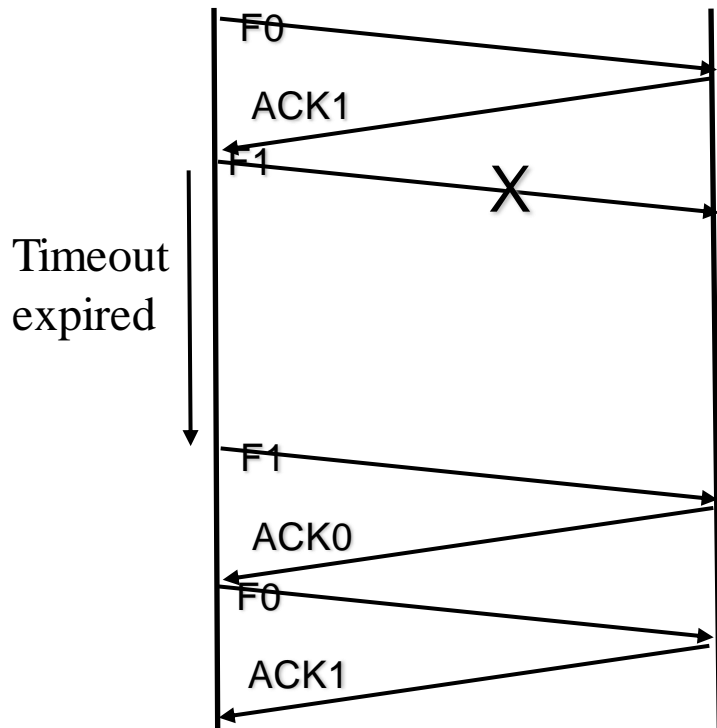
ACK-frame tới trễ



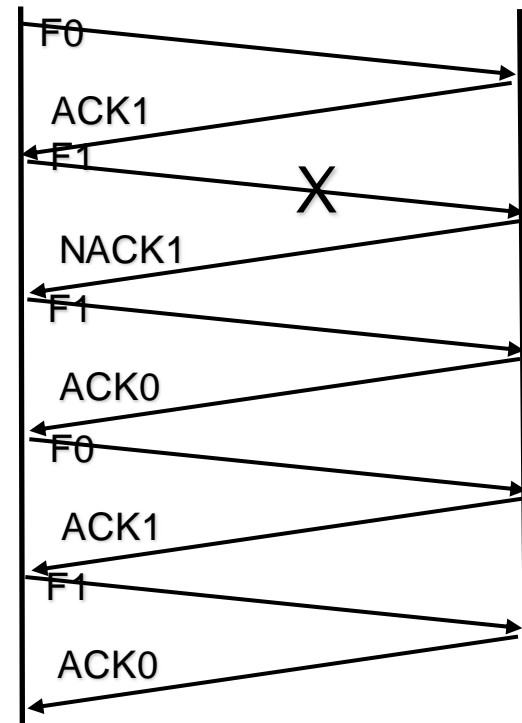
Tầm quan trọng của việc đánh số ACK

Stop and Wait ARQ hiểu ngầm và tường minh

minh



Stop and wait hiểu ngầm (không có NACK)



Stop and wait tường minh (có NACK)

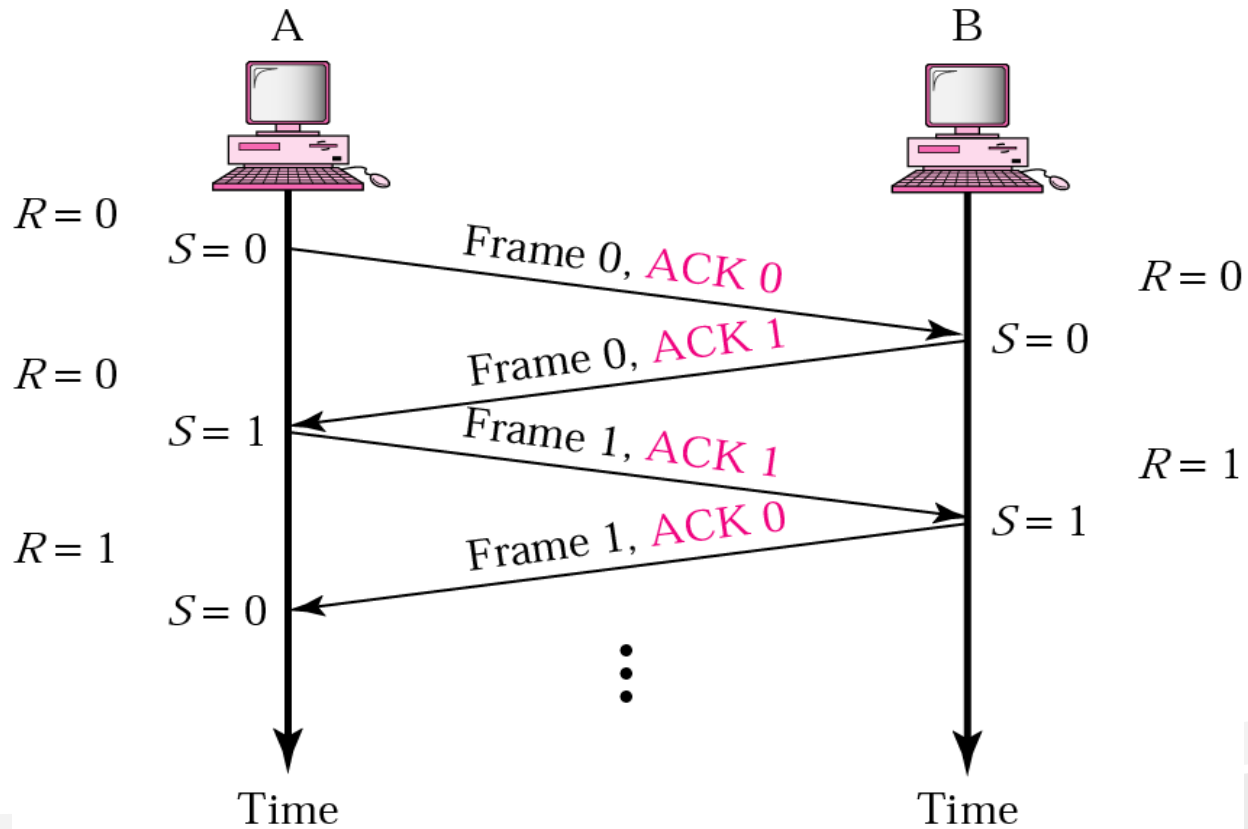
- Lỗi khung (Data bị sai)

- Sử dụng NACK để cải tiến hiệu suất. NACK được thu trước khi hết timeout.

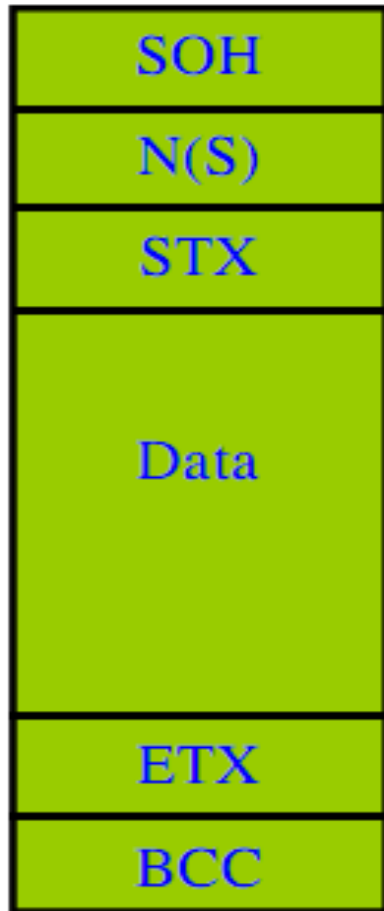
Duplex Stop-and-Wait

Kết hợp

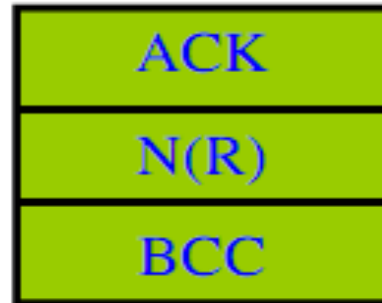
- Kết hợp data với ACK (giảm overhead & tiết kiệm BW)



Cấu trúc các loại khung



I - Frame



ACK - Frame



NACK - Frame



Hạn chế của Stop-and-Wait ARQ

- Sau mỗi một khung gửi đi, Host phải chờ 1 ACK
 - Không hiệu quả sử dụng bandwidth
- Để cải thiện hiệu quả, ACK nên được gửi sau một số khung, gọi là Continuous ARQ.



Nội dung

5.1

Kiểm soát lỗi

5.2

Idle ARQ

5.3

Continuous ARQ



Continuous ARQ

- P gửi liên tục các I-frame không cần chờ ACK-frame
- P duy trì một bản sao các I-frame trong một danh mục truyền lại hoạt động theo nguyên tắc FIFO
- S trả về một ACK-frame mỗi khi nhận được một khung đúng
- Mỗi I-frame chứa một định danh duy nhất sẽ được trả về trong các ACK tương ứng
- S duy trì một danh mục theo thứ tự tức danh sách thu gồm n khung thu tốt sau cùng
- P sẽ loại bỏ các I-frame sau khi nhận được các ACK-frame tương ứng

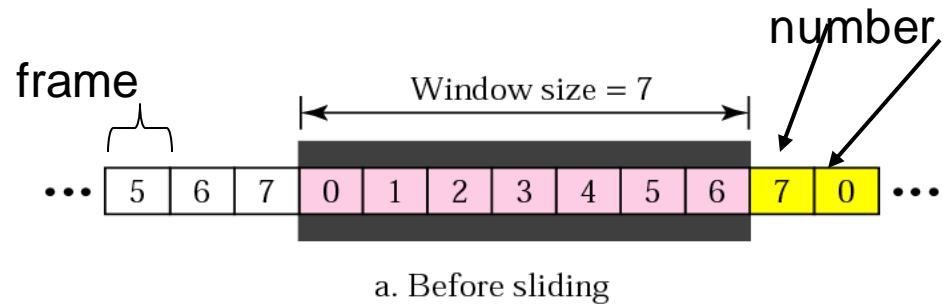


Continuous ARQ

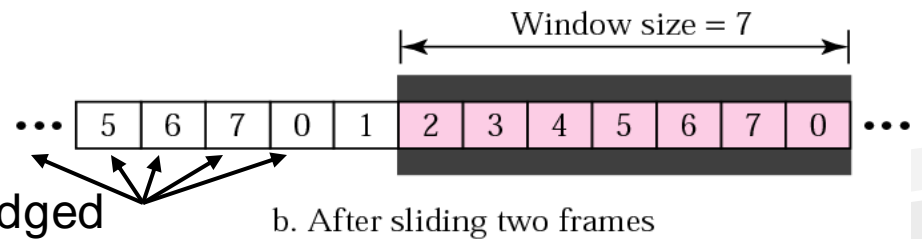
- 2 loại Continuous ARQ (Sliding Window protocols):
 1. Go-back- N ARQ
 2. Selective Repeat ARQ
- Loại Go-back- N , S phát hiện việc nhận không đúng thứ tự của P và yêu cầu P truyền lại tất cả các khung từ khung cuối cùng thu tốt cho đến khi khôi phục lại được khung truyền đúng thứ tự bị mất
- Loại Selective Repeat ARQ, S phát hiện và yêu cầu truyền lại chỉ những khung nào bị sai

Sliding Window Protocols

- Số tuần tự
 - Các khung phát được đánh số tuần tự
 - Số tuần tự của khung được lưu ở header của khung
 - Nếu số bit trong header là m thì số tuần tự đếm từ 0 đến $2^m - 1$

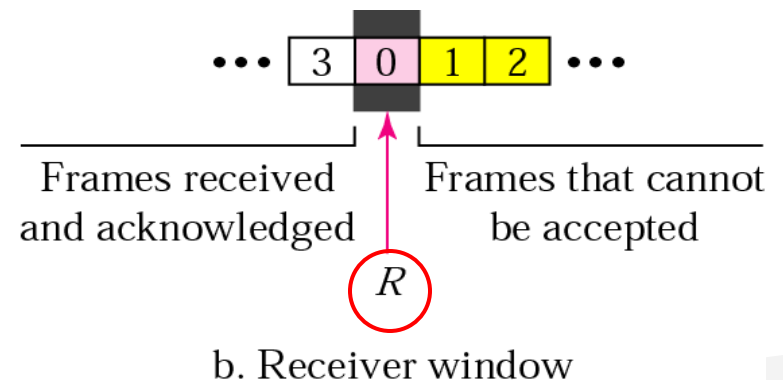
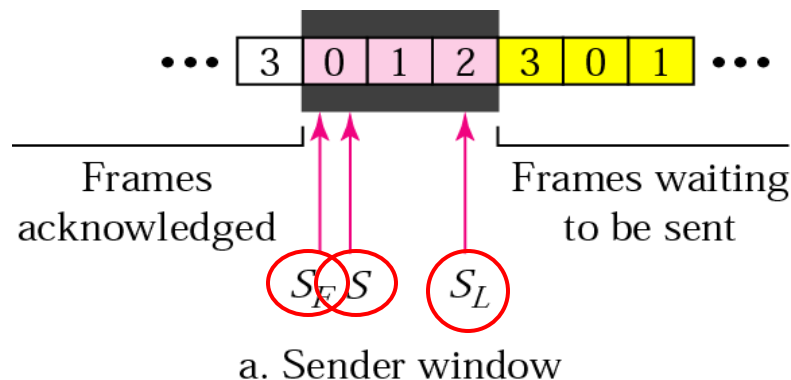


- Cửa sổ trượt
 - Để giữ các khung chưa báo phát
 - Kích thước cửa sổ nhận acknowledged frames luôn là 1



Go-back- N

- S - chỉ số tuần tự của khung đang được phát
- S_F - chỉ số tuần tự của khung đầu tiên trong cửa sổ
- S_L - chỉ số tuần tự của khung cuối cùng trong cửa sổ
- R - chỉ số tuần tự của khung đang chờ thu



Tại sao có tên Go-back- N ?

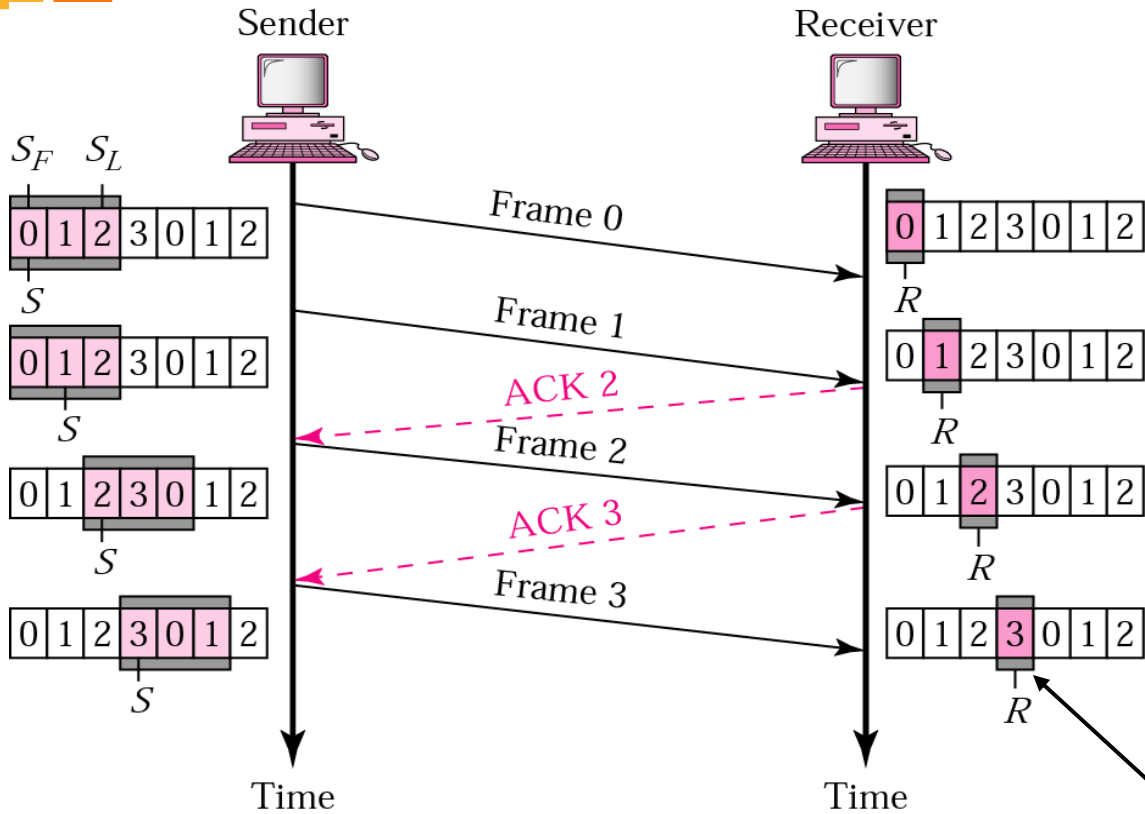
- Phát lại khung
 - Khi một khung bị hỏng, đầu phát sẽ **quay lại** và phát lại một tập hợp các khung tính từ khung không có báo phát (ACK)
 - Số lượng khung được phát lại là **N**

Ví dụ:

Kích thước cửa là 4.

Đầu phát vừa phát khung 6 và hết thời gian đếm khung 3 (khung 3 không có ACK). Đầu phát sẽ phát lại các khung 3, 4, 5, 6.

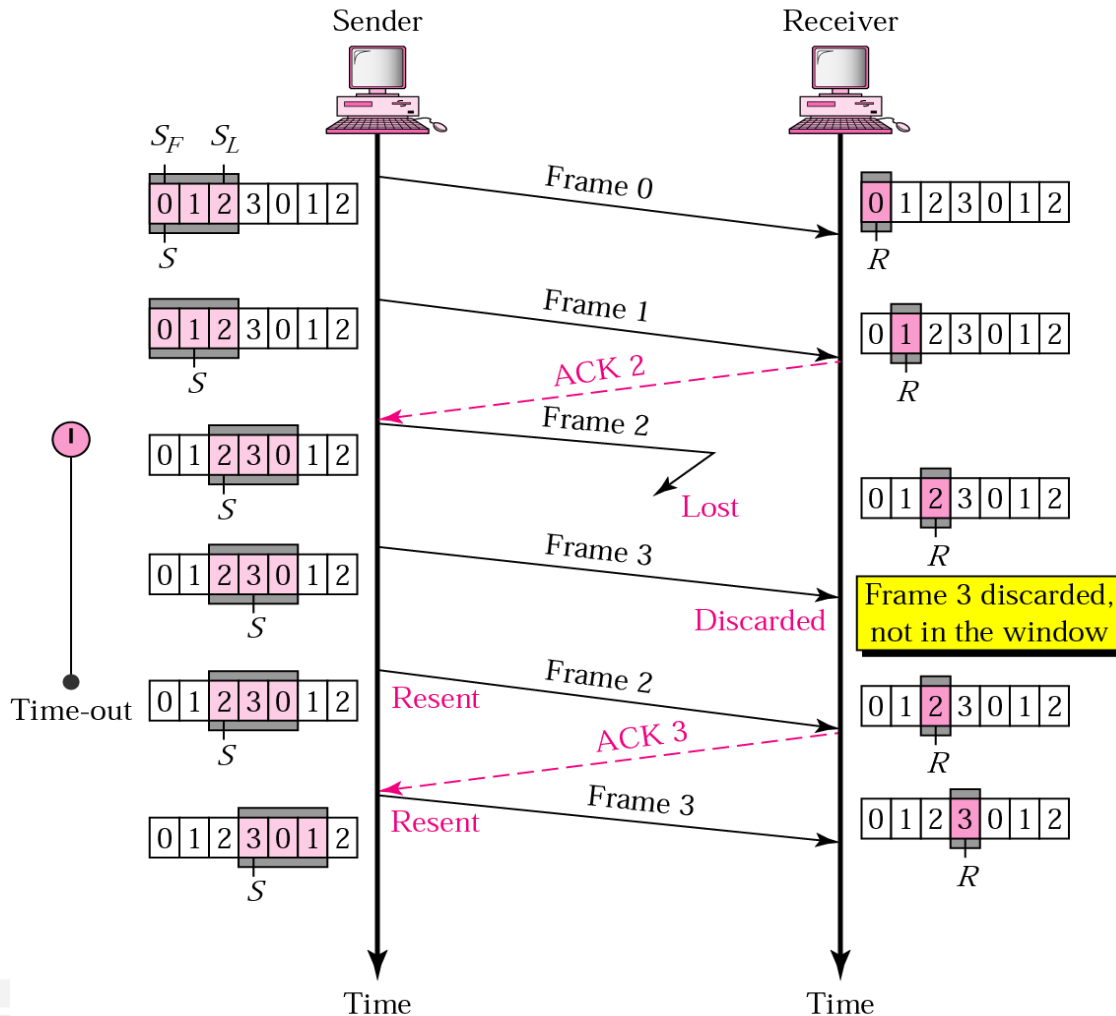
Go-back-N – trường hợp bình thường-



- Có bao nhiêu khung có thể phát mà không chờ ACK?
- ACK1 – không cần thiết nếu ACK2 được phát đi.

expected sequence number

Go-back-N - mất khung hoặc lỗi khung-

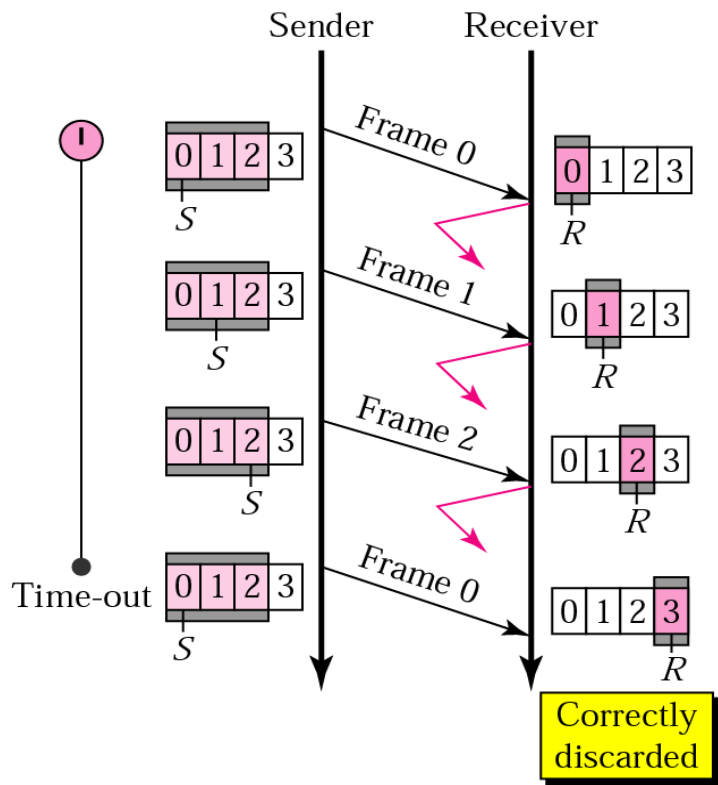


Các khung hỏng bị loại bỏ!

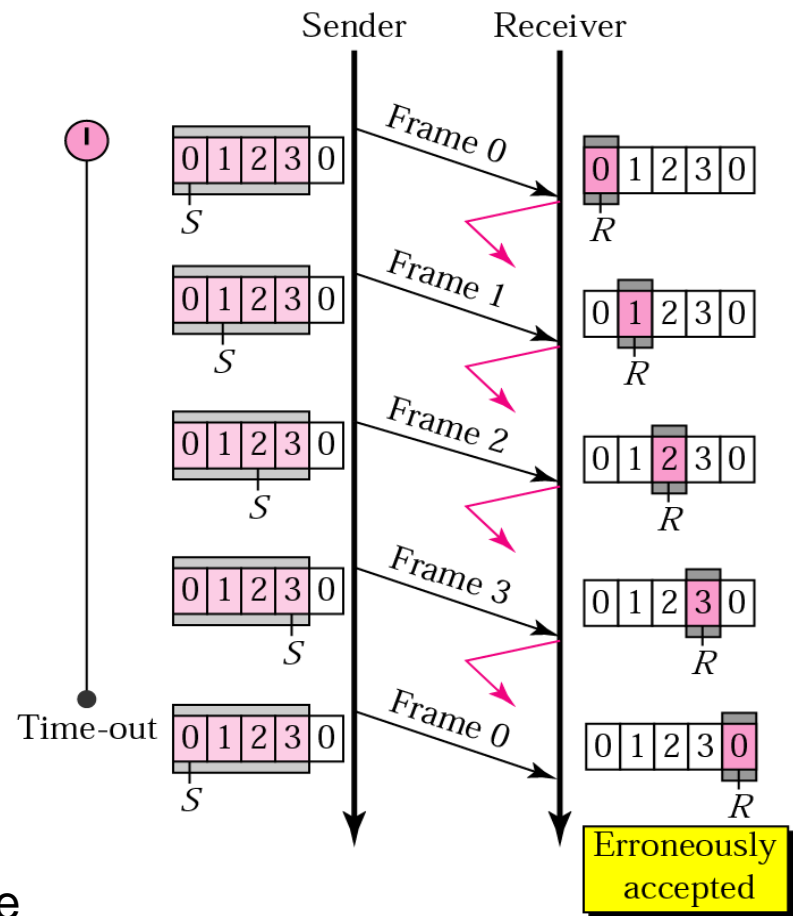
Tại sao các khung vẫn được thu đúng thứ tự tuy không có bộ đệm?

Nhược điểm của phương pháp này là gì?

Go-back-N – kích thước cửa sổ gửi-



a. Window size $< 2^m$ ← sequence number



b. Window size $= 2^m$

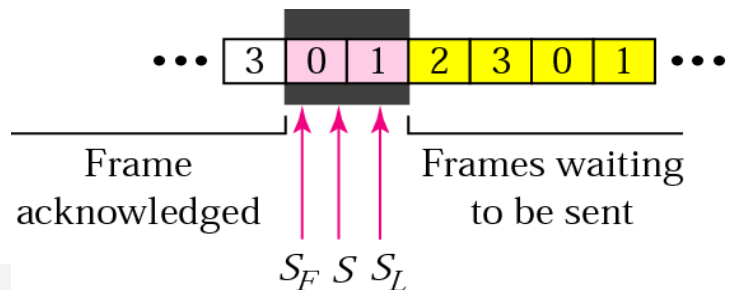


Nhược điểm của Go-back- N

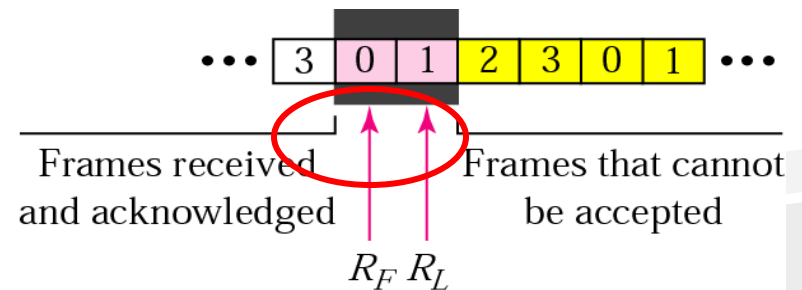
- Không hiệu quả
 - Tất cả các khung không đúng thứ tự đều phải phát lại
- Nếu liên kết có tạp âm sẽ gây ra vấn đề:
 - Nhiều khung phải phát lại -> tốn bandwidth
- Giải pháp
 - Chỉ phát lại những khung hỏng
- Selective Repeat ARQ
 - Chống lại việc phát lại không cần thiết

Selective Repeat ARQ

- Xử lý tại đầu thu phức tạp hơn
- Kích thước của sổ giảm xuống $\leq 2^{m-1}$
- Đầu phát và đầu thu có kích thước của sổ như nhau
- Đầu thu chờ thu một tập hợp khung trong một phạm vi của số tuần tự

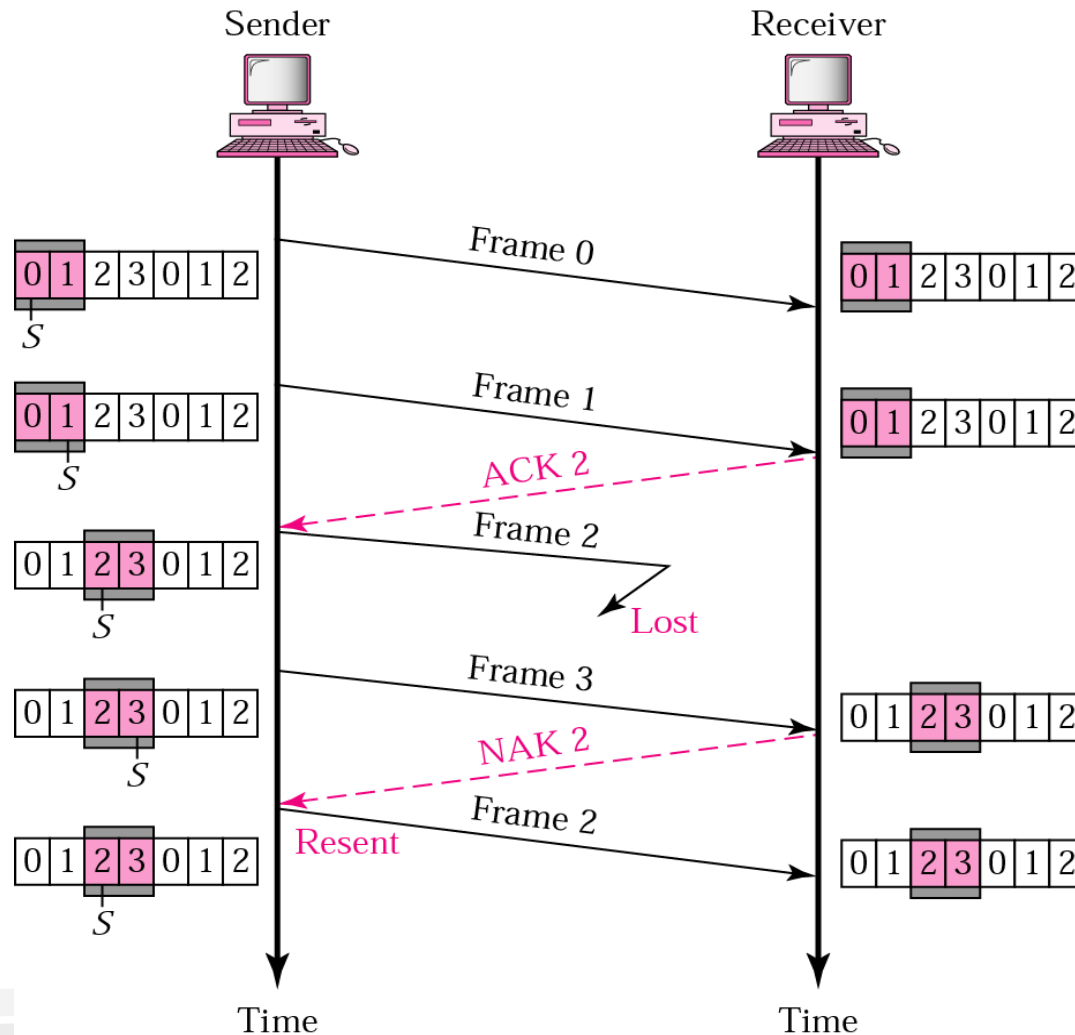


a. Sender window

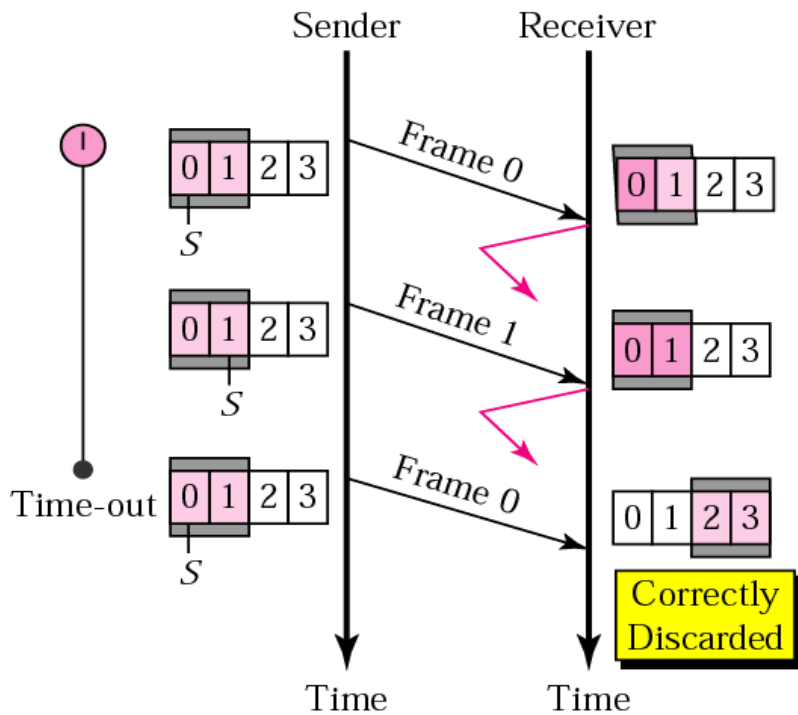


b. Receiver window

Selective Repeat ARQ – mất khung-



Selective Repeat ARQ-kích thước cửa sổ gửi-



a. Window size = 2^{m-1}