

# Übung 1

## Gruppe Hosebei

Michele Luongo, s99-713-190

Franziska Zumsteg, s99-717-084

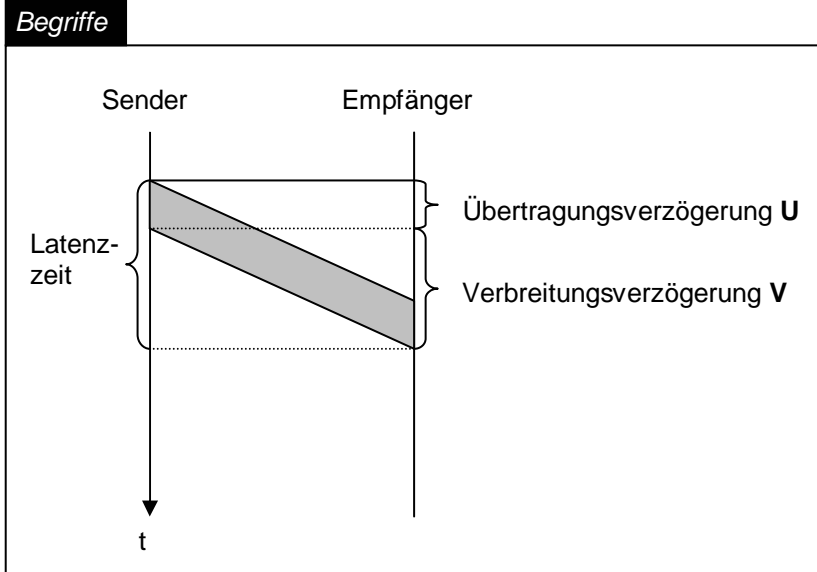
Philip Iezzi, s99-714-354

**Aufgabe 1**

Ann.:

$$1KB = 2^{10} \cdot 8bit$$

$$1Mbit = 10^6 bits$$

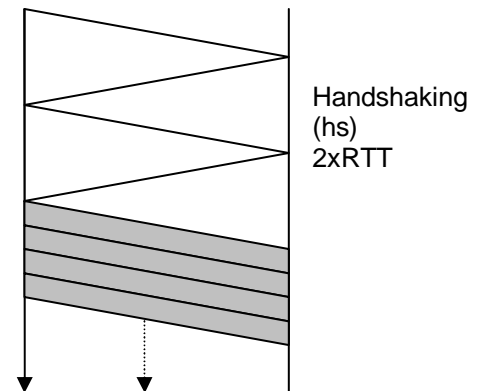


**Lösungen:**

- a) Bandbreite = 1.5 Mbit/s = 1'500'000 bit/s  
Datei = 1000 KB = 8'192'000 bit

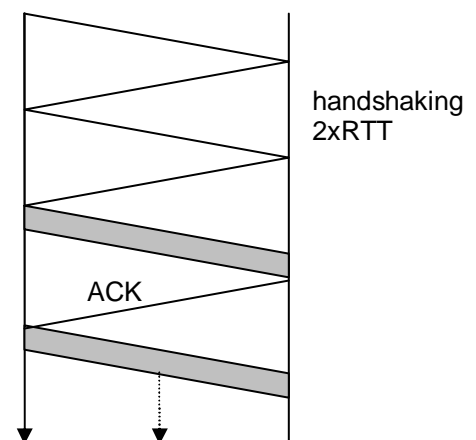
$$t = 2 \cdot 0.1 + \frac{8192 \cdot 1000}{1500000} + 0.05 = \underline{\underline{5.7113 \text{ s}}}$$

|              |                           |                 |
|--------------|---------------------------|-----------------|
| hs           | 2 * RTT =                 | 0.2 s           |
| U            | 8.192 Mbit / 1.5 Mbit/s = | 5.4613 s        |
| V            | ½ RTT =                   | 0.05 s          |
| <b>Total</b> |                           | <b>5.7113 s</b> |



- b)  $t = 2 \cdot 0.1 + 1'000 \cdot \left( \frac{8192}{1'500'000} + 0.1 \right) - 0.05 = \underline{\underline{105.6113 \text{ s}}}$

|              |                           |                   |
|--------------|---------------------------|-------------------|
| hs           | 2 * RTT =                 | 0.2 s             |
| U            | 8.192 Mbit / 1.5 Mbit/s = | 5.4613 s          |
| V            | 999½ RTT =                | 99.95 s           |
| <b>Total</b> |                           | <b>105.6113 s</b> |

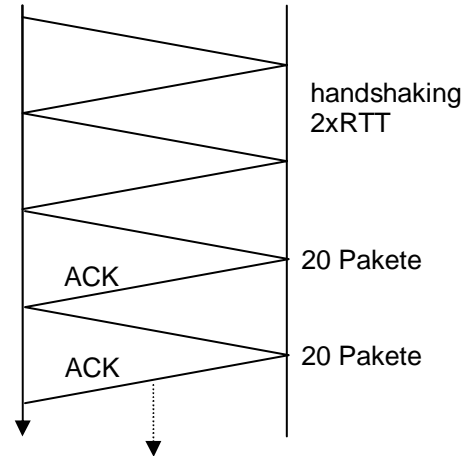


- c) Bandbreite =  $\infty$   
 → Übertragungsverzögerung = 0

$$t = 2 \cdot 0.1 + \frac{1000}{20} \cdot 0.1 - 0.05 = \underline{\underline{5.15 \text{ s}}}$$

|              |             |               |
|--------------|-------------|---------------|
| hs           | 2 * RTT =   | 0.2 s         |
| U            |             | 0 s           |
| V            | 50 * 0.05 = | 2.50 s        |
| W            | 49 * 0.05 = | 2.45 s        |
| <b>Total</b> |             | <b>5.15 s</b> |

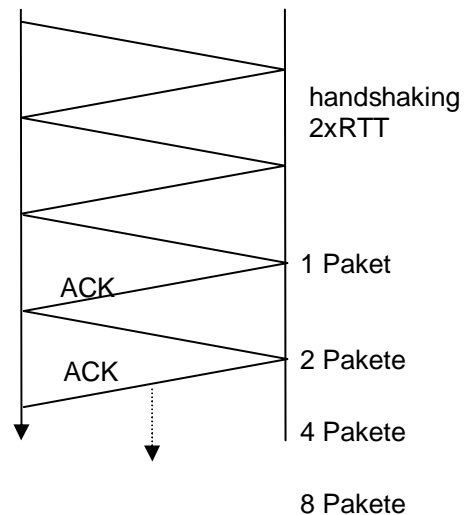
[W = wait for ACK]



- d) 1, 2, 4, 8, 16, 32, 64, 128, 256, 512 (genutzt: 489)  
 → 10xRTT

$$t = 2 \cdot 0.1 + 10 \cdot 0.1 - 0.05 = \underline{\underline{1.15 \text{ s}}}$$

|              |             |               |
|--------------|-------------|---------------|
| hs           | 2 * RTT =   | 0.2 s         |
| U            |             | 0 s           |
| V            | 10 * 0.05 = | 0.5 s         |
| W            | 9 * 0.05 =  | 0.45 s        |
| <b>Total</b> |             | <b>1.15 s</b> |



**Annahme a)-d):**

Sender erwartet bei letztem Paket keine Bestätigung, daher nur halbe RTT bei letztem Paket.

## Aufgabe 2

**STDM** = Synchronous Time Division Multiplexing (Zeitmultiplexen)

**FDM** = Frequency Division Multiplexing (Frequenzmultiplexen)

**Sprachtelefonnetz: STDM**

Verkehrsmuster: Ein Sender kommuniziert mit einem Empfänger. Dabei wird ein bidirektionaler Kommunikationskanal benötigt.

Dabei stellt STDM synchronisierte direkte Verbindungen zur Verfügung, welche genügen, die Sprache verständlich zu übertragen. Sprache ist stark redundant. Eine minimale Verzögerung kann in Kauf genommen werden.

Gleicher Kanal wird nur von wenigen (2) Benutzern verwendet. Es wird jedoch so schnell "geswitched", dass fast gleichzeitig viele Verbindungen hergestellt werden können.

### Fernseh- und Radionetze: FDM

Verkehrsmuster: Wenige Sender übertragen gleichzeitig mehrere Kanäle, welche von vielen Empfängern gleichzeitig benutzt werden können. Dabei reicht eine unidirektionale Verbindung. Für die Empfänger sind alle Kanäle stets verfügbar, dabei muss nicht "geswitched" werden, was eine gewaltige Reduktion des Auswandes ist.

Beim Sprachtelefonnetz würde es sich nicht lohnen, das verfügbare Frequenzband aufzuteilen, da dies eine viel zu grosse Menge an Frequenzen wäre. Technisch wäre dies wohl gar nicht machbar.

Bei beiden Arten von Verkehrsmuster wird das Problem auf die jeweils effizienteste Art gelöst. Dabei werden bei beiden Arten eine grosse Anzahl von Kunden befriedigt. Deshalb können wir hier von "kostengünstig" sprechen, obschon die Technik an sich eigentlich teuer ist.

### Aufgabe 3

|   |  |
|---|--|
| a) Öffnen einer Datei                         | Wohl eher auf <b>Verzögerung</b> empfindlich, da der ganze Inhalt der Datei beim Öffnen noch nicht bekannt sein muss. Öffnen kann als Anfrage alleine verstanden werden, ohne Übertragung der gesamten Datenmenge. |
| b) Lesen des Inhalts einer Datei              | auf <b>Bandbreite</b> empfindlich. Je nach Dateigrösse kann die Bandbreite grosse Auswirkung haben, da nun die gesamte Datenmenge eingelesen werden muss.  |
| c) Auflisten des Inhalts eines Verzeichnisses | keine wesentlich grosse Menge an Daten wird übertragen. Die <b>Verzögerung</b> spielt hier die grössere Rolle.   |
| d) Anzeigen der Attribute einer Datei         | wie c)<br><b>Verzögerung</b>   |

### Aufgabe 4

| Paketgrösse | Nutz-Daten | Anz. Pakete | Overhead-Bytes | Verlust-Bytes | Total "Müll"-Bytes |
|-------------|------------|-------------|----------------|---------------|--------------------|
| 1000        | 900        | 1112        | 111200         | 900           | <b>112100</b>      |
| 5000        | 4900       | 205         | 20500          | 4900          | <b>25400</b>       |
| 10000       | 9900       | 102         | 10200          | 9900          | <b>20100</b>       |
| 20000       | 19900      | 51          | 5100           | 19900         | <b>25000</b>       |

→ Wir würden also 10'000 Byte als Paketgrösse wählen!

Annahme:

Nicht vollständig gefüllte Datenpakete (hier: das letzte Datenpaket) gelten als vollständige Datenpakete, d.h. gleiche Menge an Nutzdaten in jedem Paket.

## Aufgabe 5

| Anforderungen an...<br>Anwendung                                       | Ø Bandbreite  | Spitzenbandbreite   | Latenz   | Jitter  | Verlusttoleranz                                 |
|--|---|---|--|---|---|
| <b>a. Sprachverkehr</b>  | mässig  | hoch<br>z.B.<br>Natelverkehr<br>an Silvester                    | hoch<br>(real-time)  | mässig  | niedrig<br>(Sprache ist<br>redundant)           |
| <b>b. Echtzeitübertragung<br/>von Musik</b>                            | hoch  | hoch  | hoch<br>(real-time)  | hoch<br>(verzerrte<br>Klänge<br>verunstalten<br>die Musik!) | hoch<br>(Annahme:<br>sensibles<br>Gehör)        |
| <b>c. Datei-Server</b>   | hoch  | hoch  | gering   | gering  | hoch  |
| <b>d. Druck-Server</b>   | mässig  | mässig<br>(der<br>Flaschenhals<br>ist der<br>Drucker<br>selbst) | gering   | gering  | hoch  |
| <b>e. Digitale Bibliothek</b>  | hoch  | hoch  | mässig   | gering  | hoch  |
| <b>f. Routineüberwachung<br/>von entfernten<br/>Wetterinstrumenten</b> | gering  | mässig  | gering<br>(Routine: Ob<br>Messdaten<br>verspätet<br>ankommen<br>spielt keine<br>Rolle) | gering  | hoch  |
| <b>g. Videoüberwachung<br/>eines Warteraums</b>                        | mässig<br>Annahme:<br>Keine hohe<br>Bildqualität/<br>Bildwiederholrate<br>nötig | mässig  | mässig   | mässig  | mässig  |
| <b>f. Fernsehsendung</b>   | hoch  | hoch  | hoch<br>(v.a. falls<br>Live-<br>Sendung)   | mässig  | niedrig<br>(kleines<br>Flackern<br>stört nicht) |

**Definition:**

*Latenz:* Totale Übertragungs- und Verbreitungsverzögerung (siehe Grafik in Aufg. 1)

*Jitter:* Latenzzeitschwankungen (Variabilität der Antwortzeit)

*hohe Anforderung an Latenz* = geringe Latenz

*hohe Anforderung an Jitter* = keine grossen Latenzzeitschwankungen

*hohe Anforderung an Verlusttoleranz* = möglichst wenig Datenverlust

Annahme:

Alle Anwendungen werden über digitale Datenleitungen verbreitet.

## Aufgabe 6

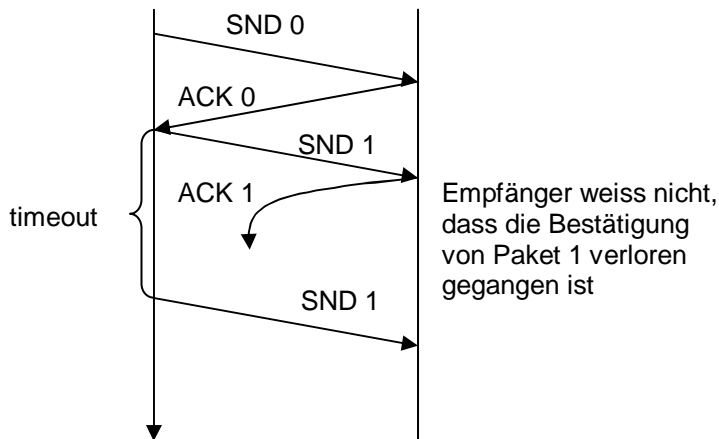
Paket-Grösse: 1 KB  
„Stop-and-Wait“

### a) zuverlässige Verbindung

Da das nächste Paket erst übertragen wird, sobald das vorherige erfolgreich bestätigt wurde, wird keine Sequenznummer benötigt. Wir gehen von einer zuverlässigen Verbindung aus, also muss stets auf genau ein gesendetes Paket genau eine Bestätigung folgen. Sobald das nächste Paket auf die „Reise“ geht, ist das vorherige bereits beim Empfänger. Ein „Überholen“ über eine kürzere Route ist somit gar nicht erst möglich.

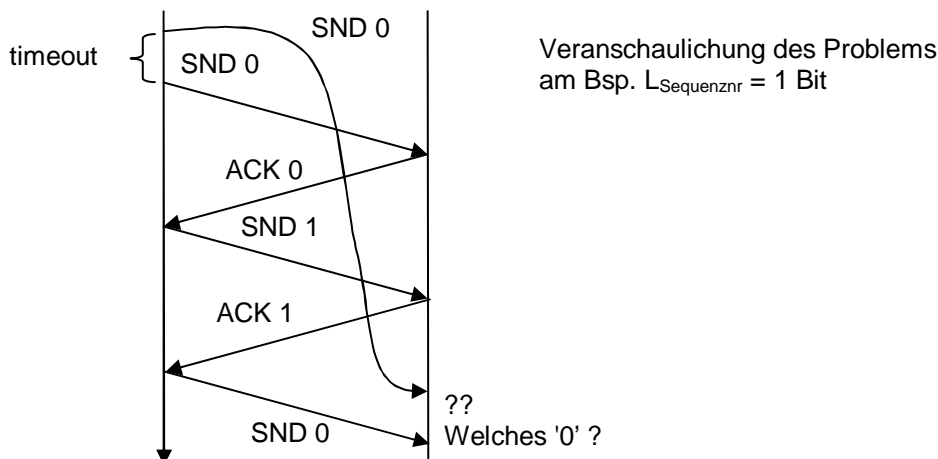
### b) minimale Länge der Sequenznummer

Die minimale Länge der Sequenznummer bei einer Verbindungsleitung, bei der gelegentlich Pakete verloren gehen, beträgt 1 Bit. Theoretisch reicht es aus, zwischen zwei aufeinander folgenden Datenpaketen unterscheiden zu können (Sequenznummer 0 und 1). Dank der Sequenznummer merkt der Empfänger, dass die Bestätigung verloren gegangen ist, da er nochmals ein Paket mit der selben Sequenznummer erhält.



### c) verspätetes Eintreffen von Paketen

Die Anforderungen an die Sequenznummer wächst gewaltig. Die Länge der Sequenznummer sollte ausreichen, alle Pakete, welche während einer Minute versendet werden, mit einer eindeutigen Nummer zu versehen. Je nach Medium wird eine grosse Anzahl Bits benötigt.



## Aufgabe 7

---

**Round Robin** kann mit **STDM** verglichen werden. Der Sender muss auf den Token warten, der in regelmässigen Abständen an ihm „vorbeikurvt“. Beim **STDM** werden genauso fixe Zeitschlitzte vergeben, was mit der Wartezeit auf den Token verglichen werden kann.

Bei **Round Robin** kann es jedoch zu längeren Wartezeiten auf den freien Token kommen, da dieser ja von einem anderen Sender besetzt sein kann. Nur bei freiem Token ist die Wartezeit regelmässig. Dies also ein Unterschied zu **STDM**, bei welchem wir eine fixe Zeitverteilung haben.

Mit **FDM** hat **Round Robin** nicht viel gemeinsam. Der Kanal wird nach Frequenzen unterteilt und nicht nach Zeitschlitzten. Während die Aufteilung bei **Round Robin** und **STDM** „sequentiell“ erfolgt, erfolgt sie bei **FDM** „parallel“.