



Technische  
Universität  
Braunschweig



Institut für Betriebssysteme  
und Rechnerverbund  
Connected and Mobile Systems



# Computernetze 1

## Übung 5

### Data Link Layer – Flusskontrolle und Sendewiederholungen

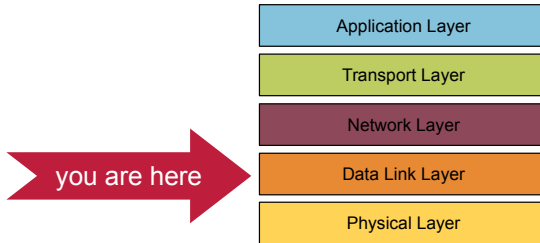
Fynn Schulze, 22. Mai 2025

Technische Universität Braunschweig, IBR

# Überblick

- 1) Kommunikationsarten
- 2) Sendewiederholungen
- 3) Beispiel Sende- und Empfangsfenster beim Sliding Window
- 4) Sliding Window
- 5) Sequenznummernraum beim Sliding-Window-Verfahren
- 6) Kanalauslastung/Flusskontrollprotokolle

# Überblick

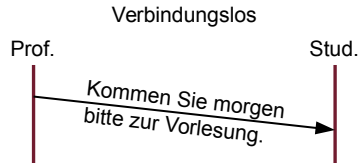
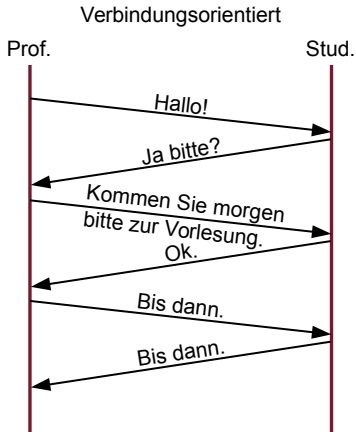


# Aufgabe 1: Kommunikationsarten

Geben Sie Beispiele für die folgenden Begriffe bzw. Verfahren, wenn alle „Netzkomponenten“ Menschen sind.

- a) Verbindungsorientierte Kommunikation
- b) Verbindungslose Kommunikation
- c) Protokoll
- d) Flusskontrolle
- e) Staukontrolle

# 1 a/b) Verbindungsorientiert/Verbindungslos

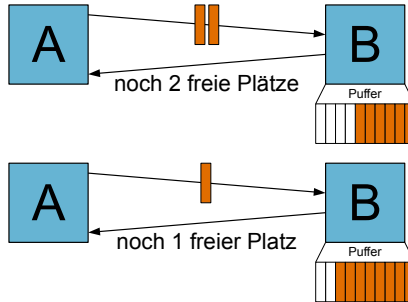


# 1 c) Protokoll

- Gemeinsame Sprache / Formulierungen
  - Buchstaben
  - Wörter
  - Sätze (Regeln für den Satzbau!)
  - Auf eine Frage folgt eine Antwort.
  - Semantik

# 1 d) Flusskontrolle

- Wenn der Zuhörer in einem Gespräch „Nicht so schnell!“ sagt



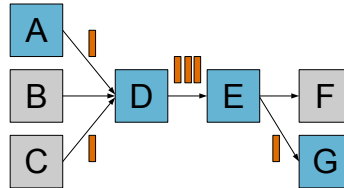
## Überlastung des Empfängers

Grundsätzlich entsteht ein Problem, wenn **A** schneller Daten sendet als **B** empfangen kann!

# 1 e) Staukontrolle

## Szenario

Zwei Personen können nicht direkt miteinander sprechen (z.B. zu große Entfernung) und bedienen sich einer dritten Person. Staukontrolle in dieser Situation ist, wenn der Übermittler „Nicht so schnell!“ sagt.



## Problem bei Überlast

Frames gehen verloren → Re-Transmissions nehmen zu → noch mehr Last ⇒ Überlast vermeiden

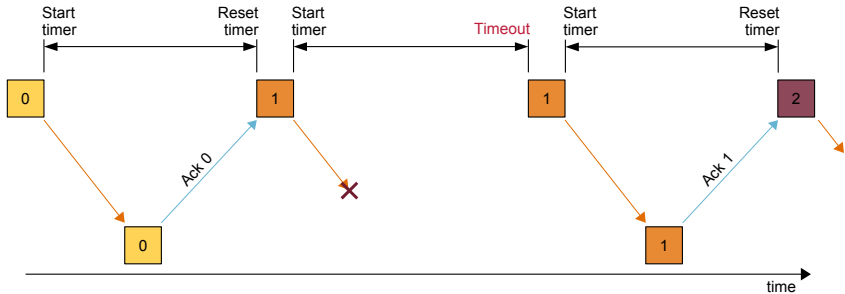


## Aufgabe 2: Sendewiederholungen

Beschreiben Sie die wesentlichen Eigenschaften und Unterschiede der folgenden Verfahren bezüglich ihrer Reaktion auf fehlerhafte Übertragungen und daraus folgenden Sendewiederholungen. Skizzieren Sie jeweils eine beispielhafte Übertragung zwischen zwei Stationen.

- Stop-and-Wait
- Go-Back-N (Frames einzeln bestätigt)
- Go-Back-N (Frames kumulativ bestätigt)
- Selective Repeat

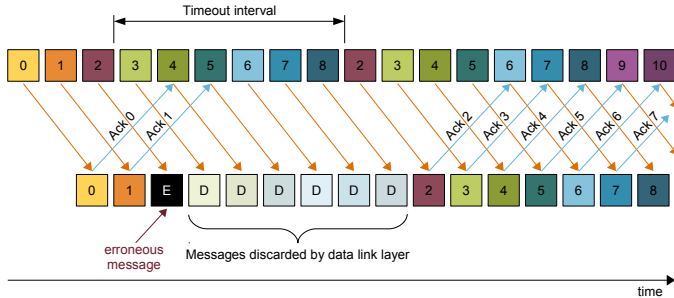
## 2) Stop-and-Wait



### Stop-and-Wait

- Jeder Frame wird einzeln bestätigt
- Nächster Frame erst nach ACK des vorherigen
- Geringer Durchsatz

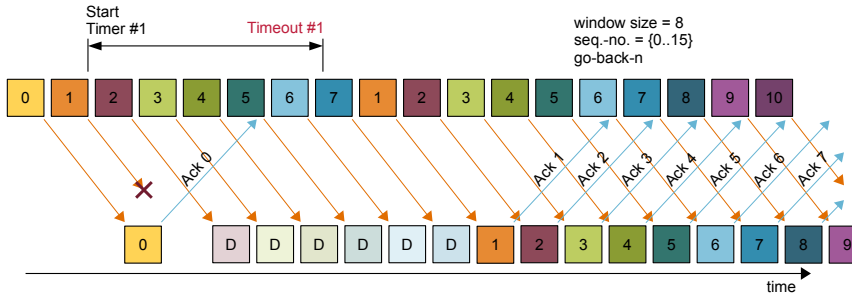
# Sliding-Window-Verfahren



## Erklärung

- Gruppe von Verfahren: Go-Back-N, Selective Repeat
- Mechanismus zur Datenflusskontrolle
- Klassisches Stop-and-Wait ist eher ineffizient
- Sender kann bestimmte Menge von Daten unbestätigt senden

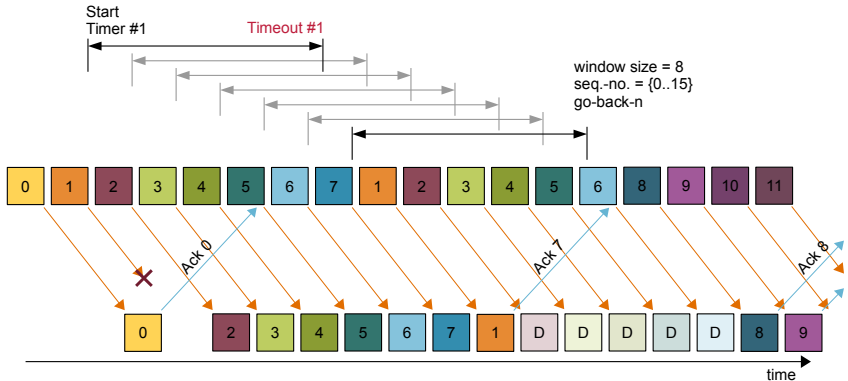
## 2) Go-Back-N



### Go-Back-N

- Jeder Frame wird einzeln bestätigt
- Es können mehrere Frames unbestätigt gesendet werden
- Out-of-order Frames werden verworfen

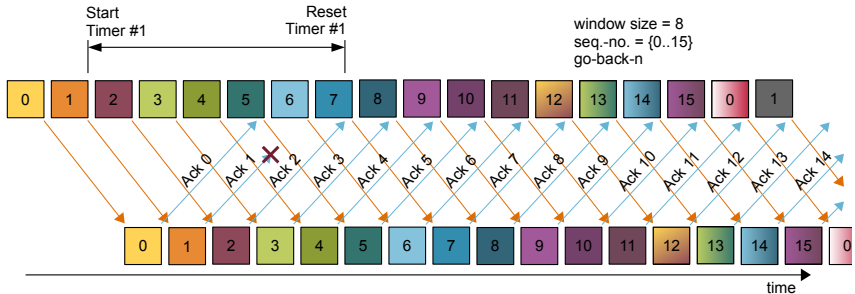
## 2) Go-Back-N (verbessert) Fall 1



### Go-Back-N (verbessert)

- Frames werden kumulativ bestätigt
- Out-of-order Frames werden für später zwischengespeichert

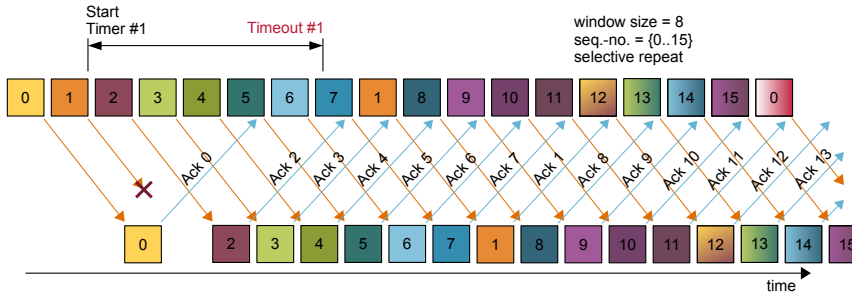
## 2) Go-Back-N (verbessert) Fall 2



### Go-Back-N (verbessert)

- Kumulative ACKs können vorhergehende ACKs ersetzen
- Neuübertragung wird verhindert

## 2) Selective Repeat



### Selective Repeat

- Frames werden einzeln bestätigt
- Es können mehrere Frames unbestätigt gesendet werden
- Sender muss für jeden Frame einzeln prüfen, ob er angekommen ist

## Aufgabe 3: Sliding Window

Gegeben sei ein Szenario mit zwei Stationen, bei der eine Station Daten zu einer anderen senden möchte. Der Datenaustausch geschieht mittels Sliding-Window-Technik zur Flusskontrolle. Dabei sollen maximal 4 Frames unbestätigt gesendet werden können. Die Sequenznummern werden modulo 8 berechnet.

Tragen Sie in der Skizze auf der nächsten Seite die jeweilige Ober- und Untergrenze des Sendefensters (links) und des Empfangsfensters (rechts) in die Uhren ein. Zeichnen Sie zusätzlich die Bestätigungen des Empfängers in die Skizze ein. Der Empfänger bestätigt hierbei alle Frames positiv und nicht kumulativ (d.h. jeden Frame einzeln, Go-Back-N). Gehen Sie weiterhin davon aus, dass auf Empfängerseite ein Puffer für maximal 4 Frames vorhanden ist.

*Hinweis: Der Frame mit der Sequenznummer 6 geht verloren, sodass es später auf Senderseite zu einem Timeout kommt!*

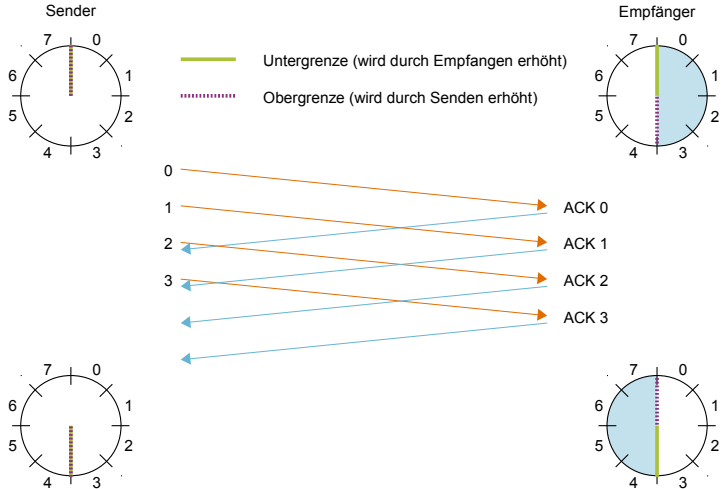


## Aufgabe 3: Erklärung

### Erklärung

- Sequenznummernraum = 8 (0, 1, 2, 3, 4, 5, 6, 7)
- Fenstergröße = 4
- Datenframes werden positiv bestätigt  $\Rightarrow$  ACKs, wenn richtig empfangen
- nicht kumulativ  $\Rightarrow$  jeder Frame wird einzeln bestätigt
- Uhren markieren die gerade verwendeten Sequenznummern

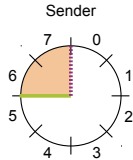
# Aufgabe 3: Schritt 1



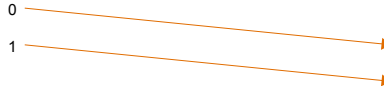
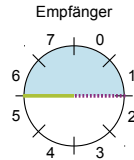
22. Mai 2025 | Fynn Schulze | Computernetze 1 | Seite 21



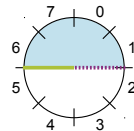
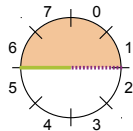
# Aufgabe 3: Schritt 3



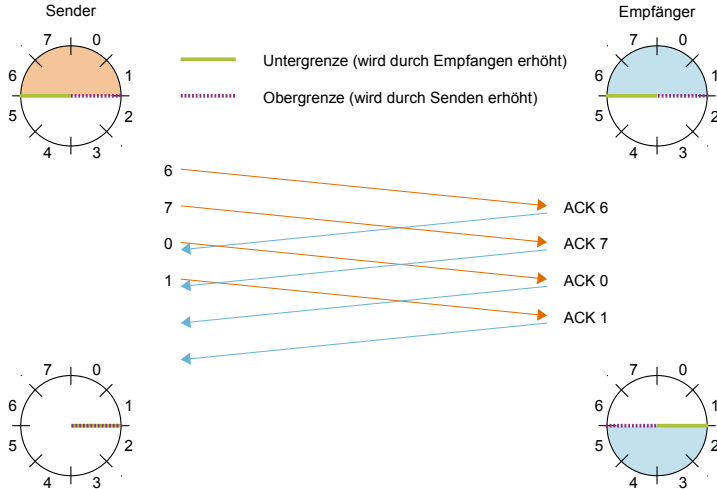
— Untergrenze (wird durch Empfangen erhöht)  
 ..... Obergrenze (wird durch Senden erhöht)



Timeout



# Aufgabe 3: Bonus – möglicher Schritt 4



## Aufgabe 4: Sliding Window

Gegeben sei ein Szenario mit *zwei Stationen*, bei der eine Station Daten zu einer anderen senden möchte. Der Datenaustausch erfolgt mittels *Sliding-Window-Technik zur Flusskontrolle*. Ein Frame besteht aus 100 Byte. Die *Fenstergröße* ist auf 300 Byte begrenzt. Die *Sequenznummern* werden modulo 8 berechnet. Zur Behebung von Frame-Verlusten wird die Go-Back-N Strategie angewendet, wobei alle Frames einzeln bestätigt werden.

Zeigen Sie den Ablauf mit mindestens 6 Frames bei dem es bei der Übertragung des 5. Frames (Seq=4) zu einem Frame-Verlust kommt.

*Hinweis: Diese Aufgabe ist ohne eine vorgegebene Skizze zu lösen!*

## 4: Wichtige Daten

### Explizit

- 2 Stationen - Sender → Empfänger
- Flusskontrolle: Sliding-Window-Technik
- Datenframe = 100 Byte
- Fenstergröße = 300 Byte
- Sequenznummern modulo 8
- Go-Back-N (Fehlerkorrektur)
- 6 Frames übertragen, Frame 5 geht verloren

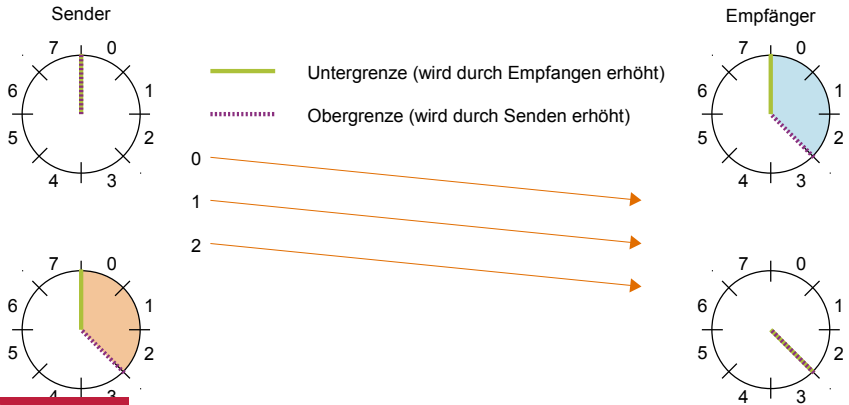
### Implizit

- Frames pro Fenster: 3
- Sequenznummern: 0 bis 7

## 4: Schritt 1

### Was passiert?

- Sender sendet 3 Frames mit jeweils 100 Byte = 300 Byte → Fenster voll

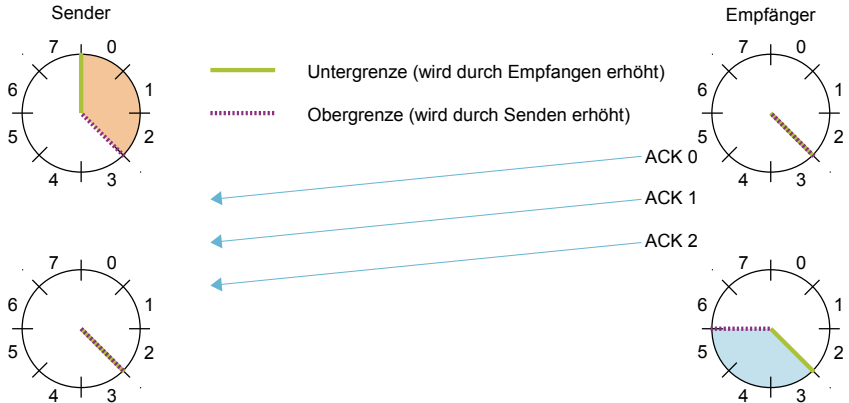




## 4: Schritt 2

### Was passiert?

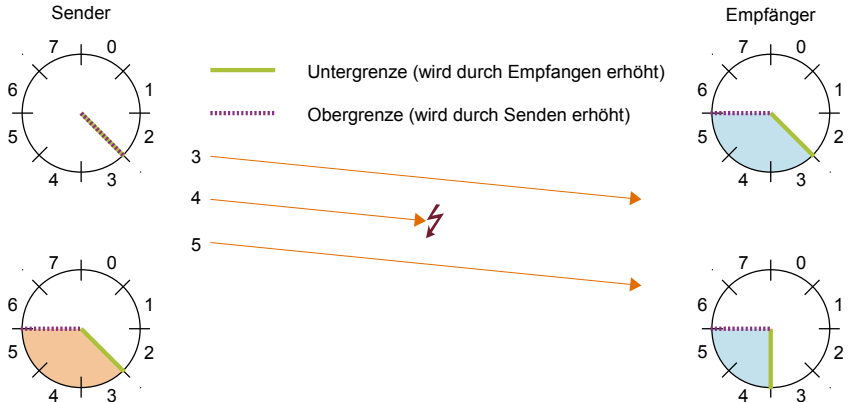
- Empfänger bestätigt jeden Frame mit einem ACK



## 4: Schritt 3

### Was passiert?

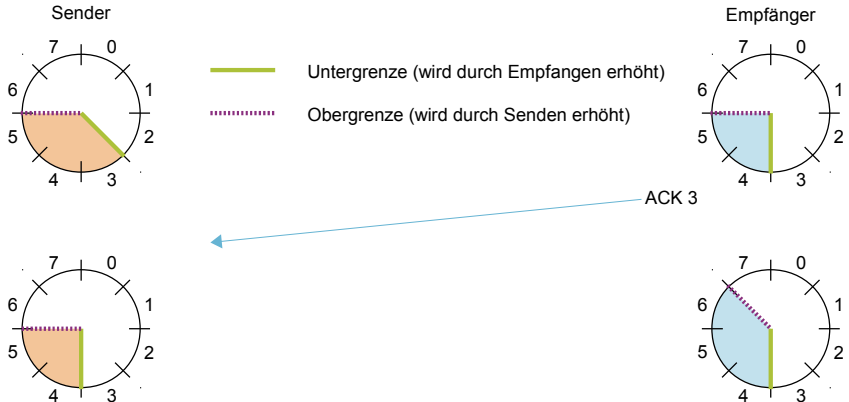
- Sender sendet 3 weitere Frames, wobei Frame 4 verloren geht



## 4: Schritt 4

### Was passiert?

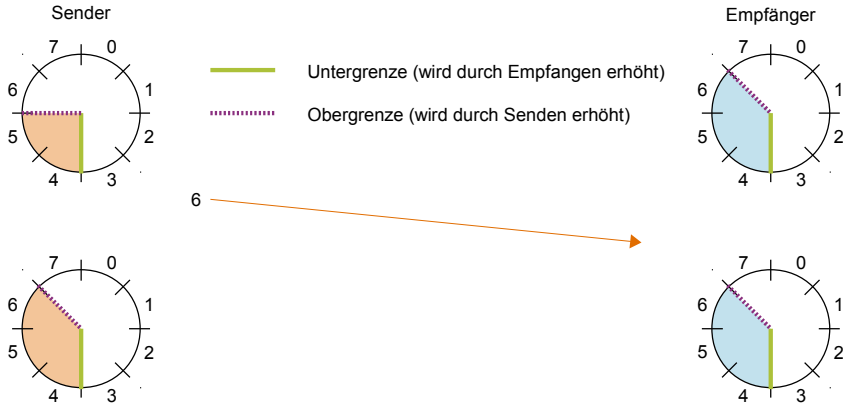
- ACK für Frame 3, Frame 5 ignoriert, da Seq.-Nr. nicht fortlaufend



## 4: Schritt 5

### Was passiert?

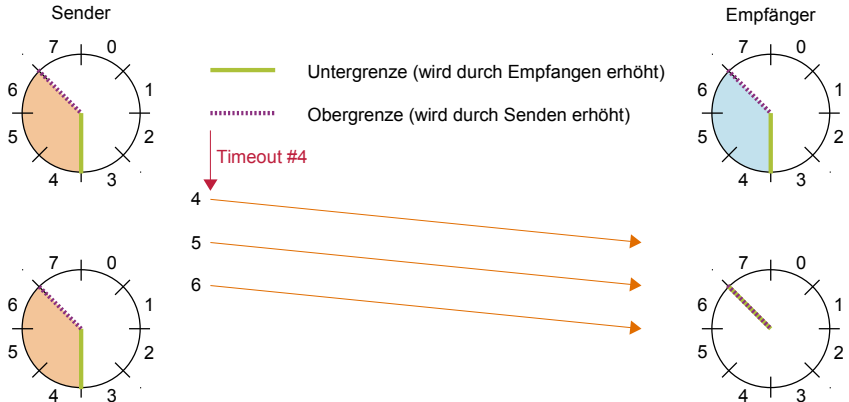
- Sender sendet Frame 6, der ebenfalls ignoriert wird (out-of-order)



## 4: Schritt 6

### Was passiert?

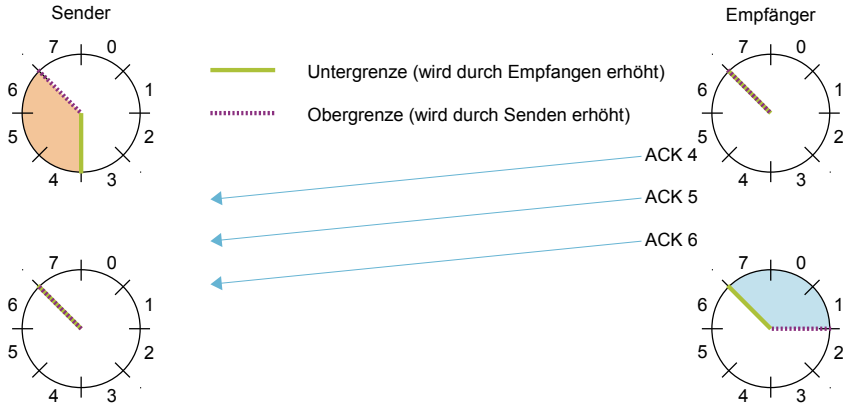
- Timeout für Frame 4 (kein ACK), Neuübertragung ab Frame 4



## 4: Schritt 7

### Was passiert?

- Empfänger bestätigt jeden Frame mit einem ACK



## Aufgabe 5: Sequenznummern

- a) Begründen Sie, warum bei einem Sliding-Window-Verfahren der verfügbare Sequenznummernraum mindestens doppelt so groß sein muss wie die maximale Fenstergröße.
- b) Skizzieren Sie eine Kommunikation zwischen zwei Stationen die aufgrund eines zu kleinen Sequenznummernraums fehlerhaft abläuft.

## 5 a) Sequenznummernraum vs. Fenstergröße

- a) Warum muss bei einem Sliding-Window-Verfahren der verfügbare Sequenznummernraum mindestens doppelt so groß sein wie die maximale Fenstergröße?

### Lösung

Ist der Sequenznummernraum weniger als doppelt so groß wie die maximale Fenstergröße, dann kann in bestimmten Situationen eine Übertragungswiederholung nicht eindeutig zugeordnet werden. Dies kann zu einem Deadlock führen, so dass der Sender auf eine Antwort des Empfängers wartet und der Empfänger auf den Sender wartet.



## 5 b) Beispiel 1

### Beispiel 1 – Sequenznummernraum = Fenstergröße

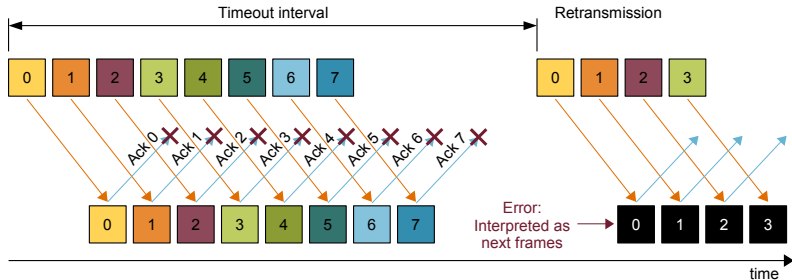
Warum eine Fenstergröße  $> \frac{\text{Seq.-Nr.-Raum}}{2}$  zu Fehlinterpretationen führt:

- Sequenznummernraum = 8
- Fenstergröße = 8
- Alle ACK-Signale gehen verloren

### Achtung

Diese Situation kann vom Empfänger nicht eindeutig erkannt werden!

## 5 b) Beispiel 1



### Beispiel 1 – Lösung

Der Empfänger müsste die *wiederholte* Übertragung der Frames 0 bis 3 bestätigen, damit der Sender fortfahren kann. Er erkennt jedoch *neue*, nicht bereits übertragene Frames.

## 5 b) Beispiel 2

### Beispiel 2 – Fenster ebenfalls zu groß

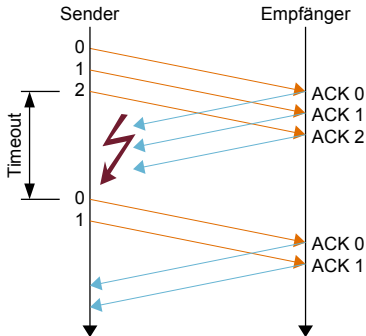
- Sequenznummernraum = 4
- Fenstergröße = 3
- Sämtliche ACKs der Frames 0 bis 2 gehen verloren

### Achtung

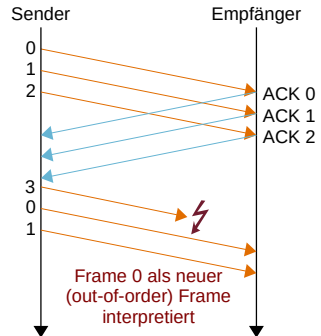
Diese Situation kann vom Empfänger nicht eindeutig erkannt werden!

## 5 b) Beispiel 2

Was passiert ist: Die Frames 0 und 1 werden wiederholt, da ACKs fehlen.



Aus der Sicht des Empfängers kann auch die folgende Situation eingetreten sein!



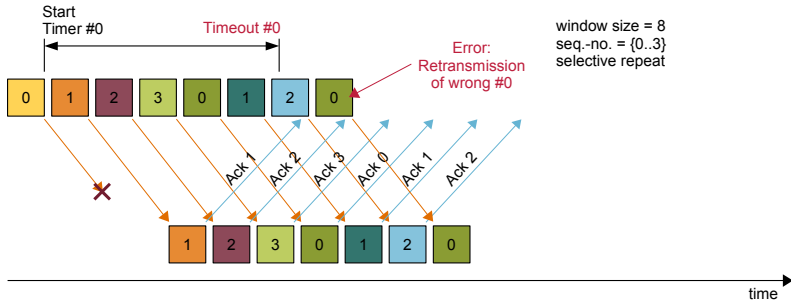
## 5 b) Beispiel 2

### Beispiel 2 – Erklärung

Der Empfänger antwortet nicht auf die erneut versendeten Frames 0 und 1, da er einen Frame mit der Sequenznummer 3 erwartet hat.

Im Falle einer Go-Back-N-Strategie muss er die Frames verwerfen, bei Selective Repeat kann er beide puffern. In jedem Fall muss er auf die Übertragungswiederholung von Frame 3 warten und im Gegensatz zur ersten Situation auf eine Aktion des Senders warten.

## 5 b) Beispiel 3



### Sequenznummernraum kleiner Fenstergröße

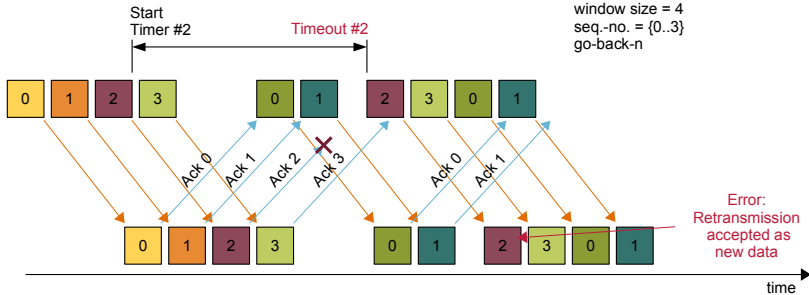
Der erste Frame 0 kann nicht mehr wiederholt werden, da es im Sendefenster bereits überschrieben wurde!

## 5 b) Beispiel 3

### Beispiel 3 – Erklärung

Der Frame 0 wurde im Sendefenster mit neuen Daten überschrieben bevor es bestätigt wurde. Es ist dem Sender somit nicht mehr möglich, die Übertragung (korrekt) zu wiederholen. Weiterhin könnte der Empfänger auch nicht erkennen, welcher der beiden Frames mit Sequenznummer 0 wiederholt wurde.

## 5 b) Beispiel 4



### Sequenznummernraum gleich Fenstergröße

Ack 2 ging verloren. Die Wiederholung konnte nicht erkannt werden, da die Sequenznummer 2 die nächste benötigte Sequenznummer war!



## 5 b) Beispiel 4

### Beispiel 4 – Erklärung

Der Empfänger konnte die Wiederholung des Frames 2 nicht als solche erkennen und hat die Daten als den nächsten erwarteten Frame akzeptiert.

## Aufgabe 6: Kanalauslastung

Zwei Stationen sind über einen Satellitenkanal mit einer Übertragungsrate von 1 Mbit/s ( $= 10^6$  Bit/s) verbunden. Der geostationäre Satellit ist jeweils 36.000 km von beiden Stationen entfernt, die Signalausbreitungsgeschwindigkeit entspricht der Lichtgeschwindigkeit (300.000 km/s). Eine Station sendet Datenframes der Größe 1.500 Bit an die zweite Station, die nur Acknowledgement-Frames der Größe 50 Bit zurücksendet.

## Aufgabe 6: Kanalauslastung

- a) Berechnen Sie die Kanalauslastung, die mit einem Stop-and-Wait-Flusskontrollprotokoll erreicht werden kann.
- b) Berechnen Sie die Kanalauslastung, die mit einem Sliding-Window-Flusskontrollprotokoll mit einer Fenstergröße von 20 Frames erreicht werden kann.
- c) Berechnen Sie die benötigte Fenstergröße, um den Kanal zu 100% auszulasten.

## 6 a) Stop-and-Wait

- a) Berechnen Sie die Kanalauslastung, die mit einem Stop-and-Wait-Flusskontrollprotokoll erreicht werden kann.

### Lösung

Entfernung:  $2 \cdot 36000 \text{ km} = 72000 \text{ km}$

Laufzeit (vgl.  $T_p$ ):  $\frac{72000 \text{ km}}{300000 \text{ km/s}} = 240 \text{ ms}$

Bei 1 Mbit/s ist der Kanal bei der Übertragung von 1500 Bit für 1.5 ms belegt, im Falle eines 50 Bit Frames für 50  $\mu\text{s}$  (vgl.  $T_x$ ).

Damit benötigt das Senden insgesamt:

$$T_p + T_x + T_p + T_{xack} = 240 \text{ ms} + 1.5 \text{ ms} + 240 \text{ ms} + 50 \mu\text{s} = 481.55 \text{ ms}$$

Davon wird der Kanal 1.55 ms belegt, also beträgt die Kanalauslastung:

$$\frac{1.55 \text{ ms}}{481.55 \text{ ms}} \approx 0.32 \%$$

## 6 b) Sliding-Window

- b) Berechnen Sie die Kanalauslastung, die mit einem Sliding-Window-Flusskontrollprotokoll mit einer Fenstergröße von 20 Frames erreicht werden kann.

### Berechnung

Vereinfachte Formel aus der Vorlesung (die eine grobe Annäherung ist, weil sie die Übertragungszeit von nur einem Frame berücksichtigt):

$$U = \frac{kT_x}{T_x + 2T_p}$$

### Lösung

$$U = \frac{20 \cdot 1.5 \text{ ms}}{1.5 \text{ ms} + 2 \cdot 240 \text{ ms}} \approx 6.2 \%$$

## 6 c) benötigte Fenstergröße

c) Berechnen Sie die benötigte Fenstergröße, um den Kanal zu 100% auszulasten.

### Lösung

Gesucht: Fenstergröße (k)

$$\frac{k \cdot 1.5 \text{ ms}}{1.5 \text{ ms} + 2 \cdot 240 \text{ ms}} = 1 \Leftrightarrow k = \frac{480 \text{ ms} + 1.5 \text{ ms}}{1.5 \text{ ms}} = 321$$

⇒ 100 % Kanalauslastung ab einer Fenstergröße von 321

# Zusammenfassung

## Flusskontrolle vs. Staukontrolle

- Flusskontrolle sorgt dafür, dass der Empfänger nicht überlastet wird
- Staukontrolle sorgt dafür, dass das Netz nicht überlastet wird

## Sendewiederholungen

- Es gibt verschiedene Verfahren um verlorene Frames zu behandeln
- Sie unterscheiden sich in der Komplexität und ihrer Leistungsfähigkeit

## Sliding Window

- bessere Kanalauslastung als Stop-and-Wait
- Sequenznummernraum muss richtig gewählt werden

# Nächste Übung

## 05. Juni 2025