Übertragungsdauer

$$T_{ges} = T_s + T_a$$

Ausbreitungsgeschwindigkeit In üblichen Medien (Kabel, Glasfaser) ist

$$v \approx \frac{2}{3}\, c \approx 200\ 000 \frac{km}{s}$$

Bandbreiten-Verzögerungs-Produkt ist das Datenvolumen, das sich während der Übertragung auf dem Medium befindet.

$$L = d \cdot T_a$$

Auslastung des Mediums ist das Verhältnis von tatsächlicher zu möglicher Datenrate.

$$U = \frac{d}{d_{max}}$$

2 Sicherungsschicht (Data Link Layer)

Aufgaben der Sicherungsschicht.

- Fehlererkennung, z.B. durch CRC
- Fehlerbehebung, z.B. durch Quittungen und Sendewiederholungen
- Strukturierung des Datenstroms, z.B. durch Sequenznummern
- Medienzugangskontrolle, z.B. durch CSMA/CD

Unterteilung. Die Sicherungsschicht kann in zwei Unterbereiche eingeteilt werden:

Logical Link Control Sicherung der Punkt-zu-Punkt-Verbindung gegen Verfälschung, Verlust und Reihenfolgenvertauschung; Flusskontrolle, Strukturierung der Übertragung

Medium Access Control Zugangskontrolle für ein geteiltes Medium

2.1 Fehlererkennung

Paketfehlertypen. Ein zuverlässiger Dienst vermeidet folgende Fehlertypen:

- Verlust
- Duplizierung
- Vertauschung der Reihenfolge
- Phantom-Dateneinheiten

Paritätsbit. Mit einem Paritätsbit kann man eine ungerade Anzahl von Fehlern erkennen.

Frame Check Sequence bezeichnet eine Prüfsumme, die zu einem Datenblock (Frame) hinzugefügt wird, um Fehlererkennung oder Fehlerkorrektur zu ermöglichen.

Cyclic Redundancy Check Hierbei wird die Datensequenz als Polynom betrachtet und ein Generatorpolynom festgelegt, das Sender und Empfänger bekannt ist. Die Datensequenz wird vor dem Senden durch das Generatorpolynom divivdiert. Der Rest der Division wird an die Datensequenz (als *FCS*) angehängt und beides wird übermittelt. Der Empfägner führt wiederum die Polynomdivision durch G durch. Falls hierbei kein Rest bleibt, wurde die Sequenz entweder fehlerfrei übertragen, oder die Sequenz ist fehlerhaft, hat aber weiterhin G als Faktor, was sehr unwahrscheinlich ist.

Gegeben sei die Datensequenz M und das Generatorpolynom G.

1. Multipliziere M mit 2^d, wobei d der Grad von G ist. Das entspricht dem Anhängen von d Nullen an M.
$$M' = M \cdot 2^d$$
2. Dividiere M' modulo durch G.
$$C = M' \ mod \ G$$
3. Füge C an die Datensequenz an.
$$\tilde{M} = M' + C$$
4. Übertrage $\tilde{M}$
5. Dividiere die empfangene Sequenz modulo durch das Generatorpolynom.
$$C' = \tilde{M} \ mod \ G$$
Wenn $C' = 0$, dann war die Übertragung mit hoher Wahrscheinlichkeit fehlerfrei.

CRC: Implementierung als Schieberegister

2.2 Flusskontrolle

Ziel. Flusskontrolle dient der Anpassung der Übertragungsrate an die Verarbeitungskapazitäten des Empfängers.

Anzahl der Sequenznummern mit Sliding Window. Gegeben seien die Datenrate des Mediums d , die Größe einer Dateneinheit S, sowie die Lebenszeit einer Dateneinheit T_l. Eine Sendewiederholung findet maximal nach T_w statt und der Empfänger quittiert den Erhalt der Dateneinheit nach maximal T_q. Dann ist eine untere Schranke für die benötigte Bitanzahl n gegeben durch
$$2^n \ Dateneinheiten \geq (2T_l + T_w + T_q) \cdot D$$
wobei $D = \frac{d}{S}$ die Dateneinheiten-Rate auf dem Medium ist.

Implizite Flusskontrolle bezeichnet die Drosselung der Datenrate durch das Zurückhalten der Quittungen. Dieses Verfahren ist problematisch: Wartet der Sender zu lange auf eine Quittung, so wird nach dem Timeout die Dateneinheit erneut gesendet, was die Resourcen von Sender und Empfänger sowie die Kapazität des Mediums beansprucht.

Automatic Repeat Request steht für die automatische Wiederholung verlorengegangener oder beschädigter Dateneinheiten.

2.2.1 Stop-and-Wait

Stop-and-Wait ist ein einfaches ARQ-Verfahren. Dabei wartet der Sender auf die Quittung zu einer gesendeten Dateneinheit, bevor er die nächste verschickt. Falls vor dem Timeout keine Quittung eintrifft, wird die Dateneinheit wiederholt.

Übertragunszeit bei Stop-and-Wait. Unter der Annahme, dass Verarbeitungsdauer und Größe der Quittung vernachlässigbar sind, erreicht die Quittung den Sender nach $T_s + 2 \cdot T_a$. Damit ist die gesamte Übertragungszeit für n Dateneinheiten

$$T_{ges} = (T_s + 2 \cdot T_a) \cdot n$$

Auslastung bei Stop-and-Wait. Die Auslastung des Mediums beim Stop-and-Wait-Verfahren ist das Verhältnis von Sendezeit zur Summe aus Sende- und Wartezeit:

$$U = \frac{T_s}{T_s + 2T_a}$$

Leistungsfähigkeit im Fehlerfall. Gegeben sei die Fehlerwahrscheinlichkeit p.

$$U = \frac{1-p}{2a+1}$$

2.2.2 Go-Back-N

Prinzip. Der Sender kann eine gewisse Anzahl von Dateneinheiten (Fenstergröße W) senden, bevor er eine Quittung empfangen muss. Nach dem Senden der Dateneinheiten $D_i, \ldots, D_{i+W-1}$ wartet er auf das Eintreffen einer Quittung. Sobald die Quittung ACK_i eintrifft, schickt er D_{i+W}. Hat der Sender nach Ablauf des Timers keine Quittung für D_i erhalten, wiederholt er alle Dateneinheiten $D_i, \ldots, D_{i+W-1}$.

Beim Erhalt einer erwarteten Dateneinheit D_i schickt der Empfägner die Quittung ACK_i. Erhält er eine Dateneinheit D_j mit $j > i+1$, so wird sie verworfen und die Quittung ACK_i wiederholt, solange bis D_{i+1} eintrifft, die dann mit ACK_{i+1} quittiert wird.

Auslastung in Abhängigkeit von der Fenstergröße. Das Fenster kann so groß sein, dass der Sender kontinuierlich senden kann, oder er muss nach Ausnutzung des Fensters auf die Quittung warten. Im ersten Fall wird die Kapazität des Mediums voll ausgenutzt.

$$U(W,a) = \begin{cases} 1 & \text{wenn } W \geq 2a+1 \\ \frac{W}{2a+1} & \text{wenn } W < 2a+1 \end{cases}$$

$$a = \frac{T_a}{T_s}$$

2.3 Medienzugangskontrolle

2.3.1 Medienzugangsverfahren

Raummultiplex Einteilung des Raumes in Sektoren durch gerichtete Antennen, dedizierte Leitungen etc; Beispiel: Zellenstruktur in Mobilfunknetzen

Zeitmultiplex Kanal belegt gesamten Frequenzraum für eine gewisse Zeit; Beispiel: Ethernet

Frequenzmultiplex Einteilung der verfügbaren Bandbreite in Frequenzabschnitte; Beispiel: UKW-Radio

Codemultiplex Kanäle operieren gleichzeitig auf derselben Frequenz; durch Verknüpfung des Signals mit einem eindeutigen Code können die Kanäle vom Empfänger wieder getrennt werden

2.3.2 CSMA/CD (Carrier Sense Multiple Access with Collision Detection)

CSMA/CD ist ein Zeitmultiplex-Verfahren bei konkurrierendem Zugriff auf das Medium.

Ablauf.

1. *Listen before Talk*: Endsystem prüft, ob das Medium frei ist
2. Falls Medium frei beginnt das Endsystem zu senden
3. *Listen while Talk*: Zur Erkennung von Kollisionen hört der Sender während des Sendens das Medium ab
4. im Kollisionsfall: Sendeunterbrechung, Jamming-Signal
5. Wiederholung der Sendung, geregelt durch Backoff-Algorithmus

Mindestlänge der Dateneinheit. Eine Voraussetzung für CSMA/CD ist, dass das Senden der Dateneinheit nach der zweifachen Signallaufzeit noch nicht beendet sein darf. Daraus ergibt sich eine Mindestlänge für die Dateneinheit, die durch das Padding-Feld der Dateneinheiten sichergestellt wird.

Abkürzung	Länge [bit]	Beschreibung
PR	56	Präambel zur Synchronisation
SD	8	Start-of-Frame Delimiter zeigt Beginn an
DA	16/48	Destination Address, MAC-Adresse des Ziels
SA	16/48	Source Address, MAC-Adresse der Quelle
Length	16	Anzahl der Oktette im Datenfeld
Data	≤ 12000	Datenfeld, max. 1500 Oktette
PAD	optional	Padding um auf nötige Mindestlänge zu ergänzen
FCS	32	Frame Check Sequence, CRC-32

Table 3: Felder einer CSMA/CD-Dateneinheit

CSMA/CD-Dateneinheit

2.4 Lokale Netze

2.5 Ethernet

Topologie. Beim Ethernet sind die Endsysteme in Sterntopologie verbunden.

2.6 Token Ring

Topologie. Beim Token Ring sind die Endsysteme jeweils Punkt-zu-Punkt zu einem Ring verbunden.

Ablauf. Token Ring ermöglicht kontrollierten Zugriff auf das Medium mittels eines zirkulierenden Senderechts (Token). Empfängt ein Endsystem das Token, so hat es für eine bestimmte Zeit das Recht zu senden (Token Holding Time). Die gesendeten Daten kommen nach dem Umlauf im Ring wieder beim Sender an, der diese wieder vom Ring nimmt. Danach gibt er das Token an das nachfolgende System weiter.

Echtzeit. Token Ring ist für Echtzeitanwendungen geeignet, da eine maximale Wartezeit zwischen Sendewunsch und Anfang des Sendens garantiert werden kann:

$$Wartezeit_{max} = (n-1) \cdot Sendezeit(Datenmenge_{max}) + Tokenumlaufzeit$$

Quittierung. Ein Endsystem im Token Ring kann die Dateneinheit ändern, bevor es diese weitergibt. Der Empfänger kann also den Erhalt einer Dateneinheit quittieren, indem er z.B. ein bestimmtes Bit setzt. Der Sender erkennt diese Quittung während er die Dateneinheit vom Ring nimmt.

Aktiver Anschluss ans Medium. Beim Token Ring sind die Endsysteme aktiv ans Medium angeschlossen. Jedes System regeneriert die erhaltene Dateneinheit und schickt sie weiter, agiert somit als Verstärker. Das ermöglicht größere Leitungslänge. Die Regenerierung ist allerdings mit Verzögerung verbunden. Für den Anschluss neuer Systeme an das Medium muss das Medium unterbrochen werden.

2.7 High-Level Data Link Control

Phasen.

1. Verbindungsaufbau
2. Datentransfer
3. Verbindungsabbau

Flag. Die eindeutige Bitsequenz

`01111110`

wird verwendet, um Anfang und Ende einer Dateneinheit zu signalisieren. Ein Autreten des Flags in den Nutzdaten wird durch *Bitstopfen* verhindert. Hierbei wird nach 5 aufeinanderfolgenden Einsen eine Null eingefügt, die vom Sender wieder entfernt wird.