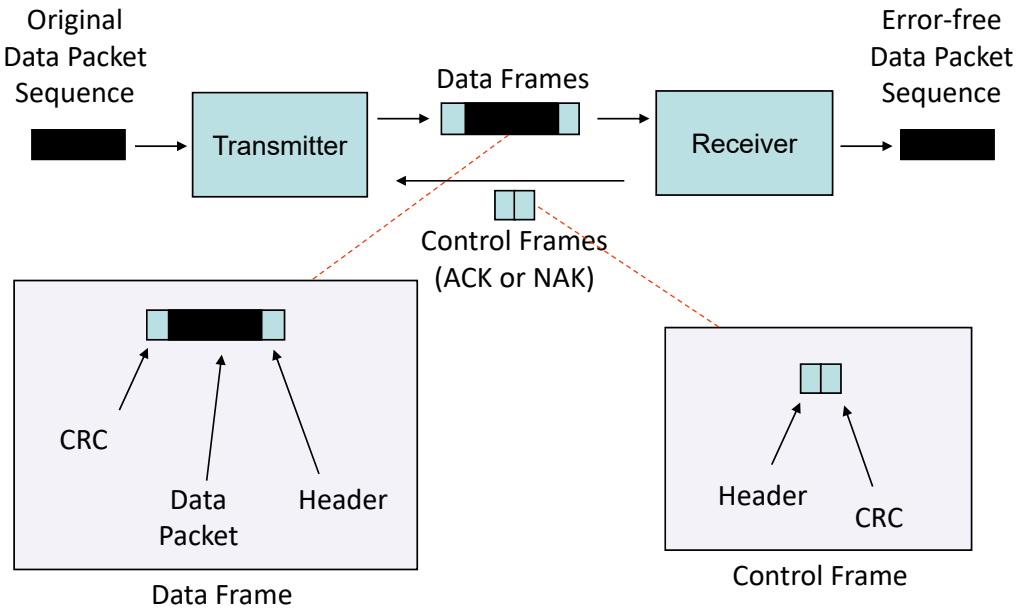
Automatic Repeat reQuest (ARQ)



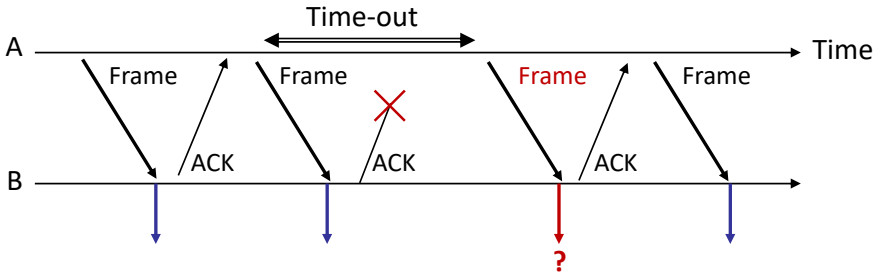
Design Goals of ARQ Protocols

- Three Design Goals of ARQ Protocols
  - **Goal #1:** to ensure that each packet is delivered error-free
  - **Goal #2:** to ensure that each packet is delivered exactly once without duplication
  - **Goal #3:** to ensure that packets are delivered in order

1. Stop-and-Wait ARQ (S&W)

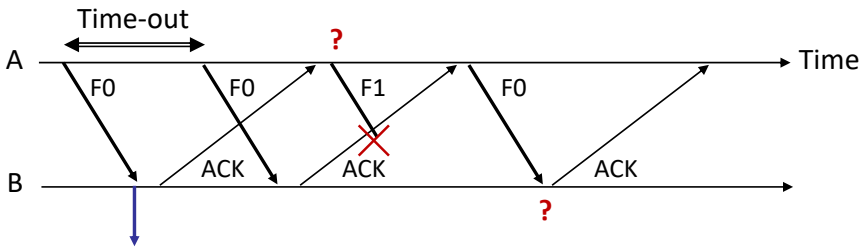
- The transmitter and receiver work on the delivery of one frame at a time through alternation of actions
  - *Essential Components:* ACK, timeout, sequence numbering
    - ACK acknowledges reception of a frame
    - Needs 1-bit sequence numbering to remove ambiguities

Problematic Scenario without Sequence Numbering



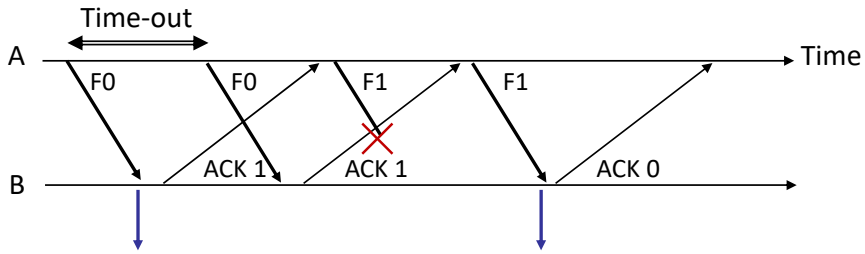
- Problem: B receives the 2nd frame twice
- Solution: Add frame sequence number (S_{recent}) in the frame header

Problematic Scenario without Sequence Numbering



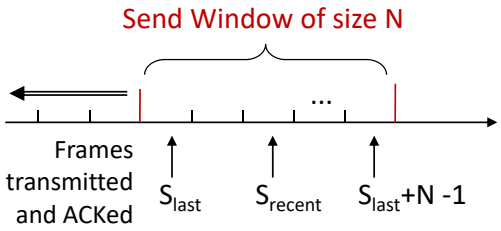
- Problem: A assumes that F1 was delivered error-free
  B discards the new F0 by mistake
- Solution: Add frame sequence number (R_{next}) in the ACK header
  - R_{next} is the sequence number of next frame expected by the receiver
  - It acknowledges reception of all prior frames implicitly

Correct Operation with Sequence Numbering

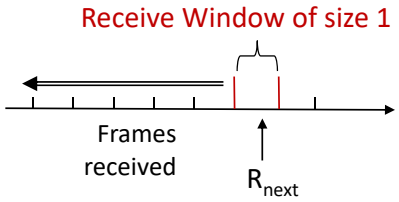


2. Go-Back-N ARQ (GBN)

- GBN improves S&W by keeping the channel busy when the transmitter waits for acknowledgment from the receiver
  - At the transmitter side: Allow up to N outstanding frames



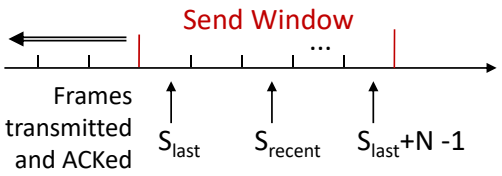
  - At the receiver side: Receive window of size 1



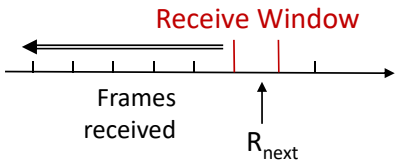
GBN Protocol

- *Essential Components:* ACK, timeout, sequence numbering
  - ACK acknowledges reception of all prior frames implicitly
- Upon timeout:
  - Frame in error and all subsequent frames are retransmitted
- Needs m-bit sequence numbering to remove ambiguities
  - What is the minimum value for m?

GBN Transmitter & Receiver



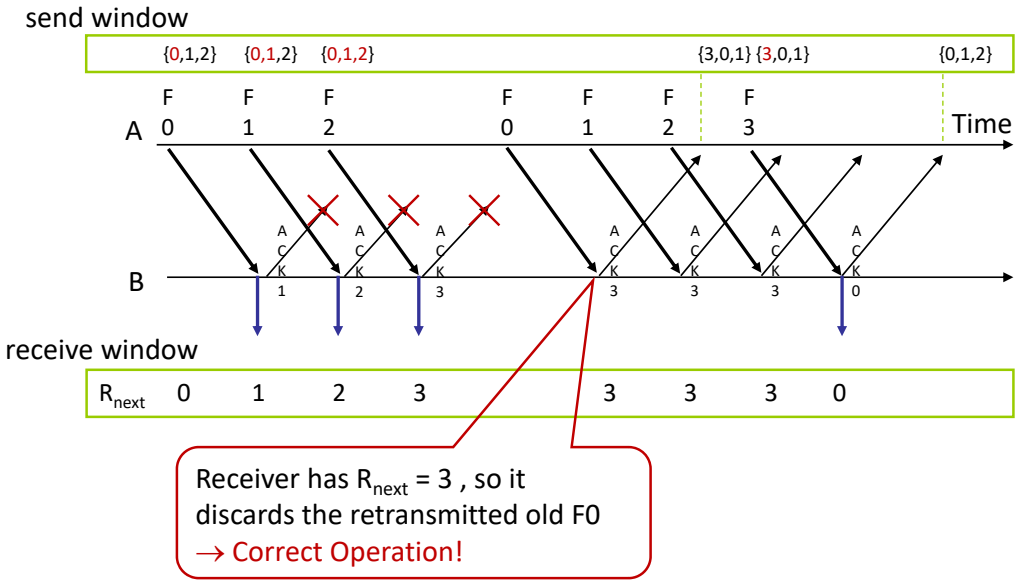
1. Transmitter waits for error-free ACK with: $R_{next} \in [S_{last}, S_{recent}+1]$
2. When such ACK arrives, send window slides forward: $S_{last} = R_{next}$
3. When timer expires for S_{last}, transmitter go-back-N to retransmit S_{last} and all subsequent frames



1. Receiver only accepts error-free frame with sequence number R_{next}
2. When such frame arrives, R_{next} is incremented by one, meaning that receive window slides forward by 1: $R_{next} = R_{next} + 1$
3. Erroneous frames and error-free frames with sequence number $\neq R_{next}$ are discarded
4. ACK is sent for each error-free frame received

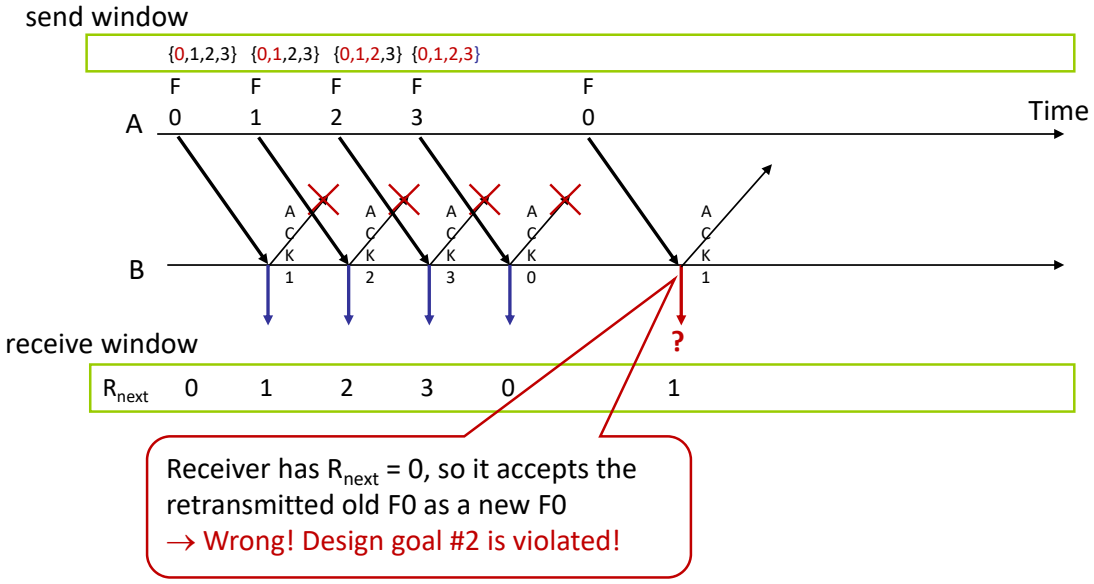
$N + 1 \leq 2^m$

- Example: 2-bit (m = 2) sequence numbering suffices for Go-Back-3 (N = 3) ARQ



$N + 1 \leq 2^m$

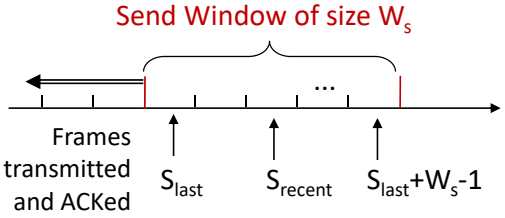
- Example: 2-bit (m = 2) sequence numbering is inadequate for Go-Back-4 (N = 4) ARQ



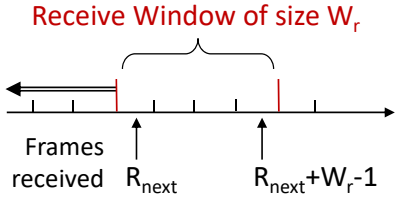
Receiver has R_{next} = 0, so it accepts the retransmitted old F0 as a new F0
→ Wrong! Design goal #2 is violated!

3. Selective Repeat ARQ (SR)

- SR improves upon GBN by buffering at the receiver side
  - Allow up to W_s outstanding frames



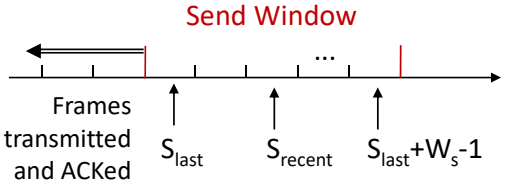
  - Allow a receive window of size W_r (>1)
    - Receiver buffers the error-free frames with sequence number $\in [R_{next}+1, R_{next}+W_r-1]$



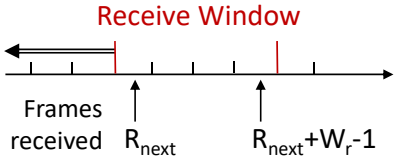
SR Protocol

- *Essential Components:* ACK, NAK, timeout, sequence numbering
  - NAK is sent when an error-free out-of-sequence frame is received
  - ACK is sent for all other error-free frames
  - Both ACK and NAK acknowledge reception of all prior frames
- Frame in error is retransmitted upon
  - Timeout or reception of NAK
  - Only the frame in error is retransmitted
- Needs m-bit sequence numbering to remove ambiguities
  - What is the minimum value for m?

SR Transmitter & Receiver



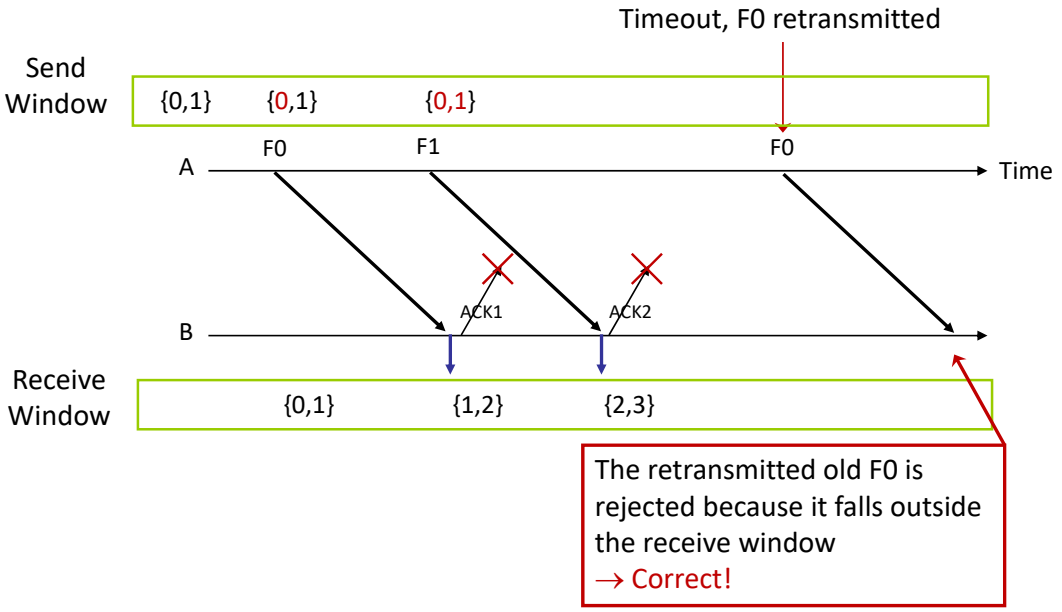
1. If an error-free ACK or NAK with $R_{next} \in [S_{last}, S_{recent}+1]$ arrives, send window slides forward: $S_{last} = R_{next}$
2. When timer for a frame expires or when a NAK arrives, transmitter retransmits the corresponding frame only



1. Receiver only accepts error-free frames with sequence number $\in [R_{next}, R_{next}+W_r-1]$
2. When frame with sequence number R_{next} arrives, R_{next} is incremented to a proper value $\Rightarrow$ receive window may slide forward by more than one
3. Erroneous frames and error-free frames with sequence number $\notin [R_{next}, R_{next}+W_r-1]$ are discarded
4. NAK is sent when an error-free out-of-sequence frame is received
5. ACK is sent for all other error-free frames received

$W_s + W_r \leq 2^m$

- Example: 2-bit (m = 2) sequence numbering, $W_s = W_r = 2$



$W_s + W_r \leq 2^m$

- Example: Problem when $W_s+W_r > 2^m$
  2-bit (m = 2) sequence numbering, $W_s = W_r = 3$