

Teil II vertieft und erläutert die Grundlagen der Telematik. Der Aufbau dieses Teils orientiert sich streng nach den Schichten des ISO/OSI-Modells, wobei die näheren Erläuterungen zu den Kabeln vorangestellt werden.

7. Übertragungsmedien

Die wichtigsten physikalischen Eigenschaften und Begriffe im Zusammenhang mit Kupfer- und Glasfaser-Kabel werden erklärt. Einige Beispiele von Kabeln und deren Normierung runden die Ausführungen in diesem Kapitel ab.

8. Übertragungsarten

Serielle und parallele Übertragungen, synchrone und asynchrone Übertragungen sowie Zeichen- und Bit-orientierte Übertragung werden erläutert. Die Technologie des Multiplexens rundet die Betrachtung der grundsätzlichen Übertragungsarten ab.

9. Bitströme und Signale

Die Entstehung von Bitströmen aus analogen Signalen in der Schicht 1 und deren Übertragung als codierte oder modulierte Signale wird an dieser Stelle erklärt.

10. Strukturierung der Bitströme (Framing)

Das Framing und die Gründe für eine Strukturierung der Bitströme und die damit im Zusammenhang stehenden Funktionen der Schicht 2 werden erläutert.

11. Vermittlung von Netz-Verbindungen

Das Versenden von Nachrichten über ein Netz (Vermittlung) und die damit im Zusammenhang stehenden Funktionen der Schicht 3 werden erläutert.

12. Sicherung des Nachrichten-Transportes

Die Sicherung der Nachrichtenübertragung im Netz wird erklärt.

13. Anwendungsorientierte Funktionen der Übertragung

Die Sitzungsfunktionen, Darstellungsfunktionen und anwendungsspezifische Funktionen in Netzen werden beschrieben.

7 Übertragungsmedien

In diesem Kapitel werden die verschiedenen Übertragungsmedien und deren Eigenschaften besprochen, die in der Telematik für die Übertragung von Daten zur Verfügung stehen. Es werden die dazu notwendigen physikalischen Begriffe, die im Zusammenhang mit den Übertragungsmedien benötigt werden, erläutert. Einige Übertragungsmedien werden exemplarisch vorgestellt, wobei die Normierung von Übertragungsmedien in Netzen im Vordergrund stehen soll. Einige Beispiele von gebräuchlichen Steckern und Verbindungen sowie die möglichen Störungen auf den Übertragungsmedien bilden den Abschluss des Kapitels.

7.1 Physikalische Begriffe bei Übertragungsmedien

In Kabel-Datenblättern werden wir auf die Begriffe Dämpfung, Reflexion, Übersprechen, Bandbreite und Verzerrung stoßen. Diese Begriffe sollte man erklären können. Bei den Lichtwellenleitern kommen Begriffe wie optische Dichte, Lichtbrechung und Reflexion, Dämpfung, Streuung, Absorption, Übertragungsfenster und Dispersion vor.

7.1.1 Bandbreite B (engl. bandwidth)

Die Bandbreite eines Signals ist der Frequenz-Bereich von der tiefsten bis zur höchsten Frequenz, die in diesem Signal vorkommen. Bei analogen technischen Übertragungseinrichtungen wird das gesamte zur Verfügung stehende Frequenzband mittels Filtern (Hoch- und Tiefpassfilter) in Bänder aufgeteilt (Breitbandnetz) und den unterschiedlichen Technologien zur Verfügung gestellt (Tabelle 7.1).

Beim Kabel-TV-Netz zum Beispiel wird das gesamte Frequenzband von etwa 41 bis 854 MHz in verschiedene Kanäle unterteilt. Diese Kanäle haben eine bestimmte Bandbreite (etwa 7 MHz), in welcher das TV-Signal übertragen wird. In diesen Kanälen werden sowohl die UKW-Frequenzen, als auch analoge und vermehrt auch digitale TV-Signale übertragen.

Der Begriff Bandbreite wird in der Praxis leider oft total falsch verwendet. Sehr oft wird damit die Übertragungsrate zwischen zwei Kommunikationsteilnehmern bezeichnet, was natürlich nicht stimmt. So kann es sein, dass eine Übertragungsstrecke eine installierte Bandbreite von 6 MHz hat, die Übertragungsrate jedoch 10 Mbit/s beträgt.

Applikation

7 Anwendung

6 Darstellung

5 Sitzung

4 Transport

3 Vermittlung

2 Sicherung

1 Bitübertragung

Übertragungsmedien

Praxis-Hinweis:

Die an dieser Stelle beschriebenen Begriffe finden sich in Katalogen und anderen Hersteller-Unterlagen wieder. Dort finden sich auch genaue Zahlenwerte für die hier beschriebenen physikalischen Eigenschaften der Übertragungsmedien.

Bandbreite in Hz ist nicht gleich der Übertragungsrate in Bit/s!

Bezeichnung	Kanäle	Frequenz	Wellenlänge	Kanalbreite
Langwelle	-	151 - 285 kHz	2000 - 1050 m	9 kHz
Mittelwelle	-	536 - 1605 kHz	560 - 189 m	9 kHz
Kurzwelle	-	5,95 - 26,1 MHz	50 - 11,5 m	9 kHz
VHF Band I	2-4	41 - 68 MHz	6,35 - 4,4 m	CCIR-Norm abhängig
UKW	2-56	87,5 - 100 MHz	3,4 - 2,9 m	300 kHz
Digital Radio	S 2-S 5	111 - 139 MHz	2,7 - 2,15 m	7 MHz
USB	S 1-S 10	104 - 174 MHz	2,9 - 1,7 m	CCIR-Norm abhängig
VHF Band II	5-12	174 - 223 MHz	1,7 - 1,3 m	CCIR-Norm abhängig
OSB	S 11-S 20	230 - 300 MHz	1,3 - 1 m	CCIR-Norm abhängig
ESB	S 21-S 37	302 - 446 MHz	1 - 0,64 m	CCIR-Norm abhängig
UHF Band IV	21-37	470 - 606 MHz	64 - 49,5 cm	CCIR-Norm abhängig
UHF Band V	38-69	606 - 854 MHz	49,5 - 35 cm	CCIR-Norm abhängig
S-Band	-	2 - 3,5 GHz	normalerweise Interkontinentalverbindungen	
C-Band	-	3,6 - 4,2 GHz	Satelliten-TV	
X-BAND	-	7,25 - 8,4 GHz	nur für militärische Zwecke	
KU FSS-Band	-	10,7 - 11,7 GHz	Satelliten-TV	
KU DBS-Band	-	11,7 - 12,5 GHz	Satelliten-TV	
KU SMS-Band	-	12,5 - 12,75 GHz	Satelliten-TV	

Tabelle 7.1: Die kommerzielle Rundfunk-Kanaleinteilung

7.1.2 Basisband (engl. baseband)

Nutzt man in einem Kabel nur einen Teil der verfügbaren Bandbreite (z.B. nur den untersten Kanal eines Breitbandnetzes), so spricht man vom Basisband-System. Die vom Sendesignal eingenommene Bandbreite ist direkt abhängig von der Übertragungsgeschwindigkeit. Sie kann aber je nach verwendetem Übertragungsverfahren stark variieren. Die digitalen Signale werden direkt in Form von Impulsen in das Kabel eingespeist und belegen die gesamte zur Verfügung stehende

Bandbreite des Kabels oder einen Teil davon, wobei der andere Teil nicht mehr für andere Dienste nutzbar ist. Basisband-Systeme bieten also nur einen Kanal. Abbildung 7.1 zeigt den Unterschied zwischen Breitband- und Basisband-Systemen.

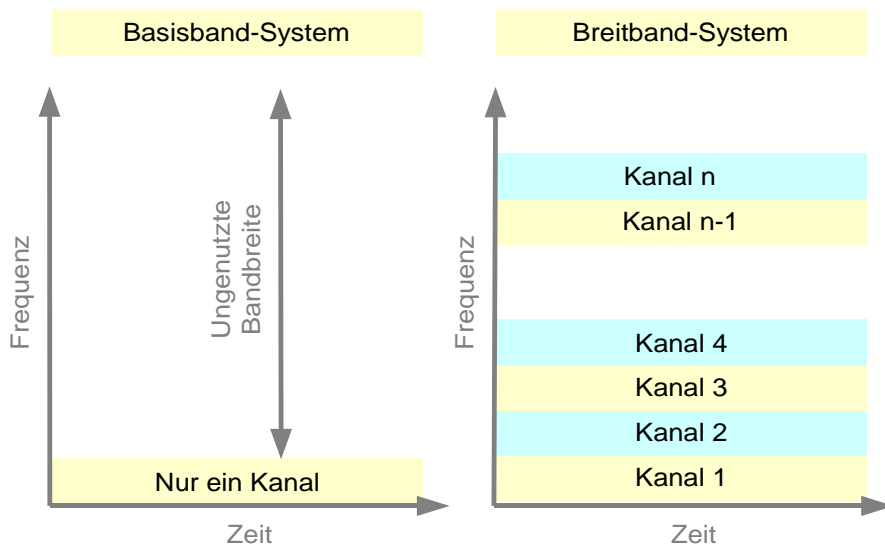


Abbildung 7.1: Vergleich zwischen Basisband- und Breitband-System

Als Beispiel sei hier das in LANs eingesetzte Ethernet erwähnt. Diese Technologie ist eine typische Basisband-Technologie, wird doch im untersten Band nur ein Kanal mit einer Bandbreite von ca. 30 MHz benutzt und die restliche zur Verfügung stehende Bandbreite von 30 MHz bis unendlich bleibt ungenutzt.

7.1.3 Dämpfung a (engl. attenuation)

Die zu übertragenden Signale werden von einem Sender mit einer Leistung P_i in ein Kabel eingespiesen. Am anderen Ende der Leitung empfängt der Empfänger das Signal mit einer kleineren Empfangsleistung P_o . Die Differenz zwischen eingespiessener Leistung und empfangener Leistung bezeichnet man als Verlust. Dieser muss in regelmässigen Abständen durch Verstärker kompensiert werden. Dies ist ein Grund, weshalb mit Kupferleitungen Daten nur über eine gewisse Strecke übermittelt werden können und weshalb Kabel mit möglichst kleiner Dämpfung bevorzugt werden. Der Verlust wird als Dämpfung (a) bezeichnet und kann aus den Leistungen P_i , P_o berechnet werden.

Stehen für die Berechnungen die Spannungen U_i und U_o zur Verfügung, so müssen die Leistungen mit dem Zusammenhang $P = U^2/R$ aus den Spannungen errechnet werden oder der Logarithmus des Quotienten U_i/U_o muss mit 20 multipliziert werden. Die Dämpfung wird mit der Einheit Dezibel (dB) angegeben. Abbildung 7.2 zeigt die

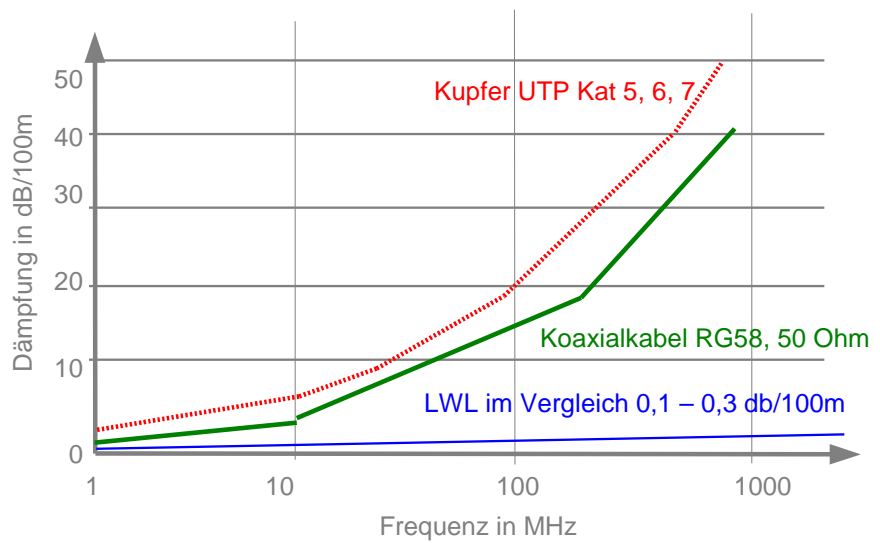


Abbildung 7.2: Dämpfungen verschiedener Übertragungsmedien

Dämpfungen von Unshielded Twisted Pair-Kupferkabeln (UTP), Koaxialkabeln und Lichtwellenleitern (LWL) im Vergleich.

7.1.4 Übersprechen / Nebensprechdämpfung NEXT

(engl. crosstalk)

Mit der Dämpfung verwandt ist die so genannte Nebensprechdämpfung (siehe Abbildung 7.3). Diese Verluste entstehen in nebeneinander liegenden Signalleitungen, indem die Signale der einen Leitung auf die daneben liegende Leitung durch Kopplung (elektromagnetische Übertragung) eingestreuert werden und somit die Datensignale beider Leitungen „gestört“ werden.

Diesen Effekt kann man unter Umständen in extremer Form (Übersprechen) beim Telefonieren erleben, wenn man plötzlich ein Ge-

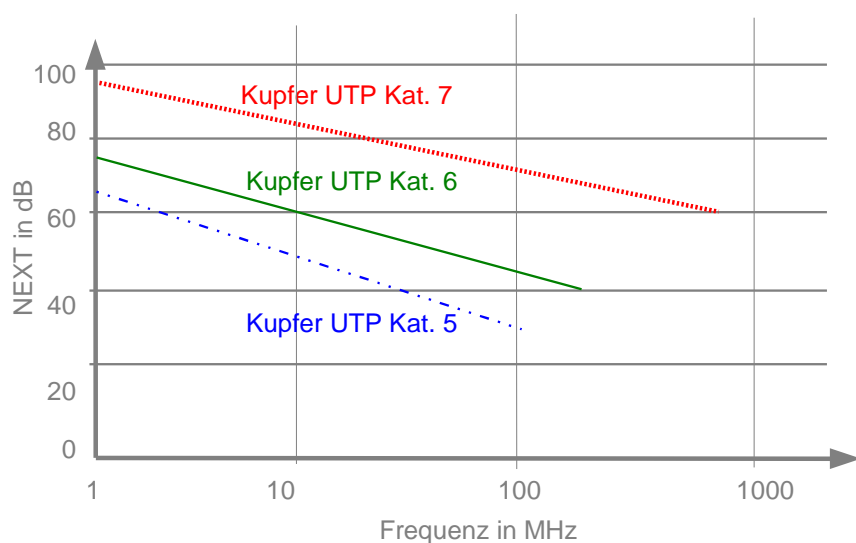


Abbildung 7.3: NEXT für verschiedene verdrehte Kupfer-Kabel

sprach von zwei anderen Telefonierenden im Hintergrund des eigenen Gespräches mithören kann.

Man definiert zwei Arten der Nebensprechdämpfung:

- Nahnebensprechdämpfung, wenn das Nebensprechen nahe beim Sender stattfindet (engl. near-end-crosstalk, NEXT).
- Fernnebensprechdämpfung, wenn das Nebensprechen auf der Seite des Empfängers stattfindet (engl. far-end-crosstalk, FEXT).

In der Praxis muss darauf geachtet werden, dass NEXT möglichst gross ist, das heisst, dass möglichst kein Übersprechen stattfindet.

7.1.5 Dämpfung-Nebensprech-Verhältnis

(engl. attenuation to crosstalk ratio) (ACR)

Das Verhältnis Dämpfung/Nahnebensprechdämpfung, das so genannte ACR (engl. Attenuation to crosstalk ratio), ist ein weiterer wichtiger Wert. ACR ist ein Mass für die qualitative Bewertung einer Übertragungsstrecke und nicht für Kabel.

Der ACR-Wert wird in den Verkabelungsstandards (z.B. ISO/IEC 11801, EIA/TIA 568 und EN 50173) für die Qualitätsbewertung von Ende-zu-Ende-Verbindungen spezifiziert.

7.1.6 Reflexion

Werden in einem Kabel Signale mit einer gewissen Frequenz eingespielen, so laufen diese Signale bis an das Ende des Kabels (wie Wellen auf dem Wasser ans Ufer). Am Ende des Kabels werden sie umkehren (reflektiert) und den nachfolgenden Signalen während des Zurücklaufens überlagert, was im Extremfall zu einer vollständigen Auslöschung des Nutzsignals führen kann.

Damit dies verhindert werden kann, befinden sich bei allen Datenbus-Kabeln so genannte Abschlusswiderstände (Abbildung 7.4).

Sie können dieses Phänomen selber beobachten: Wenn Sie die Wellen eines Sees an einer senkrechten Mauer beobachten, werden Sie feststellen, dass diese Wellen über die nachfolgenden Wellen zurückgeworfen werden und diese in ihrer Ausbreitung stören. Beobachten Sie hingegen Wellen, die auf ein flaches Ufer im Strandbad treffen, so werden Sie sehen, dass diese Wellen auslaufen und somit die nachfolgenden nur unwesentlich stören.



Abbildung 7.4: SCSII-Gerät mit Abschlusswiderstand (siehe Pfeil)

Praxis-Hinweis:

Abschlusswiderstände brauchen unter anderem alle ISDN-Installationen, Koaxial-Ethernet-Bus-LANs und USB-Systeme. Man erkundige sich immer bei Fachleuten, ob die Abschlusswiderstände in einer Installation vorhanden sind, bevor man ein solches System in Betrieb nimmt – instabile oder nicht funktionierende Anlagen wären die Folge von fehlenden Widerständen.

7.1.7 Verzerrungen (engl. distortion), Jitter

Verzerrungen treten dann auf, wenn die Grund-Frequenz der Signale gestört wird. Dies kann besonders bei hohen Bitraten (Bit/s) und schlechten Abtastraten der Fall sein (wie oft tastet der Empfänger das erhaltene Signal ab).

Jitter zum Beispiel ist ein Fehler in der Zeitbasis. Er wird verursacht durch Zeitverschiebungen in den Schaltkreisen der Kommunikations-Komponenten. Die zwei häufigsten Gründe für Jitter sind schlecht implementierte Zeitbasis-Elemente in den Schaltkreisen und Verzerrungen in der Wellenform aufgrund schlecht angepasster Leitungswiderstände, was wiederum zu Reflexionen in den Datenleitungen führt. Abbildung 7.5 veranschaulicht die Verzerrung der Wellenform infolge Veränderung der Zeitbasis.

Die obere Abfolge repräsentiert ein perfektes, in der Zeitbasis einheitliches Digitalsignal des Wertes 010101. Wird dieses perfekte Signal durch eine Leitung falscher Impedanz geleitet, so kommt es zu Verundungen sowie zu Reflexionen, die die exakten Nulldurchgänge „verschmieren“, es entsteht eine Unschärfe. Speziell die Nulldurchgänge sind nicht mehr scharf. Trotzdem repräsentiert das verzerrte Signal denselben Wert 010101. Ab einem bestimmten Mass der Verzerrung sind diese Fehler hörbar, z.B. in Form von Clicks im Musiksignal einer CD, und beruhen nicht automatisch auf Fehlern in der Informationsspur der CD.

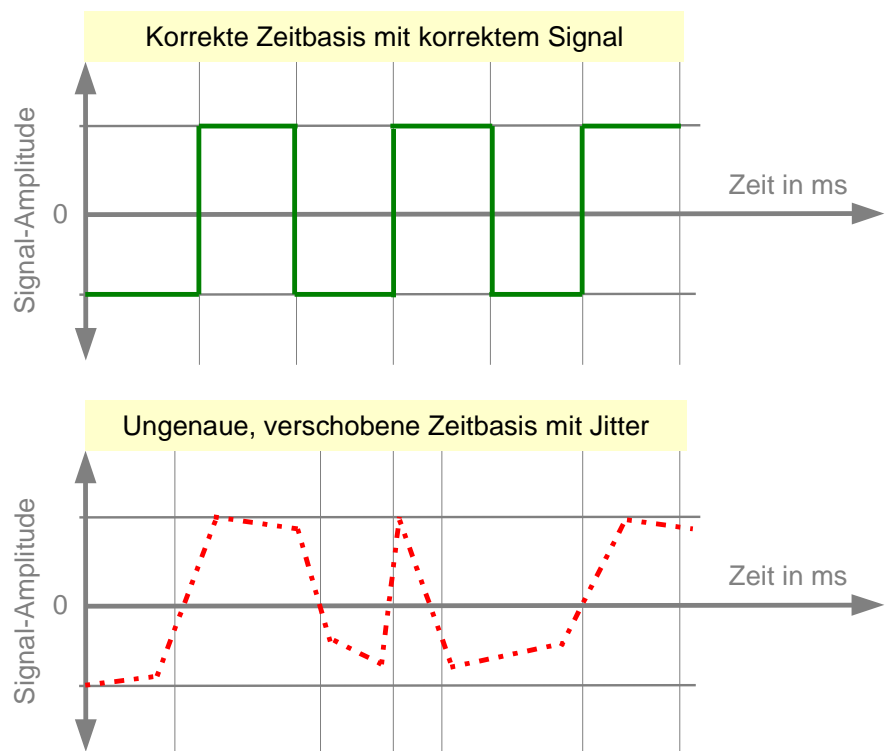


Abbildung 7.5: Jitter infolge einer schlechten Zeitbasis

7.1.8 Physikalische Zusammenhänge bei Lichtwellenleitern

Damit die Wirkungsweise von Lichtwellenleitern verstanden werden kann, müssen einige physikalische Zusammenhänge erklärt werden. Um die digitale Information über Lichtwellenleiter zu verbreiten, müssen die elektrischen Signale zuerst in Puls-Lichtsignale umgewandelt werden.

7.1.8.1 Optische Dichte

Die Dichte eines optischen Mediums bestimmt die Geschwindigkeit, mit der sich das Licht im Medium ausbreiten kann. Das Mass für diese Dichte ist der Brechungsindex. Je höher der Brechungsindex, desto höher ist die optische Dichte des Stoffes. So ist der Brechungsindex in Vakuum = 1 und derjenige in Glas 1,5 bis 1,9.

Die unterschiedliche optische Dichte von Glas und Luft (Vakuum) spielt besonders bei Steckverbindungen eine Rolle.

Ein Lichtstrahl tritt unter einem gewissen Winkel aus dem einen Steckerteil in die Luft zwischen den beiden Steckerteilen aus und wird abgelenkt (gebrochen). Im zweiten Steckerteil wird der Lichtstrahl von der Luft wieder in das Glas eingeleitet und in der anderen Richtung gebrochen (Abbildung 7.6). Qualitativ schlecht verarbeitete Steckverbindungen mit Winkelfehlern können daher grosse Verluste verursachen, wenn der Lichtstrahl derart stark abgelenkt wird, dass er aus dem Stecker austritt.

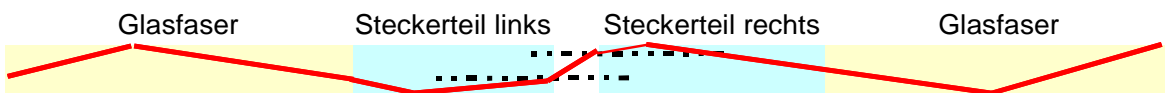


Abbildung 7.6: Der Strahlengang des Lichtes durch eine Steckerverbindung

7.1.8.2 Reflexion

Gelangt ein Lichtstrahl unter einem sehr flachen Winkel an die Grenzfläche zwischen Glas und Luft (oder Kunststoff, dem Material der Glasfaserummantelung), wird der Strahl den LWL nicht verlassen können, weil er total reflektiert wird (Abbildung 7.7).

Diese letzte Eigenschaft der Reflexion wird in Lichtwellenleitern dazu benutzt, dass die Information nicht unterwegs verloren geht. Die ge-

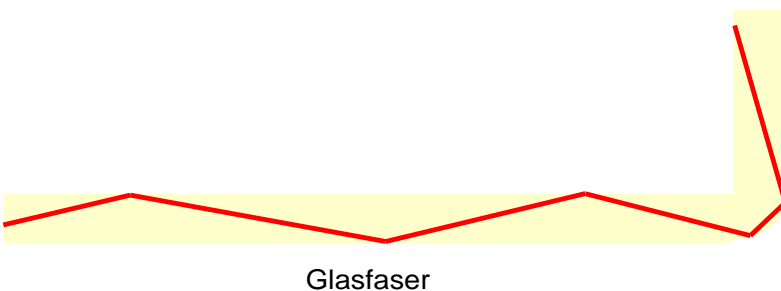


Abbildung 7.7: Totalreflexion in der Glasfaser

pulsten Lichtwellen, welche die Information übertragen, werden an den Grenzflächen der Lichtwellenleiter immer in das Innere der Glasfaser zurückgeworfen. Wird ein LWL zu stark gebogen oder gar geknickt, kann es zu Lichtaustritt kommen und somit zu Verlusten in der Datenübertragung.

7.1.8.3 Dämpfung (Streuung, Absorption, Dispersion)

LWL sind nicht anfällig auf äussere Störungen wie elektromagnetische und elektrische Felder. Feuchtigkeit kann den LWL jedoch schaden, weshalb die Glasfasern mit einem speziellen Gel geschützt sind.

Verluste können hingegen trotzdem entstehen. Man spricht in diesem Zusammenhang von Dämpfung. Für die Dämpfung sind die physikalischen Phänomene Streuung, Absorption und Dispersion verantwortlich:

Streuung

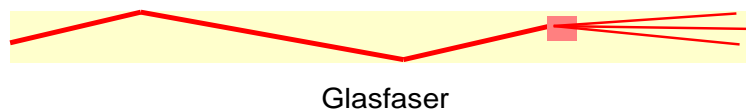


Abbildung 7.8: Streuung in der Faser

An defekten oder verunreinigten Stellen im Glas (vergleiche Abbildung 7.8) kann der eng gebündelte Lichtstrahl gestreut werden. Dabei dürfte klar sein, dass ein gestreuter Strahl nicht mehr die gleich hohe Signalintensität aufweist wie ein gut gebündelter Strahl.

Absorption

Alle Stoffe absorbieren (verschlucken) Licht. Sie tun dies bei unterschiedlichen Lichtwellenlängen unterschiedlich stark (vergleiche Abbildung 7.9). Bei ganz speziellen Wellenlängen wird viel Licht absorbiert und man versucht, diese Wellenlängen bei der Signalübertragung zu vermeiden, indem man so genannte (normierte) Übertragungsfenster definiert⁴⁵.

Moderne Lichtwellenleiter-Herstellungsverfahren zielen darauf ab, den Effekt der Absorption zu minimieren. Durch umfangreiche Forschung konnten diese Herstellungsverfahren so weit optimiert wer-

⁴⁵ Mit zunehmender Wellenlänge nimmt der Streuverlust ab. Verunreinigung, z.B. OH-Ionen, die bei der Faserherstellung in die Faser gelangen, absorbieren das Licht bei verschiedenen Wellenlängen. Bedingt durch die OH-Absorptionsspitzen gibt es Dämpfungsspitzen (bei ca. 950, 1.200 und 1.400 nm) und günstige Wellenlängenbereiche, die auch Fenster oder Arbeitswellenlängenbereiche genannt werden. Folgende Fenster (Wellenlängenbereiche) werden heute zur optischen leistungsgebundenen Signalübertragung mittels LWL-Systemen genutzt: 850 nm, 1300 nm und 1550 nm.

den, dass im Bereich 1200 nm bis 1600 nm eine konstante Dämpfung von 1 dB/km erreicht werden kann (gestrichelte Linie).

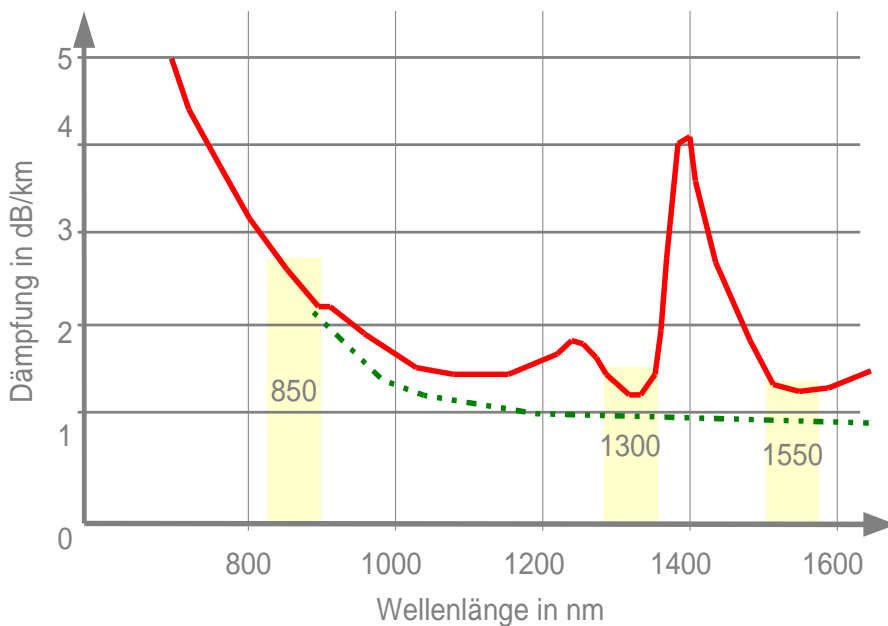


Abbildung 7.9: Die optischen Fenster in der optischen Übertragungstechnik

Dispersion

Jeder Lichtstrahl verbreitert sich mit zunehmender Länge der Übertragungsstrecke. Diese Eigenschaft nennt man Dispersion. Man kann dies mit dem Lichtstrahl einer Taschenlampe leicht nachprüfen. Die Qualität der optischen Fasern wird daher auch durch den Grad der Dispersion bestimmt.

7.2 Verschiedene Übertragungsmedien

Für die Übertragung von Daten in den verschiedenen Formen von Rechnernetzen und insbesondere lokalen Netzen stehen folgende Medien zur Verfügung:

- verdrehte Leitung mit oder ohne Abschirmung (Twisted Pair)
- Koaxialkabel
- Lichtwellenleiter (Glasfaser)
- Luft (Funk).

Für die korrekte Verkabelung in Gebäuden steht eine Europa-Norm (EN) zur Verfügung (EN 50173).

7.2.1 Kupferkabel

Bei den Kupferkabeln werden Niederfrequenzkabel und Hochfrequenzkabel unterschieden.

Praxis-Hinweis:

Man beachte unbedingt die neuesten Kabelkataloge der Hersteller.

7.2.1.1 Niederfrequenzkabel (NF-Kabel)

Man unterscheidet folgende Grundtypen:

1. Shielded Twisted Pair (STP), mit zwei einzeln mit Folie abgeschirmten Paaren, einem Gesamtschirm und einer Wellenimpedanz (Hochfrequenz-Übertragungseigenschaften eines Kabels) von 150 Ohm.
2. Unshielded Twisted Pair (UTP), völlig ohne Abschirmung, mit vier Paaren und einer Wellenimpedanz von 100 Ohm.
3. Screened Unshielded Twisted Pair (S-UTP), bei dem der Gesamtschirm entweder als Folie (St) oder als Folie und Geflecht (St-C) ausgeführt ist. Es gibt zwei oder vier Paare mit einer Wellenimpedanz von 100 Ohm.
4. Screened Shielded Twisted Pair (S-STP), mit zwei oder vier einzeln mit Folie abgeschirmten Paaren, Gesamtschirm als Geflecht und Wellenimpedanz 100 Ohm.
5. Sternvierer, als Besonderheit der Telefonie. Hier sind alle vier Adern gemeinsam mit sich selbst verdrillt. Die Übertragungskapazität ist sehr gering und nur für die Telefonie geeignet.

Die verschiedenen Kabel werden in Klassen von A bis G und Kategorien 1 bis 8 eingeteilt. Generell gilt, je grösser die unverstärkte Übertragungstrecke und je grösser der Datendurchsatz (Bit/s), desto besser muss das Kabel sein (höhere Kategorie). Höhere Kabelkategorien sind auch weniger anfällig auf Störungen von aussen und haben einen kleineren Übertragungswiderstand.

<i>Klasse</i>	<i>Anwendungen</i>	<i>Kategorie</i>	<i>Stand</i>
Class A	Sprach-/Datenverbindungen für niederfrequente Anwendungen bis 100 KHz für Telefon und ISDN	Kat. 1&2	gültig
Class B	Datenverbindungen mit mittleren Datenraten bis 1 MHz für Telefon und ISDN	Kat. 1&2	gültig
Class C	Datenverbindungen bis 16 MHz für Telefon, ISDN, Token Ring, Ethernet	Kat. 3	gültig
Class D	Datenverbindungen bis 100/125 MHz für Telefon, ISDN, Token Ring, Ethernet (GigaBit Ethernet), FDDI, TPDDI, 100 VG Anylan	Kat. 5 (Kat5e)	gültig
Class E	Datenverbindungen bis 250 MHz für Class D plus ATM und GigaBit Ethernet	Kat. 6	gültig
Class F	Datenverbindungen bis 600 MHz	Kat. 7	gültig
Class G	CATV-Anlagen (Video) bis 1200 MHz bei max. 50 m Kabellänge	Kat. 8	In Diskussion

Tabelle 7.2: Die verschiedenen Kategorien und Klassen der Kupferkabel

Neue Verkabelungen sollen, falls sie in Kupfer ausgeführt werden, generell mit Kategorie-5-Kabeln verlegt werden.

Die verschiedenen Kabelarten und ihre physikalischen Eigenschaften findet man in den aktuellen Katalogen.

7.2.1.2 Hochfrequenzkabel (HF-Kabel)

Für Übertragungsraten über 100 MBit/s werden oft Koaxialkabel eingesetzt. Zum Beispiel:

1. Das 50 Ohm RG 58 Koaxialkabel nach der Norm IEEE 802.3 für 10 Base 5 und 10 Base 2
2. Das 75 Ohm RG 59 Koaxialkabel nach der Norm IEEE 802.7 für Breitbandnetzwerke (beispielsweise Kabelfernsehen)
3. Das 93 Ohm RG 62 Koaxialkabel für IBM 3270 Terminalverkabelung (ARCNet).

Das RG 58 Kabel der IEEE-Norm 10 Base 5 (Yellow Cable) ist ca. 10 mm dick und darf beim Verlegen nicht mit Radien kleiner als 250 mm gebogen werden, da sonst die Impedanz der Isolation verändert wird, was zu Störungen in der Datenübertragung führen kann.

Weil das yellow cable relativ schwer ist, existiert noch das so genannte RG 58 (Thin Wire) Kabel. Dieses Kabel hat eine Impedanz von 50 Ohm. Der Durchmesser ist etwa 5 mm und sein minimaler Biegeradius beträgt 80 mm (Reflexionsgefahr).

7.2.2 Lichtwellenleiter (LWL)

Lichtwellenleiter gewinnen sehr rasch an Bedeutung. Für universelle Gebäudeverkabelungen sind diese Übertragungsmedien nicht mehr wegzudenken, weil sie eine sehr grosse Störsicherheit gewährleisten. Lichtwellenleiter bestehen im Prinzip aus einem sehr dünnen Glasfaserkern aus hochreinem Quarzglas mit Dotierstoffen. Dieser Kern ist umgeben mit einem optischen Mantel aus Quarzglas. Die gesamte Faser ist mit einer Acrylatbeschichtung (Aussenhülle) versehen. Diese Fasern werden zu Kabeln mit einer, zwei oder mehreren Fasern verbunden. Auch kombinierte Glas/Kupferkabel existieren.

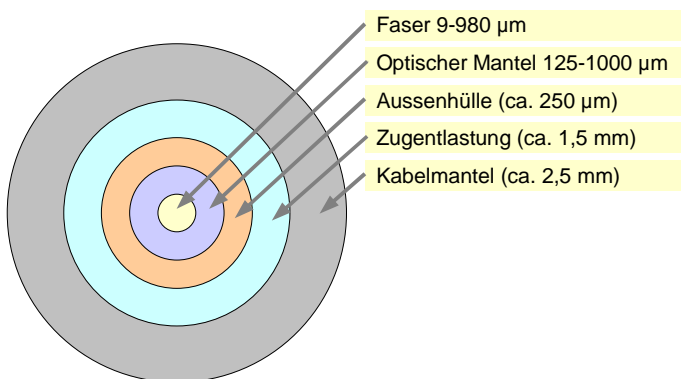


Abbildung 7.10: Der prinzipielle Aufbau eines LWL-Kabels

Es werden vier Grundaufbauten für das LWL-Kabel unterschieden:

- Festader oder auch Vollader
- Kompaktader
- Hohlader gefüllt oder ungefüllt
- Bündelader gefüllt oder ungefüllt.

7.2.2.1 Festader / Vollader

Die Festader (auch Vollader genannt) besteht aus einer Glasfaser und einer sie fest umgebenden Hülle.

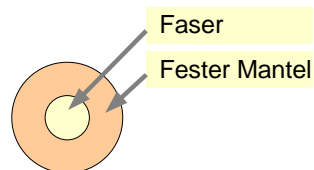


Abbildung 7.11: Festader

7.2.2.2 Hohlader gefüllt

Die gefüllte Hohlader besteht aus einer Glasfaser und einer lose umgebenden Schutzhülle, wobei der Zwischenraum zwischen Glasfaser und Hülle mit einem wasserabweisenden Gel gefüllt ist. Diese Faser ist zwar von den Ausmassen grösser als eine Festader, hat aber meist bessere Eigenschaften bezüglich wirkender Kräfte auf die Hülle, z.B. Temperaturschwankungen und Zugkräfte. Das Füllmaterial schützt u.a. auch vor Längswasser, Querwasser und Druck.

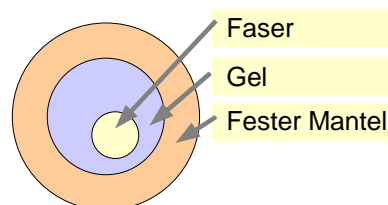


Abbildung 7.12: Gefüllte Hohlader

7.2.2.3 Hohlader ungefüllt

Die ungefüllte Hohlader ist eine Ader mit einer losen umgebenden Hülle um die Glasfaser und ohne Füllmaterial zwischen Faser und festem Mantel.

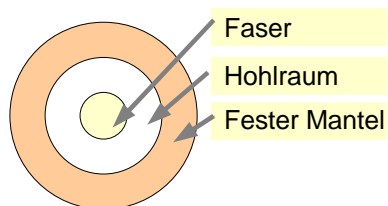


Abbildung 7.13: Ungefüllte Hohlader

7.2.2.4 Kompaktader

Die Kompaktader ist vom Aufbau eine Mischung zwischen der Festader und der Hohlader mit dem Unterschied, dass die Schutzhülle nicht fest, sondern lose um die Glasfaser liegt.

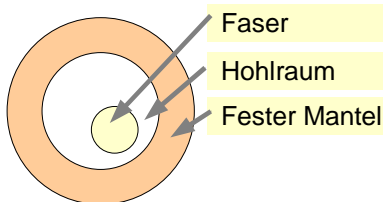


Abbildung 7.14: Kompaktader

7.2.2.5 Bündelader gefüllt

Die gefüllte Bündelader besteht aus mehreren Fasern mit einer gemeinsamen Schutzhülle, wobei auch hier der Zwischenraum mit einem wasserabweisenden Gel gefüllt ist. In der Regel werden zwei bis zwölf Fasern kräftefrei gebündelt. Zur Unterscheidung der Lichtwellenleiter sind die Fasern farblich unterschiedlich.

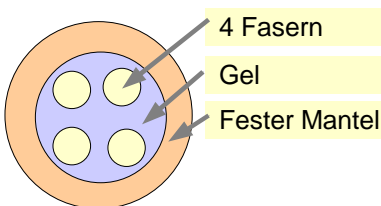


Abbildung 7.15: Gefüllte Bündelader

7.2.2.6 Bündelader ungefüllt

Bei der ungefüllten Bündelader ist der Zwischenraum zwischen den Fasern und der umgebenden Schutzhülle nicht mit Gel gefüllt.

7.2.2.7 LWL-Kabel

Im weiteren Verlauf bei der Herstellung eines LWL-Kabels werden eine oder mehrere Adern (Voll-, Kompakt-, Hohl- oder Bündelader) und eventuell Blindelemente mit einem Stützelement und einer Zugentlastung in einem Kabelmantel verseilt. Die Verseilungshohlräume sind meistens zum Schutz vor Längswasser mit einem wasserabweisenden Gel gefüllt. Das Stützelement ist ein Element, das in axialer Richtung Zug- und/oder Stauchkräfte aufnehmen kann. Dieses Element befindet sich üblicherweise in der Kabelmitte und besteht meistens aus einem dielektrischen Epoxy-Glasfiberstab.

Blindelemente werden eingesetzt, falls die Anzahl der Adern nicht aufgeht, um das Stützelement zentral in der Kabelseele zu installieren.

7.2.3 Typen von LWL

Für den Einsatz in optischen Übertragungssystemen gibt es heute drei Typen von Glasfasern:

- Multimode-Stufenindexfaser
- Multimode-Gradientenindexfaser
- Singlemode / Monomode-Stufenindexfaser.

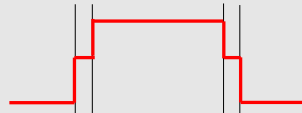
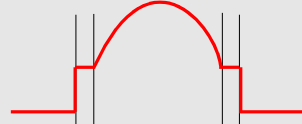
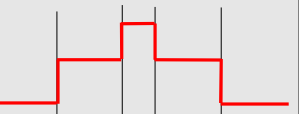
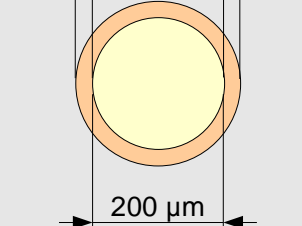
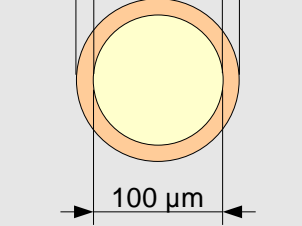
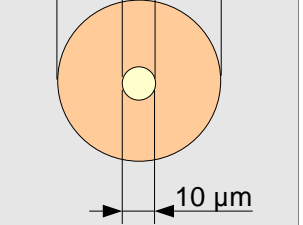
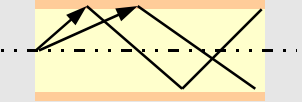
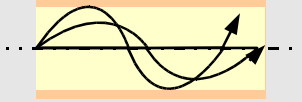

Art	Multimode		Monomode
	Stufenprofil	Gradientenprofil	Stufenprofil
Verlauf der Brechzahl			
Querschnitt			
Strahlenverlauf			
Modendispersion	50 ns/km	1 ns/km	0
Dämpfung bei 850 nm	5 bis 30 dB/km	3 bis 10 dB/km	2 bis 5 dB/km
Strahlenführung	Totalreflexion	Brechung	Wellenführung

Tabelle 7.3: Multimode- und Monomode-Fasern mit ihren Eigenschaften

7.2.3.1 Multimodefaser (Stufen- und Gradientenfaser)

(engl. multimode fiber)

Bei diesem Lichtwellenleiter (siehe Tabelle 7.3) werden mehrere diskrete⁴⁶ Moden⁴⁷ zur Signalübertragung benutzt, d.h., die Lichtstrahlen

⁴⁶ Diskretheit (von lat. discretus = unterschieden, getrennt) bezeichnet allgemein eine räumliche oder zeitliche Trennung von Objekten oder Ereignissen. Diskretheit ist nicht zu verwechseln mit Diskretion, der Geheimhaltung von Wissen.

Ein diskretes Signal besteht aus zeitlich oder räumlich getrennten Teilen, zum Beispiel sind Rauchzeichen und Morsezeichen diskret.

⁴⁷ Moden sind die diskreten Wellen, die zur Signalübertragung benutzt werden. Bei Monomode- oder Singlemodefasern breitet sich nur eine Welle aus. Moden sind Eigenwellen (Licht-Eigenschwingungen) im LWL.

werden an der Grenzschicht zwischen Kern und Mantel häufig und unterschiedlich reflektiert, was unterschiedliche Laufzeiten der Strahlen bedingt. Die Multimodefaser ist entweder eine *Stufenindex-Profilfaser* mit einem typischen Kerndurchmesser von 100, 120 oder 400 μm , mit einem Bandbreitenlängenprodukt von weniger als 100 MHz x km und einer Dämpfung von ca. 6 dB /km oder eine *Gradientenindex-Profilfaser* mit typischen Kerndurchmessern von 50 μm , 62,5 μm , 85 μm oder 100 μm und Manteldurchmessern von 125 μm oder 140 μm . Die Dämpfungswerte liegen bei 3 dB/km (LED⁴⁸ 850 nm), wodurch eine repeaterlose Übertragung von bis zu 10 km möglich ist. Das Bandbreitenlängenprodukt liegt hier wegen der besseren Unterdrückung der Modendispersion zwischen 200 MHz x km bei 850 nm und 500 MHz x km bei 1.300 nm.

7.2.3.2 Monomodefaser (Stufenprofilfaser)

(Engl. monomode fiber)

Die Monomodefaser ist ein Lichtwellenleiter mit Stufenindex-Profil, bei dem durch einen sehr kleinen Kerndurchmesser, der bei 8 bis 10 μm liegt, das Licht praktisch nur in einem Mode, der quasi parallel zur Achse verläuft, übertragen wird (siehe Tabelle 7.3).

Die Monomodefaser zeichnet sich dadurch aus, dass sie praktisch keine Laufzeitunterschiede aufweist (Modendispersion 0,1 ns/km), da das Licht ja nur in einer Ausbreitungsrichtung den Lichtwellenleiter durchläuft, das Impulsverhalten dadurch formgetreu ist und dass sie über die geringsten Dämpfungswerte aller Lichtwellenleiter verfügt. Dies drückt sich in einer Dämpfung von 0,1 dB /km (LED 1300 nm), einem Bandbreitenlängenprodukt von >10 GHz x km und einem Bitratenlängenprodukt von 250 GHz x km aus. Es können Entfernungen bis zu 50 km ohne Repeater überbrückt werden. Die Faser wird mithilfe von speziellen Pump-Lasern⁴⁹ für Übertragungsstrecken über 5000 km eingesetzt (Transatlantikstrecken). Der Manteldurchmesser der Monomodefaser liegt typischerweise bei 125 μm , der Kerndurchmesser typischerweise bei 10 μm .

⁴⁸ LED: Light Emitting Diode, eine Diode, die gepulstes Licht aussenden kann.

⁴⁹ Die Raman-Verstärkung [in Raman-Pump-Lasern] basiert auf dem Raman-Effekt, einem nichtlinearen Effekt in der Glasfaser, bei dem Photonen kürzerer Wellenlänge ihre Energie an langwelligere Photonen abgeben und die Verstärkung bewirken. Zur Raman-Verstärkung sind starke Pumplaser im Watt-Bereich erforderlich, aber keine spezielle Glasfaser als „Verstärkungs-Medium“. Das Maximum der Verstärkung liegt etwa 100 nm oberhalb der Pumpwellenlänge. Durch geschickte Anordnung mehrerer Pumpen kann ein sehr breites Band mit WDM-Kanälen verstärkt werden, das prinzipiell auf den gesamten Übertragungsbereich einer SMF von 1,3 μm bis 1,6 μm ausgedehnt werden kann. Bei der so genannten „Verteilten Raman-Verstärkung“ erfolgt die Verstärkung entlang der Übertragungsfaser, sodass keine diskreten Streckenverstärker notwendig sind. (Erklärung Fraunhofer Institut Nachrichtentechnik, Heinrich Herz-Institut)

Die folgende Tabelle 7.4 fasst einige typischen Eigenschaften der Lichtwellenleiter zusammen:

Fasertyp	Multimode (Gradientenindexprofil)		Monomode (Stufenindexprofil)
	50/125	62.5/125	9/125
Optische Daten			
Dämpfung (dB) (bei 1300 nm)	< 0.8	< 0.7	< 0.38
Bandbreite (MHz * km)	> 800	> 600	-
Dispersion (ps/[nm * km])	-	-	< 3.50
num. Apertur	0.20 ± 0.015	0.275 ± 0.015	13 ± 0.015
Physikalische Daten			
Kern-Ø (mm)	50 ± 3	62.5 ± 3	8.3
Mantel-Ø (mm)	125 ± 2	125 ± 2	125 ± 2
Coating-Ø (mm)	250 ± 15	250 ± 15	245 ± 10
Brechungsindex			
Kern n1	1.46	1.47	1.470
Mantel n2	1.45	1.46	1.456
Prüflast/Prüfdauer [%]	1.0 / sec (100 kpsi)	1.0 / sec (100 kpsi)	1.0 / sec (100 kpsi)
Faserbezeichnung	G 50/125	G 62.5/125	E 9/125

Tabelle 7.4: Eigenschaften der Lichtwellenleiter

7.2.4 Vergleich LWL – Kupferkabel

Die folgende Tabelle 7.5 zeigt einige vergleichende Eigenschaften der beiden Übertragungsmedien LWL und Kupferkabel.

Kupfer	LWL
Leichte Verlegbarkeit	Grosse Übertragungsbandbreite
Geringe Konfektionskosten	Niedrige Signaldämpfung
Günstige Arbeitsplatzausstattung	Kein Nebensprechen
Kostengünstige Montage	Keine Beeinflussung durch äussere elektrische Störfelder
Einfach integrierbar in bestehende Netztopologien	Schutz vor Potentialübertragung (Blitzeinschlag)
Montagefreundlich	Einsetzbar im explosionsgefährdeten Umfeld
	In vielen Fällen bessere Wirtschaftlichkeit
	Glasfaser ist leicht, dünn, flexibel
	Abhörsicher

Tabelle 7.5: Vergleich Kupfer / LWL - Kabel

Abbildung 7.16 zeigt die verschiedenen Übertragungsmedien im Vergleich. Es zeigt sich, dass die Monomodefaser eindeutig für grosse Distanzen und hohe Übertragungsraten geeignet ist. Multimodefasern und Kupferkabel werden bei kleineren Distanzen eingesetzt. Die Begrenzungen durch Dämpfung und Dispersion können heute durch geeignete Verstärkungsmassnahmen und spezielle Laser hinausgeschoben werden.

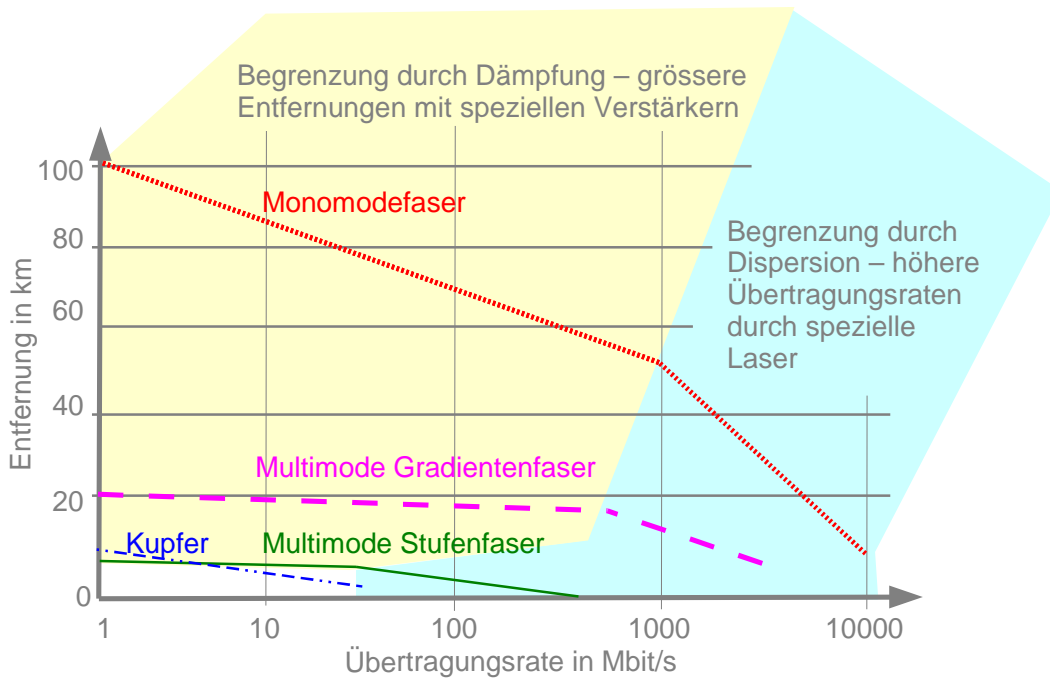


Abbildung 7.16: Verschiedene Übertragungsmedien im Vergleich

7.3 Verwendete Stecker und Verbindungen

An dieser Stelle sollen nur einige wenige Anmerkungen zu den im Einsatz stehenden Steckern für Kupferkabel und Lichtwellenleitern gemacht werden. Weitere Hinweise finden sich in der einschlägigen Literatur und den Katalogen der Hersteller.

7.3.1 Stecker bei UTP, STP und Koaxialkabeln

Es kommen die folgenden Stecker zur Anwendung:

- UTP-Kabel werden mit RJ45-Steckern (8-polige US-Telefonstecker) versehen.
- STP-Kabel werden mit dem IBM-Datenstecker versehen (manchmal auch RJ45)
- Koaxialkabel werden mit BNC- (Bayonet Nut Coupling) Stecker versehen.
- In der Telefonie wird in der Schweiz der CH-Stecker TT83 vermehrt durch RJ45 abgelöst.

7.3.2 Stecker bei LWL



Abbildung 7.17: BNC und RJ45-Stecker

Die Verbindung von Lichtwellenleitern ist ein heikles Kapitel. Eine gute Verbindung wird nur mit speziellen Apparaturen und Steckern erreicht. Generell gilt, dass jede Verbindung wegen dem Übergang Glas-Luft mit einem Verlust behaftet ist (0.3 bis 0.4 dB). Auch Fehler wie radialer, axialer und Winkel-Versatz treten bei unsachgemäss ausgeführten Verbindungsstellen auf und verursachen grosse Fehler und Verlustquellen.



Abbildung 7.18: SC, ST, E2000 und LWL-BNC-Stecker

Moderne Steckverbindungen für LWL enthalten zur Verbesserung der Kopplung und zur Verminderung von Verlusten ein Gel.

7.4 Normierung von Übertragungsmedien in Netzen

An dieser Stelle sollen einige Anmerkungen zu den gebräuchlichen Normenbezeichnungen für LAN-Kabel (Ethernet) und LWL gemacht werden. Ausführlichere Darstellungen finden sich in der einschlägigen Normen-Literatur und dem Internet.

In LANs gilt folgende Normbezeichnung: **nn Base k**, wobei „nn“ für die Übertragungsrate in MBit/s, „Base“ für den Begriff Basisband und „k“ für die maximale Segmentlänge in 100 m oder die Kabelart steht. In der Praxis wird für nn die Übertragungsrate in MBit/s angegeben. Dies ist jedoch ungeschickt, da die effektive Nutzdaten-Übertragungsrate viel kleiner ist als die theoretische Übertragungsrate und dies bei der Berechnung der Leistungsfähigkeit der Netze zu Missverständnissen führt.

In der Praxis erreichbare Übertragungsraten auf einem 100 MBit/s-Basisband bewegen sich somit zwischen 25 bis 50 MBit/s. In Bytes

ausgedrückt ergibt dies (8 Bit pro Byte angenommen): ca. 3000 bis 5500 kByte/s.

Beispiele für Hochfrequenz-Koaxialkabel (Bus-Topologie)

- 10 Base 5: 10 MBit/s Basisband, 500 m (100 Knoten/Segment) (IEEE 802.3⁵⁰)
- 10 Base 2: 10 MBit/s Basisband, 200 m (30 Knoten/Segment) (IEEE 802.3a)

Beispiele für Twisted Pair-Kabel (Stern-Topologie)

- 10 Base T: 10 MBit/s Basisband, Twisted Pair, 100 m (1024 Knoten/Segment) (IEEE 802.3i)
- 100 Base TX: 100 MBit/s Basisband, Twisted Pair, 100 m (IEEE 802.3u)

Beispiel für LWL (Stern-Topologie)

- 100 Base FX: 100 MBit/s Basisband, Fibre, 2000 m (1024 Knoten/Segment) (802.3u)
- 1000 Base LX: 1000 MBit/s Basisband, Fibre, 5000 m (802.3z)
- (Knoten = angeschlossene Stationen, PCs)

7.4.1 Spezielle LWL-Normen

Lichtwellenleiter-Kabel erfüllen eine oder mehrere der folgenden Normen:

DIN VDE 0888

DIN VDE 0899

DIN VDE 0472

DIN VDE 0473

EN 50 173

EN 187 000

EN 188 000

CCITT rec G651 bis G654

IEC 60793 bis 60794

7.5 Überlegungen zur physikalischen Verbindung

Mit der physikalischen Verbindung (Verkabelung) von WAN und GAN können sich aus Kostengründen nur grosse, international tätige Firmen befassen. Beim Aufbau eines LAN wird jede Firma auf das Problem der wirtschaftlich sinnvollen Verkabelung stossen. Möchte man eine Verkabelung realisieren, müssen einige Überlegungen in die Planung einfließen.

- Wenn man grössere Probleme im LAN ausschliessen will, dann darf an der Verkabelung auf keinen Fall gespart werden.

⁵⁰ IEEE: Institute of Electrical and Electrical Engineers, des Verbandes der Amerikanischen Elektro- und Elektronik-Ingenieure. Diese haben Normen herausgegeben, die alle unter der IEEE 800-Norm angesiedelt sind und die verschiedenen Übertragungsmedien normieren.

- Eine gute Verkabelung kostet einiges und ist auf lange Sicht trotzdem wirtschaftlich vertretbar.
- Verkabelungen sind nach wie vor abhörsicherer als drahtlose Übertragungen (siehe Wireless LAN).

Man unterscheidet zwei verschiedene Arten von Verkabelungen:

1. Sehr oft werden in einer kleinen Firma vorerst nur einige wenige Computer zu einer Arbeitsgruppe (Workgroup) zusammengeschlossen. Wächst die Firma, so wächst auch das Netz. Diese Art der Verkabelung wird *Bedarfsverkabelung* genannt und ist sehr häufig anzutreffen.
2. Die zweite Möglichkeit wäre die *Vollverkabelung*. Bei dieser Art der Verkabelung werden ganze Gelände, Gebäude, Etagen und Räume gezielt und gut geplant verkabelt. Diese Verkabelungen dienen dann nicht nur der Übertragung von Computerdaten, sondern auch der Telefonie und der Bilddaten- oder Videoübertragung.

Die Verkabelung eines Gebäudes bedingt ein geplantes Vorgehen und wird von Spezialisten ausgeführt. Welche Vorgehensweise und welche Planungshilfen dabei verwendet werden, spielt dabei keine Rolle. Die eingesetzte Methode sollte aber mindestens die folgenden Phasen aufweisen:

1. Erfassen und Dokumentieren des Ist-Zustandes
 - Situation im Gebäude
 - Standorte von Server, Workstations und Hubs (Pläne)
 - Vorhandene Kabelkanäle
 - Dimension des Netzes (Kabellängen)
 - Elektromagnetische und andere Arten von Störungen
 - Wie kann die Anlage geerdet werden?
 - Wie kann man Störungen von aussen verhindern?
 - Art der Übertragungsmedien
 - Glas
 - Kupfer, Koaxial
 - Kupfer Twisted Pair (mit und ohne Abschirmung)
 - Funk
 - Satellit
 - Aufbau der Verkabelung
 - Physikalische Topologie
 - Flexibilität, Ausbaubarkeit
 - Bandbreite der Datenübertragung aufgrund des Mengengerüsts
2. Erfassen der Benutzerbedürfnisse
3. Vorschlagen von Soll-Zustands-Varianten und Auswahl einer Ausführungsvariante
4. Planung der Umsetzung (mit Budget, Zeitplan, Ressourcenplan)

Praxis-Hinweis:

Für die Verkabelung von Gebäuden und für den Aufbau von Netzen existieren Normen.

5. Detailplanung (konkrete Arbeitsschritte)
6. Tests (eine Verkabelung, die nicht ausgemessen ist, birgt ein grosses Betriebsrisiko)
7. Dokumentation
8. Organisation des Betriebes des Netzes.

7.6 Aufgaben

1. Erklären Sie den Unterschied zwischen Bandbreite und Übertragungsgeschwindigkeit.
2. Warum bieten Basisband-Systeme nur einen Kanal?
3. Was sagt die Abbildung 7.3 aus?
4. Wie kann Jitter verursacht werden?
5. Erklären Sie den Vorteil der Qualitätsnorm LWPF anhand der Abbildung 7.9.
6. Was versteht man unter Raman-Verstärkung und was unter verteilter Raman-Verstärkung?
7. Wo werden Monomode-LWL eingesetzt?
8. Welches sind die Vorteile von LWL gegenüber Kupferkabeln?

Lösungen unter www.sauerlaender.ch/downloads

8 Signale und Bitströme

Nachrichten werden zwischen zwei Kommunikationsteilnehmern auf Übertragungsmedien mithilfe von Signalen ausgetauscht. In einigen Fällen wird die Nachricht mit analogen Signalen erfasst, so zum Beispiel beim Telefon, wo die Sprache von einem Mikrofon in elektrische Spannungen umgewandelt wird. Andere Nachrichten liegen in digitaler Form vor, wie zum Beispiel Nachrichten in Computern. In einer zunehmend von Computern dominierten Kommunikationstechnik macht es Sinn, möglichst alle Daten in digitaler Form, als Bitströme, verarbeiten zu können. Auf den Übertragungsmedien müssen diese Daten hingegen in analoger Form, als Signale, vorliegen. Die Umwandlung von analogen Signalen in digitale Daten und umgekehrt spielt somit in der Kommunikationstechnik eine zentrale Rolle.



8.1 Umwandlung analoger Signale in digitale Daten

Analoge Signale werden mittels Analog/Digital-Wandlern in digitale Daten umgewandelt.

Die Digitalisierung der analogen Signale erfolgt durch periodisches Messen der Amplituden⁵¹ und anschliessendem Umwandeln dieser Messwerte in binäre Werte. Wie die Abbildung 8.1 zeigt, wird beim

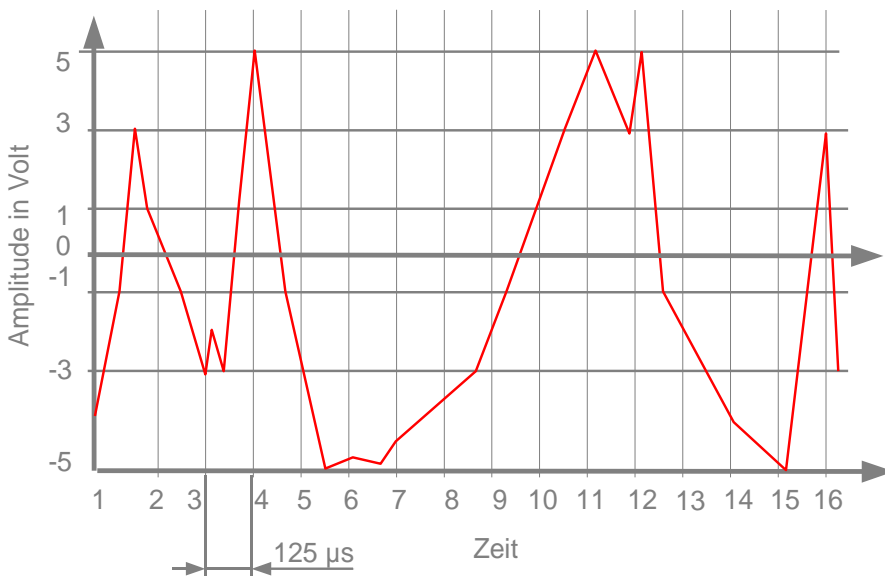


Abbildung 8.1: Das analoge Signal

⁵¹ Das periodische Messen der Amplituden wird im Englischen mit *sampling* bezeichnet. Das *Sampling* erfolgt mit einer bestimmten Abtastrate (*sampling rate*), die etwa der doppelten Signalfrequenz entspricht. Bei einer Bandbreite der Telefonie von ca. 3000 Hz sind somit 8000 Hz als Abtastrate genügend. Zu kleine Abtastraten verursachen eine schlechte Qualität der digitalen Daten und zu grosse Abtastraten liefern zu viele Daten, die unnötig Speicherplatz und Übertragungsbandbreite benötigen.

ISDN-Netz die Messung des Analogsignals im 8-kHz-Takt vorgenommen, das heisst alle 125 µs entsteht ein Messwert. Das Mikrofon im Telefonhörer soll im dargestellten Fall ein analoges Signal zwischen -5 und +5 Volt erzeugen.

Die Umwandlung der Messwerte in digitale Werte geschieht mit einer bestimmten Bit-Breite, beim ISDN beträgt diese 8 Bit. Der dezimale Wertebereich, der mit diesen 8 Bit dargestellt werden kann, ist somit von 0 bis 255 und soll den -5 bis +5 Volt entsprechen.

Damit kann eine Auflösung von $10V / 255 = 0,03921$ Volt erreicht werden.

Tabelle 8.1 zeigt die ermittelten Messwerte in Volt, deren digitalen Dezimalwerte und deren binären Werte.

Messpunkt	Messwert in Volt	Berechnung des dezimalen Wertes	Dezimaler Wert	Binäre Werte (8Bit)
1	- 4	$-4 - (-5) / 0,03921 =$	26	0001 1010
2	0	$0 - (-5) / 0,03921 =$	128	1000 0000
3	- 3	$-3 - (-5) / 0,03921 =$	51	0011 0011
4	+ 5	$+5 - (-5) / 0,03921 =$	255	1111 1111
5	- 3	$-3 - (-5) / 0,03921 =$	51	0011 0011
6	- 5	$-5 - (-5) / 0,03921 =$	0	0000 0000
7	- 4,5	$-4,5 - (-5) / 0,03921 =$	13	0000 1101
8	- 3,5	$-3,5 - (-5) / 0,03921 =$	38	0010 0110
9	- 1,5	$-1,5 - (-5) / 0,03921 =$	89	0101 1001
10	+ 1	$+1 - (-5) / 0,03921 =$	153	1001 1001
11	+ 4,5	$+4,5 - (-5) / 0,03921 =$	242	1111 0010
12	+ 4	$+4 - (-5) / 0,03921 =$	230	1110 0110
13	- 2	$-2 - (-5) / 0,03921 =$	77	0100 1101
14	- 4	$-4 - (-5) / 0,03921 =$	26	0001 1010
15	- 5	$-5 - (-5) / 0,03921 =$	0	0001 0000
16	+ 3	$+3 - (-5) / 0,03921 =$	204	1100 1100

Tabelle 8.1: Die Wertetabelle für die Messwerte aus Abbildung 8.1

Die binären Werte, die alle zuvor ermittelten Messresultate repräsentieren, werden anschliessend als Bitstrom übertragen, indem dieser als elektrisches Signal codiert und durch die Übertragungsleitungen geschickt wird (Abbildung 8.2⁵²).

⁵² ISDN benutzt einen anderen Leitungscode als der hier dargestellte. Aus Gründen der Verständlichkeit wurde ein einfacher Leitungscode benutzt. Die richtigen ISDN-Leitungscode werden weiter hinten abgebildet.

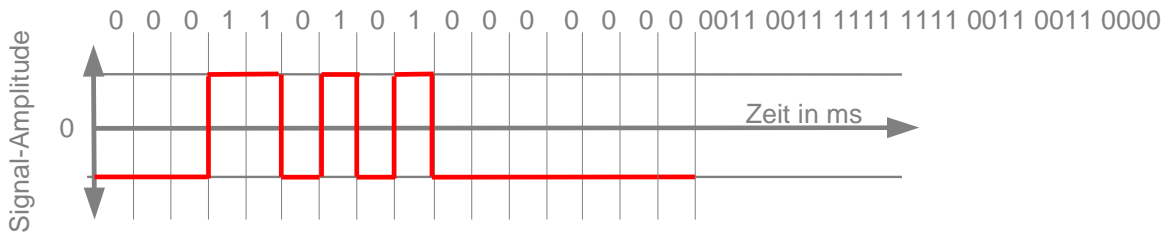


Abbildung 8.2: Leitungscodierung des Bitstromes

Für die Rückwandlung ins analoge Signal wird das elektrisch codierte Bitmuster vom Empfänger ausgelesen und in Dezimalwerte und letztlich in Spannungswerte zurückgewandelt.

Dies führt beim Empfänger zur folgenden Tabelle 8.2 (die Spannungswerte wurden wieder auf -5 bis +5 Volt normiert):

Binäre Werte (8Bit)	Dezimaler Wert	Berechnung des Spannungswertes	Spannungswerte in Volt	Stützstelle
0001 1010	26	$0,03921 \times 26 = 1$	$1 - 5 = - 4$	1
1000 0000	128	$0,03921 \times 128 = 5$	$5 - 5 = 0$	2
0011 0011	51	$0,03921 \times 51 = 2$	$2 - 5 = - 3$	3
1111 1111	255	$0,03921 \times 255 = 10$	$10 - 5 = + 5$	4
0011 0011	51	$0,03921 \times 51 = 2$	$2 - 5 = - 3$	5
0000 0000	0	$0,03921 \times 0 = 0$	$0 - 5 = - 5$	6
0000 1101	13	$0,03921 \times 13 = 0,5$	$0,5 - 5 = - 4,5$	7
0010 0110	38	$0,03921 \times 38 = 1,5$	$1,5 - 5 = - 3,5$	8
0101 1001	89	$0,03921 \times 89 = 3,5$	$3,5 - 5 = - 1,5$	9
1001 1001	153	$0,03921 \times 153 = 6$	$6 - 5 = + 1$	10
1111 0010	242	$0,03921 \times 242 = 9,5$	$9,5 - 5 = + 4,5$	11
1110 0110	230	$0,03921 \times 230 = 9$	$9 - 5 = + 4$	12
0100 1101	77	$0,03921 \times 77 = 3$	$3 - 5 = - 2$	13
0001 1010	26	$0,03921 \times 26 = 1$	$1 - 5 = - 4$	14
0001 0000	0	$0,03921 \times 0 = 0$	$0 - 5 = - 5$	15
1100 1100	204	$0,03921 \times 204 = 8$	$8 - 5 = + 3$	16

Tabelle 8.2: Wertetabelle auf der Seite des Empfängers

Die ermittelten Spannungswerte werden an den Stützstellen (im Abstand von $125 \mu\text{s}$) als Pulse dargestellt. Diese Pulsfolge (Pfeile in Abbildung 8.3) wird durch einen Tiefpassfilter gesandt und es entsteht wieder ein hörbares analoges Signal (siehe durchgezogene Linie in Abbildung 8.3 - als Vergleich ist das ursprüngliche Signal gestrichelt eingezeichnet).

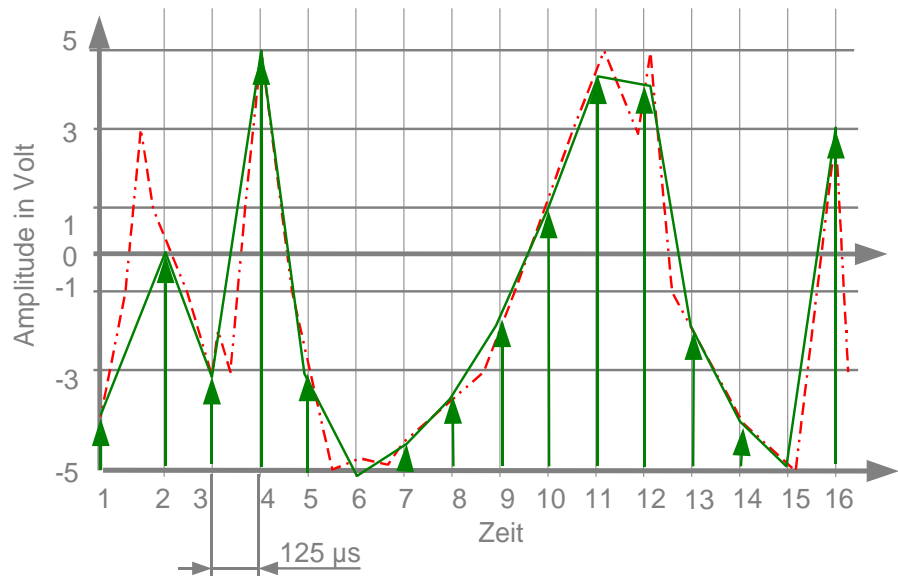


Abbildung 8.3: Ein analoges Signal entsteht aus einem Bitstrom

Wie man unschwer erkennen kann, besitzt ein zurückgewandeltes Signal nicht mehr die genau gleiche Qualität wie das ursprüngliche Signal – einige Spitzen sind abgeschnitten. Diese Qualitätsminderung ist jedoch weder am Telefon, noch auf digitalen Tonträgern wie Musik-CD oder DVD zu hören oder zu sehen – unser Ohr oder unser Auge gleicht diese Ungenauigkeiten wieder aus.

8.2 Transport von Bitströmen

Damit digitale Bitströme transportiert werden können, werden diese über mehrere Stufen codiert, bis daraus Basisband-Signale entstehen. Anschliessend können diese Basisband-Signale moduliert über analoge Leitungen übertragen werden.

Eine vollständige Übertragung einer Nachricht bedingt somit je nach Übertragungsstrecke unterschiedliche Nachrichten- respektive Signal-Aufbereitungsmethoden.

Meistens werden die Nachrichten zuerst quellencodiert, anschliessend kanalcodiert und zum Schluss leitungscodiert. Dies liefert ein Basisbandsignal, das anschliessend mithilfe von Modulationen auf Kanälen oder Leitungen von analogen Übertragungssystemen übertragen werden kann. In lokalen Netzwerken (LAN) werden die Signale oft in Form von Basisband-Signalen übertragen – es steht dabei nur ein Basisband zur Verfügung (vergleiche Ethernet).

Alle diese Codier- und Modulierverfahren werden im Folgenden dargestellt und erklärt. Abbildung 8.4 zeigt eine Übersicht über die Möglichkeiten.

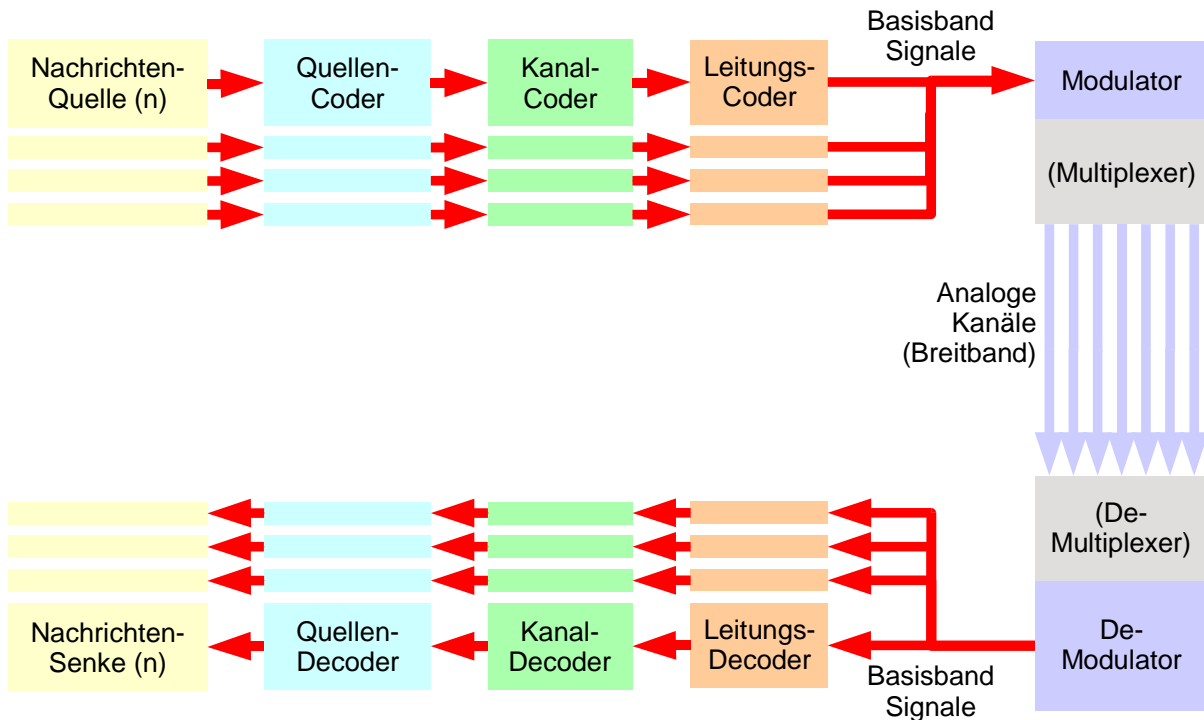


Abbildung 8.4: Mögliche Codier- und Modulier-Verfahren

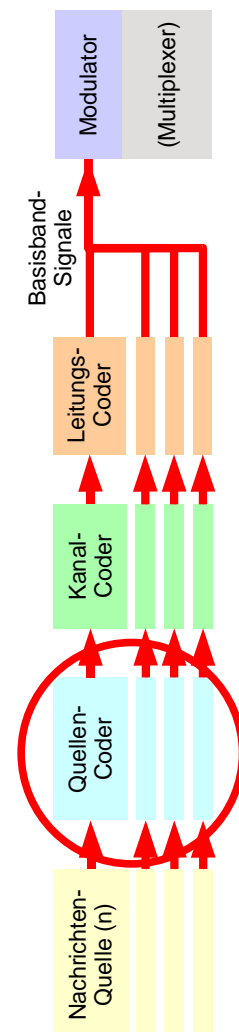
8.2.1 Quellen-Codierung / Quellen-Decodierung

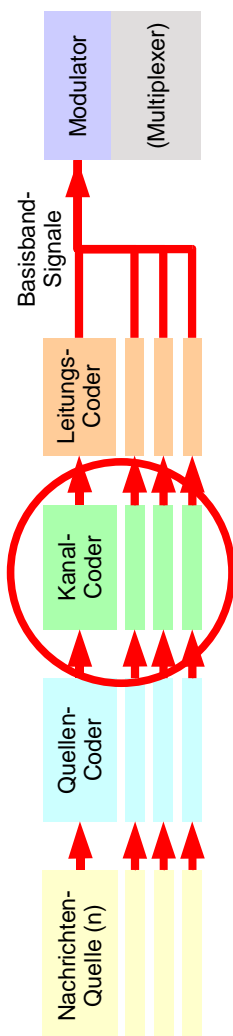
Einer der ältesten Quellencodes stellt der Morsecode dar. Dieser Code codiert die Zeichen der Nachrichten mithilfe von Punkten und Strichen (Tabelle 8.3).

Zeichen	Morsecode	Zeichen	Morsecode
A	. _	B	_ . . .
C	_ . _ .	D	_ . .
E	.	F	. . _ .

Tabelle 8.3: Einige Morsezeichen

Moderne Kommunikationseinrichtungen benutzen ausschliesslich den binären Code, der alle Zeichen in Nullen und Einsen darstellt. Zur Anwendung gelangen z.B. der ASCII-Code oder der ANSI-Code. Vollständige Code-Tabellen finden sich in der einschlägigen Literatur oder im Internet.





ASCII	ANSI	Bedeutung	Unicode	hex
128	199	Ç C mit Cedille	0 199	C7
129	252	ü Umlaut ü	0 252	FC
132	228	ä Umlaut ä	0 228	E4
134	229	å a mit Ringel	0 229	E5
135	231	ç c mit Cedille	0 231	E7
136	234	ê ^e	0 234	EA
137	235	ë e mit Trema	0 235	EB
138	232	è `e	0 232	E8
139	239	ï i mit Trema	0 239	EF
140	238	î ^i	0 238	EE
141	236	ì `i	0 236	EC
142	196	Ä Umlaut Ä	0 196	C4
143	197	Å A mit Ringel	0 197	C5
144	201	É ´E	0 201	C9
145	230	æ ae Ligatur	0 230	E6
146	198	Æ AE Ligatur	0 198	C6
147	244	ô ^o	0 244	F4
150	251	û ^u	0 251	FB
151	249	ù `u	0 249	F9
152	255	ÿ y mit Trema	0 255	FF
153	214	Ö Umlaut Ö	0 214	D6
154	220	Ü Umlaut Ü	0 220	DC
155	162	¢ t'	1 101	0165
156	163	£ Brit. Pfund	0 163	A3

Tabelle 8.4: Auszug aus der ANSI – ASCII-Tabelle

8.2.2 Kanal-Codierung / Fehlerbehandlung

Diese Codierung wird vor allem zur Fehlererkennung eingesetzt. Bei der Kanalcodierung geht es darum, Nachrichten so zu codieren, dass sie auch nach dem Passieren eines Kanals, der die Nachricht unter Umständen verzerrt bzw. verrauscht überträgt, wieder entziffert werden können.

Kanalcodierung wird in allen modernen Übertragungssystemen, z.B. im Mobilfunk bei GSM und UMTS, bei Kabelmodems und generell in Computer-Netzwerken eingesetzt.

Paritybits, Hamming Code, Cyclic Redundancy Check (CRC) und andere Codes sind typische Vertreter solcher Kanalcodierungen und dienen hauptsächlich der Fehlererkennung und Korrektur. Abbildung 8.5 zeigt die Wichtigkeit einer Fehlererkennung in der Datenübertragungstechnik!

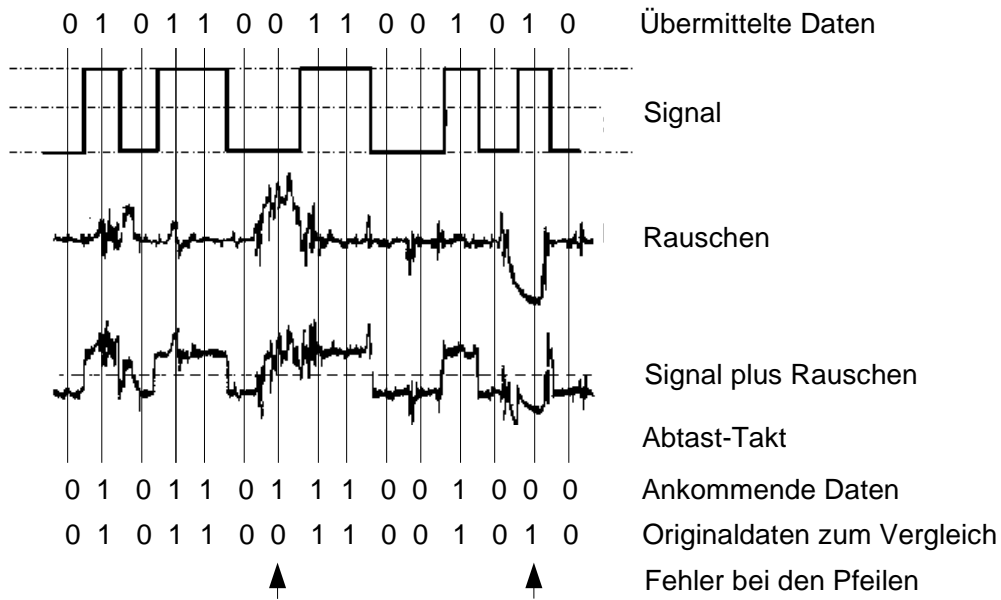


Abbildung 8.5: Die Entstehung von Fehlern auf Kommunikationsleitungen (basierend auf realen Messresultaten)

8.2.2.1 Möglichkeiten der Fehlerbehandlung

Die Fehlerbehandlung erfolgt nach zwei Grundstrategien.

1. Entweder wird vom Sender mit jedem Datenblock (Frame oder Character) genügend redundante Information (zusätzliche Information) mitgeliefert (codiert), damit der Empfänger Fehler erkennen und korrigieren kann (Forward Error Control, oder auch Forward Error Correction, FEC).
2. Die zweite Möglichkeit besteht darin, dass vom Sender nur gerade so viele Informationen mitgeliefert werden (codiert werden), um die Fehler zu erkennen; eine Korrektur ist jedoch nicht möglich (Feedback Error Control). Die Information muss erneut gesendet werden (Retransmission). Welche Art der Fehlererkennung und -korrektur in der Praxis angewendet wird, hängt vom Einsatzort der Übertragung ab. Die folgenden Überlegungen helfen, die Problematik zu verstehen.

Auf Leitungen mit guter Übertragungsqualität können relativ grosse Datenblöcke (Frames) mit Feedback Error Control kostengünstig übertragen werden, weil die wenigen zu erwartenden Fehler nur selten zum erneuten Senden ganzer Datenblöcke führen. Je schlechter

die Leitung, desto kleiner sollen die Datenblöcke gewählt werden und um so eher lohnt sich ein Forward Error Control.

Bei Simplex-Verbindungen und bei Broadcast-Kommunikation kommt hingegen nur die Forward Error Control in Frage, da der Empfänger keine Möglichkeit hat, dem Sender mitzuteilen, welches Frame er noch einmal senden soll.

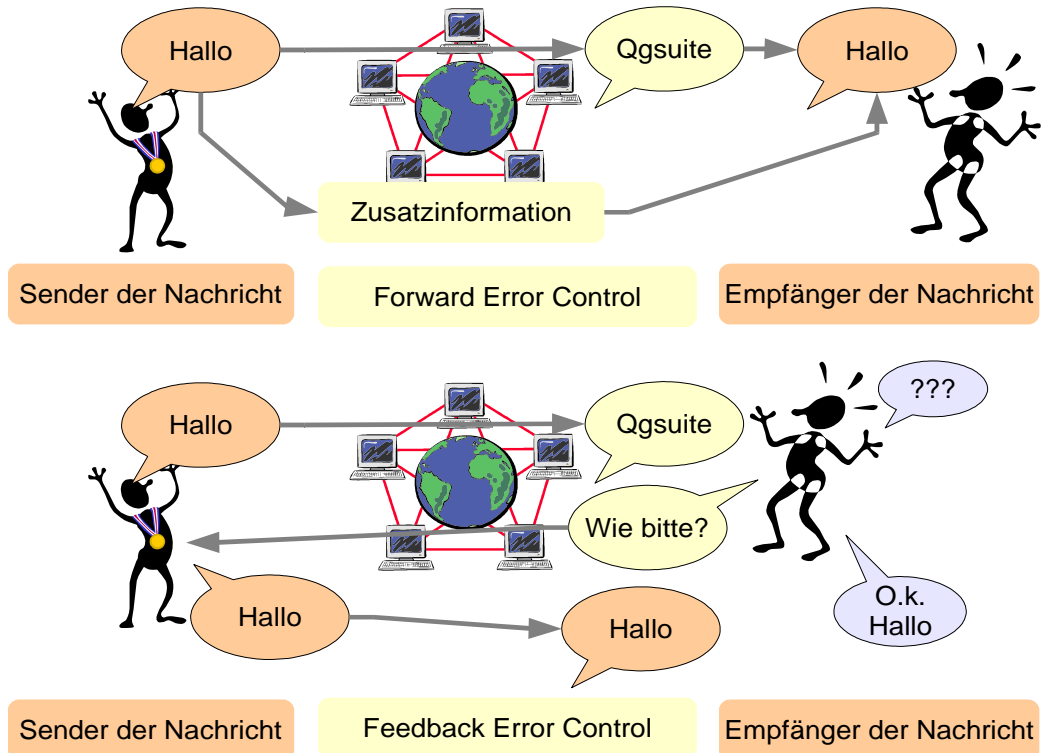


Abbildung 8.6: Die beiden Fehlerbehandlungsmethoden im Überblick

8.2.2.2 Forward Error Control and Correction (FEC)

Dies ist ein Verfahren zur Fehlerkorrektur, bei dem der Empfänger in der Lage ist, fehlerhafte Daten wiederherzustellen.

Der Sender hängt der Meldung genügend zusätzliche Informationen an (doppelt oder mehrfach eingefügte Prüfbits, Redundanz), die es dem Empfänger erlauben, die fehlerhafte Meldung wiederherzustellen (Error Correction). Der dazu notwendige mathematische Aufwand ist gross.

Anwendung findet dieses Verfahren z.B. bei Simplex-Verbindungen, aber auch in CD-Playern, beim so genannten Reed-Solomon-Code, RS-Code. Auch andere Speichermedien wie HDD, FDD, Bänder (Magnetic Tapes) benutzen solche Codes.

8.2.2.2.1 Generierung von Zusatzinformationen

Die für das FEC-Verfahren benötigten Zusatzinformationen müssen vom Sender generiert und der Nachricht angehängt werden. Der Empfänger kann aus diesen Informationen die fehlerhaften Bit erken-

nen und korrigieren. Grundsätzlich kann auch der Cyclic Redundancy Check (CRC) Folgefehler erkennen und Einzelbit-Fehler korrigieren. Eine Fehlerkorrektur für mehrere Bit ist jedoch nicht möglich. Daher kommt CRC für die FEC nicht in Frage. Ein geeignetes Verfahren stellt der Hamming-Code dar.

8.2.2.2.2 Hamming-Code

Ein weit verbreiteter Forward Error Correction-Code ist der Hamming-Code, eigentlich ein Fehlererkennungscode, der mächtig genug ist, auch Fehler zu korrigieren.

Als stark vereinfachtes Beispiel soll die Übertragung eines 8-Bit-Datenfeldes gezeigt werden: 10011001 seien die zu übertragenden Daten. Ein Einzelbit-Korrektor, auf dem Hamming-Verfahren basierend, fügt zusätzliche Bits (mit „x“ bezeichnet) an den Positionen mit Zweierpotenzen ein:

Speicher Platz	12	11	10	9	8	7	6	5	4	3	2	1
Bitstrom	1	0	0	1	x	1	0	0	x	1	x	x

Abbildung 8.7: Ausgangslage

Weil die Positionen 1, 2, 4 und 8 in diesem Fall Zweierpotenzen darstellen⁵³, sitzen dort die Prüfbits.

Die vier Prüfbits rechnet der Sender wie folgt aus: Die binären Zahlen aller Stellen mit einer 1 werden spaltenweise zusammengezählt:

Stelle	Binärzahl der Stelle
12	1100
9	1001
7	0111
3	0011

Resultat: 0001⁵⁴

Die vier Prüfbits werden an den mit „x“ markierten Stellen eingefügt. Das zu übermittelnde Codewort lautet somit:

Speicher Platz	12	11	10	9	8	7	6	5	4	3	2	1
Bitstrom	1	0	0	1	0	1	0	0	0	1	0	1

Abbildung 8.8: Zu übermittelnde Daten

⁵³ $1 = 2^0, 2 = 2^1, 4 = 2^2, 8 = 2^3$

⁵⁴ Kleiner Trick: Zur Bestimmung des XOR: Zählen Sie die Anzahl der Einsen in jeder Spalte. Ist das Resultat eine gerade Anzahl, dann schreiben Sie im Resultat eine 0, bei einer ungeraden Anzahl Einsen schreiben Sie 1.

Der Empfänger zählt wieder alle Stellen mit einer 1 im Bitmuster mit Modulo 2 zusammen. Ergibt in unserem Fall die Summe 0000, so stimmt die Übertragung:

Stelle	Binärzahl der Stelle
12	1100
9	1001
7	0111
3	0011
1	0001

Resultat: 0000

Speicher Platz	12	11	10	9	8	7	6	5	4	3	2	1
Bitstrom	0	0	0	1	0	1	0	0	0	1	0	1

Abbildung 8.9: Fehler in der Übertragung

Nun untersuchen wir den Fall, wo an erster Stelle keine 1, sondern eine 0 übermittelt wird (ein Fehler):

Stelle	Binärzahl der Stelle
9	1001
7	0111
3	0011
1	0001

Resultat: 1100 = 12

Das Resultat ist Dezimal 12 und der Decoder weiss, dass die 0 an der zwölften Stelle eine 1 sein muss. Dieser Code kann aber wie gesagt nur ein (1) falsches Bit erkennen. Sind zwei Bits fehlerhaft, so wird es für das Einbit-Hamming-Verfahren unmöglich, zu korrigieren. Erweiterungen des Hamming-Verfahrens erlauben es, auch mehrere fehlerhafte Bit zu erkennen und zu korrigieren. Diese Demonstration mit dem Einbit-Hamming-Verfahren zeigt jedoch, wie Fehler nach der FEC-Methode korrigiert werden können, und wo deren Grenze liegt.

8.2.2.3 Feedback Error Control

Bei diesem Verfahren muss der Empfänger die fehlerhaften Daten nur erkennen können. Die Fehler können vom Empfänger nicht korrigiert werden. Die fehlerhaften Frames müssen noch einmal gesendet werden (engl. Retransmission).

8.2.2.3.1 Einfache Paritätskontrolle mit dem Paritätsbit (Paritybit)

Dieses Codier-Verfahren wird beispielsweise in Übertragungssoftware von Modems angewendet. Daher muss auch bei einer Übertragung mit Modem jeweils angegeben werden, wie viele Paritätsbit man wünscht.

Das Verfahren funktioniert sehr einfach und recht effizient: Für die Übertragung eines Zeichens werden zum Beispiel sieben Datenbits benötigt (ASCII-Code, 7 Bit). Das achte Bit ist das Paritätsbit (parity bit), das mit den anderen sieben Bits gleichzeitig übertragen wird. Das Paritätsbit wird am siebten Datenbit angehängt:

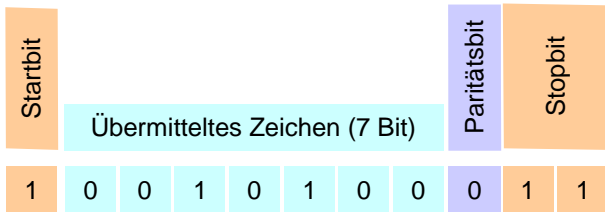


Abbildung 8.10: Paritätsbit als Prüfsumme

Der Wert des Paritätsbits wird durch mehrere aufeinander folgende Exklusiv-ODER-Verknüpfungen gebildet, indem man die ersten zwei Datenbits mittels einer Exklusiv-ODER-Funktion (XOR) verarbeitet, und danach das Resultat mit dem nächsten Datenbit auf gleiche Weise wiederholt, bis alle Datenbits verarbeitet sind, und daraus das Paritätsbit resultiert.

Die Funktionstabelle für Exklusiv-ODER lautet:

Bit 1	Bit 2	XOR
0	0	0
0	1	1
1	0	1
1	1	0

In der Praxis wird die Bestimmung des Paritybits mithilfe der folgenden Hardware-Schaltung realisiert.

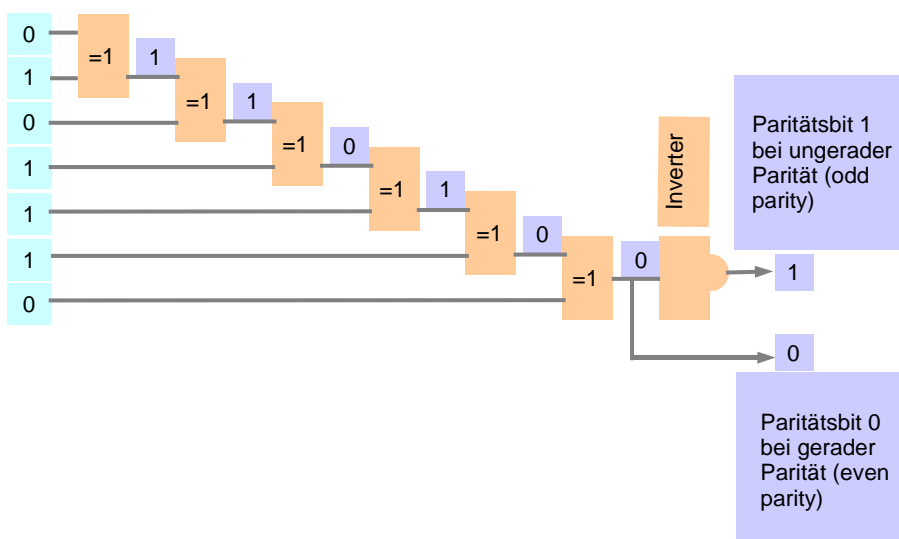


Abbildung 8.11: Digitale Schaltung zur Ermittlung des Paritybits mit XOR

Die übermittelten Daten werden je nach gewünschter Art der Parität am Ausgang der Schaltung mit einer 0 oder einer 1 versehen: 01011101 bei ungerader Parität (odd) und 01011100 bei gerader Parität (even).

Der Empfänger behandelt die empfangenen Daten mit der gleichen Hardware-Schaltung und vergleicht das Resultat seiner Schaltung mit dem Wert des Paritätsbits des Senders. Stimmen die Bits überein, wird er die Daten als fehlerfrei anerkennen, ansonsten verwerfen.

Mit dieser Methode lassen sich nur Einzelfehler oder eine ungerade Anzahl von Fehlern erkennen. Eine gerade Anzahl von Fehlern wird hier die Kontrolle unerkannt passieren. Eine Korrektur von Fehlern ist nicht möglich, da nicht bestimmt werden kann, welches Bit falsch ist.

8.2.2.3.2 BSC (Block Sum Check)

Statt einzelner Zeichen (oder Frames) können auch Zeichenblöcke (oder Frameblöcke) übermittelt werden. Der Sender ordnet die Zeichen (Frames) untereinander, in Form eines Blockes an. Es werden jeweils die Zeilen- und die Spaltenparitäten ermittelt. Diese Methode erlaubt immerhin schon die sichere Erkennung von zwei Fehlern. Das folgende Bild zeigt einen Zeichenblock aus sechs Zeichen à sieben Bit mit einem STX (Start of Text) und einem ETX (End of Text). Die Zeilenparität ist ungerade (odd) und die Spaltenparität gerade (even). Die Paritybits der Spalten werden auch als Block Check Character (BCC) bezeichnet.

Zeilen- parität (odd)	Bit	Bit	Bit	Bit	Bit	Bit	Bit	
	6	5	4	3	2	1	0	
0	0	0	0	0	0	1	0	= STX
1	0	1	0	1	0	0	0	Zeichen oder Frameinhalte
1	1	0	1	0	1	1	0	
0	0	1	1	1	0	1	1	
1	0	0	1	0	0	1	0	
0	1	0	0	1	0	1	0	
1	0	1	0	1	1	1	0	
1	0	0	0	0	0	1	1	= ETC
1	0	1	1	0	0	1	0	= BCC
Spaltenparität (even)								

Abbildung 8.12: Block Sum Check-Verfahren

8.2.2.3.3 CRC (Cyclic Redundancy Check)

Zyklischer Redundanz-Code oder Polynom-Code

Die Parity-Bit-Methode oder die Block-Parity-Bit-Methode können aufeinander folgende Fehler nicht sicher erkennen. Mit der CRC-Methode lassen sich mithilfe von standardisierten Codewörtern (so genannte Generatorpolynome) auch solche Fehler feststellen.

Die genaue theoretische Erklärung des CRC ist nicht ganz einfach. Der mathematische Zusammenhang ist ziemlich kompliziert und bedarf einiger Mathematikkenntnisse.

Die Methode basiert darauf, dass man die zu übertragenden Daten durch ein genormtes Bitmuster dividiert und den Divisionsrest mitüberträgt. Der Empfänger kann die Fehler anhand der übertragenen Daten erkennen. Als Bitmuster kommen die erwähnten Generatorpolynome zum Einsatz.

Es existieren genormte Generatorpolynome: Die Umsetzung der Polynome in maschinenlesbare Bitmuster erfolgt gemäss folgendem Beispiel:

$$\text{CRC-12} (= x^{12} + x^{11} + x^3 + x^2 + x + 1)$$

$$\text{CRC-16} (= x^{16} + x^{15} + x^2 + 1)$$

$$\text{CRC-CCITT} (= x^{16} + x^{12} + x^5 + 1)$$

$$\text{CRC-32} (= x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1)$$

Abbildung 8.13: Die Generatorpolynome

Alle Summanden im Polynom haben die Form x^n . Alle x im Polynom stehen für eine 1 oder eine 0. Die Exponenten (n) geben die Stelle im Bitmuster an (im Beispiel hat das Bitmuster 10 Stellen, Stelle 0 bis Stelle 9).

Beispiel: $x^9 + x^7 + x^3 + x^2 + 1$ bedeutet 1010001101

Der erste Summand (von rechts) repräsentiert die Stelle 0 im Bitmuster, die 1 im Polynom ergibt bei der Umwandlung eine binäre +1, da $1^0 = 1$.

Die Stelle 1 im Bitmuster fehlt im Polynom (x^1), weil im Bitmuster eine Null ($0^1 = 0$) steht. Im Polynom werden keine Nullen abgebildet, weil ein Summand mit Null-Wert die Gesamtsumme nicht ändern kann!

Die Stelle 2 (x^2) ist im Polynom wieder vertreten, weil im Polynom eine Eins des Bitmusters ($1^2 = 1$) repräsentiert werden muss.

Es ist nun leicht nachzuvollziehen, wie die Stellen 3, 7 und 9 des 10-stelligen Bitmusters in diesem Beispiel umgesetzt werden müssen.

Die zu übertragenden Daten werden beim CRC der folgenden, umfangreichen Behandlung unterzogen:

Der Sender generiert eines der genormten Generatorpolynome.

1. Der Sender gibt dem Empfänger das Generatorpolynom bekannt, das er für die bevorstehende Übertragung verwenden will.
2. Durch Dividieren der gültigen Daten mit dem Generatorpolynom erhält der Sender eine Prüfsumme.
3. Die Prüfsumme wird den Daten angehängt und mit den Daten übermittelt.
4. Der Empfänger dividiert die erhaltenen Daten durch das vorher abgemachte Polynom.
5. Gibt der Divisionsrest Null, so sind mit sehr grosser Wahrscheinlichkeit keine Fehler in den Daten.

Abbildung 8.14 zeigt das Verfahren schrittweise.

Das Bitmuster 11100110 soll mit dem Generatorpolynom 11001 übertragen werden. Weil im folgenden Beispiel das Generatorpolynom fünf Bit lang ist, werden dem Bitmuster vor der Division vier Nullen (0000) angehängt = 11100110 0000.

Das neue Bitmuster wird durch das Generatorpolynom mit Modulo 2 dividiert. Der Rest 0110 wird nun zu 111001100000 dazugezählt, was 1110011000110 als zu übertragendes Bitmuster ergibt.

Auf diese Weise erreicht man, dass das zu übertragende Bitmuster durch das vorher bestimmte Generatorpolynom ohne Rest teilbar ist. Weil aber dem Empfänger das Generatorpolynom ebenfalls bekannt ist (es wird immer ein genormtes Polynom verwendet), kann er jetzt alle empfangenen Bitmuster durch das Generatorpolynom teilen, und wenn der Rest der Division 0 ist, dann sind die Daten mit grosser Wahrscheinlichkeit korrekt.

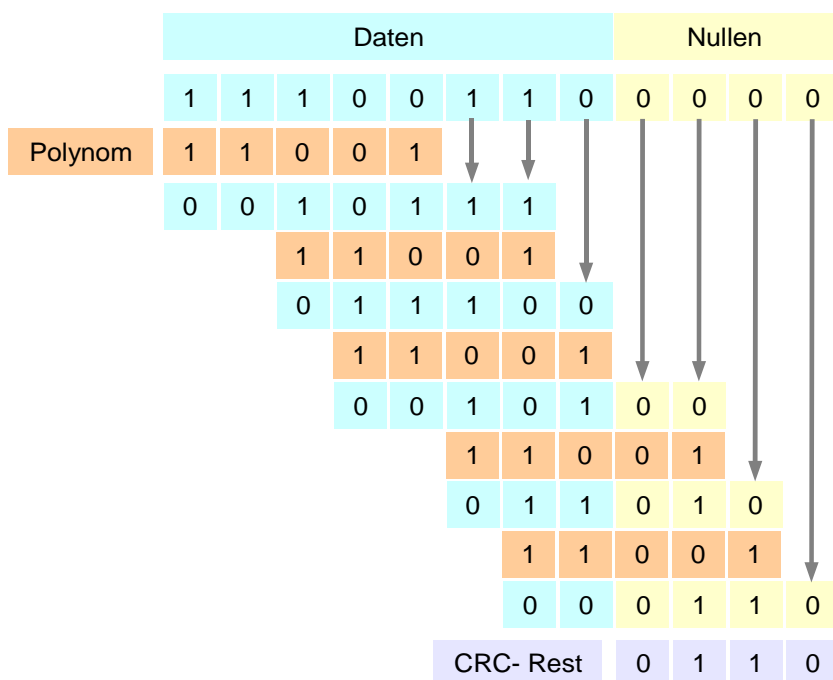


Abbildung 8.14: CRC-Ermittlung

Durch eine einfache Schieberegisterschaltung lässt sich die Methode mit Hardware realisieren.

CRC kann Folgefehler erkennen, deren Länge kleiner ist als die Anzahl der an die Daten bei der Division angehängten Nullen (Länge des Generatorpolynoms – 1).

8.2.3 Leitungscodierung / Leitungscodierung

Diese Technologie wird mittels CoDec-Geräten realisiert. Dies sind Geräte, die Bitströme in Leitungscodes codieren. Diese elektrischen Leitungscodes (Signale) werden durch Signalleitungen vom Sender zum Empfänger übermittelt.

Leitungscodes weisen einige wichtige Eigenschaften auf. Die wichtigste ist wohl die Taktrückgewinnung: Es muss möglich sein, dass den Signalwerten der Takt (engl. Clock) entnommen werden kann. Wäre dies nicht möglich, müsste eine separate Taktleitung zwischen dem Sender und dem Empfänger zur Verfügung stehen. Der Takt eines Codes sollte möglichst unabhängig vom Inhalt der übertragenen Daten sein.

Im Weiteren ist es wünschenswert, dass die Übertragung gleichstromfrei ist. Auf manchen Übertragungstrecken darf wegen der angeschlossenen Geräte kein Gleichstrom auftreten.

Die Übertragungreichweite hängt von der Betriebsdämpfung ab. Generiert der Code zu hohe Frequenzen werden die Signale stärker gedämpft und die Reichweite der Übertragung wird vermindert.

In einigen Fällen kann in einem Signalwert mehr als ein Zeichenwert codiert werden. Zudem ist es wünschenswert, wenn der Code eine Resynchronisation des Empfängers erlauben würde. Dies wird meist durch Rahmenbildung ermöglicht (siehe Layer 2).

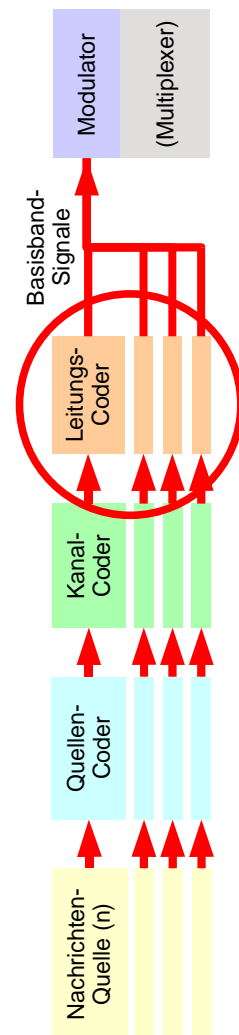
Es existieren einige Code-Varianten, um Bitmuster als elektrische Signale darzustellen. Man unterscheidet meistens zwischen binären Leitungscodes, biphasen Leitungscodes, ternären Leitungscodes und Blockcodes.

8.2.3.1 Binäre Leitungscodes

Bei binären Leitungscodes werden die Symbolwerte durch den Signalwert bestimmt. Im Folgenden sind einige typische binäre Leitungscodes dargestellt.

8.2.3.1.1 Non Return to Zero (NRZ)

Die einfachste Art, Bitmuster zu codieren, ist der Non Return to Zero-Level (NRZI-L) Code. Diesen gibt es in einer unipolaren und einer polaren Variante⁵⁵ (siehe Abbildungen 8.16. und 8.15).



⁵⁵ Unipolar = Spannungspegel entweder + oder -
Polar = Spannungspegel + und -

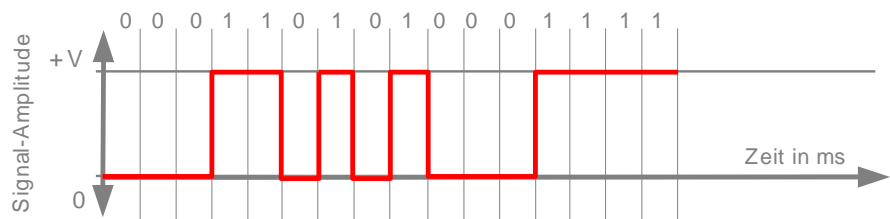


Abbildung 8.15: Unipolarer Code (NRZ-L)

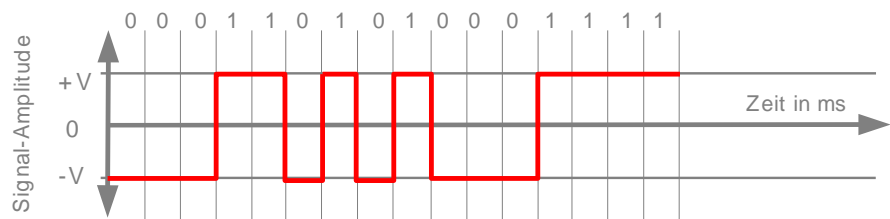


Abbildung 8.16: Polarer Code (NRZ-L)

Das Verfahren ist störanfällig, weil bei konstanter Bitfolge (z.B. alles 1 oder 0) dauernd eine Gleichspannung an der Datenleitung anliegen würde. Die beiden Kommunikationsteilnehmer können aufgrund der fehlenden Signalwechsel auch schlecht synchronisiert werden und es muss daher eine spezielle Synchronisierleitung zwischen Sender und Empfänger gelegt werden.

Dieses Verfahren wird z.B. bei allen digitalen Schaltungen, bei der seriellen RS232-Übertragung oder beim CAN-Bus⁵⁶ verwendet.

Eine Variante ist der NRZ-I (Non Return to Zero-Invers), bei dem bei jeder 1 der Pegel wechselt. Eingesetzt wird NRZ-I beim USB und beim FDDI und bei der Aufzeichnung von Daten auf CD-ROM und Festplatte. (Abbildung 8.17).

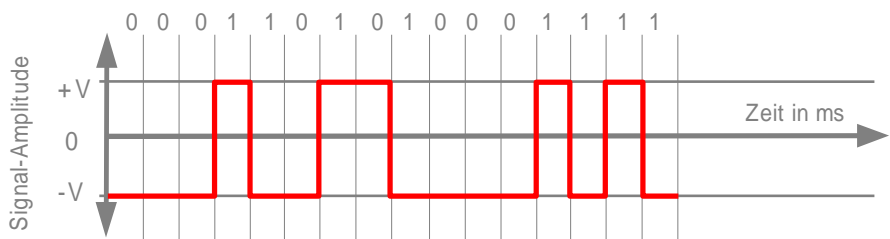


Abbildung 8.17: Der NRZ-I-Code

⁵⁶ Der CAN-Bus (Controller Area Network) gehört zu den Feldbussen. Es handelt sich dabei um ein serielles Bussystem, das von BOSCH für die KFZ-Automatisierung entwickelt wurde und 1985 zusammen mit Intel vorgestellt wurde, um die Kabelbäume (bis zu 2 km pro Fahrzeug) zu reduzieren und dadurch Gewicht zu sparen.

8.2.3.1.2 Return to Zero (RZ)

Die Return to Zero-Codierung ist eine Weiterentwicklung der NRZ-Codierung.

Bei dieser Codierung ist es möglich, den Takt aus dem Signal zurückzugewinnen. Der RZ-Code ist gekennzeichnet durch einen Rechteckimpuls in der ersten Hälfte des Bitintervalls für das Datenelement 1. Danach erfolgt eine Rückkehr in den Grundzustand (zero, hier -V).

Der Code existiert in zwei Varianten. Einerseits als polarer RZ-Code, bei dem die Einsen immer als positiven Pegel dargestellt werden (Abbildung (8.18)), und andererseits als bipolarer, differenzieller⁵⁷ RZ-Code, der die Einsen abwechslungsweise als positiver Pegel und als negativer Pegel darstellt (Abbildung 8.19).

Der Nachteil des RZ-Codes ist, dass für die Pegelwechsel bei der 1 eine doppelte Signal-Bandbreite nötig ist.

Der RZ-Code gelangt in digitalen Schaltungen für die Industrie und sogar im Flugzeugbau zum Einsatz. Die ganz neuen DWDM⁵⁸-Übertragungen benutzen auch den RZ-Code.

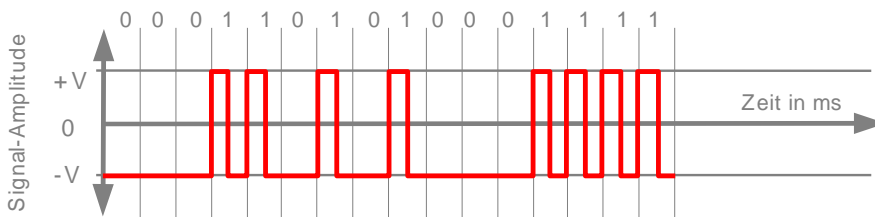


Abbildung 8.18: Polarer RZ-Code

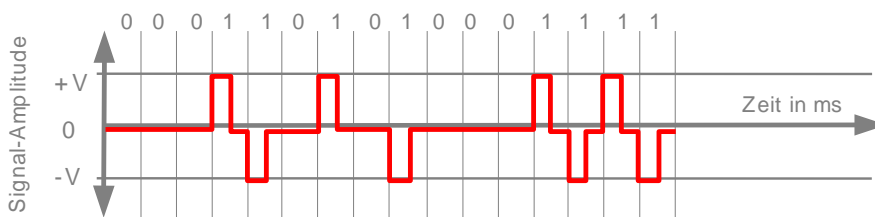


Abbildung 8.19: Bipolarer RZ-Code

⁵⁷ Differenziell bedeutet, dass z.B. die Einsen abwechslungsweise als positiver Pegel oder als negativer Pegel dargestellt werden. Man erreicht dadurch besonders bei gestörten Leitungen eine Signalwiederherstellung beim Empfänger.

⁵⁸ DWDM = Dense Wave Division Multiplex, eine Lichtwellenleiter-Übertragungstechnik

8.2.3.2 Biphase Leitungscodes

Bei biphasen Leitungscodes werden die Symbolwerte durch Pegelwechsel codiert. Im Folgenden sind einige typische biphase Leitungscodes dargestellt.

8.2.3.3 Manchester-Code

Der Manchester-Code ist ein sehr typischer biphaser Leitungscode, da er die Einsen als Pegelwechsel von plus nach minus codiert und die Nullen in umgekehrter Richtung.

Die Manchestercodierung wird bei 10 MBit-Ethernet für die Taktrückgewinnung aus dem Bitstrom verwendet. Weil das Signal zwischen den Takten den Pegel wechselt, wird im schlimmsten Fall die doppelte Anzahl Signalwechsel benötigt. Das bedeutet, dass bei 10 MHz nur 5 Mbit/s übertragen werden könnten. In der Praxis wird Ethernet mit einer Bandbreite von 30 MHz betrieben, was eine Datenrate von 10 Mbit/s erlaubt.

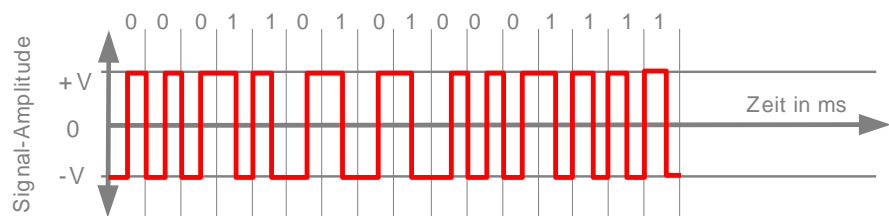


Abbildung 8.20: Der Manchester-Code

8.2.3.3.1 Differential-Manchester-Code

Beim Differential-Manchester-Code zeigt das Vorhandensein eines Signalwechsels am Anfang des Taktsignals, ob eine 0 oder eine 1 folgt. Wechselt das Signal von High zu Low, so folgt eine 0. Wechselt das Signal nicht, so folgt eine 1 (Bsp. Token-Ring).

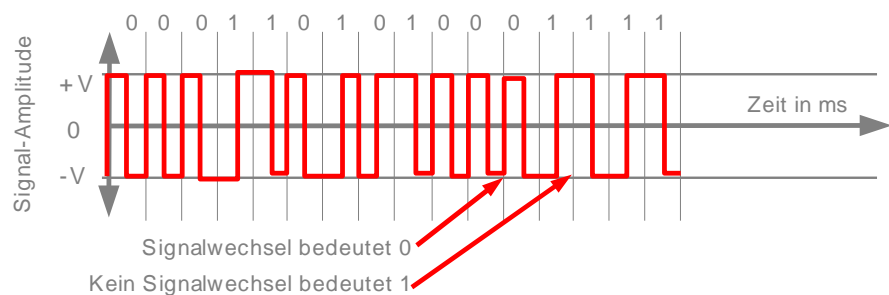


Abbildung 8.21: Der Differential-Manchester-Code

8.2.3.4 Ternäre Codes

Bei ternären Codes werden die beiden Symbolwerte 0 und 1 durch 3 Codiersymbole -1, 0 und +1 abgebildet.

8.2.3.4.1 Der High Definition Bipolar-Code (HDB3-Code)

Der HDB3-Code gleicht dem Bipolar-AMI-Code. Der Unterschied besteht darin, dass bei 4 und mehr aufeinander folgenden Nullen absichtlich nach drei Nullen ein Signalwechsel in der gleichen Richtung wie die letzte Eins (Verletzung der Regel, engl. violation) stattfindet, damit das Problem des Gleichspannungsanteiles minimiert und eine hohe Störanfälligkeit erreicht werden kann.

(Bsp. in WANs auf 2,048 MBit/s Übertragungsstrecken (E1))

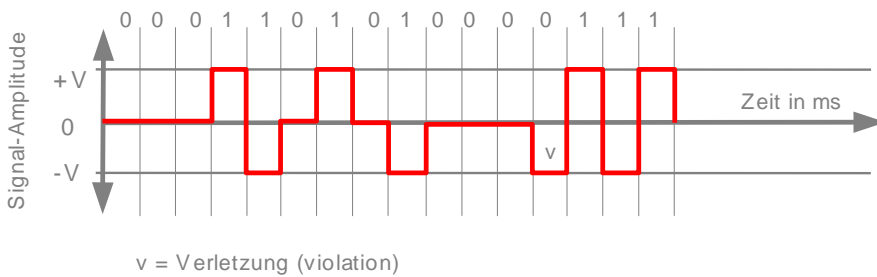


Abbildung 8.22: Der HDB3-Code

8.2.3.4.2 Der 2B1Q (2Binär/1Quarternär) Code

Der 2B1Q-Code, wie er im ISDN-Netz angewendet wird, überträgt pro Taktsignal 2 Bit und benutzt dazu vier Spannungsniveaus. Weitere ähnliche Codes, z.B. der 4B/3T-Code (4Binär/3Ternär), überträgt pro Taktsignal 4 Bit auf drei Spannungsniveaus (Ternär) (z.B. ISDN).

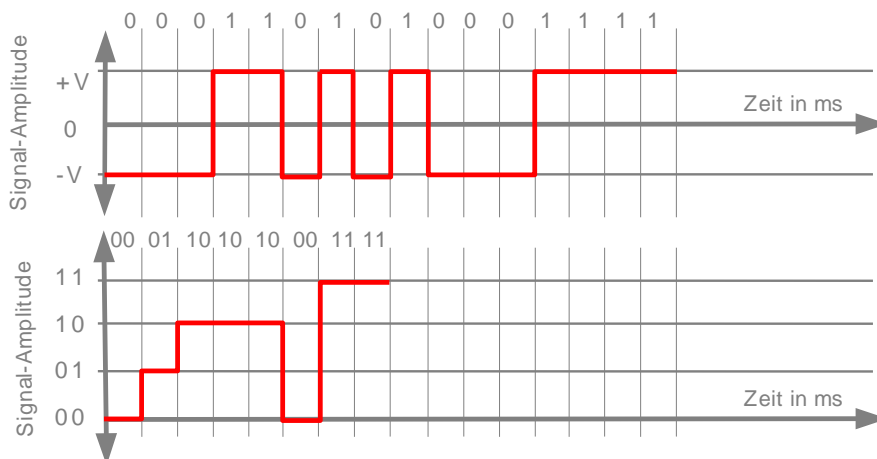


Abbildung 8.23: Der 2B1Q-Code

8.2.3.4.3 Bipolar-Alternate Mark Inversion-Codierung (AMI)

Hier wird versucht, den Gleichspannungsanteil zu minimieren, indem eine 0 mit 0V dargestellt wird und die 1 jeweils alternierend mit +5V oder -5V dargestellt wird (Bsp: WAN-Verbindungen (E1, T1)).

8.2.3.5 Block-Codes

Block-Codes sind Codes, bei denen m-Bits als Block zusammengefasst und zu einem neuen Block der Länge n codiert werden

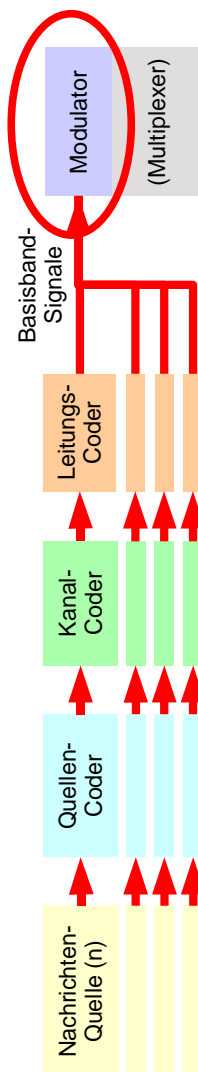
8.2.3.5.1 4B5B-Code / 8B/10B-Code

Gigabit Ethernet, ATM und andere Hochgeschwindigkeits-Übertragungen auf der Basis der Fibre Channel-Norm, welche für Übertragungen auf LWL aufgestellt wurde, benutzen den 4B/5B- oder den 8B/10B-Code, der vier respektive acht Bit mithilfe von ausgeklügelten Code-Tabellen auf fünf respektive 10 Bit erweitert und damit lange 1- oder 0-Intervalle im Bitstrom verhindert. Dazu werden jeweils 4 Daten-Bit in 5 Signal-Bit codiert. Dabei darf es nicht mehr als eine führende „0“ und nicht mehr als 2 abschliessende „0“ geben. Anschliessend werden die Bits mit dem NRZ-I-Code übertragen.

Bedingt durch das Einfügen von Redundanz erreicht man nur eine Effizienz von 80%.

Der 4B5B-Code wird zum Beispiel bei FDDI verwendet.

Auch 100-MBit-Ethernet verwendet zur Taktrekonstruktion eine 4Bit/5Bit-Codierung. Die Übertragungsrates auf dem Kabel beträgt daher 125 MBit/s. Die Übertragung wird auf drei Spannungsstufen im Wechsel 0,1,0,-1 durchgeführt, wobei die Information durch Halten einer Stufe bei 0 Bits übertragen wird (MLT-3-Verfahren⁵⁹). Dieses Verfahren hat den Vorteil einer guten Frequenzausnutzung.



8.2.4 Modulieren / Demodulieren

Diese Technologie wird mittels MoDem-Geräten realisiert. Dies sind Geräte, die Bitströme in Wellenform umwandeln. Sodass sie anschliessend auf analogen Leitungen übertragen werden können.

Sollen die digitalen Daten in analoger Form übertragen werden (Telefon, Funknetze), müssen diese ebenfalls zuerst aufbereitet werden. Typische Anwendungen dafür sind die Datenübertragung mit Modems und dem analogen Telefonnetz oder neuerdings die Digital Subscriber Line-Übertragungen (xDSL).

Die digitalen Daten müssen in diesem Fall in analoge Signale umgewandelt werden. Dies wird dadurch erreicht, dass auf einem Dauerton von beispielsweise 1000 bis 2000 Hz, dem so genannten Sinuswellenträger (sine wave carrier), die Amplitude, die Frequenz oder die Phasenlage moduliert wird (ein Bipolar-AMI-Code ist zur Verdeutlichung mit abgebildet).

1. Die Amplitude (Spannungspegel) der analogen Sinus-Schwingung wird verändert.

⁵⁹ MLT-3 (Multilevel Transmission Encoding - 3 levels) ist ein in der Nachrichtentechnik zur Datenübertragung über elektrische Kabel verwendetes Verfahren mit drei Spannungspegeln (+,0,-) (ternäres Signal) - im Gegensatz zu zweier-tigen Verfahren wie NRZ, NRZ-I oder RZ.

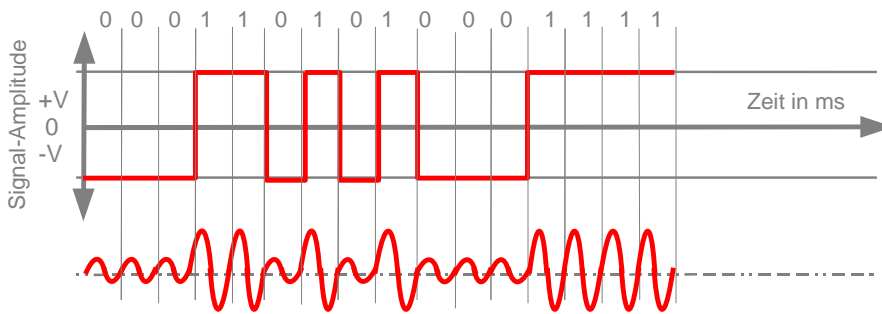


Abbildung 8.24: Die Amplitudenmodulation

2. Die Frequenz (Tonhöhe) der analogen Sinus-Schwingung wird verändert.

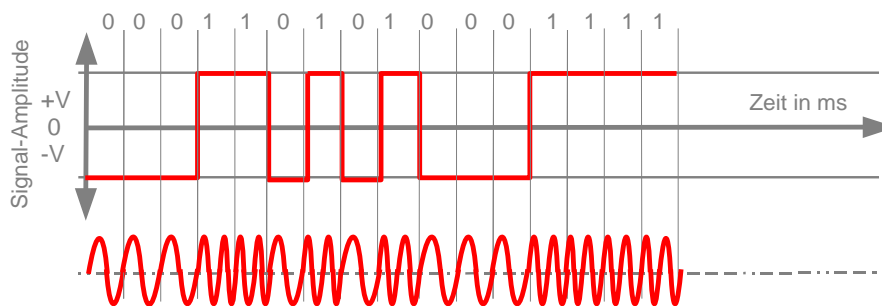


Abbildung 8.25: Die Frequenzmodulation

3. Die Phasenlage der analogen Schwingung wird systematisch, in gleichmässigen Intervallen, geändert. In diesem Beispiel alle 180° . Andere Intervalle, z.B. 30° , 45° oder 60° , sind ebenfalls möglich.

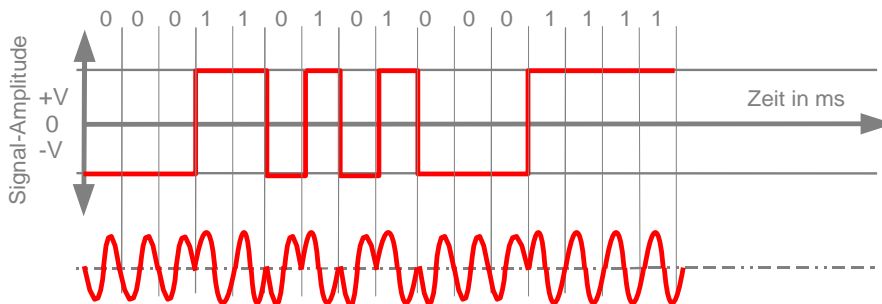


Abbildung 8.26: Phasenmodulation (180°)

Die drei Modulationsarten haben den grossen Vorteil der fehlenden DC-Komponente (Direct Current = Gleichstrom) im Netz.

8.2.4.1 QPSK / QAM

In der Praxis werden die drei Modulationsarten kaum angewendet, da diese die zur Verfügung stehenden Bandbreiten schlecht ausnützen. So wird für eine Übertragungsrate von beispielsweise 10 MBit/s eine Kanal-Bandbreite von 10 MHz benötigt. (Anmerkung: In den USA wird die Übertragungsrate mit bps (bit per second) abgekürzt.)

Die ersten Modems für die Datenübertragung auf Telefonleitungen arbeiteten mit 300 Bit/s, oder wie man damals sagte mit 300 Baud. Die Kanalbandbreite von 3000 Hz des Telefonnetzes wurde damit nicht ausgeschöpft. Aber bereits mit den 2400 Bit/s-Modems wurde die Grenze des Telefonnetzes erreicht. Bis zu dieser Übertragungsgeschwindigkeit könnte man auch die oben erwähnten Modulationsverfahren einsetzen, weil jeder Zustand der Modulation entweder eine 1 oder eine 0 repräsentierte.

Praxis-Hinweis:

Schnelle Telefonie-Modems, die neuen xDSL-Technologien und TV-Kabelmodems verfügen neben unterschiedlichen Datenkompressionsmöglichkeiten über die Quadratur Phasen Modulation (QPSK, Quadratur Phase Shift Keying) oder Quadratur Amplituden Modulation (QAM) Technologien.

Selbstverständlich werden heute grössere Datenübertragungsraten gewünscht. Es ist daher unerlässlich, einen Ausweg zu suchen. Eine Möglichkeit wäre, die Telecom-Betriebe zu überzeugen, dass sie die Bandbreite erweitern. Das kommt aber nicht in Frage und wäre auch nicht nötig, da einige findige Köpfe eine Lösung gefunden haben, um mit der heutigen Bandbreite grosse Datenmengen zu übertragen und somit die restliche Bandbreite für andere Zwecke einzusetzen.

Die beiden Methoden sind vom PSK und ASK abgeleitet. Wird mit einer Amplitude und verschiedenen Phasenlagen gearbeitet, spricht man von QPSK. Werden noch verschiedene Amplitudenwerte zur Übertragung verwendet, dann spricht man von QAM.

Das folgende Beispiel der QPSK zeigt, wie, ähnlich dem 2B1Q-Code, der Bitstrom in Bitmuster à 2 Bit aufgeteilt wird. Jedem Bitmuster wird eine eindeutige Phasenlage (Winkel) zugeordnet. Der Empfänger muss somit nur die Phasenlage im Signal überprüfen und kann somit das Bitmuster wieder rekonstruieren.

Eine übersichtliche Darstellung der Phasenlagen erreicht man mit Vektoren. Die Länge der Vektoren symbolisiert die Amplitudenwerte, und die Winkel der Vektoren im Koordinatensystem entsprechen der Phasenlage. Eine solche Darstellung wird Phasendiagramm genannt. Abbildung 8.28 zeigt eine 16-QAM. Es wird wiederum das Phasendiagramm verwendet. Im Unterschied zum QPSK variieren hier auch die Amplituden (durch verschieden lange Vektoren symbolisiert).

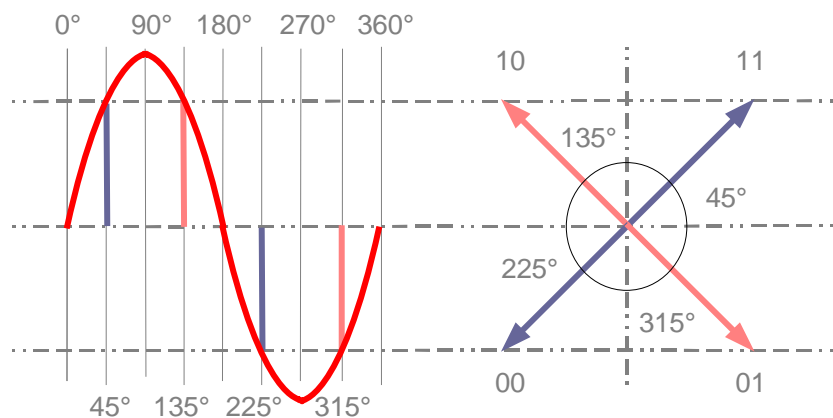


Abbildung 8.27: Ableitung der Vektordarstellung beim QPSK (4-QAM)

Es sind hier 3 Amplituden und 12 Phasenwinkel eingesetzt: Das ergibt die nötigen 16 Zustände, um 4 Bit pro Signalwechsel zu übertragen (1 Bit pro Signalwechsel nennt man 1 Baud).

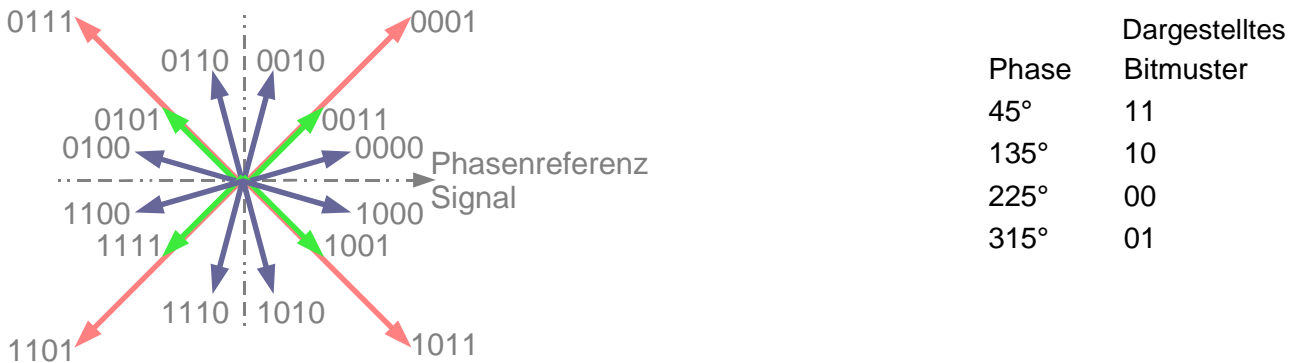


Abbildung 8.28: Phasendiagramm eines 9'600 Bit/s-Modems (16-QAM)

Die Zustandstabelle sieht wie folgt aus:

Phase in Grad	Amplitude	Bitmuster
15°	1	0000
45°	1,5	0001
75°	1	0010
45°	0,5	0011
165°	1	0100
135°	0,5	0101
105°	1	0110
135°	1,5	0111
345°	1	1000
315°	0,5	1001
285°	1	1010
315°	1,5	1011
195°	1	1100
225°	1,5	1101
255°	1	1110
225°	0,5	1111

Tabelle 8.5: Die Zustandstabelle des 16-QAM

Selbstverständlich kann man berechnen, wie viele Vektoren (Längen und Winkel) notwendig sind und wie viele Stellen das Bitmuster haben muss, um andere Datenraten mit 2400 Baud zu übertragen.

Die Datenrate (R) in Bit/s gibt an, wie viele Bit pro Sekunde durch eine Übertragungsleitung gesendet werden können.

Die Modulationsrate (D) in Baud ist meistens durch die Bandbreite der Übertragungsleitung gegeben.

Die Anzahl (n) Bit pro Baud, die notwendigerweise moduliert werden müssen, um Datenraten zu übertragen, welche grösser als die Modulationsrate sind, können mit folgender Formel berechnet werden.

$$n = \frac{R}{D} \quad n = \frac{9600 \text{ Bit/s}}{2400 \text{ Baud}}$$

Abbildung 8.29: Formel für die Berechnung der Anzahl Bit mit Beispiel

Für unser Beispiel heisst das, dass 9600 Bit/s nur übertragen werden können, wenn mit jedem Baud vier Bits übertragen werden können. Daher kommt in unserem Beispiel die 16-QAM-Codierung.

Wenn man nun die Anzahl der notwendigen Vektoren eruieren will (Signalelemente [N]), so kommt folgende Formel zur Anwendung:

$$N = 2^n \quad N = 2^4 = 16$$

Abbildung 8.30: Berechnung der Anzahl der Signalelemente mit Beispiel

Tabelle 8.6 zeigt die verschiedenen QAM-Modulationen.

QAM	Anzahl Punkte	Bemerkung
4-QAM	4	2 Bits
8-QAM	8	3 Bits, entspricht 8-PSK
16-QAM	16	4 Bits
32-QAM	32	5 Bits (es wären 36 Punkte möglich, Eckpunkte des Quadrates werden nicht benutzt, um auf 32 Punkte zu kommen).
64-QAM	64	6 Bits
128-QAM	128	7 Bits (es wären 144 Punkte möglich, Eckpunkte des Quadrates werden nicht benutzt, um auf 128 Punkte zu kommen).
256-QAM	256	8 Bits
512-QAM	512	9 Bits, hier ist die Störanfälligkeit bereits so gross, dass dieses Modulationsverfahren kaum angewendet wird.

Tabelle 8.6: QAM-Modulationsverfahren im Überblick

Praxis-Hinweis:

Für unser 9600 Bit/s-Modem haben wir 2400 Baud zur Verfügung. Mit obiger Formel kann man eruieren, dass 16 Signalelemente notwendig sind. Welche Vektorlängen und Winkel zum Einsatz kommen, hängt jedoch noch von anderen Überlegungen ab (z.B. Eindeutigkeit der Zuweisungen).

8.2.4.2 OFDM (Orthogonal Frequency Division Multiplexing)

Beim OFDM werden statt der einzelnen Signalträger mehrere Träger gleichzeitig moduliert. Jeder einzelne Träger ist phasen- und (ab 4 bit pro Symbol zusätzlich) amplitudenmoduliert und trägt von daher die Information von mehreren Bits.

Dieses Modulationsverfahren liefert viel stabilere Signale als alle anderen Verfahren. Dies, weil die Bits nicht auf einem seriell geschalteten Träger übertragen werden, sondern auf parallelen Trägern gleichzeitig. Diese Technik bewirkt, dass das resultierende einzelne Hochfrequenzsignal viel länger vom Empfänger abgetastet werden kann, nämlich so lange, bis alle parallel übertragenen anderen Signale ebenfalls abgetastet worden sind, ohne dass die Übertragungsleistung beeinträchtigt würde.

Ein Beispiel mit 8192 Trägern, einer 64-QAM-Modulation pro Träger-signal (entspricht 6 Bit pro Träger) und einer Taktrate (Symbol-dauer) von einer Millisekunde: Damit lassen sich $8192 * 6 * 1 / 1.e-3 = 49152000$ Bit/s (ca. 50 Mbit/s) übertragen.

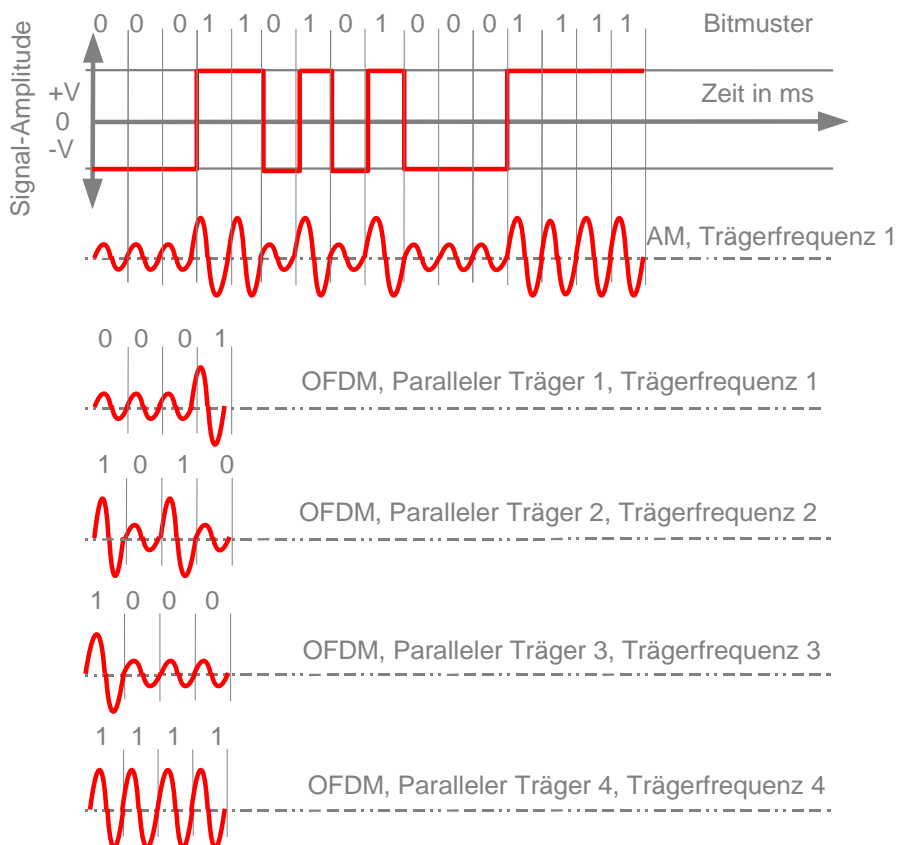


Abbildung 8.31: Das Prinzip der parallelen Übertragung bei OFDM

8.2.4.3 Anwendungsbeispiele

Standard	Verfahren
DVB-S	Für den Satellitenkanal: wenig Laufzeitverzerrung, schwache Signale. Die Modulation ist QPSK; die Symbolrate liegt bei 20...30 MSymbole/s. Eine doppelte FEC, bestehend aus einem Blockcode [Reed-Solomon (188,204,8)], einem Interleaver und einem Faltungscode mit Punktierungen zwischen 1/2 und 7/8, reduziert den Einfluss des additiven Rauschens maximal.
DVB-C	Für das Breitbandkabel: keine Laufzeitverzerrung, sehr wenig Bandbreite. Die Modulation ist QAM, mit 16, 32, 64, 128 oder 256 Symbolen. Die Symbolrate liegt bei knapp 7 MSymbole/s für ein 8-MHz-Raster. Es wird nur der Blockcode [Reed-Solomon (188,204,8)] als FEC verwendet, da die Signale im Kabel sehr störungsfrei sein sollten.
DVB-T	Für die terrestrische Ausstrahlung speziell zu mobilen Empfängern: sehr starke Laufzeitverzerrung, wenig Bandbreite, Fading. Zur Laufzeitverzerrung wird eine OFDM mit 1705 oder 6817 Trägern verwendet; die Modulation auf jedem Träger ist QPSK oder QAM mit 16 oder 64 Symbolen. Die Symbolrate (Summe aller Träger) variiert je nach Kanalbandbreite und Schutzintervall um 5...7 MSymbole/s. Es wird dieselbe doppelte FEC wie bei DVB-S verwendet.
ADSL	Asymmetric Digital Subscriber Line OFDM mit 32 Trägern für den Up- und 190 für den Downstream (jeweils 4,3125 kHz über ca. 1 MHz Bandbreite)
WLAN	54 Mbps-WLAN OFDM mit 52 Trägern nach IEEE 802.11g und nach IEEE 802.11a

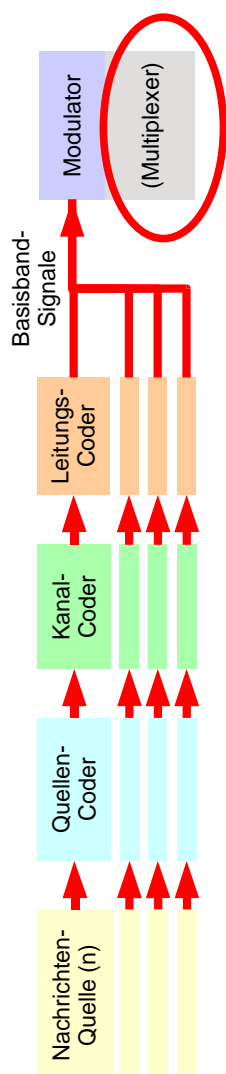


Tabelle 8.7: Einige Anwendungsbeispiele von Modulationen

8.3 Multiplexen

Dieses Verfahren wird hauptsächlich in den Zentralen der grossen Netze angewendet, wo die Daten von mehreren Leitungen mit kleiner Bandbreite auf eine einzige Leitung mit grosser Bandbreite aufgeschaltet werden. Siehe dazu auch das Synchrones Digitale Hierarchie (SDH) in Teil I des Buches. Zwischen den Zentralen werden die Daten auf Netzen mit grosser Bandbreite transportiert (Breitbandnetze) und auf der anderen Seite werden mehrere Teilnehmer von einer Zentrale aus mit Daten versorgt.

Grundsätzlich kann das Multiplexen (Konzentrieren, Verdichten) auf zwei verschiedene Arten geschehen:

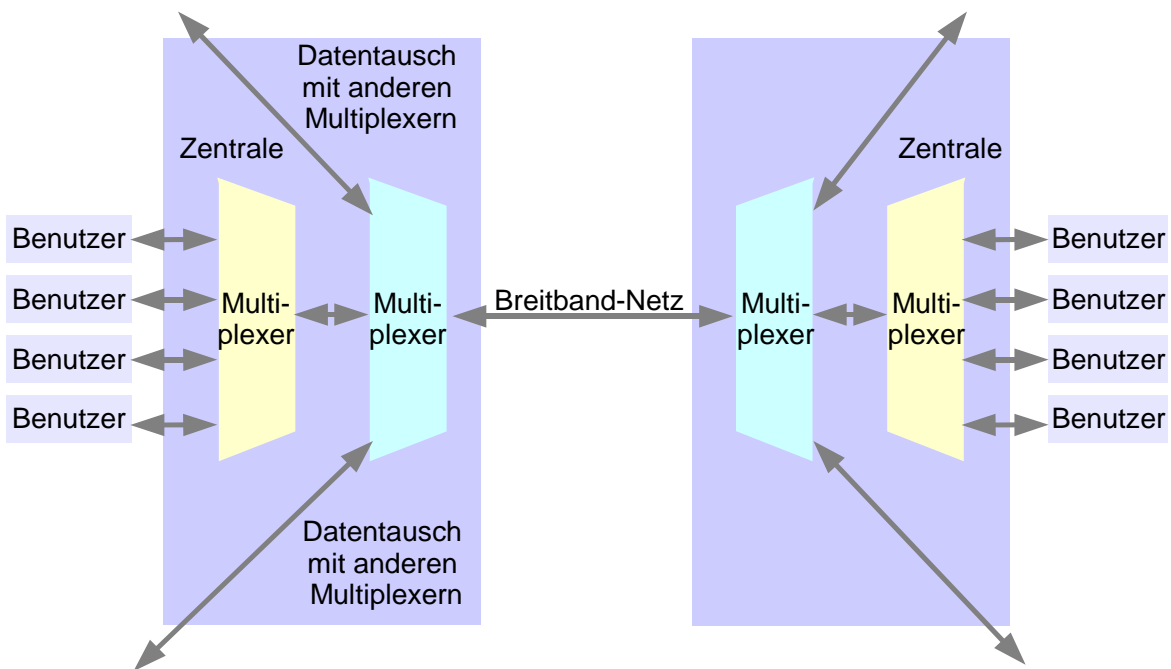


Abbildung 8.32: Grundsätzliches Konzept des Multiplexens

8.3.1 Frequenzmultiplexen (FDM)

Beim Frequenzmultiplexverfahren (Frequency Division Multiplexing, FDM) wird jedem Teilnehmer ein Teil der gesamten Bandbreite (ein Kanal) zur Verfügung gestellt. Die Übertragungsrate ist vor und hinter dem Multiplexer gleich gross.

Breitbandnetze, wie sie die Kabelfernsehbetreiber (CATV) im Einsatz haben, nutzen die Technik des FDM, da es wenig sinnvoll ist, von der Zentrale aus zu jedem Teilnehmer für jeden Fernsehkanal eine eigene Leitung zu unterhalten.

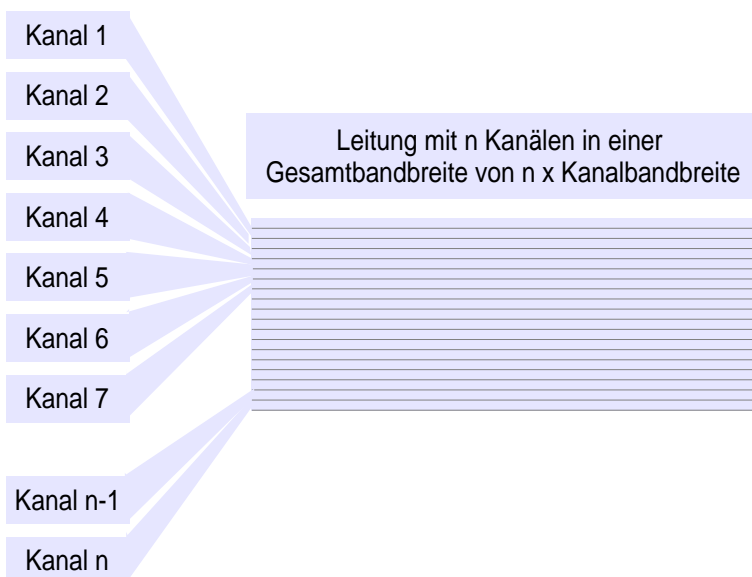


Abbildung 8.33: Das Frequenzmultiplexen

8.3.2 Zeitmultiplexen (TDM)

Beim Zeitmultiplexverfahren (Time Division Multiplexing, TDM) werden von jedem Teilnehmer reihum eine gewisse Zeit lang Daten auf die Leitung geschickt. Er hat somit zeitweise die ganze Bandbreite zur Verfügung. Die Übertragungsrate ist hinter dem Multiplexer n Mal grösser ($n = \text{Anzahl Kanäle vor dem Multiplexer}$). Es können nur digitale Daten mit TDM übermittelt werden.

Digitale Dienste, wie zum Beispiel das ISDN, nutzen diese Technik.

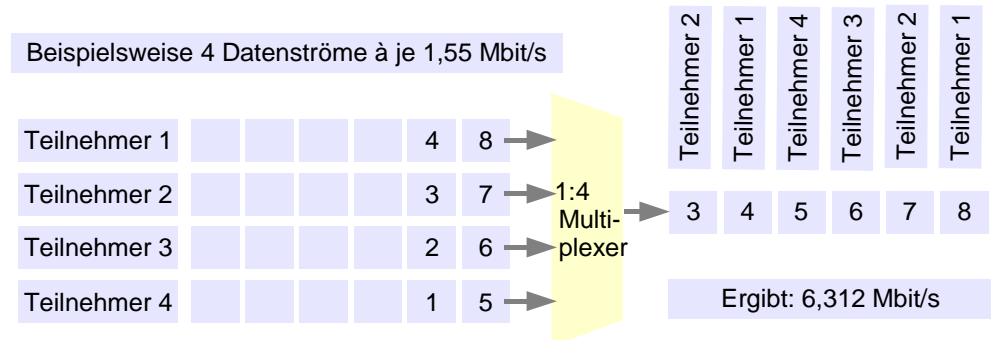


Abbildung 8.34: Das Zeitmultiplexen

Es sind Kombinationen dieser beiden Verfahren möglich, indem auf einem Breitbandnetz die Frequenz-Känäle (FDM) nach einem TDM-Verfahren in so genannte Zeitschlitze eingeteilt werden. Die GSM-Telefonie (Global System for Mobile Communication) benutzt 124 Frequenzkanäle (Duplex) à 200 kHz Bandbreite. Jeder Frequenzkanal wird mit TDM in 8 Zeitschlitze unterteilt (1:8 TDM). Dies ergibt theoretisch $124 \times 8 = 992$ GSM-Kanäle (Duplex) pro GSM-Zelle.

8.3.3 Wellenlängenmultiplexverfahren (WDM)

Beim Wellenlängenmultiplexverfahren (Wavelength Division Multiplexing, WDM) wird das FDM auf Glasfaserkabel angewendet, indem jeder Kanal mit einer eigenen Licht-Wellenlänge in einer einzigen Glasfaser übertragen wird. Die Aufteilung der Kanäle erfolgt in optischen Prismen oder Beugungsgittern.

Im Prinzip können damit mehrere SDH-Kanäle (z.B. OC-48) auf mehreren Wellenlängen auf dem gleichen LWL übertragen werden.

Eine Weiterentwicklung ist das Dense Wavelength Division Multiplexing (DWDM), das heute mehr als 64 Wellenlängenkanäle à je 40 Mbit/s im Protected Mode (links und rechts herum im Ring) übertragen kann. Dies führt zu Übertragungsraten im Terabit/s-Bereich!

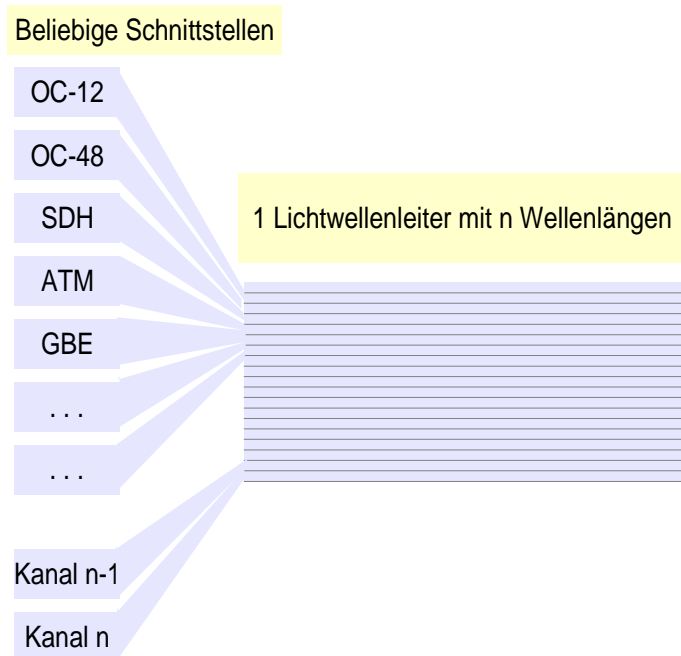


Abbildung 8.35: Das Prinzip des DWDM

8.4 Aufgaben

1. Eine Übertragung von 2 Gbit/s soll auf einer Übertragungsstrecke mit einer Kanal-Bandbreite von 500 MHz übertragen werden. Wie viele Signalelemente sind dazu notwendig? (Rechnungsgang angeben)
2. Ein OFDM mit 512 Trägern, einer 64-QAM-Modulation pro Trägersignal und einer Taktrate (Symboldauer) von zwei Millisekunden sei gegeben. Wie viele Bit/s lassen sich damit übertragen? (Rechnungsgang angeben)
3. Weshalb ist QPSK mit 4-QAM identisch? Bitte in wenigen Sätzen begründen.

Lösungen unter www.sauerlaender.ch/downloads

9 Übertragungsarten

Informationen, die ein Benutzer mithilfe einer Maschine übertragen will, werden letztlich in binärer Form mithilfe eines elektrischen Leitungssystems transportiert. Die Datenübertragung kann seriell oder parallel, durch Multiplexen sowie asynchron oder synchron geschehen. Hier geht es um Übertragungsarten auf der Schicht 1.

9.1 Serielle und parallele Übertragung

Datenströme können sowohl seriell als auch parallel übertragen werden. Serielle Übertragungen werden in vielen Schnittstellen am PC aber auch im LAN eingesetzt.

9.1.1 Serielle Übertragung

Bei der seriellen Übertragung werden die Daten in Bitströme zerlegt und hintereinander bitweise übertragen (Abbildung 9.1).

Die weit verbreitete RS-232-Schnittstelle verwendet dieses Verfahren, wie auch deren Nachfolgerin, die RS-449 oder der Universal Serial Bus (USB) an den PCs.

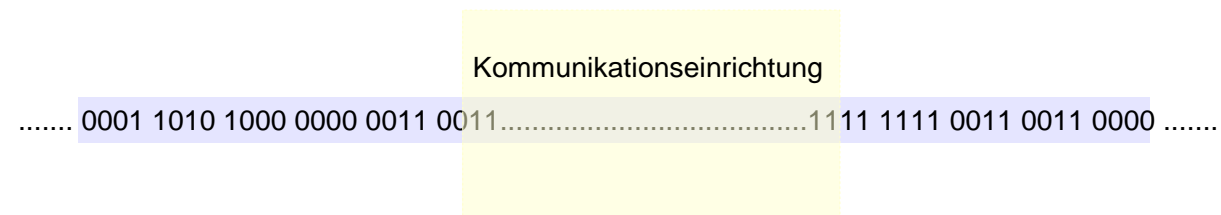


Abbildung 9.1: Serielle Übertragung eines Bitstromes

9.1.1.1 Beispiel RS 232

Die Norm RS-232 wurde von der Electronic Industries Association (EIA) aufgestellt und existiert mittlerweile in der Fassung RS-232 C. In der Norm sind die mechanische, die elektrische, die funktionale und die verfahrenstechnische Spezifikation festgelegt.

Die *mechanische Spezifikation* definiert die Masse des 25- respektive 9-poligen Steckers.

Die *elektrische Spezifikation* legt fest, dass eine Spannung von weniger als -3 bis $-15V$ eine binäre Eins und eine Spannung von mehr als $+3$ bis $+15V$ eine binäre Null ergibt.

Die *funktionale Spezifikation* legt fest, welche Signale an welchem Anschluss (Pin) des Steckers anliegen.

9.1.1.2 Beispiel USB

Der USB (Universal Serial Bus) hat sich zu einer Standardschnittstelle für periphere Geräte aller Art entwickelt. Die heutigen Multimediageräte weisen vorwiegend USB-Anschlüsse auf.

Applikation
7 Anwendung
6 Darstellung
5 Sitzung
4 Transport
3 Vermittlung
2 Sicherung
1 Bitübertragung
Übertragungsmedien

Praxis-Hinweis:

Die maximalen Datenraten sind 20 kBit/s und es sind Kabellängen bis zu 15 m erlaubt.

Die Geräte können dank Hot-Plugging zu jeder Zeit während dem Betrieb des PCs in den USB-Hostadapter des Rechners eingesteckt werden. Durch Plug&Play werden die eingesteckten Komponenten sofort erkannt und die Grundeinstellungen werden dann durch das Betriebssystem vorgenommen. Es müssen keine spezifischen Einstellungen mehr vorgenommen werden wie Jumperbelegung, korrekte Terminierung oder Protokolleinstellungen.

9.1.1.2.1 Host

Es gibt nur einen Host in jedem USB-System. Das USB-Interface zum Host-Computer-System wird als Host-Controller bezeichnet, welcher normalerweise als Kombination von Hardware, Firmware und Software implementiert wird. Ein so genannter Root-Hub ist im Host-System integriert, um schon Anschlussmöglichkeiten für ein oder mehr Endgeräte zu ermöglichen.

Am Hostadapter (Anschluss Steckertyp A) können bis zu 127 Geräte (Steckertyp B) mithilfe eines oder mehrerer USB-HUBs angeschlossen werden.

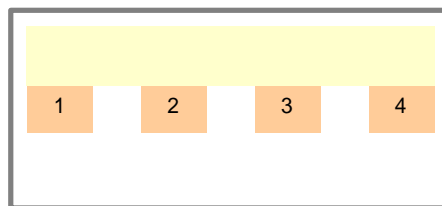


Abbildung 9.3: Steckertyp A

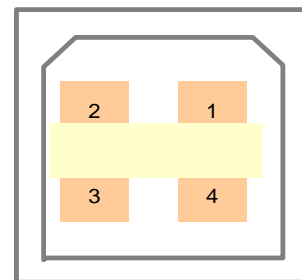


Abbildung 9.2: Steckertyp B

Pin	Beschreibung	Farbe
1	Vbus 5V DC	Rot
2	D- Daten-	Weiss
3	D+ Daten+	Grün
4	GND Erde/Masse	Schwarz

Tabelle 9.1: Die Pinbelegung der Stecker

Für die modernen mobilen Geräte sind kleine Anschlüsse gefordert. Diese wurden mit Mini-A, Mini-B oder Mini-AB realisiert.

Geräte, die wenig Strom verbrauchen (bis 500mA), wie USB-Memorystick, Cardreader oder kleine Festplatten, können durch die von der Schnittstelle gelieferte Spannung von 5V ohne weitere Stromversorgung betrieben werden. Die Daten werden über ein Paar bidirektionale Leitungen gesendet (PIN 2+3). Das Kabel ist ein 4-adriges Twisted-Pair-Kabel.

9.1.1.2.2 Datenübertragung

Die Datenübertragung erfolgt paketerorientiert. Die einzelnen Frames werden bei USB 1.x im Millisekundentakt übertragen, während USB 2.0 diese jeweils weiter in 8 Hi-Speed-Frames zu 125µs unterteilt.

USB	Geschwindigkeit	Übertragungsrate	Länge	Kabel
1.1	Low	1.5Mb/s	3m	UTP
1.1	Medium	12Mb/s	5m	STP
2.0	High	480Mb/s		STP

Tabelle 9.2: Die verschiedenen USB-Varianten (USB 2.0 ist abwärtskompatibel.)

9.1.1.2.3 USB-Baum

Ein Anschluss an einem Hub wird als Port bezeichnet. In einem USB-System kann es mehrere Hubs geben. Der Upstream-Port verbindet den Hub mit einem anderen Hub näher am Host oder direkt mit dem Host. Alle weiteren (Downstream-)Ports ermöglichen den Anschluss eines beliebigen USB-Geräts.

9.1.1.2.4 USB-Hub

Der USB-Hub besteht hardwaremässig aus dem Hub-Controller und dem Hub-Repeater. Ein Repeater ist ein protokollgesteuerter Schalter zwischen Upstream- und Downstream-Ports. Er besitzt ausser-

Praxis-Hinweis:

Hubs sind in der Lage, neu angeschlossene oder wieder entfernte Geräte automatisch zu erkennen und stellen die Energieversorgung für das entsprechende Gerät sicher. Ports von langsamen und normalen USB-Geräten werden voneinander isoliert.

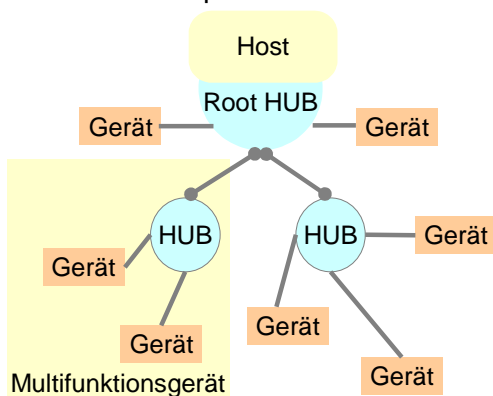


Abbildung 9.4: Der USB-Baum

dem Reset- und Energiesparfunktionen. Der Hub-Controller stellt Register zur Verfügung, um die Kommunikation mit dem Host zu ermöglichen. Spezifische Kontrollbefehle erlauben es dem Host, den Hub zu konfigurieren und seine Ports zu überwachen.

9.1.1.3 Beispiel IEEE1394 (Firewire)

Die IEEE 1394-Technologie, zunächst „Firewire“ genannt, bezeichnet eine verhältnismässig neue (seit 1995) serielle Schnittstellentechnologie für Computer- und Videogeräte zur Übertragung digitaler Daten mit bis zu 400 Mbit/s (IEEE1394A) und 800 Mbit/s (IEEE1394B). Die

Initiative und das Grundkonzept gehen auf die Firma Apple zurück. 1997/1998 benennt Sony die IEEE 1394-Schnittstellen der firmeneigenen Produkte von „FireWire“ in „i.LINK“ um.

9.1.1.3.1 IEEE-1394 im Überblick

1. Paketorientierte Datenübermittlung
2. Geräte-Adressierung über Software
3. Hot-Pluggable
Der Anwender kann 1394-Geräte ohne Werkzeug während des Systembetriebs anschliessen oder entfernen.
4. 63 Geräte anschliessbar
5. Bidirektional (Datenübertragung in beide Richtungen)
6. Datentransferraten von 100, 200, 400 und 800 Mbit/s beziehungsweise 12,5, 25 oder 50 MByte/s
7. Gemischter Betrieb unterschiedlich schneller Geräte mit 100, 200, 400 und 800 Mbit/s möglich
8. Dünne und preiswerte serielle Kabel
9. Einfache Konfiguration, da keine Abschlusswiderstände, Geräte-IDs oder Einstellungsverfahren notwendig sind.
10. Die Spannungsversorgung der Geräte ist über das Datenkabel möglich. Dafür sind zwischen 8 bis 40 Volt bei maximal 1,5 Ampere vorgesehen.
11. Als Peer to Peer-Netzwerk benötigt 1394 keinen dedizierten Host. Bei USB fungiert der PC als Host.

Die Firewire-Schnittstelle wurde ursprünglich benutzt, um digitale Camcorder an digitale Video-Hardware anzuschliessen. Es können aber auch Festplatten über Firewire angeschlossen werden.

IEEE 1394-1995 ermöglicht theoretisch eine Datentransferrate von bis zu 50 MB/s, mit IEEE 1395b bis 400 MB/s.

Prinzipiell sind Firewire, IEEE 1394-1995, i.Link und Lynx (Luchs) kompatibel zueinander oder nur andere Bezeichnungen für dasselbe. Allerdings unterscheidet sich i.Link schon äusserlich durch den kleineren vierpoligen Steckverbinder von den sechspoligen IEEE- und Firewire-Originalen. Der Unterschied: i.Link bietet keine Spannungsversorgung für externe Geräte über das 1394-Kabel.

9.1.1.3.2 Anschluss der Geräte

Prinzipiell ist 1394 eine simple Sache. Ein oder mehrere Geräte mit dem seriellen Kabel an den PC/Controller anschliessen und fertig. Abgesehen von den Unwägbarkeiten eines Plug&Play-Betriebssystems, gibt es noch einige Einschränkungen. Die Geräte sind seriell hintereinander geschaltet wie in einer Kette. Bei Geräten mit zwei oder mehr 1394-Ports sind Verzweigungen möglich.

Praxis-Hinweis:

Datenübertragung

Die Übertragung erfolgt paketorientiert. Ab 2002 erfolgt die Übertragung mit 8B10B-Leitungscode, welcher eine grössere Leitungslänge zulässt.

Der Anschluss erfolgt über ein flexibles 6-adriges STP-Kabel (4 Adern für Datentransfer, 2 Adern für Stromversorgung) oder 4-adriges Kabel (nur Signalleitungen).

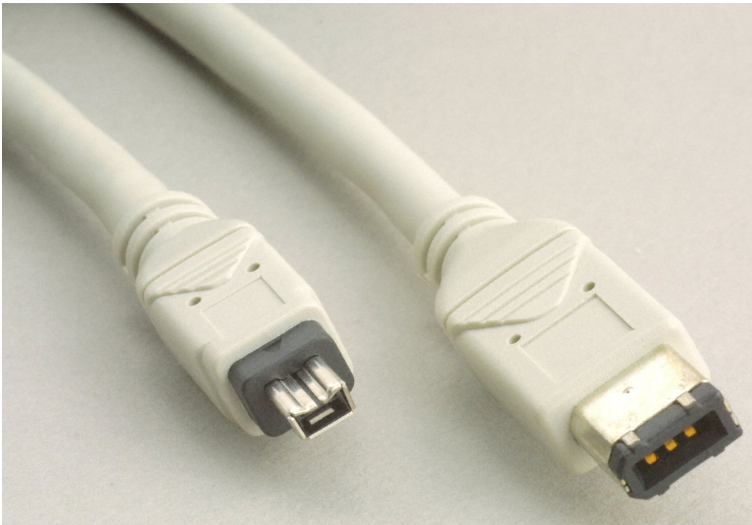


Abbildung 9.5: Das IEEE 1394-Kabel

9.1.2 Parallele Übertragung

Bei der parallelen Übertragung werden Informationen über mehrere Leitungen parallel übertragen. Ein Byte (8 Bit), Wortlängen von 16 Bit oder noch grössere Multiple eines Bytes werden gleichzeitig übermittelt (Abbildung 9.6).

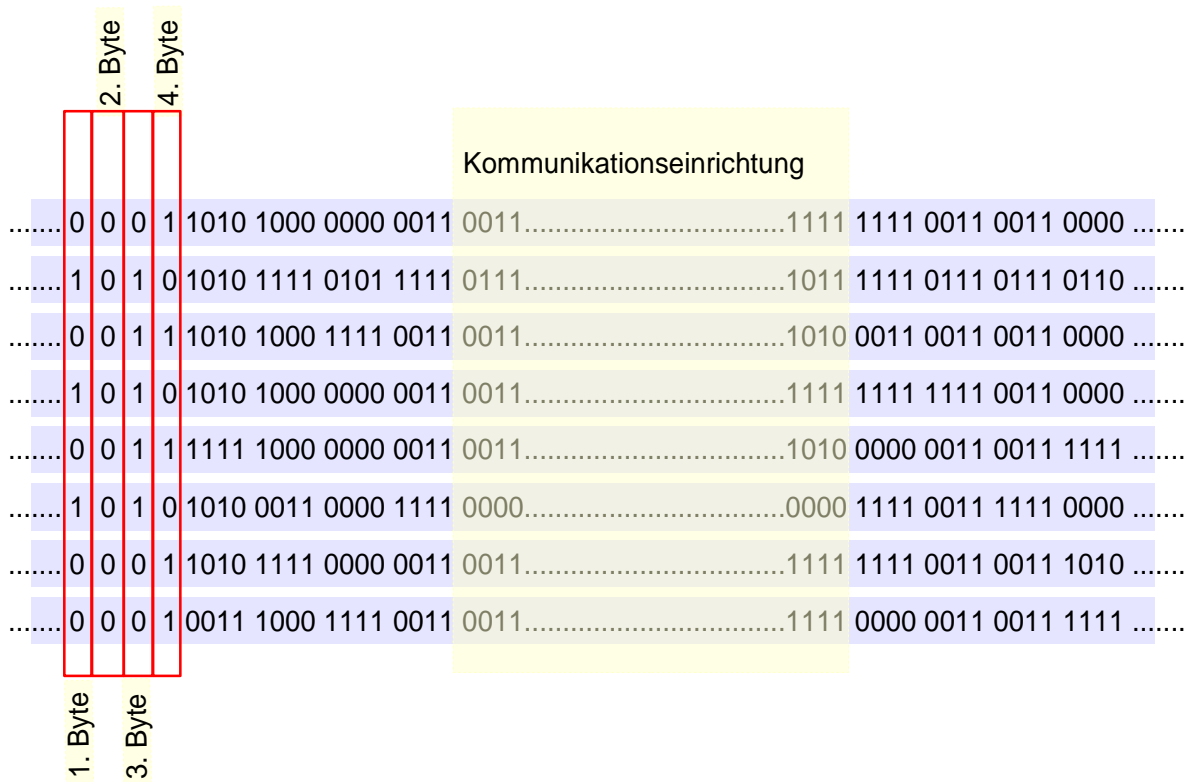


Abbildung 9.6: Parallele Übertragung von 8 Bitströmen (= 1Byte) gleichzeitig

Anwendungen: Centronics-Schnittstelle, IEC-Bus (auch bekannt unter dem Namen IEEE-488 oder GP-IB) und sämtliche System- und Peripheriebusse in Computern (Micro Channel (MCA), (E)ISA, PCI, VME-Bus, SCSI, Multibus I + II, NUBUS etc.).

9.2 Zeichenorientierte/Bitorientierte Übertragung

Es gibt grundsätzlich die Möglichkeit, Daten als Zeichen oder als Bitstrom zu übertragen. Beide Möglichkeiten sind in der Praxis häufig anzutreffen und werden im Folgenden erklärt. In beiden Fällen ist eine Strukturierung der Ströme sinnvoll für die effizientere Handhabung. Die Strukturierung der Zeichenströme in Worte mit klar definiertem Anfang und Ende sowie die Strukturierung der Bitströme in Frames (Rahmen) macht die Handhabung einfacher. Dies ist vergleichbar mit dem Sprechen in Worten und Sätzen – man stelle sich vor, die Menschen würden in Buchstaben und Leerzeichen kommunizieren!

Praxis-Hinweis:

Die Übertragung erfolgt meistens asynchron. (Eine synchrone Übertragung ist jedoch auch denkbar.) Der Empfänger kann bei der asynchronen Übertragung die Zeichen anhand der Start- und Stopp-Bits unterscheiden.

9.2.1 Zeichenorientierte Übertragung

Es existieren Geräte, deren Daten als Zeichen anfallen. Anzutreffen ist dies z.B. in Tastaturen, Steuerworte für die Steuerung von Maschinen und Inhalte von Bildschirmmasken in EDV-Anlagen. Die Datenkommunikationseinrichtungen, welche solche Nachrichten übermitteln, tun dies auf der Basis von einzelnen Zeichen. Die Zeichen (englisch: character) werden anhand einer Code-Tabelle (zum Beispiel dem ASCII-Code) in 7-Bit, 8-Bit oder neuerdings noch mehr Bit umcodiert. Acht-Bit codierte Zeichen heissen auch Byte. Diese codierten Zeichen werden als Bitstrom übertragen und vom Empfänger anhand der gleichen Code-Tabelle wieder in Zeichen umgewandelt. Damit der Sender in einer zeichenorientierten Übertragung den Anfang und das Ende eines Textes markieren kann, werden besondere Steuerzeichen verwendet: STX für Start of Text und ETX für End of Text. In einer beliebigen zeichenorientierten Übertragung können aber STX und ETX zufälligerweise auch vorkommen. Es ist unschwer zu erkennen, dass dies beim Empfänger zu einer grösseren Katastrophe führen würde. Aus diesem Grund werden vom Sender vor allen Steuerzeichen DLE (Data Link Escape) gesetzt, um die Steuerzeichen eindeutig zu markieren. Leider können auch diese DLE in einem beliebigen Text vorkommen. Damit auch in diesem Fall keine Fehlinterpretationen beim Empfänger vorkommen können, wird das Verfahren des character stuffing (= Zeichen stopfen) eingesetzt. Der Sender fügt somit nicht nur vor jedem STX oder ETX ein DLE ein, sondern auch vor jedem DLE, das er vor dem „DLE-Stopfen“ im Text findet. Der Empfänger entfernt vor dem Lesen der Nachricht alle DLE

vor den Steuerzeichen. Findet er eine Folge von zwei DLE, weiss er, dass das erste gestopft ist und das zweite zum Text gehört.

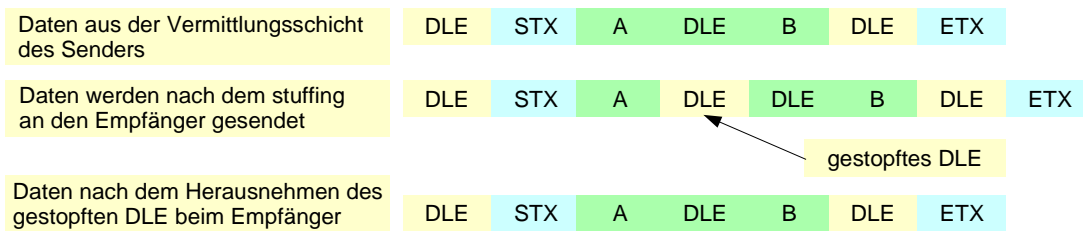


Abbildung 9.7: Stopfen von DLE im Zeichenstrom

9.2.2 Bit orientierte Übertragung

Bei der bitorientierten Übertragung liegen die Daten beim Sender schon als Bitstrom vor und könnten problemlos übertragen werden. Damit die Datenübertragung grosser Bitströme (grosse Dateien mit mehreren MByte) abgesichert erfolgen kann, muss der Bitstrom – ähnlich den Zeichen oder Bytes bei der zeichenorientierten Übertragung – strukturiert werden. Dies ist notwendig, damit zum Beispiel bei einem Unterbruch oder einem Fehler in der Übertragung die Kommunikation nicht neu gestartet und somit die ganze Datei erneut übertragen werden muss. Eine Einteilung in kleinere Einheiten ermöglicht nach dem erneuten Aufbau der Leitung oder der Beseitigung der Störung ein Fortführen der Übertragung.

In der Telematik existieren zwei Möglichkeiten zur Strukturierung der Bitströme. Die Bits werden in Frames (Rahmen) eingepackt oder in Cells (Zellen) organisiert. Diese Strukturierung erfolgt in der Schicht 2 des ISO/OSI-Modells. Die damit zusammenhängenden Verfahren und Protokolle werden in den nächsten zwei Kapiteln näher erläutert. Damit der Empfänger erkennen kann, wann ein Frame beginnt und wann es zu Ende ist, werden Eröffnungs- und Endmarken in dem Bitstrom eingefügt (siehe Abbildung 9.8). Diese Eröffnungsmarken (opening flag) und Endmarken (closing flag) haben beispielsweise das Bitmuster 01111110.

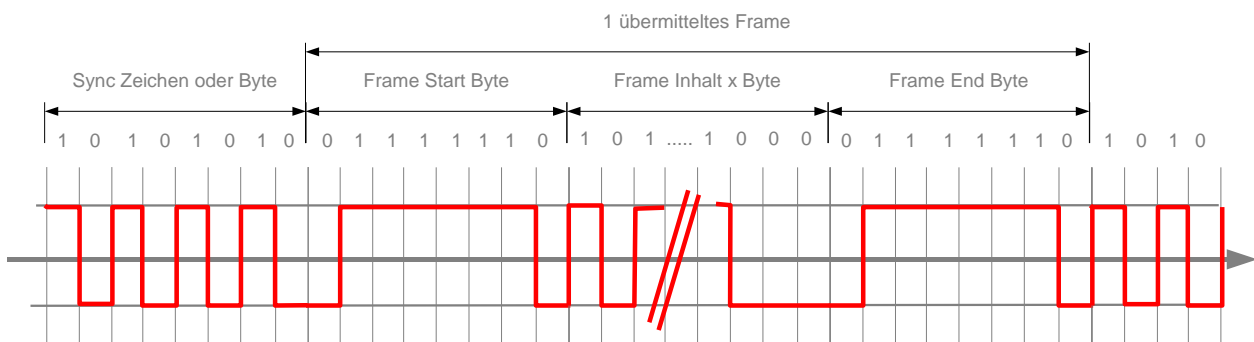


Abbildung 9.8: Die Unterteilung des Bitstromes in Frames (Rahmen)

Dummerweise ist es keine Seltenheit, dass in einem normalen Bitstrom das Bitmuster 01111110 vorkommt. Der Empfänger würde dieses Bitmuster natürlich als Marke erkennen und die Übertragung würde zusammenbrechen. Um dies zu verhindern, wird das bit stuffing (= Bit stopfen) eingesetzt (siehe Abbildung 9.9). Sobald der Sender in einem Bitstrom eine Folge von mehr als 5 Einsen erkennt, fügt er, vor der Einteilung des Bitstromes in Frames, nach 5 Einsen eine 0 ein. Der Empfänger entfernt nach der Erkennung der Frames alle Nullen, welche nach einer Folge von 5 Einsen im Bitstrom ankommen und erhält somit den ursprünglichen Bitstrom.

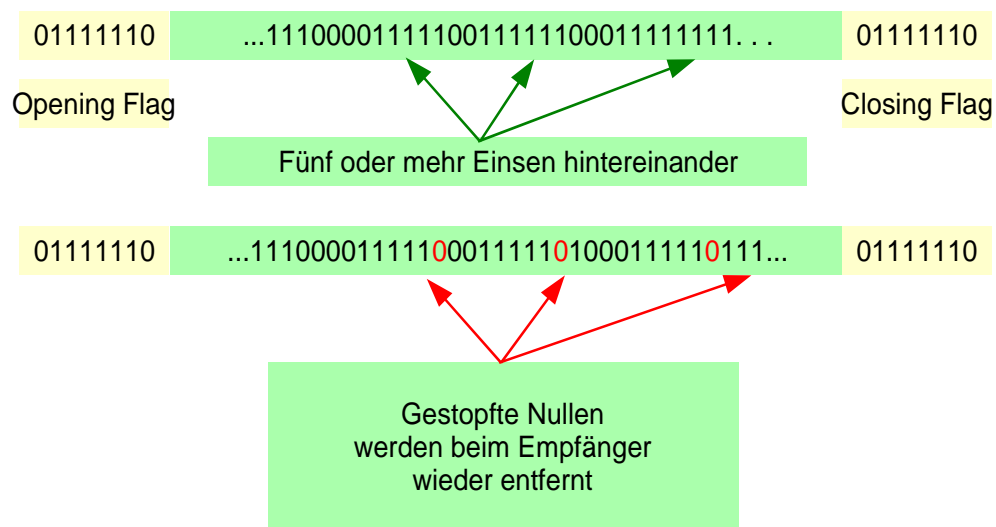


Abbildung 9.9: Bit stuffing zur Vermeidung von Missverständnissen

9.3 Asynchrone und synchrone Übertragung

Es ist unschwer einzusehen, dass es Geräte gibt, die konstant Daten generieren (wie zum Beispiel Messwerte einer Temperaturmessung) und solche, bei denen die Daten nicht kontinuierlich anfallen (wie zum Beispiel die Bewegung einer Maus oder die Daten aus einem Keyboard). Aus diesem Grund existieren auch zwei unterschiedliche Datenübertragungsarten – für den ersten Fall die synchrone und für den zweiten Fall eher die asynchrone Übertragung. Beide Fälle werden im Folgenden beschrieben.

9.3.1 Asynchrone Datenübertragung

Diese Übertragungsart wird häufig dort eingesetzt, wo die zu übertragenden Daten nur zeitweise anfallen (asynchron), wie z.B. bei der Dateneingabe per Tastatur. Hier ist typisch, dass nach dem Drücken einer Taste eine gewisse Zeit verstreicht bis zur nächsten Eingabe und die Übermittlung zum Computer somit nicht kontinuierlich erfolgt. Für die Datenübertragung heisst das, dass nach dem Übertragen einer Bitfolge (zum Beispiel acht Bit) eine gewisse Wartezeit (engl. id-

ling) vorliegt. Damit der Empfänger im Computer weiss, wann er das nächste Byte empfangen soll, muss das Byte mit einem Startbit angemeldet werden und am Schluss des Byte mit Stopbits abgemeldet werden.

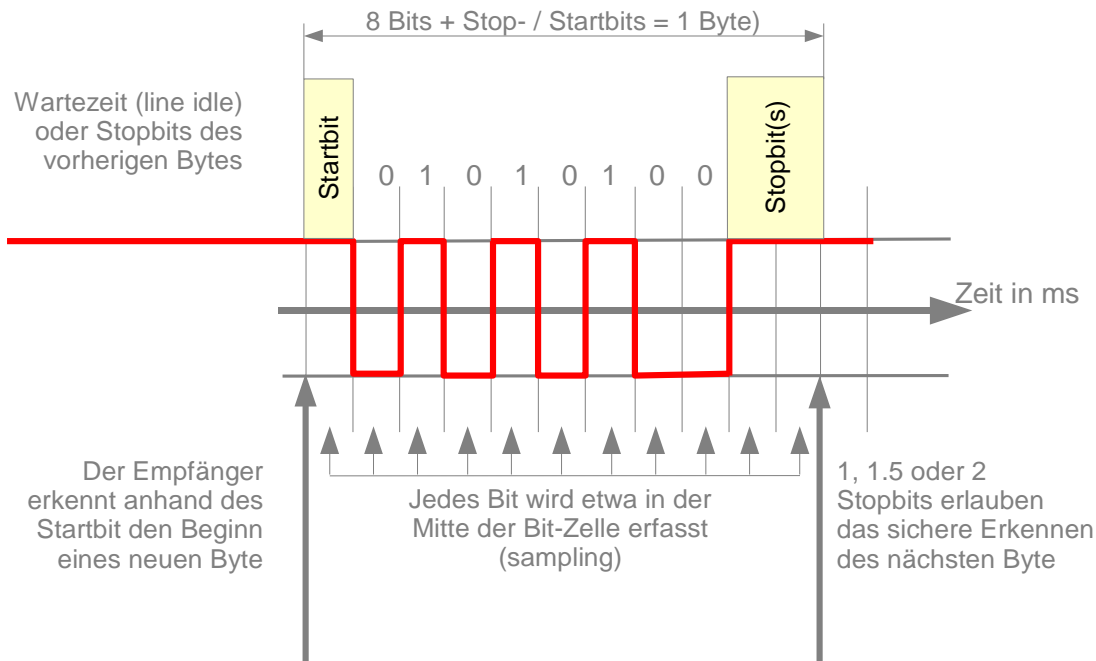


Abbildung 9.10: Die Übertragung eines Byte

In Abbildung 9.10 wird dieser Zusammenhang grafisch dargestellt (hier wird ein Byte, bestehend aus acht Bit, übertragen).

9.3.2 Synchrone Datenübertragung

Diese Übertragungsart eignet sich besonders für grössere Dateien oder grössere Übertragungsraten.

Im Gegensatz zur asynchronen Übertragung wird hier ein kontinuierlicher Datenstrom übertragen. Dies führt in der Praxis zu einer Schwierigkeit: Die Übertragung der Daten (Bits) erfolgt mittels elektronischer Schaltungen. Diese Schaltungen müssen auch bei hohen Übertragungsraten sowohl beim Sender als auch beim Empfänger genau wissen, wann gültige Bits zu schreiben respektive zu lesen sind. Die beiden Kommunikationsgeräte müssen das „gleichzeitig“, eben synchron tun. Damit der Empfänger weiss, wann gültige Bits ankommen, muss eine Synchronisationsinformation (Clock) mit übertragen werden.

Dies kann auf verschiedene Weise geschehen:

1. Abbildung 9.11 zeigt, wie auf einem separaten Leiter die Synchronisationsinformation parallel zu den Daten übertragen wird (out band).

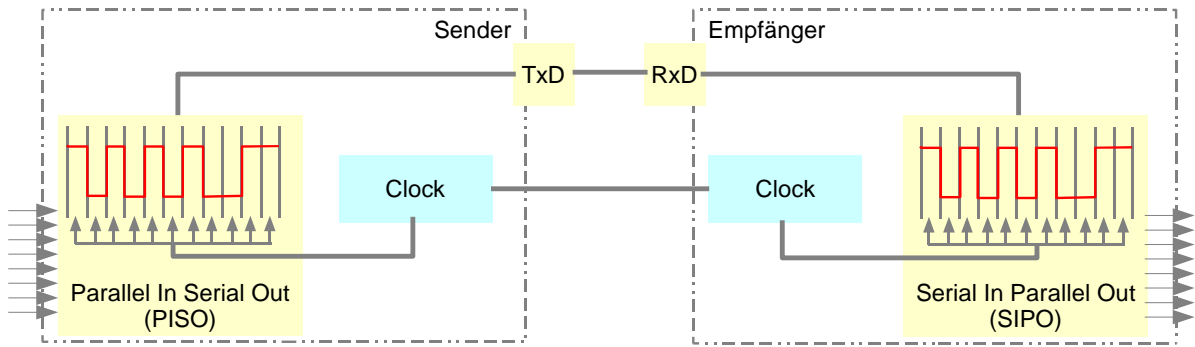


Abbildung 9.11: Clockdaten out band

2. Es werden spezielle Zeichen für die Clock-Information reserviert (in band).

3. Es werden einzelne spezielle Bits dazu verwendet (in band).

Damit diese in band clock-Übertragung möglich wird, muss der Bitstrom unterteilt und in so genannten Frames (Rahmen) zusammengefasst. Wenn keine Daten zur Übermittlung anfallen, werden die Frames mit Synchronisationszeichen (sync characters oder sync bytes) oder Synchronisationsbytes aufgefüllt. Jeder Frame hat ein Startzeichen oder einige Startbytes und ein Endzeichen oder Endbytes (siehe Abbildung 9.8).

Eine Möglichkeit, wie sie in der Praxis anzutreffen ist, um Datenströme synchron auf einer Leitung zu übertragen, zeigt Abbildung 9.12.

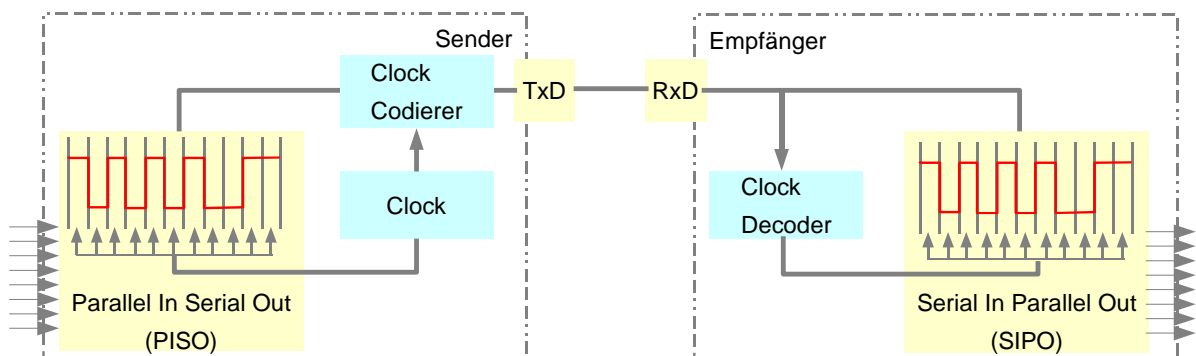


Abbildung 9.12: Das Clocksignal wird mit Frames in den Datenstrom gepackt.

Die oft parallel anfallenden Daten werden in einem Parallel/Seriell-Wandler (PISO, Parallel In Serial Out) umgewandelt und in einem Codierer mit einer Taktinformation (Clock) versehen. Die Daten werden als Frames strukturiert durch den Ausgang, Transmit Data (TxD), zum Eingang, Receive Data (RxD), geschickt. Der Empfänger muss diese Taktinformation mit einem Clock Decoder wieder herausfiltern, um den ursprünglichen Datenstrom zu erhalten.

9.4 Ankoppelung an die Schicht 2

Die Schicht 1 wird unterteilt in die beiden Teilschichten „Physical Medium Attachment“ (PMA) und „Physical Layer Signaling“ (PLS). In der ersten Teilschicht werden die Schnittstellen zu den verschiedenen Übertragungsmedien und in der zweiten Teilschicht wird die Signalerzeugung und die Ankoppelung an die Schicht 2 (Medium Access Control, MAC) definiert. In den Normen werden beispielsweise die Pinbelegung der Buchsen und Stecker, die Anzahl der benutzten Leitungen, die Art der Übertragung (seriell, parallel), die Funktionen der Steuerleitungen, die Trägerfrequenz und andere Funktionen beschrieben. Auf keinen Fall sind die Übertragungsmedien selbst Gegenstand der Schicht 1!

Abbildung 9.13 zeigt in einer groben Übersicht die Zusammenhänge der beiden Teilschichten und einige in der Praxis bei verschiedenen Implementationen eingesetzte Protokolle (AUI, MAU und MDI).

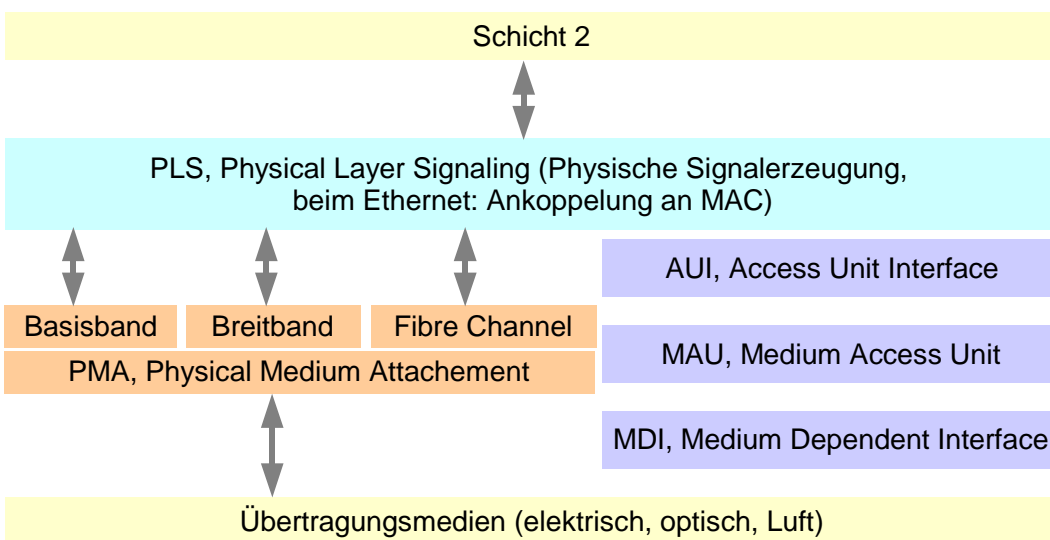


Abbildung 9.13: Die Ankoppelung der Schicht 2

9.5 Aufgaben

1. Gegeben sei eine asynchrone Datenübertragung mit serieller Schnittstelle (RS232):
Ein Zeichen von 8Bit wird mit einer Übertragungsrate von 19200 Bit/s im Paritätsmodus übertragen. Wie gross ist die effektive Zeichenübertragungsrate?
2. Welche zusätzliche Steuerinformation wird für eine synchrone Datenübertragung benötigt?
3. Wie viele Hosts können in einem USB-System aktiv sein?
4. Wie erfolgt die Datenübertragung in einer USB-Verbindung?
5. Wie kann ein USB-Anschluss erweitert werden?
6. Wie wird die USB-Übertragungsklasse erkannt?
7. Wie viele Geräte können an einem USB-System angeschlossen sein?

Lösungen unter www.sauerlaender.ch/downloads

10 Übertragung der Bitströme

Das Kapitel behandelt die Strukturierung und Sicherung der Bitströme. Diese Verfahren sind notwendig, damit die Schicht 2 ihre Hauptfunktion des Aufbaus, des Betriebs und des Abbaus der Punkt-Punkt-Verbindung wahrnehmen kann.

Die Strukturierung geschieht mithilfe von Rahmen (engl. frames) oder Zellen (eng. cells).

Die Übertragungssicherung wird mit Übertragungs-Steuerungen erreicht. Die verschiedenen Möglichkeiten werden erläutert.

Im Weiteren werden die Übertragungsprotokolle beschrieben.

10.1 Strukturierung der Bitströme

Die Übertragung endloser unstrukturierter Bitströme ist nicht empfehlenswert. Dies wäre vergleichbar mit einer Rede, die nicht in Worte, Sätze und Abschnitte gegliedert ist, sondern als eine Aneinanderreihung von Buchstaben gehalten würde. Die Zuhörer hätten wohl kaum eine Chance, den Sinn der Rede erfassen zu können. Auch eine Fehlererkennung wäre problematisch. Genau so würde es aber einer Kommunikationseinrichtung ergehen, wenn man die Bitströme nicht strukturieren würde.

10.1.1 Rahmen (engl. frames)

Eine häufige Art der Strukturierung erfolgt mit Frames. Solche Frames sind z.B. im HDLC (High Level Datalink Control) Protokoll beschrieben. Dieses Protokoll benutzt die folgende Frame-Struktur:

Marke Flag	Adressen	Steuerung	Daten / Information	Rahmen Prüfsumme	Marke Flag
01111110	Sender Empfänger	RR	xyzxyzxyzxyz	CRC	01111110
8 Bit	8 oder mehr Bit	8 oder 16 Bit	Mehr als 0 Bit	16 oder 32 Bit	8 Bit

Abbildung 10.1: Ein HDLC-Frame

Die **Marke** (Flag) ist mit 01111110 festgelegt. Alle Stationen, die an das Netz angeschlossen sind, versuchen kontinuierlich, diese Flags zu finden und sich damit zu resynchronisieren. Selbstverständlich kann dieses Bitmuster auch innerhalb der Information vorkommen. Damit in diesem Fall keine Fehler auftreten, wird Bit-stuffing angewendet.

Applikation

7 Anwendung

6 Darstellung

5 Sitzung

4 Transport

3 Vermittlung

2 Sicherung

1 Bitübertragung

Übertragungsmedien

Das **Adressfeld** gibt an, welche Station (Empfänger) das Frame erhalten soll oder welche Station (Sender) das Frame generiert und auf das Netz gegeben hat. 11111111 ist eine reservierte Adresse und wird für das Broadcasting in Netzen verwendet (eine Station sendet eine Nachricht an alle).

Das Feld **Steuerung** bedeutet: Ein HDLC-Frame kann für verschiedene Aufgaben eingesetzt werden:

1. Es kann für die Übertragung von Nachrichten (Daten) benutzt werden. Dies mit so genannten Informations-Frames.
2. Die Verbindung kann mit Überwachungs-Frames überwacht werden.
3. Der Aufbau, der Betrieb und der Abbau der Verbindung wird mit so genannten unnummerierten (unnumbered) Frames gesteuert.

Damit der Empfänger versteht, auf welche Weise der Sender die Frames an ihn senden will, muss an dieser Stelle im Frame festgehalten werden, welcher Frametyp vorliegt.

Ein **Informations-Frame** wird Angaben zur Flusskontrolle (z.B. Sequenznummer und Angaben zum Timeout) enthalten.

Ist das Frame ein **Überwachungs-Frame**, so enthält es an dieser Stelle nur Angaben der Flusskontrolle. Es gelangen beispielsweise folgende Befehle zur Anwendung:

RR Receiver Ready
RNR Receiver Not Ready
REJ Reject
SREJ Selective Reject

Unnummerierte Frames halten den Betriebsmodus der Übertragung fest. HDLC kennt drei Betriebsmodi:

1. Den Normal Response Mode (NRM), bei dem ein Host seine Slaves zum Senden auffordern muss. Es können sowohl Punkt-zu-Punkt-Netze als auch Multipoint-Netze eingesetzt werden.
2. Den Asynchronous Response Mode (ARM), bei dem auch ein Slave ohne Erlaubnis vom Host mit der Übertragung von Daten beginnen kann. Dies erfolgt normalerweise in Duplex-Punkt-zu-Punkt-Netzen.
3. Den Asynchronous Balanced Mode (ABM), bei dem es nur gleichberechtigte Kommunikationsteilnehmer (auf Duplex-Punkt-zu-Punkt-Netzen) gibt.

Die Frames teilen den gewählten Betriebsmodus der Gegenstation mit den folgenden Anweisungen mit:

SARM Set Asynchronous Response Mode
SARME Set Asynchronous Response Mode Extended
SNRM Set Normal Response Mode

SNRME Set Normal Response Mode Extended
 SABM Set Asynchronous Balanced Mode
 SABME Set Asynchronous Balanced Mode Extended
 Zusätzlich enthalten sie die folgenden Link-Control-Befehle und -Funktionen sowie Flusskontrollzeichen:
 RSET Reset
 FRMR Frame Reject
 DISC Disconnect
 UA Unnumbered Acknowledge
 CMDR Command Reject
 DM Disconnect Mode

Die **Information** enthält die Daten (Bitfolge der Nachricht).
 Der **Prüfbitrahmen** enthält die „cyclic redundancy checksum“ (CRC, Fehlerbehandlung).

10.1.2 Zellen – Cells

Im ATM werden Zellen à 53 Byte eingesetzt. Diese Zellen haben im Gegensatz zu den HDLC-Frames eine fixe Grösse! Der Header der Zelle ist daher auch immer 5 Byte (40 Bit) gross und folgendermassen strukturiert:

GFC	VPI	VCI	PT	CLP	HEC	Daten / Information
1010	10010011	11001100 11001100	100	1	10010011	xyzxyzxyzxyz
4 Bit	8 Bit	16 Bit	3 Bit	1 Bit	8 Bit	Genau 48 Byte
Header: Genau 40 Bit = 5 Byte						
Zellengrösse: Genau 53 Byte						

Praxis-Hinweis:

Damit der Empfänger die Steuerbits der verschiedenen Frameteile (vor allem die Flags!) einwandfrei von allfällig gleichlautenden Bitfolgen im Informations-Teil (Daten) unterscheiden kann, gelangt hier das „bit stuffing“ zum Einsatz. Es ist hier noch wahrscheinlicher, dass eine Bitfolge im Datenteil genau der Bitfolge eines Flags entspricht. Der Empfänger würde die Bitfolge als Anfang oder Ende des Frames beurteilen und die Verbindung würde abgebrochen.

Abbildung 10.2: Das Format einer ATM-Zelle (UNI)

- GFC: 4 Bits für den *generic flow control*. Dieser wird lokal benutzt, um mehrere Stationen zu identifizieren, die alle das gleiche ATM Interface benutzen⁶⁰.
- VPI: 8 Bits für den *virtual path identifier*. Dieser wird im Zusammenhang mit dem Virtual Channel Identifier (VCI) benutzt, um die nächste Zellen-Destination im nächsten Switch festzulegen.
- VCI: 16 Bits für den *virtual channel identifier*. Dieser wird im Zusammenhang mit dem VPI benutzt, um die nächste Zellen-Destination im nächsten Switch festzulegen.
- PT: 3 Bits für den *payload type*. Das erste Bit wird benutzt um anzuzeigen, ob der Zelleninhalt für Steuerdaten oder Nutzdaten eingesetzt ist. Falls es sich um eine Nutzdaten-Zelle handelt, zeigt

⁶⁰ kommt nur im Betriebsmodus UNI, User-Network Interface vor.

das zweite Bit den Zustand einer allfälligen Verstopfung und das dritte Bit ob die Zelle die letzte einer ganzen Serie ist.

- CLP: 1 Bit für das *congestion loss priority*. Dieses zeigt an, ob die Zelle wegen Leitungsverstopfung weggeworfen werden soll.
- HEC: 8 Bits für den *header error control*. Dies sind Checksummen über den Header.

10.1.3 Generic Framing Procedure (GFP)

Je mehr Protokolle integriert werden müssen auf einer Plattform (z.B. Ethernet, OC-x, SDH, ATM, auf SDH oder DWDM), desto wichtiger ist es, dass eine nach ITU genormte Rahmenstruktur vorhanden ist. Der Generic Framing Procedure-Rahmen ist nach ITU-T SG15 G.7041 genormt.

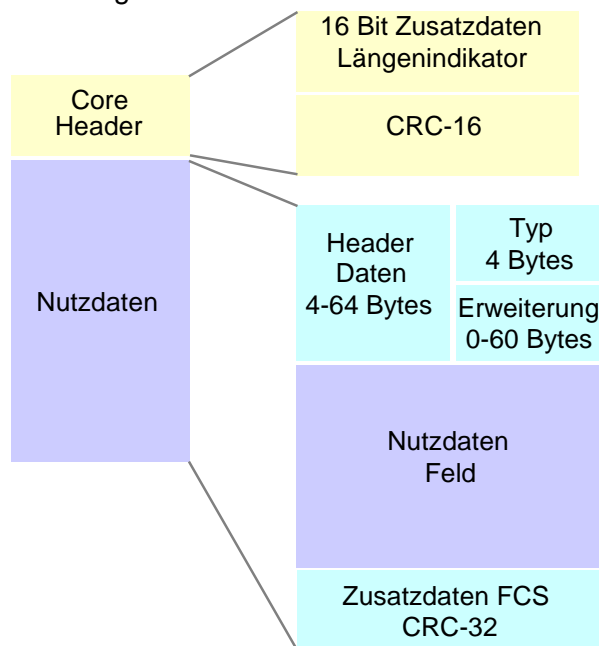


Abbildung 10.3: Der GFP-Rahmen

10.2 Sicherung der Übertragung

Für die Sicherung der Übertragung in einer Punkt-Punkt-Verbindung spielen unter anderem Begriffe wie Sequenznummern, Timeout und Flusskontrolle von Frames eine wichtige Rolle. Zudem existieren verschiedene Verfahren zur Sicherung (Resynchronisation) der Übertragung.

10.2.1 Resynchronisation

Für die Sicherung oder auch Resynchronisation der Übertragung werden die folgenden automatischen Verfahren (ARQ, Automatic Request) eingesetzt:

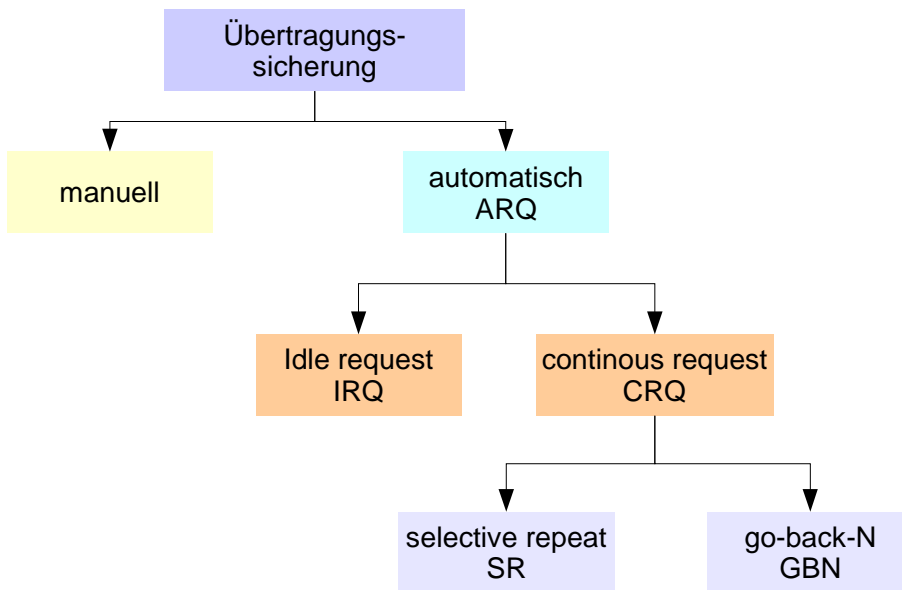


Abbildung 10.4: Übersicht über die Übertragungssicherungs-Verfahren

10.2.1.1 Idle Request (IRQ)

IRQ, Idle request, funktioniert wie folgt⁶¹:

Ein Empfänger wartet, bis er von einem Sender einen Teil einer Nachricht (beispielsweise ein Frame oder ein Zeichen) erhält. Er kontrolliert den Inhalt des Frames (beispielsweise mithilfe einer Prüfsumme). Ist der Inhalt richtig, quittiert er den Erhalt des Frames.

Das folgende Beispiel verdeutlicht den Vorgang:

Sender	Empfänger	
l	l	verstanden
c	c	verstanden
h	g	verstanden
nein h	h	verstanden

Es ist zu beachten, dass hier nur das Zeichen wiederholt wird, das die Prüfung des Empfängers als nicht richtig erkannt und somit nicht bestätigt hat. Das Wort „verstanden“ wäre hier ein Bestätigungswort für den Sender.

Diese Art der Sicherung nennt man idle request (IRQ), weil der Sender so lange sendet, bis er merkt, dass der Empfänger ein Zeichen nicht oder falsch erhalten hat, welches er dann nochmals sendet.

10.2.1.2 Continuous Request (CRQ)

Das Gegenstück zum IRQ sind die beiden Continuous Request-Verfahren, Selective Repeat (SR) und Go Back N (GBN).

Die Continuous Request-Verfahren arbeiten grundsätzlich wie folgt:

⁶¹ to idle: im Leerlauf warten, request: ersuchen, Aufforderung

Praxis-Hinweis:

Dieses Verfahren ist zwar langsam und daher vor allem für kurze Übertragungstrecken und langsame Übertragungsraten geeignet (für Modems geeignet, nicht für LANs). Dafür ist die benötigte Pufferspeicherkapazität sehr klein, weil der Empfänger immer nur das aktuelle und das letzte Zeichen im Speicher behalten muss (um Verdoppelungen zu vermeiden).

Praxis-Hinweis:

Das selective repeat hat den Vorteil, dass nur wenige Daten, nämlich die fehlerhaften, noch einmal gesendet werden müssen, wohingegen beim Go-back-N eine ganze Sequenz neu gesendet wird. Dies ist bei Systemen mit wenig Sendeenergie (Satelliten) von Bedeutung.

Praxis-Hinweis:

Dieser Mechanismus unterliegt einem Timeout, da sonst der Sender bei einem Problem des Empfängers nie mehr frei würde und demzufolge keine weiteren Empfänger bedienen könnte.

Der Sender schickt dauernd Frames in den Empfangspuffer des Empfängers, ohne auf irgendeine Bestätigung zu warten. Der Empfänger untersucht die Frames auf ihre Richtigkeit und schickt eine Bestätigung in einem Frame an den Sender der Nachricht (Acknowledge, ACK). Die Bestätigung enthält die Framenummer der Nachricht. Anhand dieser Bestätigung kann der Sender eindeutig identifizieren, ob alle seine Frames richtig angekommen sind.

Falls der Sender erkennt, dass ein Frame verlorengegangen oder falsch angekommen ist, hat er grundsätzlich zwei Möglichkeiten, um den Fehler zu korrigieren:

Beim selective repeat wird anhand der Framenummer jeweils nur gerade dasjenige Frame nochmals gesendet, das als fehlerhaft erkannt wurde.

Beim Go back N wird nach einer gewissen Zeit erkannt, dass z.B. das fünftletzte Frame falsch war. Der Sender wird dann aufgefordert, fünf Frames zurück zu gehen und ab dort noch einmal zu senden.

10.2.2 Steuerung und Flusskontrolle

Damit Sender und Empfänger wissen, um welches Frame es sich handelt, wird dem Frame sowohl beim IRQ-Protokoll als auch beim CRQ eine *Sequenznummer* mitgegeben.

10.2.2.1 Sequenznummern, Timeout, Flusskontrolle bei IRQ

Selbstverständlich können die Frames nicht einfach von eins bis n durchnummeriert werden, da sonst bei grossen Files die Nummer mit der Zeit grösser würde als das Frame.

Beim IRQ benötigt man genau zwei Frame-Sequenznummern, weil beim übernächsten Frame das Timeout des ersten sicher erreicht ist und somit die erste Nummer wieder frei wird. Typischerweise verwendet man hier 1 und 0 (siehe Abbildung 10.6, dort ist das erste Frame mit N und das zweite Frame mit N+1 bezeichnet).

Die Flusskontrolle kann hier mit Hardware realisiert werden, wenn eine RS232C-Schnittstelle mit RTS/CTS (Ready to send/Clear to send) verwendet wird oder ebenfalls mit einem Protokoll, dem X-ON- oder X-OFF-Protokoll (ASCII-Tabelle DC1, DC3 Kontrollzeichen). Falls ein Computer merkt, dass sein Puffer keine weiteren Daten mehr speichern kann, sendet er ein X-OFF an den Sender. Dieser stoppt die Übertragung und fährt wieder fort, sobald er ein X-ON erhält (empfangsbereit, Speicher verfügbar).

10.2.2.2 Sequenznummern und Flusskontrolle bei selective repeat und Go back N

Beim continuous request geht man davon aus, dass ein Fluss von Frames kontinuierlich übermittelt wird ohne dauernd auf die ACK zu warten. Die Fehlererkennung findet daher ebenfalls kontinuierlich statt. Damit zur Fehlererkennung genügend Rechenzeit zur Verfügung steht, werden bei den Teilnehmern der Kommunikation FIFO-Speicher (First In First Out-Speicher) benötigt. Damit die Puffer-Speicher und die Framenummern nicht beliebig gross werden, wird jeweils nur ein Ausschnitt aus dem Framestrom betrachtet. Dieser Ausschnitt wird „Fenster“ (engl. window) genannt. Weil sich das Fenster während der Übertragung dauernd über dem Framestrom verschiebt, benutzt man den Begriff „sliding window“ (Schiebefenster). Die Grösse des sliding window muss bei beiden Kommunikationspartnern gleich eingestellt sein, sonst ist die Übertragung nicht möglich.

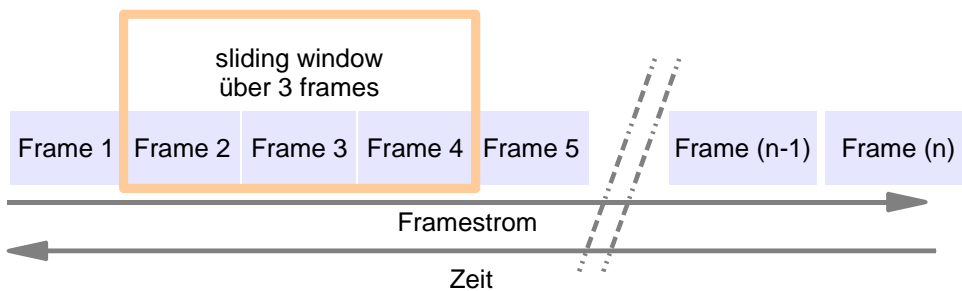


Abbildung 10.5: sliding windows

Dank der FIFO-Speicher ist es den Teilnehmern möglich, die Frames in einer beliebigen Reihenfolge und mit variablen Fenstergrössen übermitteln zu können und der Partner kann anhand der Framenummern die Nachricht wieder fehlerfrei zusammensetzen. Die Anzahl der übermittelten Frames kann beliebig gross sein. Eine Nummerierung von 1 bis n ist daher sicher unsinnig, weil mit der Zeit die Nummerngrösse den Frameinhalt übertreffen würde. Die Anzahl der Sequenznummern sind somit mindestens:

Protokoll	Anzahl Frames im gesendeten Fenster	Anzahl Frames im empfangenen Fenster	benötigte Sequenznummern
IRQ	1	1	2
selective repeat	n	n	2*n
Go back N	n	1	n+1

Tabelle 10.1: Anzahl benötigter Sequenznummern

Praxis-Hinweis:

Der grosse Nachteil beim idle request liegt darin, dass nach jeder Übermittlung eines Frames auf die Bestätigung (ACK/NAK) gewartet (engl. idling) werden muss.

Praxis-Hinweis:

In der Praxis sind die windows während einer Übertragung variabel. Die Grösse der Windows wird somit dauernd optimiert und der jeweiligen Leitungsqualität angepasst. Der Einfachheit halber benutzen in diesem Buch alle Beispiele fixe window-Grössen.

Praxis-Hinweis:

Bei der idle request-Methode kommt man mit zwei Nummern aus, weil immer gewartet wird, bis das Frame bestätigt ist und somit die verwendete Nummer wieder frei wird.

Damit selective repeat eindeutig in der Lage ist, alle n -gesendeten Frames im Schiebefenster zu bestätigen, benötigt es $2 \cdot n$ Sequenz- oder Framenummern, weil zuerst alle gesendeten Frames innerhalb des Schiebefensters vom Empfänger bestätigt sein müssen, bevor die Nummern vom Sender wieder belegt werden können. Das Go back N verlangt nur $n+1$ Nummern, weil sich beim Empfänger immer nur ein Frame im Puffer befindet. Merkt der Sender, dass der Empfänger ein Frame nicht richtig erhalten hat, so sendet er sowieso noch einmal alle Frames.

10.2.3 Genauere Betrachtung des IRQ

Jeder vom Sender abgeschickte Datenblock (Frame) wird vom Empfänger durch einen Steuerblock ACK (engl. acknowledged) bestätigt. Der Empfang von durch die Übermittlung verfälschten Frames kann dem Sender auch durch eine negative Bestätigung (NAK, engl. not acknowledged) angezeigt werden.

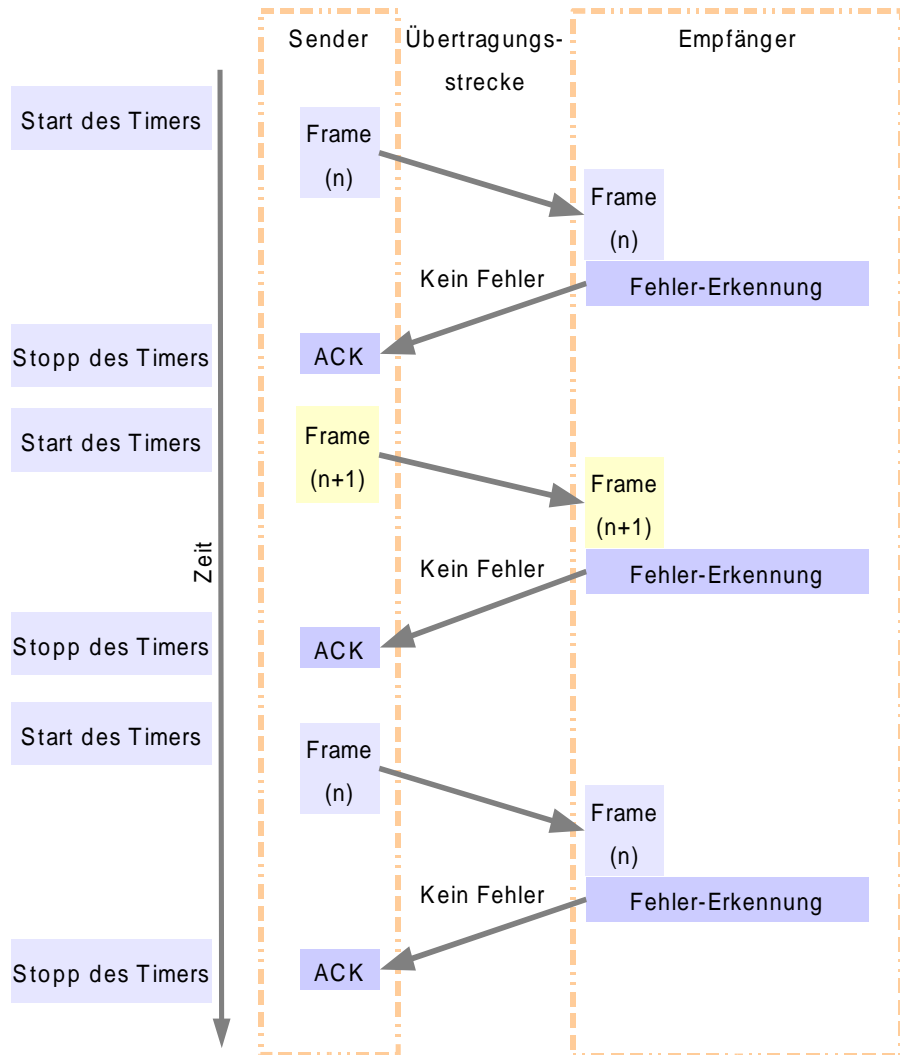
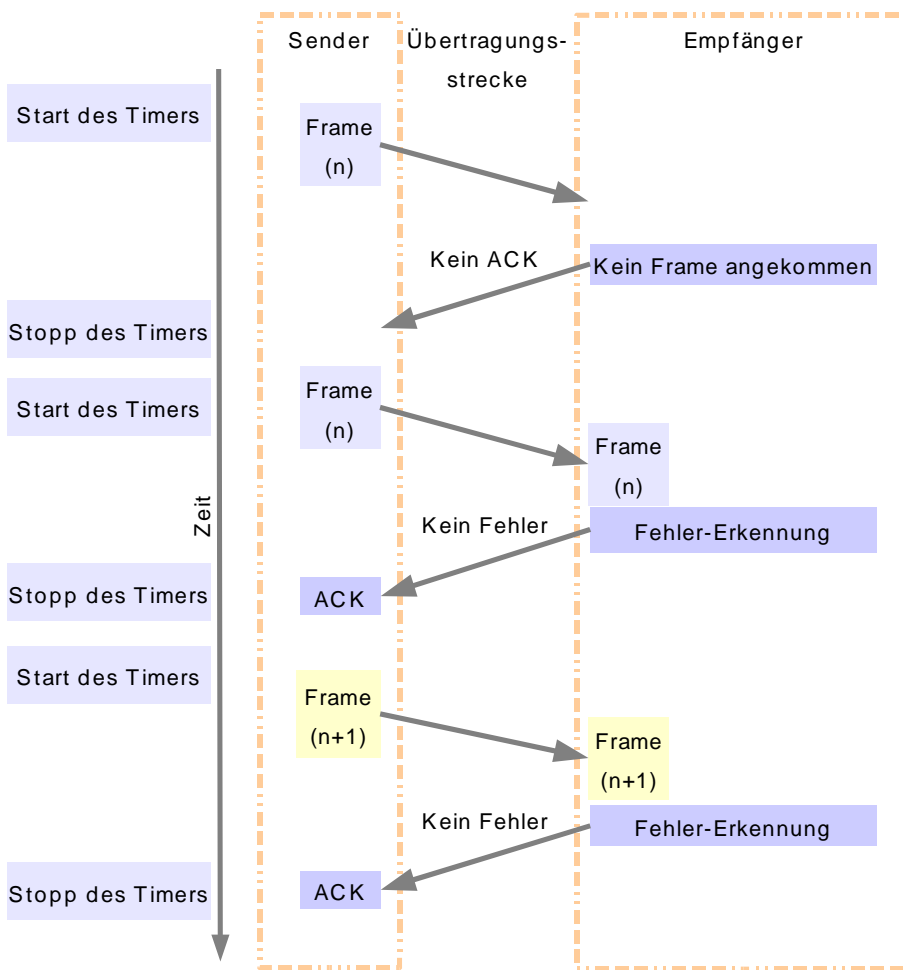


Abbildung 10.6: IRQ ohne Fehler

Abbildung 10.6 zeigt eine fehlerfreie IRQ-Übertragung.

Der Sender kann das folgende Frame, $F(n+1)$, erst senden, wenn er das ACK für das erste Frame, $F(n)$, erhalten hat. Der Sender interpretiert somit eine fehlende Bestätigung als Fehler im Frame und sendet das Frame noch einmal.

Abbildung 10.7 zeigt den Vorgang, wenn ein Frame nicht angekommen ist.



Praxis-Hinweis:

Eine Alternative in der Praxis ist die Bestätigung mit NAK: Der Empfänger sendet bei fehlerhaften Frames eine negative Bestätigung (not acknowledged, NAK) zurück. Der Sender kann somit auch fehlerhafte Frames von verloren gegangenen Bestätigungen unterscheiden.

Abbildung 10.7: Ein Frame geht verloren, es wird kein ACK gesendet.

Wenn ein Frame nicht ankommt, sendet der Empfänger kein ACK. Wenn der Sender innerhalb seiner Timeoutzeit kein ACK bekommt, sendet er das Frame noch einmal.

Was geschieht aber, wenn ein ACK verloren geht? Der Sender bekommt innerhalb seiner Timeoutzeit kein ACK und sendet das Frame noch einmal. Der Empfänger hat das Frame also doppelt erhalten. Der Empfänger muss also eine Möglichkeit haben, die doppelten Frames zu erkennen und nur eine Kopie zu behalten.

Damit einwandfreie Frames vom Empfänger aufgrund eines fehlenden ACK nicht doppelt weitergeleitet werden, muss der Empfänger mit einer Möglichkeit der Doppelerkennung ausgestattet sein.

Abbildung 10.8 zeigt eine Übertragung, bei der das ACK beim Sender nicht ankommt, das fehlerfrei übertragene Frame somit doppelt ankommt. Dies muss erkannt werden und das doppelte Frame muss eliminiert werden.

In Abbildung 10.8 wurden schliesslich die Frames n und n+1 korrekt übertragen, was der Empfänger mit ACK bestätigte.

Praxis-Hinweis:

Bei allen Beispielen wird eine Grösse des sliding window von $n = 2$ angenommen. Die Beispiele sind vereinfacht dargestellt, d.h. die in der Praxis übliche Veränderung der Fenstergrösse (window size) während der Übermittlung wird nicht dargestellt. Es wird von einer unveränderlichen Fenstergrösse ausgegangen.

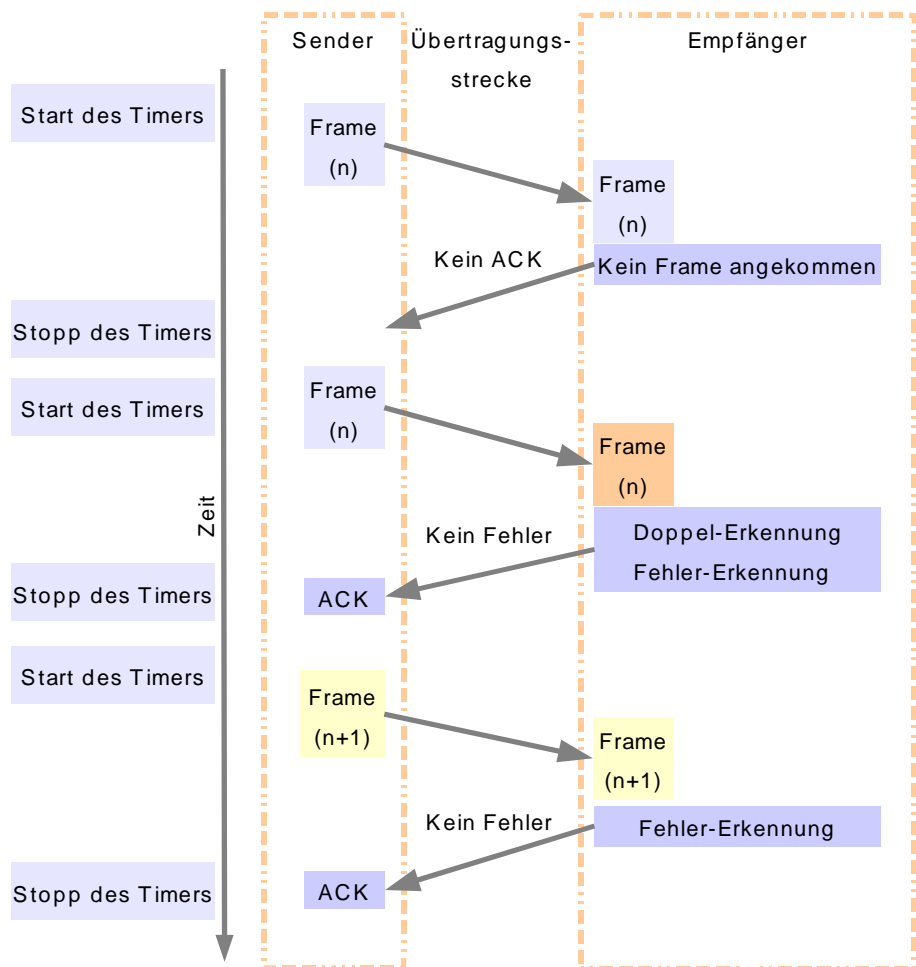


Abbildung 10.8: Ein ACK kommt beim Sender nicht an.

10.2.4 Genauere Betrachtung des CRQ

Betrachten wir wieder die verschiedenen Situationen, die auftreten können: Die Frames werden korrekt übermittelt und mit ACK oder NAK bestätigt; es gehen Frames verloren und Bestätigungen gehen verloren.

10.2.4.1 Selective Repeat

Abbildung 10.9, alle Frames werden korrekt übermittelt:

1. Zum Zeitpunkt (A) übermittelt der Sender die Frames 1 und 2 zum Empfänger. Der Empfänger speichert die Frames im FIFO-Puffer-Speicher und führt eine Fehlererkennung durch. Wenn er keine Fehler entdecken kann, teilt er dies dem Sender mit, indem er ACK1 und ACK2 sendet.
2. Zum Zeitpunkt (B) sind die ACK1 und ACK2 aufgrund der Zeitverzögerung auf dem Übertragungsmedium noch nicht beim Sender eingetroffen. Dieser schickt trotzdem die Frames 3 und 4 (window über 3 und 4). Die Frames 1 und 2 behält er noch im Speicher.

3. Zum Zeitpunkt (C) sind die ACK1 und ACK2 angekommen und der Sender kann die zugehörigen Frames aus dem Speicher löschen, damit Platz für neue Frames entsteht. Gleichzeitig kann er die neuen Frames 1' und 2' senden. Der Empfänger hat in der Zwischenzeit die Frames 1 und 2 aus dem FIFO-Speicher genommen und weiterverarbeitet, sodass auch sein Speicher wieder frei wird für neue Frames. Der FIFO-Puffer des Empfängers ist grösser als nötig, damit im Falle einer Verzögerung genügend Platz für den Empfang weiterer Frames bleibt.
4. Ab Zeitpunkt (D) wiederholt sich das Prozedere, bis der Sender alle Daten der Nachricht übermittelt hat.

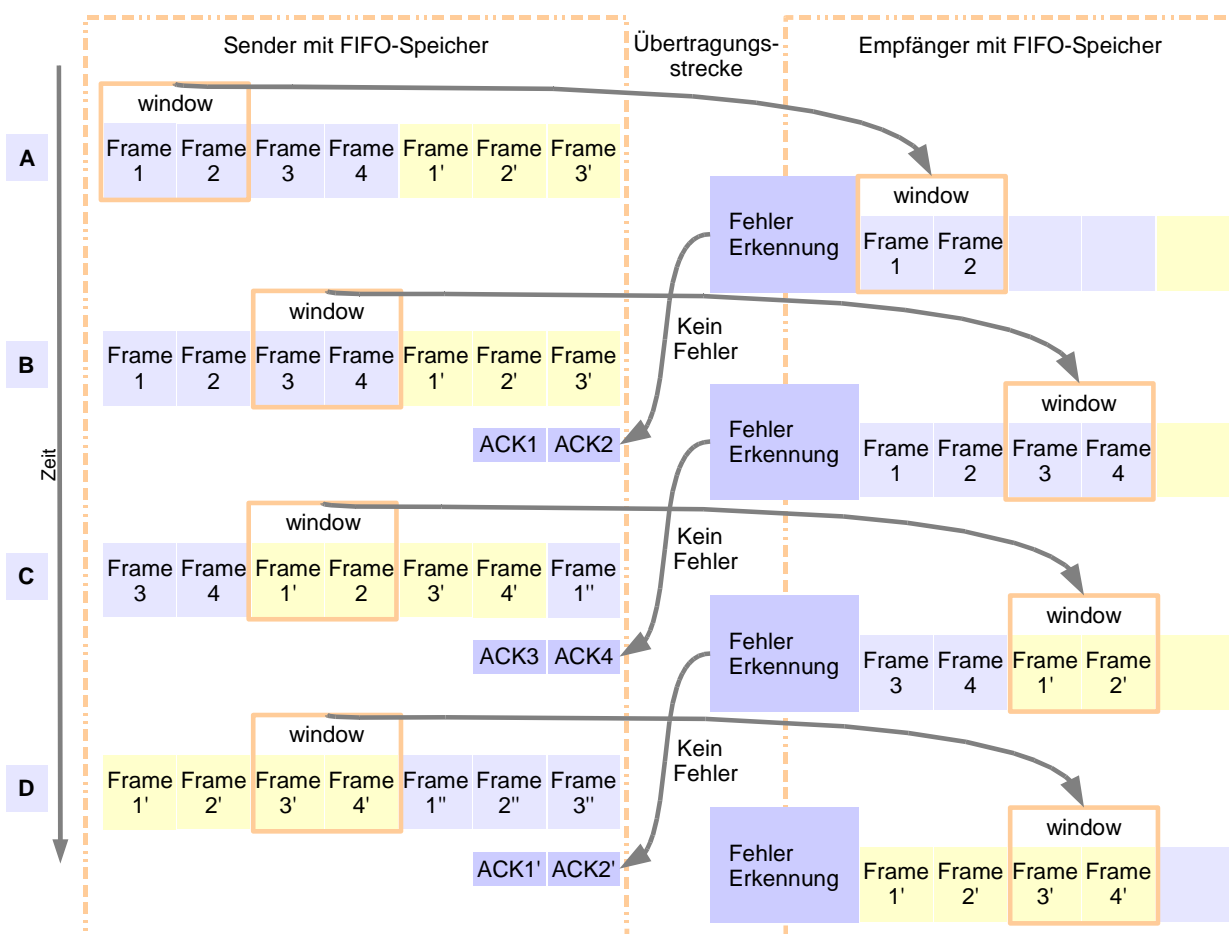


Abbildung 10.9: Korrekt übermittelte Frames mit SR

Abbildung 10.10, Frame 1 geht verloren:

1. Zum Zeitpunkt (A) sendet der Sender die Frames 1 und 2. Frame 1 geht dabei verloren.
2. Zum Zeitpunkt (B) übermittelt der Sender ohne Unterbruch die Frames 3 und 4. Der Sender erhält jetzt das ACK2 der Fehlerer-

- kennung des Empfängers. ACK1 wird keines verschickt, da das Frame 1 fehlerhaft ist. Das window des Senders bleibt über Frame 1 und 2 stehen, bis ACK1, ACK2, ACK3 und ACK4 übermittelt sind.
3. Zum Zeitpunkt (C) übermittelt der Sender das Frame 1 noch einmal und der Empfänger bestätigt das Frame 1 mit ACK1.
 4. Zum Zeitpunkt (D) muss der Sender auf das ACK1 warten und kann somit nicht senden – der Framestrom wird für kurze Zeit unterbrochen.
 5. Zur Zeit (E) kann der Sender die alten Frames 1 bis 4 aus dem Speicher entfernen, da sie alle bestätigt sind. Er kann die neuen Frames 1' und 2' senden. Der Empfänger ordnet die alten Frames 1 bis 4, gibt sie zur weiteren Verarbeitung frei und hat somit Platz für die neuen Frames. (Damit im Beispiel zwischen den „alten“ und den „neuen“ Frames 1 bis 4 unterschieden werden kann, sind die

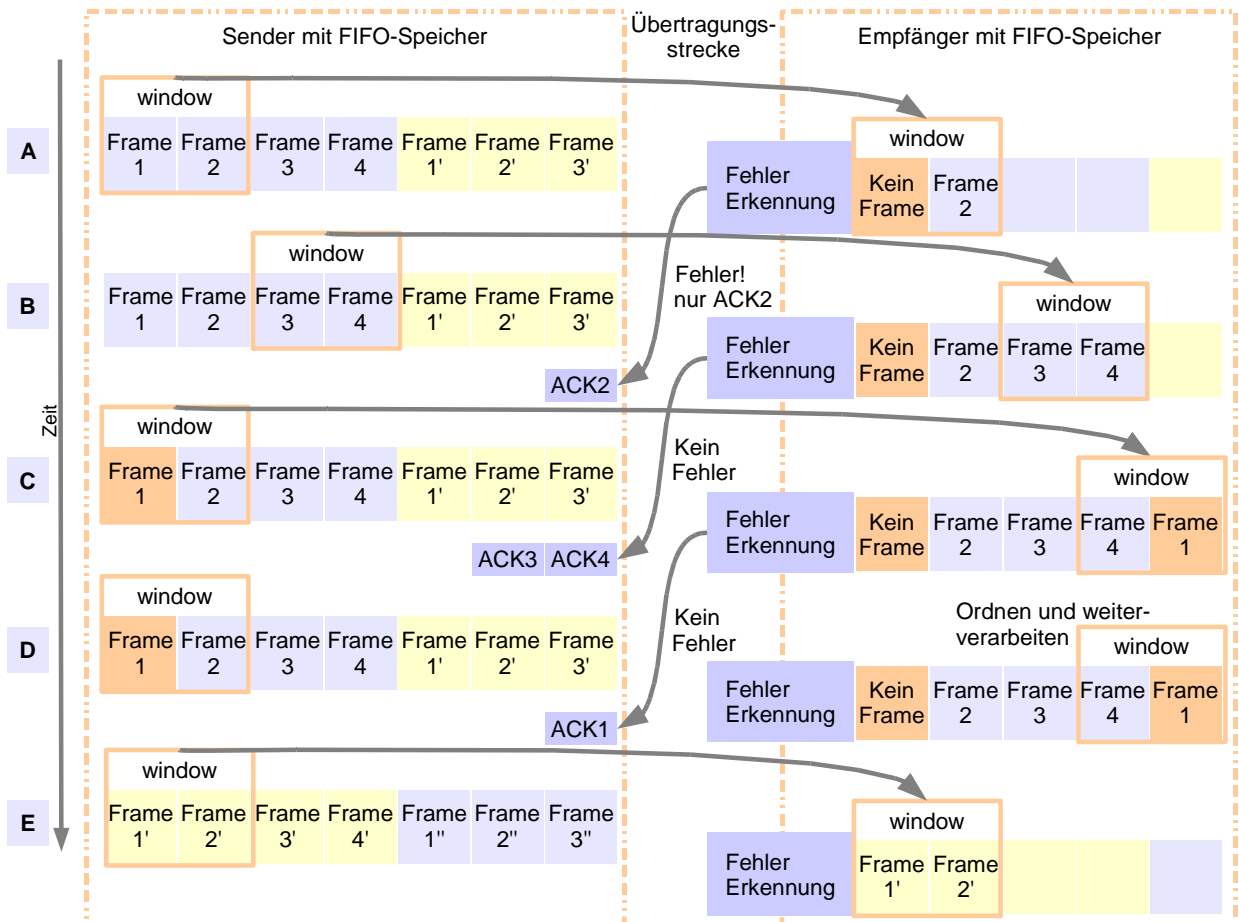


Abbildung 10.10: Frame 1 geht mit SR verloren

„neuen“ Frames mit 1' bis 4' gekennzeichnet. Es sind aber trotzdem nur vier Framenummern nötig bei einem window-size von 2.)

Abbildung 10.11, ein ACK geht verloren:

1. Zum Zeitpunkt (A) sendet der Sender die Frames 1 und 2. Die Frames werden fehlerfrei übertragen. Das ACK geht jedoch verloren.
2. Zum Zeitpunkt (B) übermittelt der Sender ohne Unterbruch die Frames 3 und 4. Der Sender erhält jetzt aber weder das ACK1 noch das ACK2 der Fehlererkennung des Empfängers. Der Sender kann nicht weiter senden und wartet.
3. Zum Zeitpunkt (C) erhält der Empfänger die ACK3 und ACK4 und ist überzeugt, dass er die Frames 1 und 2 wiederholt senden muss. Er schickt die Frames 1 und 2 noch einmal. Der Empfänger wartet auf die neuen Frames 1 und 2.
4. Zum Zeitpunkt (D) übermittelt der Sender die Frames 1 und 2 noch einmal. Da jetzt die Frames 1 und 2 beim Empfänger doppelt vorhanden sind, sendet dieser noch einmal ein ACK1 und ACK2 und löscht die doppelten Frames.

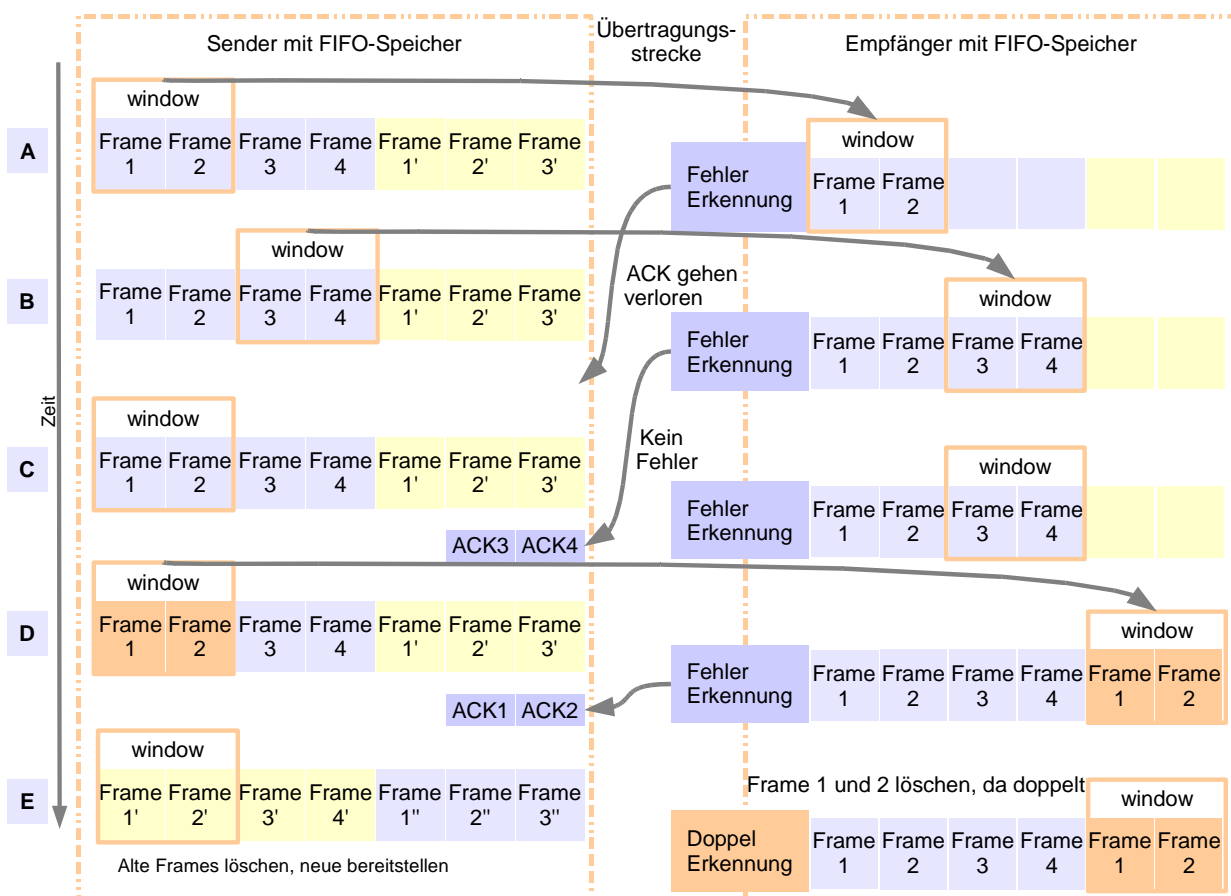


Abbildung 10.11: Ein ACK geht verloren

5. Zur Zeit (E) erhält der Sender endlich noch die Bestätigungen ACK1 und ACK2. Er kann nun die alten Frames 1, 2, 3 und 4 aus seinem Speicher löschen und die nächsten Frames 1', 2', 3' und 4' zum Senden bereitstellen. Der Empfänger gibt die alten Frames 1, 2, 3 und 4 zur weiteren Verarbeitung frei und bekommt somit Platz für die neuen Frames.

Aus diesen Beispielen ist ersichtlich, weshalb hier 2*n Sequenznummern benötigt werden!

Auch bei diesem Verfahren kann das NAK (not acknowledged) eingesetzt werden.

10.2.4.2 Go back N

Abbildung 10.12, alle Frames werden korrekt übermittelt:

1. Zum Zeitpunkt (A) übermittelt der Sender das Frame 1. Weil es richtig ankommt, sendet der Empfänger ein ACK1.

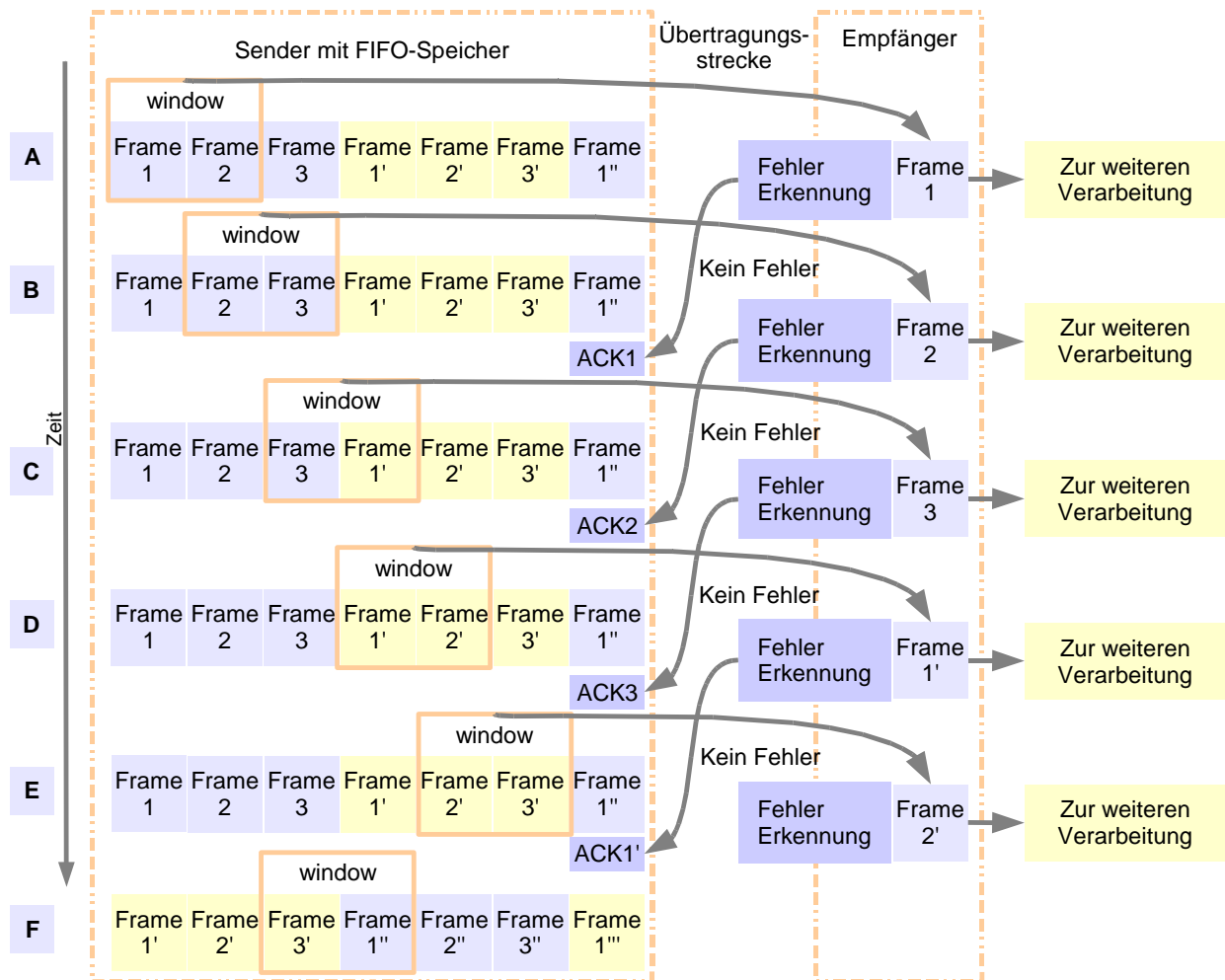


Abbildung 10.12: Korrekt übermittelte Frames mit GBN

2. Zum Zeitpunkt (B) übermittelt der Sender das Frame 2. Nach dem Senden von Frame 2 kommt die Bestätigung von Frame 1, ACK1, beim Sender an. Er weiss also, dass das Frame 1 in Ordnung ist. Er behält alle Frames in seinem Speicher. Der Empfänger hingegen gibt das Frame 1 aus seinem Speicher frei.
3. Zum Zeitpunkt (C) wird Frame 3 gesendet, Frame 2 aus dem Speicher des Empfängers entfernt und ACK2 beim Sender empfangen.
4. Dieses Prozedere wiederholt sich von nun an, bis der Sender alle Frames gesendet hat.

Abbildung 10.13, Frame 2 geht verloren:

1. Zum Zeitpunkt (A) übermittelt der Sender das Frame 1. Weil es richtig ankommt, schickt der Empfänger ein ACK1.
2. Zum Zeitpunkt (B) übermittelt der Sender das Frame 2. Das Frame geht aber unterwegs verloren. Der Empfänger behält das Frame 1 in seinem Speicher. Nach dem Senden von Frame 2 kommt die Bestätigung von Frame 1, ACK1, beim Sender an. Er weiss

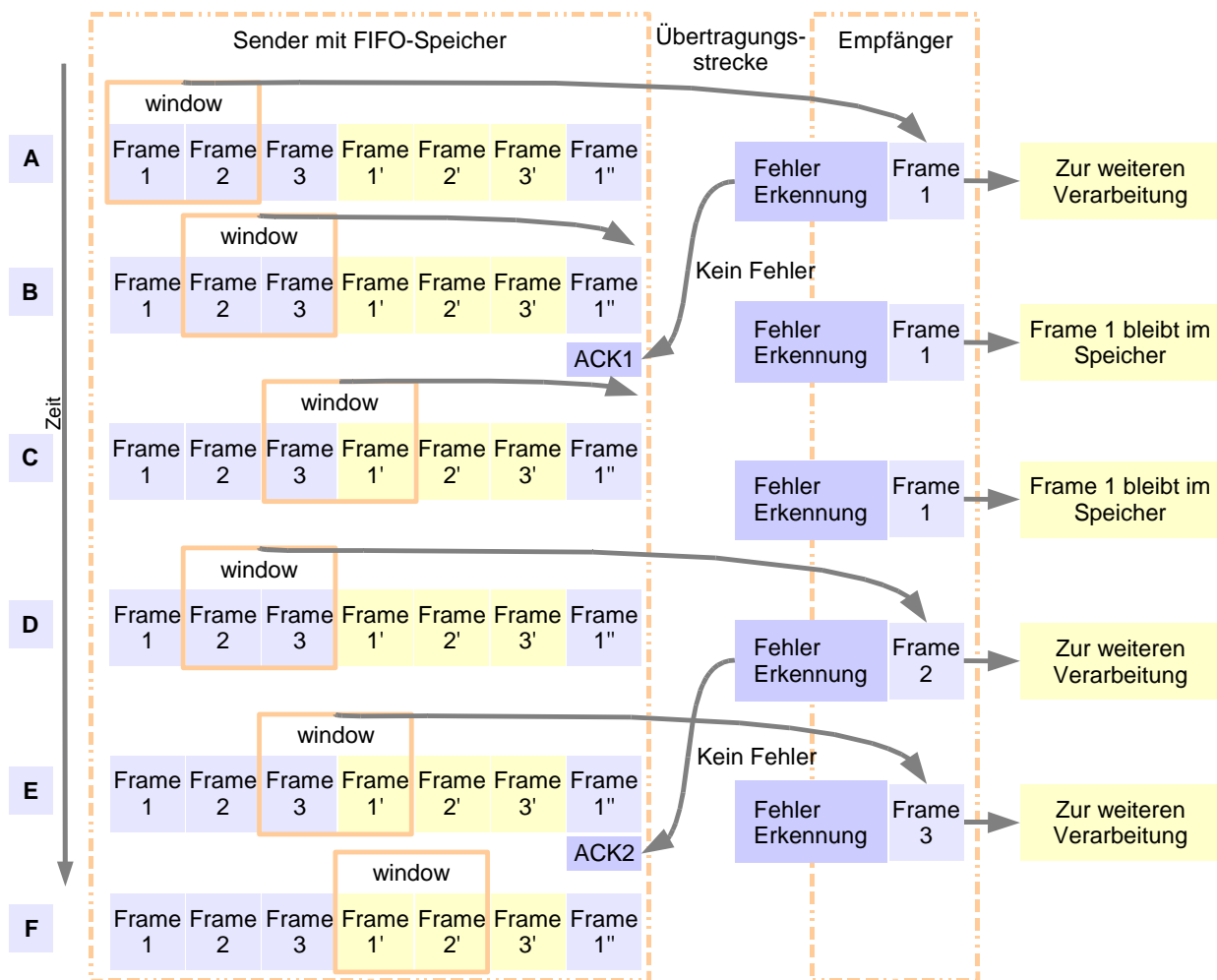


Abbildung 10.13: Frame 2 (und 3) werden nicht richtig übertragen. Daher Go Back zu Nummer 2 in Schritt D.

also, dass das Frame 1 in Ordnung ist. Er behält alle Frames in seinem Speicher.

3. Zum Zeitpunkt (C) wird Frame 3 gesendet. Das Frame wird vom Empfänger nicht angenommen, da er ja Frame 2 erwartet. Frame 1 bleibt im Speicher. Da der Sender kein ACK2 erhält, merkt er, dass beim Frame 2 etwas schief gelaufen sein muss. Er sendet nun noch einmal alle Frames ab Frame 2 neu. Er geht also zurück, bis zum falsch übermittelten Frame und sendet noch einmal alle Frames von dort an.
4. Zum Zeitpunkt (D) übermittelt der Sender somit das Frame 2, welches der Empfänger bestätigt.

Jetzt kann die Übertragung wieder normal weiter ablaufen.

Es dürfte nicht schwer fallen, zu sehen, dass das Verfahren auch bei fehlerhaften Bestätigungen (fehlende ACK) funktioniert.

Das Beispiel zeigt auch, dass hier n+1 Sequenznummern notwendig sind, um die Frames eindeutig zu unterscheiden.

10.2.5 Übertragungssicherheit – Verbindungsprotokolle

Damit wir gleich einen ersten Eindruck über die verschiedenen Protokolle erhalten, können wir folgende Zusammenfassung studieren: Verbindungsprotokolle (engl. link-protocols) werden benötigt, um die Verbindung zwischen den Kommunikationsteilnehmern aufzubauen und zu sichern. Wir unterscheiden zwischen den zeichenorientierten und bitorientierten Protokollen.

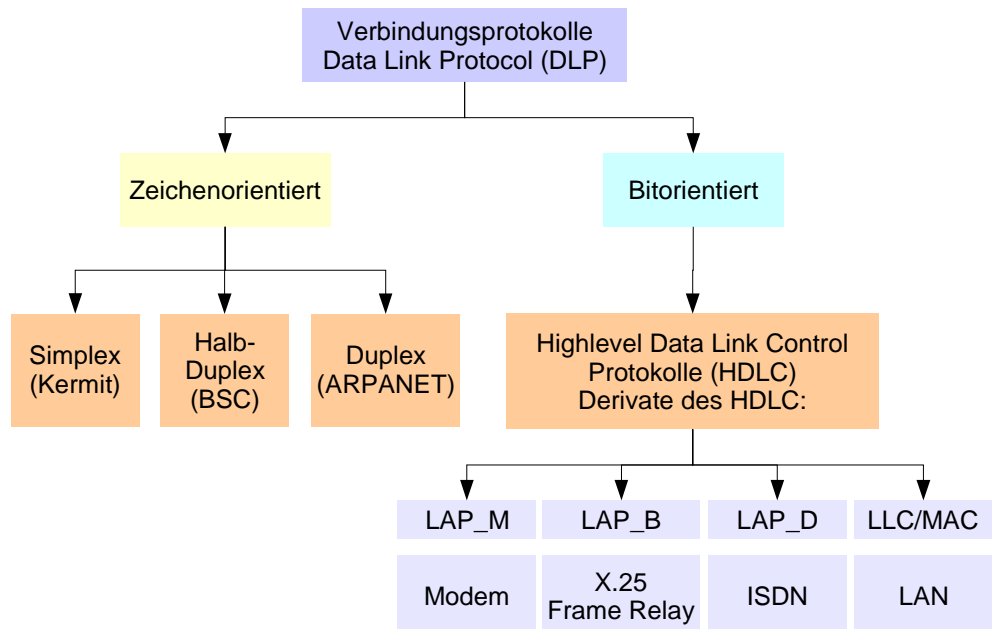


Abbildung 10.14: Übersicht über die Verbindungsprotokolle

Im Prinzip existiert bei den bitorientierten Protokollen das HDLC (High Level Data Link Control Protokoll) als wichtigstes dieser Protokolle. Die Frames des HDLC sind alle ähnlich (nach einer Norm) aufgebaut. Die verschiedenen Abarten (Derivate) des HDLC unterscheiden sich daher vor allem in den Link Access Procedures (LAP_x). Je nach Anwendung (Modem, X.25, ISDN, LAN) ist die Art und Weise, wie die Links zu Stande kommen, unterschiedlich.

10.2.5.1 Zeichenorientierte Protokolle

Die drei wichtigsten Arten der zeichenorientierten Protokolle werden hier erläutert. Oft werden diese Protokolle „totgesagt“, doch immer wieder erscheinen sie in der Technik.

10.2.5.1.1 Simplex-Protokoll (KERMIT)

Das Protokoll „Kermit“ baut eine Punkt-Punkt-Verbindung auf, wobei der eine Computer als Sender und der zweite als Empfänger konfiguriert sein muss (Simplex).

Die beiden Computer werden Data Terminal Equipment (DTE) genannt. Das Protokoll, das zwischen den beiden DTE aufgebaut wird, heisst Data Link Protocol (DLP). Die physische Verbindung kann mittels Null-Modem-Kabel (seriell), Modemverbindungen, Parallelkabel, Infrarot, Laser, Funk, USB (Universal Serial Bus) erstellt werden.

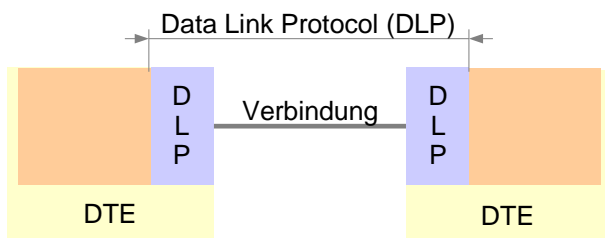


Abbildung 10.15: Verbindung zweier DTE

Das Kermit-Protokoll kommt bei Übertragungen zwischen Kleincomputern, Telefon-Endgeräten und allenfalls Taschenrechnern zum Einsatz. Ein ähnliches Protokoll ist das X-Modem-Protokoll, das in einigen Varianten auch Halbduplex-Verbindungen ermöglicht.

Es können auch Modems verwendet werden. Dabei muss das Modem des Empfängers in den Empfangsmodus gesetzt werden, sonst kann das Modem die Anrufe des Senders nicht annehmen.

Das Modem wird mit Data Circuit Termination Equipment oder manchmal mit Data Communication Equipment (DCE) bezeichnet. Die zwei Modems sind hier über ein öffentliches Telefonnetz (Public Switched Telephone Network, PSTN) verbunden.

Zur Datenübertragung muss das Kermit-Programm auf beiden Computern gestartet werden. Der Sender startet die Verbindung mit dem

Befehl „Connect“. Der Sender kann anschliessend Dateien übermitteln, indem er den Befehl „Send“, gefolgt von den Dateinamen eingibt. Die Dateien werden gesamthaft übertragen. Teile von Dateien können nicht übertragen werden. Sind die Dateien übermittelt, wird die Verbindung vom Sender mit „Exit“ abgebrochen.

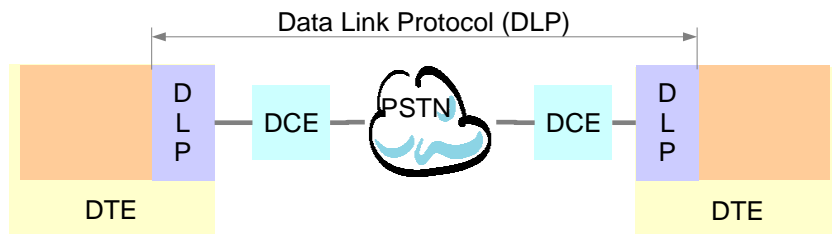


Abbildung 10.16: DLP am Beispiel einer Modemverbindung

10.2.5.1.2 Das Polling

Praxis-Hinweis:

Polling braucht immer eine zentrale (Steuer-)Station. Eingesetzt wird es nicht nur in Host-Umgebungen, sondern auch vermehrt in Kommunikations-Netzen und Client/Server-Umgebungen.

An dieser Stelle muss ein wichtiger Begriff eingeführt werden, der sowohl bei den zeichenorientierten Protokollen wie auch bei den bitorientierten zum Einsatz kommt. Damit überhaupt eine Punkt-Punkt-Kommunikation zwischen einer zentralen Station (Host oder Primary Station) und anderen am Netz angeschlossene Stationen (Slaves oder Secondary Stations) aufgebaut werden kann, muss die Primary Station die Secondary Station zuerst zur Kommunikation auffordern. Dieses Auffordern oder Anfragen heisst im englischen Sprachgebrauch „polling“.

In der Telematik ist dieser Begriff eng mit den Protokollen der Schicht 2 verbunden. In dieser Schicht werden die Punkt-Punkt-Verbindungen zwischen Host und Slave aufgebaut. Irgendwie müssen die Slaves ja „merken“, wann sie Nachrichten an die Gegenstelle senden dürfen und wann nicht. Polling kann auch in den Schichten 3, 4 und 7 vorkommen, da auch diese Schichten Verbindungen aufbauen können zwischen den Endgeräten (Schicht 3/4) respektive den Anwendungsprotokollen der Clients in Schicht 7.

10.2.5.1.3 Halb-Duplex-Protokoll (Binary Synchronous Control, BSC)

Dieses Protokoll wurde für die Kommunikation zwischen Terminals und ihren Hosts entwickelt.

Das von IBM entwickelte BISYNC-Protokoll (Binary Synchronous Communication, Binäre Synchronkommunikation) ist in der Industrie verbreitet. Es ist für Leitungen gedacht, die im Halbduplex-Modus arbeiten, sowohl für Gruppen- als auch Direktverbindungen. Es werden zwei verschiedene Netz-Topologien eingesetzt: Das Multipoint-Netz und das Multidrop-Bus-Netz.

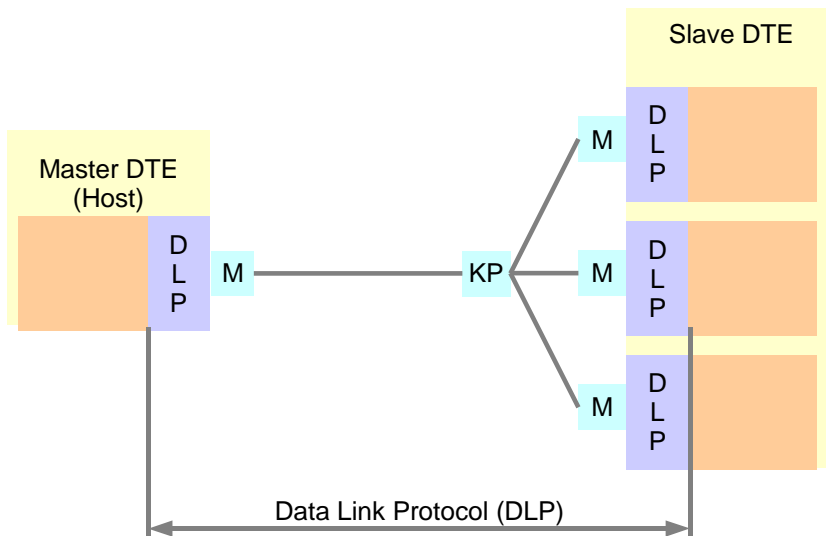


Abbildung 10.17: Das Multipoint-Netzwerk

Beim Multipoint-Netz überwacht ein Host (Master DTE) die Slave DTE. Der Host und die Slaves sind je über ein Vierdraht-Modem (M) mit dem Knotenpunkt (KP) verbunden.

Beim Multidrop-Bus-Netz werden die Slaves mit dem Host über je ein Line Driver/Receiver (LD/R) mit Twisted Pair-Kabel verbunden.

Diese Art Netze kommt beispielsweise in Warenhäusern vor, wobei ein Zentralcomputer die Registrier-Kassen-Computer (POS, Point Of Sale) überwacht und die Daten abrufe. Diese Netze wurden früher mit Idle request (IRQ)/BSC-Protokollen betrieben. Heute wird ein HDLC-basiertes NRM-Protokoll (Normal Response Mode) eingesetzt.

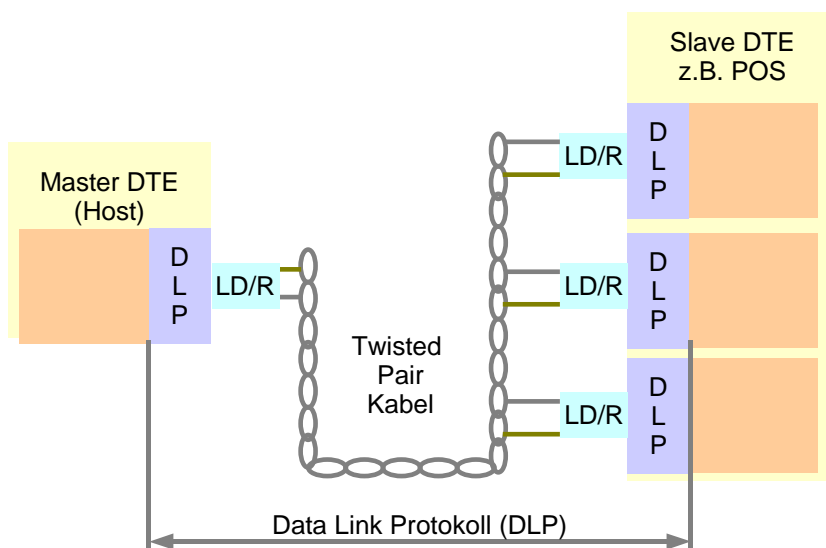


Abbildung 10.18: Das Multidrop-Netzwerk

10.2.5.1.4 Duplex-Protokoll (IMP/IMP-Verbindung im ARPANET)

In den Anfängen des ARPANET (Vorläufer des Internet) wurden Interface Message Processors (IMP) zu einem Netz verbunden. Jeder IMP sollte mit mindestens zwei weiteren IMPs verbunden sein, um eine möglichst hohe Sicherheit gegen Ausfälle zu erhalten. Ein Host konnte Nachrichten mit bis zu 8064 Bit (1008 Zeichen) an einen IMP senden und dieser beförderte die Nachricht in Paketen von 1008 Bit (126 Zeichen) mit speziellen Duplex-Protokollen (IMP/IMP-Protokollen).

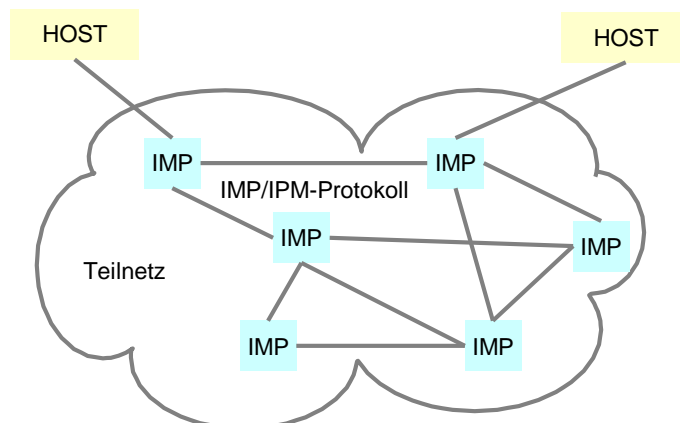


Abbildung 10.19: Das IMP/IMP-Protokoll beim ARPANET

Die IMP/IMP-Protokolle benutzen Continuous Request Control-Protokolle (CRQ) und werden heute noch sowohl für Verbindungen auf der Erde als auch für Verbindungen zu Satelliten angewendet. Der Aufbau und die Funktionsweise der zeichenorientierten Duplex-Protokolle ist relativ kompliziert. Eine genaue Beschreibung ist in der Literatur unter den Stichworten ARPANET, IMP to IMP oder DLP zu finden.

10.2.6 Bit orientierte Protokolle

HDLC ist ein internationaler Standard, der für point to point- (Punkt-zu-Punkt) und Multipoint-Verbindungen entwickelt wurde. IBM's Synchronous Data Link Control Protokoll (SDLC), der Vorläufer des HDLC, und das Advanced Data Communications Control Procedure (ADCCP) des American National Standards Institute (ANSI) werden ebenfalls noch eingesetzt. HDLC ist die Grundlage für verschiedene Data Link Control-Protokolle. Frameformat siehe Abbildung 10.1.

10.2.6.1 Einige Derivate des HDLC

Der HDLC-Frame ist genormt. Die verschiedenen Protokolle, die auf diesem Frameformat beruhen, unterscheiden sich aber in der Art des Verbindungszuganges (Link Access). Der Hauptunterschied beruht darauf, ob die Verbindung durch das dazwischen liegende Netz be-

einflusst wird oder ob das Netz von der Verbindung nicht wahrgenommen wird (transparente Verbindung).

10.2.6.1.1 Link Access Procedure Balanced (LAP-B)

Das LAP-B-Protokoll kontrolliert Frames in Punkt-zu-Punkt-Datennetzen. Ein gutes Beispiel dafür ist das X.25-Paketvermittlungsnetz. Das Protokoll kontrolliert den Verkehr zwischen einem Computer (DTE, Data Terminal Equipment) und einer Datenvermittlungsstelle (PSE, Packet Switching Exchange oder DSE, Data Switching Exchange).

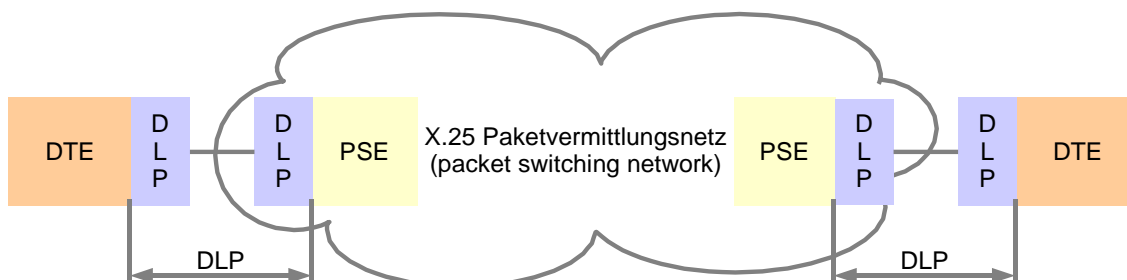


Abbildung 10.20: Der Link Access bei LAP-B

Weil LAP-B ein Unterprotokoll von HDLC ist, gelangen auch hier die gleichen Überwachungsframes wie bei den anderen Unterprotokollen zur Anwendung.

Zu beachten ist, dass alle Netze, die LAP-B einsetzen, dem DTE des Kunden nur einen Network Exchange-Zugang (hier das beschriebene PSE) ermöglichen. Dies hat zur Folge, dass das DTE des Kunden keinen Einfluss hat auf den genauen Transportweg seiner Daten oder auf die Funktionen und Dienste auf dem Netz. Man spricht hier von nicht transparenten Netz-Verbindungen.

10.2.6.1.2 Link Access Procedure für Modems (LAP-M)

Dies ist ein Protokoll, das z.B. in der Norm V.32 implementiert ist. Modems nach dieser Norm akzeptieren asynchrone Start/Stop-Daten eines fremden DTE und transportieren die Nutzdaten in Frames mit synchroner Übermittlung und einem Fehlerprotokoll auf HDLC-Basis. Abbildung 10.21 zeigt das Prinzipschema solcher Modems.

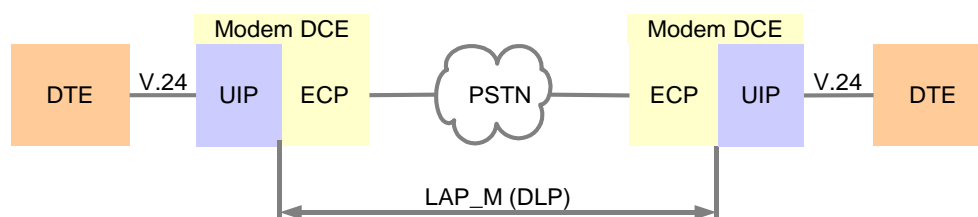


Abbildung 10.21: Der Link Access bei LAP-M

Praxis-Hinweis:

Die LAP-M-Netzwerkverbindung ist, im Gegensatz zu den Verbindungen mit LAP-B-Protokollen, transparent. Das bedeutet, dass das DTE des Senders mit dem DTE des Empfängers so verbunden ist, dass der Netzbetreiber (hier das PSTN) keinen Einfluss hat auf die Übertragungsart zwischen den Modems. Der Netzbetreiber kann höchstens die Bandbreite der Übertragung einschränken.

Das DTE ist über eine V.24-Schnittstelle mit dem Modem verbunden. Das Modem besteht aus zwei Teilen: einer Benutzerschnittstelle (UIP, User Interface Part) und einem Fehlerkorrekturteil (ECP, Error Correcting Part). Die Modems sind wiederum über das öffentliche Telefonnetz (PSTN) verbunden.

10.2.6.1.3 Link Access Procedure für D-Channel ISDN (LAP-D)

Dieses ebenfalls nicht transparente Protokoll ist auf dem D-Kanal des ISDN implementiert und überwacht den Verbindungsaufbau einer ISDN-Verbindung.

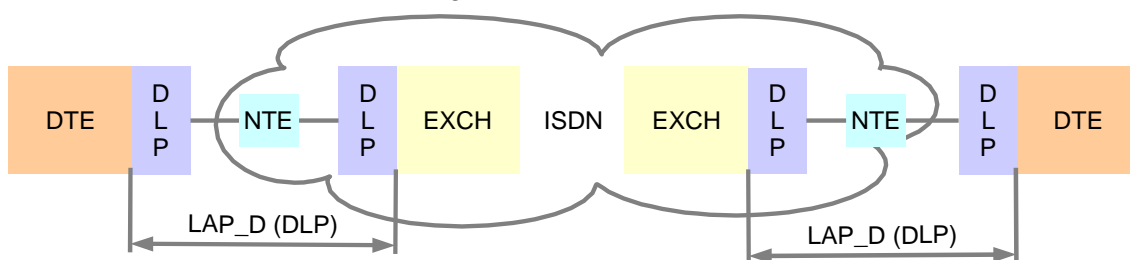


Abbildung 10.22: Der Link Access bei LAP-D

Das DTE wird hier über ein Endgerät (NTE, Network Termination Equipment) zur Vermittlungsstelle (EXCH, Switching Exchange) geschaltet. Die Daten werden über das ISDN-Netz übertragen und über EXCH und NTE zum Empfänger geleitet.

Damit die neuen Funktionen von ISDN (z.B. gleichzeitig zwei nutzbare Kanäle B1 und B2 beim Basisanschluss oder 30 Kanäle, B1 bis B30 beim Primär-Multiplexanschluss) ausgeschöpft werden können, sieht das Frame des LAP-D folgendermassen aus:

- Am Anfang steht die Marke (Flag).
- Block 2 und 3 sind Adressblöcke mit dem Service Access Point Identifier (SAPI), für die Bestimmung der Art des Endgerätes (Sprache, Daten, Sprache und Daten), dem Start/Stop-Bit (C/R), dem TEI, Terminal Endpoint Identifier (einer Endgerätezahl) und dem erweiterten Adressbit (EA).
- Blöcke 4 und 5 sind die aus dem HDLC bekannten Steuerungsblöcke.
- Information (Daten)
- Die beiden vorletzten Blöcke sind für die Fehlerkorrektur reserviert, hier CCITT 16.
- Abschluss: Flag

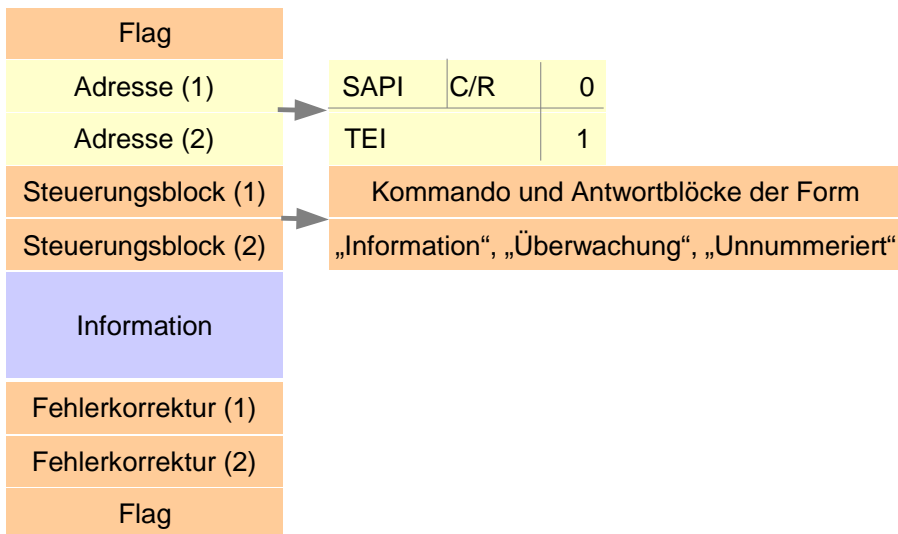


Abbildung 10.23: Der Aufbau des HDLC-Frames bei ISDN (alle Blöcke 8 Bit lang)

10.2.6.1.4 Medium Access Control (MAC)/Logical Link Control (LLC)

MAC (Medium Access Control) und LLC (Logical Link Control) sind HDLC-Derivate für LANs und bilden eine Einheit.

Das MAC-Teilprotokoll sitzt auf der physikalischen Schicht auf (Layer 1, Bitübertragung) und beinhaltet das Zugriffsverfahren (CSMA/CD und CSMA/CA).

LLC ist nach IEEE 802.2 genormt und kann als verbindungsloser Service (unzuverlässiger Datagrammdienst), als verbindungsorientierter Service mit Verbindungsaufbau oder als Service mit bestätigtem Verbindungsaufbau (bestätigter Datagrammdienst) vorkommen.

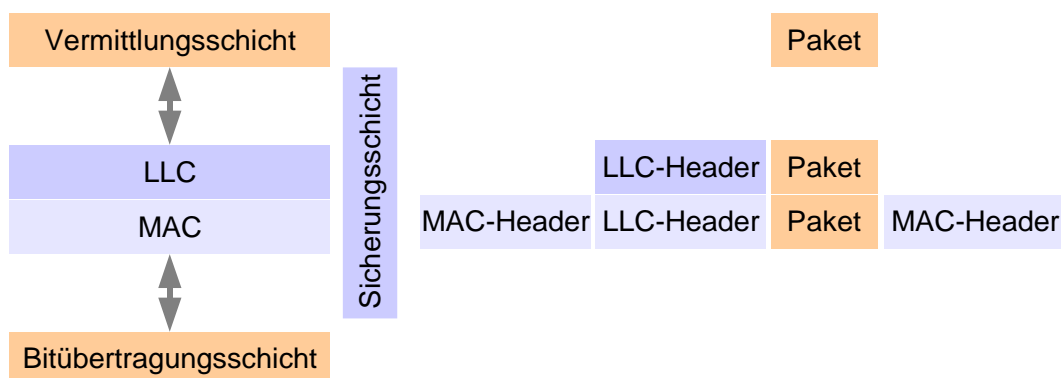


Abbildung 10.24: Der Zusammenhang zwischen LLC und MAC

Das LLC-Teil-Frame-Format ist wie folgt aufgebaut:

Ziel-Adresse	Sender-Adresse	Überwachung	Information
8-Bit	8-Bit	8-Bit	n x 8 Bit

Abbildung 10.25: Das LLC-Frame

Start mit der Zieladresse, gefolgt von der Senderadresse (je acht Bit). In den anschliessenden drei Bytes befindet sich die Überwachung mit Überwachungsfunktionen und Polling-Kontrolle (siehe HDLC). Daran anschliessend folgen die Datenblöcke.

Zu bemerken ist noch, dass die Ziel- und Senderadresse des LLC nur zwischen den LLC des Senders und des Empfängers gelten und im Netzwerk keine Funktion haben (nicht verwechseln mit der Ziel- und Quellenadresse der MAC-Teil-Schicht). Netzwerkadressen und Fehlererkennung ist Sache der MAC-Teilschicht. Dies ist der Grund, weshalb die MAC- und LLC-Teilschichten im ISO/OSI-Modell eine Einheit bilden. Der gesamte LLC-Teil-Frame wird im Nutzdatenfeld der MAC-Schicht eingefügt.

Das Frameformat der MAC-Teilschicht ist für jedes Zugriffssteuerverfahren leicht unterschiedlich.

Für IEEE 802.3 sieht das Format wie folgt aus:

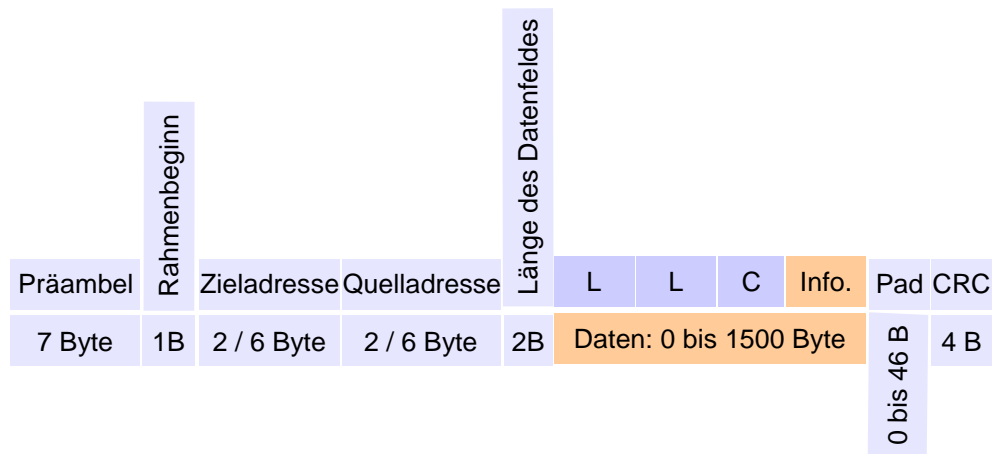


Abbildung 10.26: Das MAC-Frame mit integriertem LLC-Frame

- Die Präambel (Vorspann) besteht aus sieben Byte, die alle die Bitfolge 10101010 haben. Die Manchester-Codierung dieser Bitfolge erzeugt für die Dauer von 5,6 µs eine 10-MHz-Schwingung, woran sich der Taktgeber des Empfängers synchronisieren kann.
- Das Rahmenstart-Byte hat die Bitfolge 10101011.
- Dann folgen zwei Adressen: eine für das Ziel und eine für die Quelle. Die Norm lässt Adressen von zwei oder sechs Byte Länge zu. Das 10-MHz-Basisband benutzt sechs-Byte-Adressen. Die

Adresse, die nur aus Einsen (11111111...111) besteht, wird Broadcast-Adresse genannt. Die Rahmen, deren Ziel-Adresse aus lauter Einsen besteht, werden an alle Stationen gesandt. Die MAC-Adresse einer Netzwerkschnittstellenkarte (NIC) kann mithilfe eines Konfigurationsprogrammes des jeweiligen Karten-Herstellers in Erfahrung gebracht werden. Sie wird als Hexadezimal-Zahl angegeben und kann folgendermassen aussehen:

0000B20205ACF.

- Das Feld mit der Länge des Datenfeldes gibt an, wie gross das Datenfeld ist.
- Das Datenfeld kann eine Länge von 0 bis 1500 Byte aufweisen.
- Falls das Datenfeld die Länge 0 hat, kann dies im Netz zu Problemen führen, weshalb das Feld „Pad“ in diesen Fällen die Möglichkeit hat, 0 bis 46 Byte in den Rahmen einzufüllen.
- Die Prüfsumme wird im letzten Feld mitgegeben und basiert auf dem CRC-Verfahren.

10.2.7 Steuerung der Kommunikation nach IEEE 802

Die Steuerung der Kommunikation (Media Access Control) hängt stark vom verwendeten Netz ab:

Norm	Beschreibung
IEEE 802.3	Ethernet (nn Base k) mit CSMA/CD-Zugriffssterverfahren
IEEE 802.4	Token-Passing-Zugriffssterverfahren mit Bus-Topologie
IEEE 802.5	Token-Passing-Zugriffssterverfahren mit Ring-Topologie
IEEE 802.6	für MAN-Netze mit DQDB- Zugriffssterverfahren
FDDI	nach ANSI mit Token-Passing und Ring-Topologie

Tabelle 10.2: Die IEEE 802-Normen (Auszug)

Tabelle 10.2 beinhaltet nur eine Auswahl an Normen für Steuerungsverfahren. Je nachdem, ob ein LAN mit Kupferdrähten, LWL oder gar drahtlos betrieben wird, gelangen andere Verfahren zum Einsatz. Ebenso unterscheiden sich die Steuerungsverfahren der verschiedenen Netzarten wie LAN, WLL (Wireless Local Loop), GSM (Global System for Mobile Communication), UMTS (Universal Mobile Telecommunication System), MAN und anderen zum Teil grundsätzlich. Grundsätzlich unterscheidet man zwei verschiedene Möglichkeiten der Steuerung zwischen den Computern in einem Netz.

- Eine Möglichkeit zur Steuerung eines LANs beruht auf der Idee, dass eine Station nur senden darf, wenn sie dazu autorisiert ist. Die Station erhält als Zeichen für die Autorisierung eine Marke, ein so genanntes Token. Nur wenn eine Station ein Token hat, kann sie senden. Ohne Token muss sie zuhören, was andere sagen. Es

hat pro Netz ein Token. Das Token wird von einer Station zur anderen weitergereicht. Will eine Station Daten senden, so behält sie das Token, sendet die Daten und gibt das Token weiter. Hat die Station nichts zu senden, so gibt sie das Token nach Ablauf einer gewissen Zeit ebenso weiter. So ist gewährleistet, dass alle drankommen. Zu diesem Verfahren der Steuerung gehört das Token Passing von IBM.

- Die zweite, weit verbreitete Möglichkeit (engl. approach) beruht darauf, dass die Stationen im Wettbewerb miteinander stehen. Jede Station sendet, wann immer sie will (nicht synchronisiert). Stossen zwei Meldungen auf dem Kabel zusammen, müssen sie zeitversetzt noch einmal gesendet werden. Dieses Verfahren nennt man CSMA/CD (Carrier Sense Multiple Access mit Collision Detection), was etwa heisst:

Die Leitung wird dauernd abgehört. Viele können darauf gleichzeitig senden (zugreifen). Es findet eine Kollisions-Erkennung statt. Diese Art der Steuerung wird vorwiegend auf physikalischen Bus-, Stern- und Baum-Topologien gefahren (Ethernet).

Das Verfahren mit Token ist leistungsfähiger, aber leider auch teurer. Abbildung 10.27 zeigt die Leistungsfähigkeit der beiden Verfahren im Vergleich.

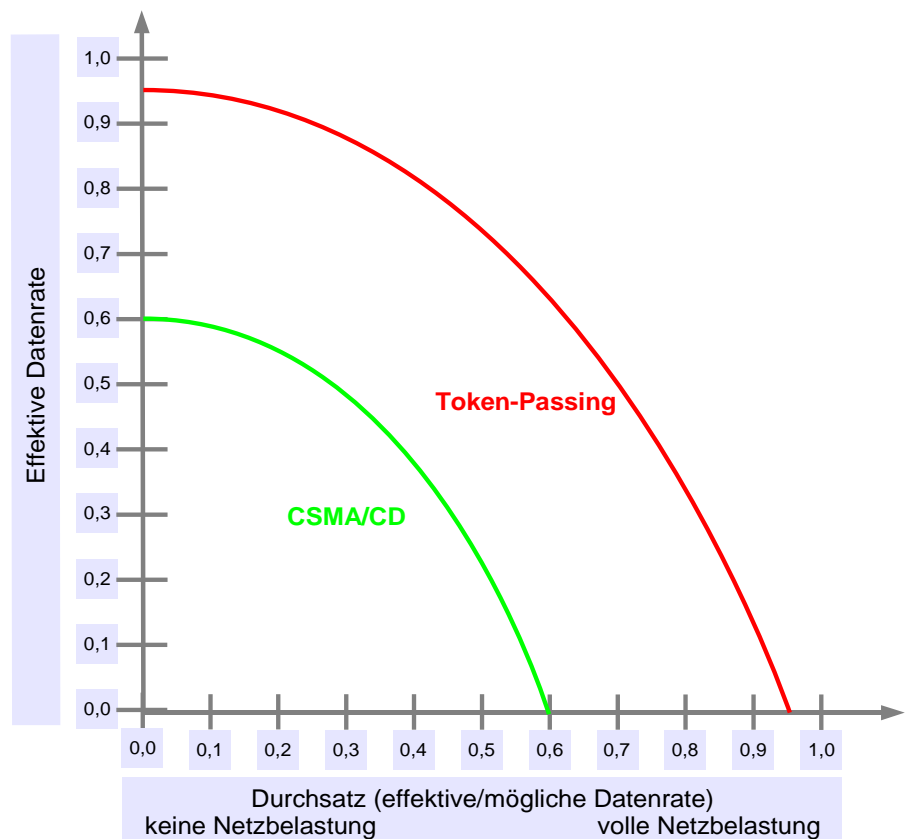


Abbildung 10.27: Die Leistungsfähigkeit der CSMA/CD- und Token-Protokolle

10.2.8 Beispiele weiterer Steuerungsverfahren

Neben dem häufig in LANs eingesetzten CSMA/CD-Verfahren werden gerade bei Glasfaserverbindungen, Funkstrecken und MAN (Metropolitan Area Networks) einige weitere Zugriffssteuerungsverfahren immer wichtiger:

Drahtlose LANs benutzen ein Verfahren, das Multiple Access with Collision Avoidance (CSMA/CA) heisst. Dieses Verfahren liegt dem IEEE 802.11 zu Grunde. Es beruht darauf, dass eine Senderstation einer Empfängerstation mit einem kurzen Rahmen ihre Absicht mitteilt. Die anderen, benachbarten Stationen hören diese Ankündigung und unterlassen das Senden während der Dauer der Übertragung des folgenden grossen Datenrahmens.

GSM-Netze (Global System for Mobile Communication) verwenden Frequenzen im 900-MHz-Band (890,2 bis 959,8 MHz). Das Frequenzband ist in 124 Uplink-Kanäle (Mobile an Basisstation) und 124 Downlink-Kanäle (Basisstation an Mobile) unterteilt (Frequency Division Multiplexing, FDM). Jeder der Kanäle ist mit dem Zeitmultiplex-Verfahren (Time Division Multiplexing, TDM) in acht getrennte Zeitschlitze unterteilt. Diese Aufteilung schliesst Kollisionen aus.

10.3 Aufbau, Betrieb und Abbau von Punkt-Punkt-Verbindungen

Die Hauptaufgabe der Schicht 2 ist der Aufbau, der Betrieb und der Abbau von Punkt-Punkt-Verbindungen im Netzwerk. Siehe dazu Abbildung 10.28.

10.3.1 Aufbau der Verbindung

Bevor Daten zwischen zwei Stationen ausgetauscht werden können, muss die Schicht 2 eine sichere Punkt-Punkt-Verbindung aufbauen.

Die Benutzersoftware des Senders schickt der Verbindungsschicht des Senders eine „Line.CONNECT.request“ Anfrage.

Die Verbindungsschicht des Senders pollt den Empfänger mit einem unnummerierten Frame an und legt den Betriebsmodus in einer Steueranweisung fest (z.B. SNRM).

Das Frame erreicht den Empfänger und die Steueranweisung wird ausgewertet.

Der Benutzersoftware des Empfängers wird durch ein „Line.CONNECT.indication“ ein Kommunikationswunsch des Senders angezeigt und wenn von dort keine abschlägige Antwort folgt, dann wird ein Frame mit der Steueranweisung UA (Unnumbered Acknowledge) zurückgesendet.

Sobald das Unnumbered Acknowledge Frame den Sender wieder erreicht hat, wird der Benutzersoftware die fertig aufgebaute Verbindung mit „Line.CONNECT.confirm“ bestätigt.

10.3.2 Betrieb der Verbindung

Die Benutzersoftware schickt den Befehl „Line.DATA.request“. Die Flusskontrolle und die Überwachung der Verbindung wird von nun an mit speziellen Überwachungs-Frames durch die Verbindungsschicht sichergestellt (wird in Abbildung 10.28 nicht gezeigt).

Wenn die Informations-Frames mit den Daten beim Empfänger ankommen, werden die Daten der Benutzersoftware mit dem Befehl „Line.DATA.indication“ angezeigt und zugestellt.

Gleichzeitig wird ein Überwachungsrahmen mit Steueranweisung (Receiver Ready) an den Sender geschickt. Dieser weiss dann, dass der Empfänger für den Empfang des nächsten Frames bereit ist.

Selbstverständlich kann nun auch der Empfänger Daten an den Sender übermitteln (die Verbindungen sind Halbduplex oder Duplex).

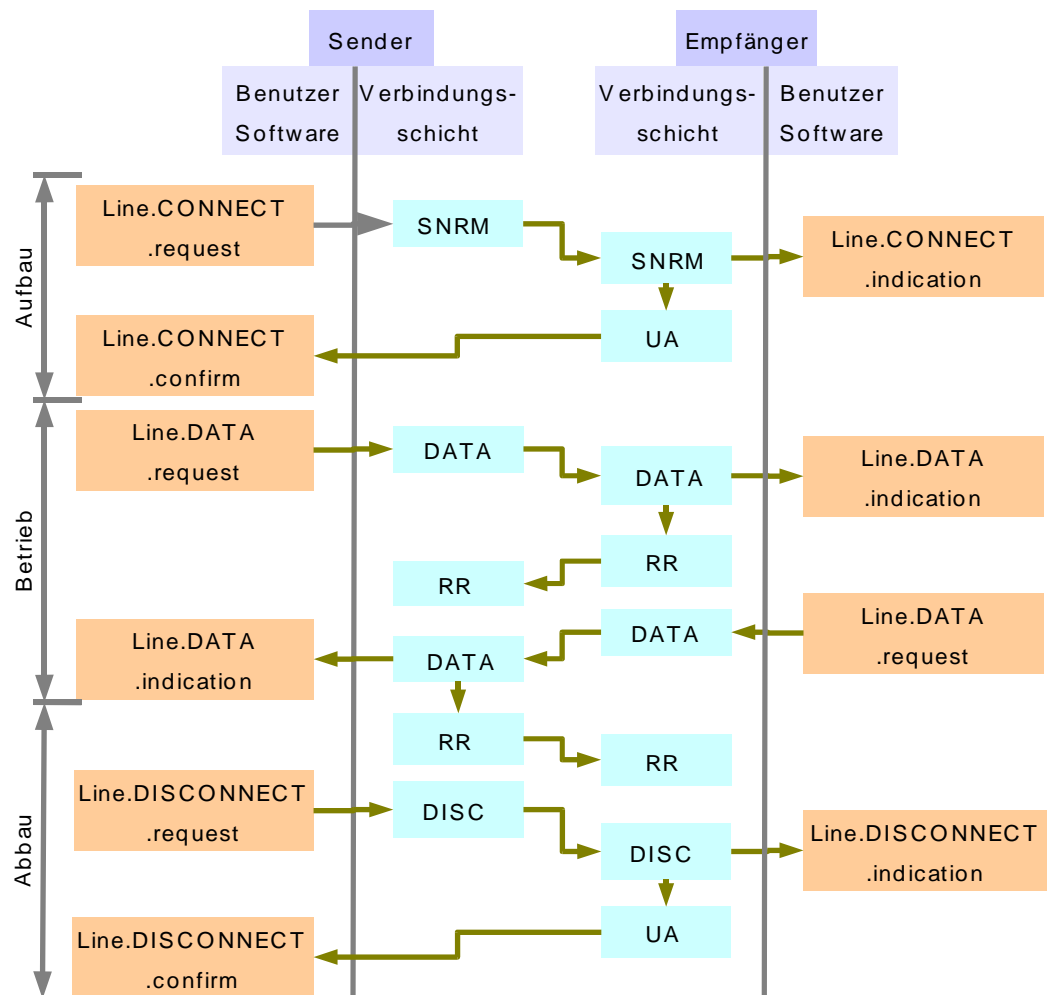


Abbildung 10.28: Aufbau, Betrieb, Abbau einer Layer 2-Verbindung

10.3.3 Abbau der Verbindung

Sobald die Daten übertragen sind, kann der Sender den Abbau der Verbindung veranlassen. Die Benutzersoftware des Senders schickt zu diesem Zweck ein „Line.DISCONNECT.request“ an die Verbindungsschicht. Diese zeigt den Abbau dem Empfänger mit der Steueranweisung DISC (Disconnect) in einem unnummerierten Frame an. Sobald die Benutzersoftware des Empfängers die Anzeige „Line.DISCONNECT.indication“ erhalten hat, wird dem Sender der korrekte Abbau mit UA (Unnumbered Acknowledge) bestätigt und die Leitung wird für andere Kommunikationen freigegeben. Die leeren Felder in Abbildung symbolisieren die Zeit, in der die Verbindungsschicht Frames auswertet.

10.4 Aufgaben

1. Wo liegt der Unterschied zwischen Packets und Frames?
2. Welches sind die Hauptfunktionen des 2. Layers im ISO/OSI-Modell?
3. Wie ist es möglich, dass der Empfänger ein falsches Bit bekommt?
4. Wie kann das Weiterleiten von zwei empfangenen Frames, die identisch sind, verhindert werden?
5. Warum wird heute meist auf Paritätsprüfung verzichtet, obwohl dieses Verfahren dank Hardware-Realisierung sehr schnell ist?
6. Warum ist Idle Request relativ langsam?
7. Wo liegt der Hauptunterschied zwischen Selective Repeat und Go back N?
8. Weshalb braucht man beim Selective Repeat $2 \cdot n$ Sequenznummern?

Lösungen unter www.sauerlaender.ch/downloads

11 Vermittlung von Netzverbindungen

Sind die Nachrichten für Kommunikations-Teilnehmer in entfernten Netzen bestimmt, so reichen Frames mit ihren Hardware-Adressen nicht aus, um den Teilnehmer sicher zu finden. Die Situation ist vergleichbar mit Ihrem Namen und Ihrer Adresse: Ihr Name entspricht der Hardware-Adresse, der in den Frames gespeichert ist. Sie können sich jedoch an verschiedenen Orten (Adressen) auf der Welt aufhalten. Somit würde Sie ein Brief nicht erreichen, da die Post (das Netz) Ihre gegenwertige Adresse nicht unbedingt kennt. Der Brief muss somit Ihre gegenwertige Adresse aufgedruckt haben und die Post muss die Adresse und den Weg zu dieser Adresse kennen, um eine sichere Zustellung zu garantieren.

Das vorliegende Kapitel beschreibt die Lösungsansätze der Telematik, um dieses grundlegende Problem zu lösen.

11.1 Die Aufgabe der Vermittlung

Das beschriebene Problem wird mithilfe der Vermittlung in Netzen gelöst.

Zu Beginn des Telefoniezeitalters erfolgte die Vermittlung von Gesprächen manuell. Man nahm den Hörer ab und wurde mit einer Telefonistin der nächst gelegenen Vermittlungsstelle verbunden. Die Vermittlerin im Amt schaltete die gewünschte Verbindung zur nächsten Zentrale durch. Dort sass wieder eine Telefonistin, welche die Verbindung weiter aufbaute. Dies wurde so lange fortgeführt, bis die Verbindung bei der Zentrale des Empfängers ankam. Das Telefon beim Empfänger läutete und die Vermittlerin vom Amt liess den Empfänger warten (online), bis sie die aufgebaute Leitung rückwärts zur Zentrale des Senders hin bestätigt hatte. Nach dieser Prozedur konnten die beiden Teilnehmer miteinander sprechen.

Diese Zentralen wurden nach und nach durch Relaissteuerungen ersetzt und in der Folge arbeiten heute alle Zentralen mit digitaler Vermittlung.

Generell gilt das folgende Schema in Abbildung 11.1, das den Stellenwert der Vermittlung darstellt.

Abkürzungen in der Abbildung:

IMP Interface Message Processor, Schnittstellenprozessor

PSTN Public Switched Telephone Network

Applikation
7 Anwendung
6 Darstellung
5 Sitzung
4 Transport
3 Vermittlung
2 Sicherung
1 Bitübertragung
Übertragungsmedien

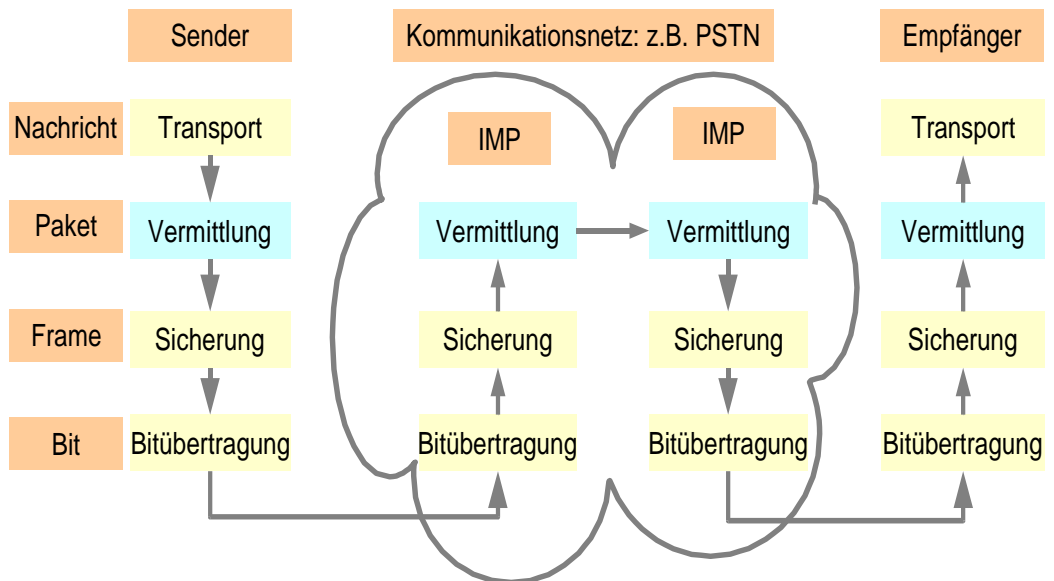


Abbildung 11.1: Das Prinzip der Vermittlung in einem Kommunikationsnetz

11.2 Verbindungsorientiert/Verbindungslos

Praxis-Hinweis:

Die ARPANET (Internet) Entwickler gingen davon aus, dass ein Subnet (Übertragungsnetz) immer etwas Unzuverlässiges sei. Aus diesem Grund verlangten sie, dass die Hostrechner im Netz (Rechner des Senders und des Empfängers) Fehlersuche, Fehlerkorrektur und Flusssteuerung durchzuführen haben.

Auf Grund historischer Zänkereien zwischen der ARPA-Internet-Gemeinde (Advanced Research Projects Agency) und anderen Netzbetreibern (zum Beispiel Telefonnetz-Betreibern) gibt es zwei Auffassungen über die genaue Aufgabe der Vermittlung.

Dies führt dazu, dass sich die Vermittlung auf SEND-PACKET und RECEIVE-PACKET beschränkt und völlig verbindungslos oder verbindungsunabhängig sein sollte. Jedes Paket muss aber eine vollständige Zieladresse haben, da es unabhängig von seinen Vorgängern oder Nachfolgern verschickt wird. Man nennt dies einen verbindungslosen Dienst.

Andere Netzbetreiber, vor allem diejenigen, die das Subnet selber verwalten (beispielsweise PSTN-Betreiber), sehen die Sache etwas anders. Hier herrscht die Ansicht vor, dass das Subnet sehr zuverlässig sei und dass die Vermittlung daher eine Verbindung aufzubauen habe, auf der dann Daten gesendet werden können. Die

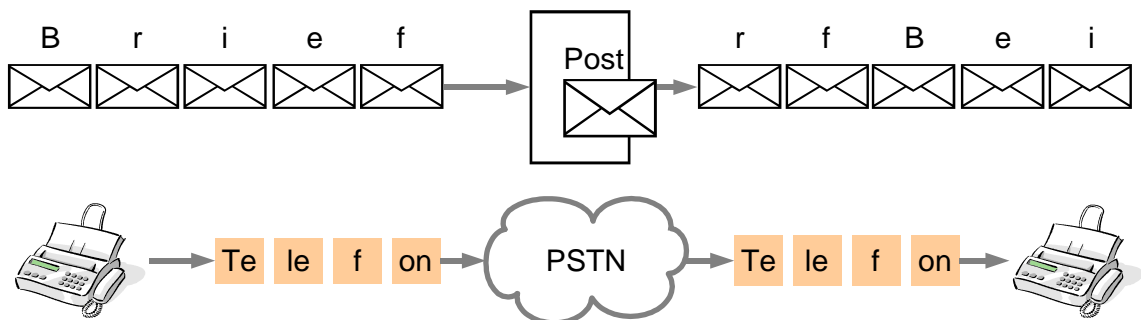


Abbildung 11.2: Verbindungslose und verbindungsorientierte Kommunikation

Verbindung sei mit einer speziellen Kennung zu versehen, welche während der gesamten Datenübertragung verwendet werden müsse. Am Schluss der Übertragung wird die Verbindung ordnungsgemäss abgebaut. Eine spezielle Flusskontrolle soll verhindern, dass der Sender seine Pakete schneller ablegt als der Empfänger. Man nennt dies *verbindungsorientierte* Verbindung.

Das öffentliche Telefonnetz ist verbindungsorientiert, die Briefpost hingegen verbindungslos. Briefe kommen nicht unbedingt in der Reihenfolge des Absenders an. Beim Telefonieren schätzen wir es, wenn die Worte des Partners am anderen Ende in der richtigen Reihenfolge ankommen!

Beim verbindungslosen Dienst hat jedes Paket eine volle Zieladresse und wird unabhängig von allfälligen weiteren Paketen auf individuellen Übermittlungswegen im Netz transportiert. Erst der Empfänger setzt die Pakete wieder in der richtigen Reihenfolge zu einer Nachricht zusammen.

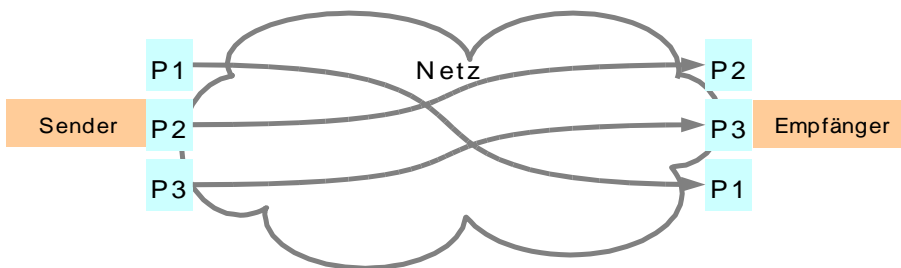


Abbildung 11.3: Verbindungslose Kommunikation

11.2.1 Verbindungsorientierte Verfahren

Eine genauere Betrachtung der verbindungsorientierten Protokolle zeigt, dass hier auf einem realen Netz mit Übertragungsleitungen und Schnittstellen-Computern eine virtuelle Verbindung aufgebaut wird,

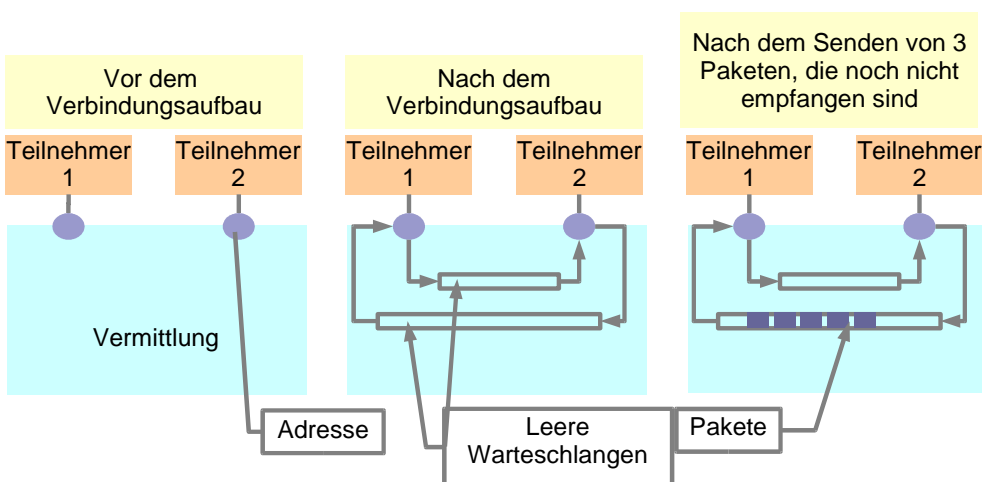


Abbildung 11.4: Verbindungsorientierte Kommunikation

die während der Dauer der Datenübertragung bestehen bleibt und anschliessend wieder abgebaut werden muss, damit andere Benutzer die Einrichtungen wieder benutzen können. Typische Beispiele aus der Praxis sind das öffentliche Telefonnetz, das ISDN-Netz, das X.25-Netz, Frame Relay und ATM-Netze.

11.2.1.1 Leitungsvermittlung im Telefon- oder ISDN-Netz

In öffentlichen Netzen (Telefon, ISDN) wird das Vermittlungsverfahren der Leitungsvermittlung eingesetzt (circuit switching).

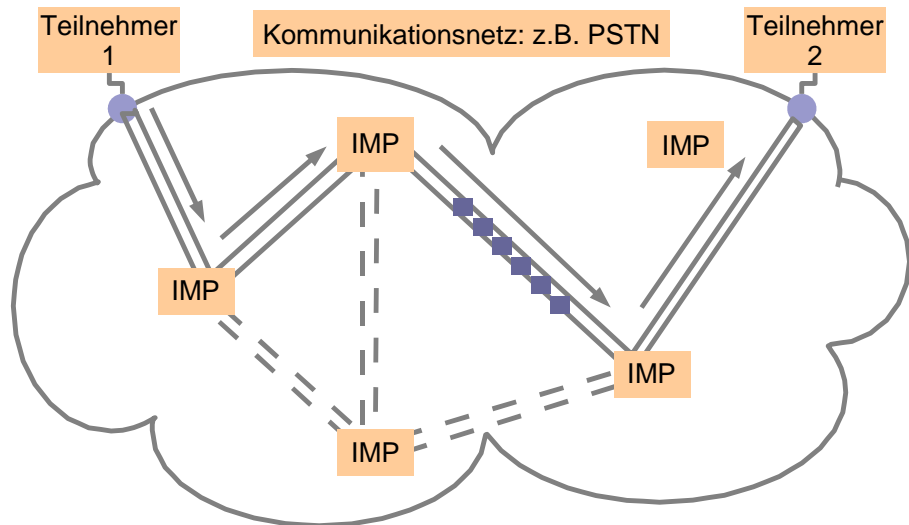


Abbildung 11.5: Leitungsvermittlung am Beispiel PSTN

Die Leitungsvermittlung besteht darin, dass vor der Datenübertragung eine virtuelle Verbindung auf physikalischen Leitungen und in den Knoten bereitgestellt (aufgebaut) wird. Dies geschieht dadurch, dass der Sender die Telefonnummer des Empfängers wählt und damit die Zentralen veranlasst, die Leitung durchzuschalten und Speicher in den Knoten bereitzustellen. Die beiden Teilnehmer (Benutzer) der Leitung senden ihre Daten über diese Leitung und wenn sie damit fertig sind, wird die Leitung wieder abgebaut und die Ressourcen (Speicher, Leitweg-Informationen ...) in den Knoten freigegeben.

11.2.1.2 Paketvermittlung mit Virtual Circuits (VC)

Die Paketvermittlung (virtual circuit packet switching) basiert darauf, dass ein Teilnehmer an seiner lokalen Paketvermittlungsstelle (PSE, Packet Switching Exchange) Pakete ins Netz speist. Die Pakete haben eine Adresse und werden auf verschiedenen Wegen über das Netz dem Empfänger zugestellt. (I = Teilinformation, As = Adresse des Senders, Ae = Adresse des Empfängers).

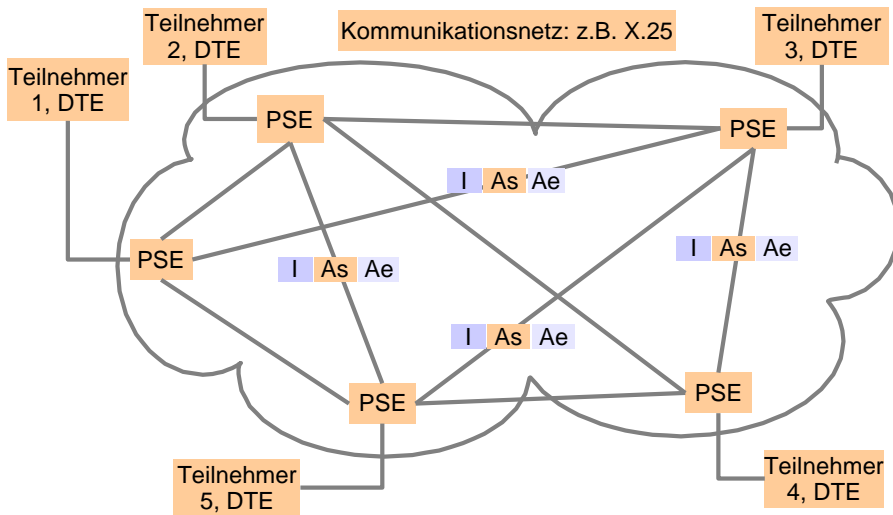


Abbildung 11.6: Paketvermittlung am Beispiel X.25

11.2.1.3 Die Paketgrösse

Die Paketgrösse spielt hier eine Rolle. Kleinere Pakete erlauben vermehrtes Sammeln der Meldungen an den Knoten und haben kleinere Absendeverzögerungen, weil sie schneller verpackt sind. Aber kleinere Pakete haben im Verhältnis zu den Nutzdaten einen grösseren Header (Nachrichtenkopf mit Steuerinformation). Abbildung 11.7 zeigt einen Vergleich zwischen verschiedenen Paketgrössen:

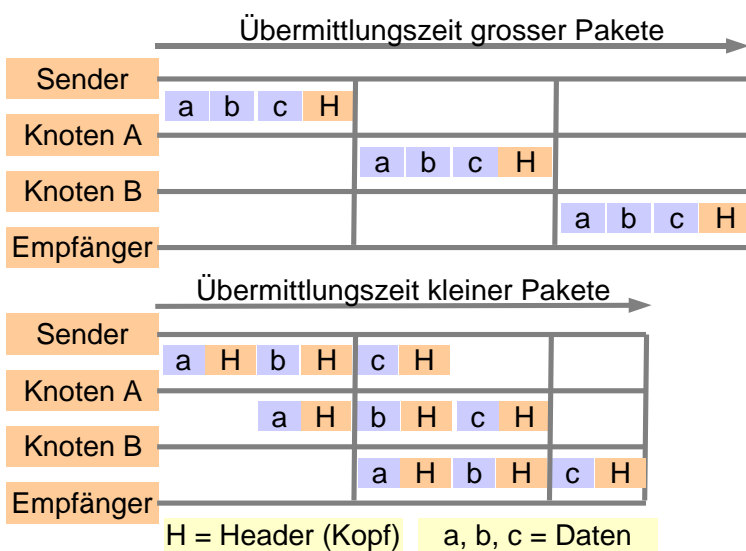


Abbildung 11.7: Der Unterschied in der Übermittlungszeit der Pakete

Kurze Pakete können schneller gesendet werden, da während der Übertragung der Daten b von Sender zum Knoten A bereits die Daten a von Knoten A zum Knoten B übertragen werden können.

11.2.1.4 Der kürzeste Pfad (Routing, Leitwege)

Bei den Virtual Circuits (VC) wird ein Leitweg vor dem Senden der Pakete ausgehandelt und dann auch benutzt. Jeder Knoten muss also wissen, wie ein Paket geroutet (gelenkt) werden soll. Nach der Benutzung wird der Leitweg (route) wieder gelöscht. Dijkstra (1959) hat ein Verfahren zur Bestimmung des kürzesten Pfades entwickelt. Eine Nachricht soll auf dem folgenden Netz von A nach D übermittelt werden. Die Pfade im Netz haben verschiedene Längen und somit werden verschiedene Leitungskosten anfallen. Ein Netzbetreiber möchte natürlich möglichst geringe Leitungskosten haben, um einen möglichst grossen Ertrag an der Übertragung der Daten zu haben. Das Problem besteht darin, wie man die einzelnen Knoten programmieren muss, damit sie die Daten auf dem kürzesten Weg übertragen. (Engpässe auf dem Netz werden in unserem Beispiel nicht betrachtet. Diese stellen dann noch eine weitere Schwierigkeit dar.)

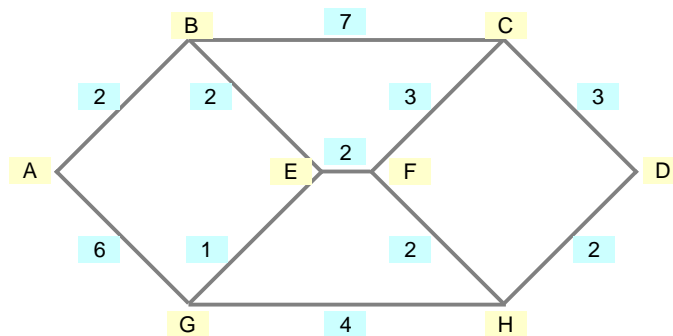


Abbildung 11.8: Ausgangslage

- a) Beginnen wir bei A und untersuchen wir für Knoten A, welches die Abstände der nächsten Knoten sind. Beschriften wir B mit (2,A), was bedeutet, dass B von A den Abstand 2 hat. G wird mit (6,A) beschriftet. Wir erkennen, dass B den kleinsten Abstand zu A hat, da B den Abstand 2 hat und G den Abstand 6.

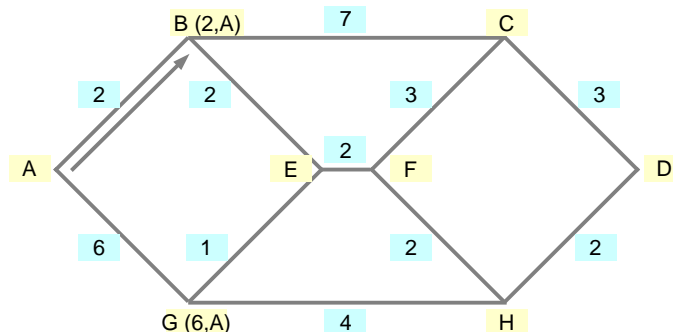


Abbildung 11.9: Situation nach dem 1. Schritt

- b) Betrachten wir alle Knotenabstände zu B, also C und E, und markieren diese Knoten wie folgt: Bis zum Punkt C sind es zusätzlich

noch 7, d.h. die totale Entfernung ist somit 9 (9,B). Bis Punkt E ist die totale Entfernung von A 4. Die Beschriftung ist somit E (4,B).

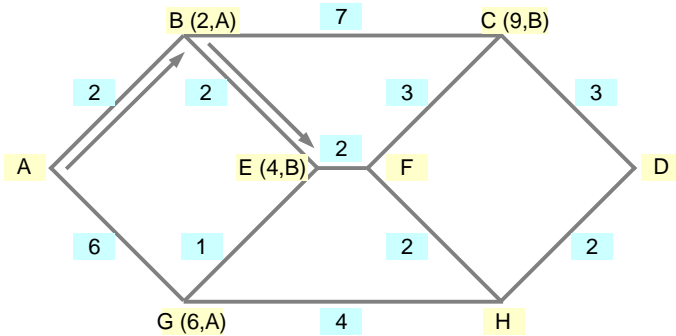


Abbildung 11.10: Situation nach dem 2. Schritt

c) Punkte um E: Wir sehen, dass sich G von G (6,A) auf G (5,E) ändert, weil der Weg über B und E kürzer ist, als von A nach G!

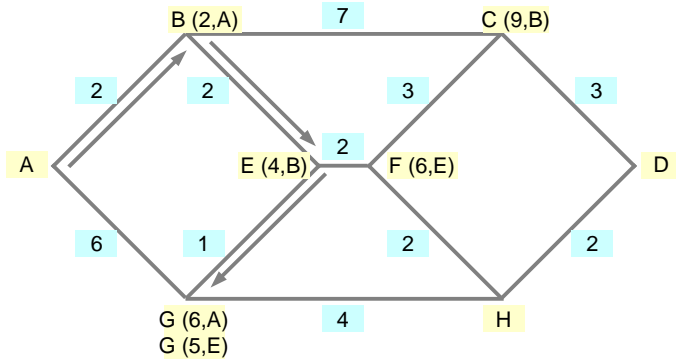


Abbildung 11.11: Situation nach dem 3. Schritt

d) Von G aus ergibt sich folgende Situation. Die zu untersuchenden Knoten sind F und H, weil E bereits untersucht wurde. Dies ergibt für H einen totalen Abstand von 9.

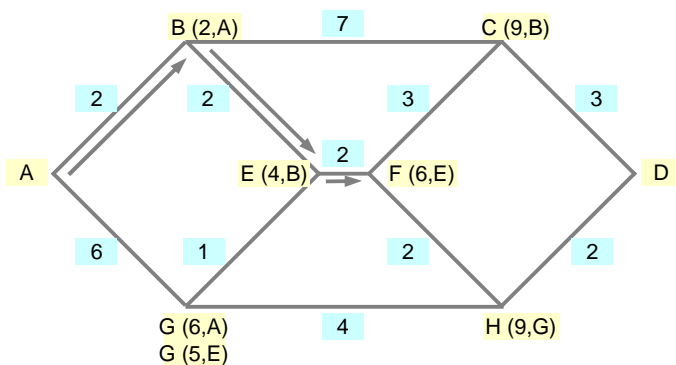


Abbildung 11.12: Korrektur im 4. Schritt

- e) Für F kommen C und H in Frage. Für H ergibt sich nun aber ein totaler Abstand von 8, da E die Daten nach der letzten Erkenntnis nicht nach G senden wird, sondern direkt nach F.

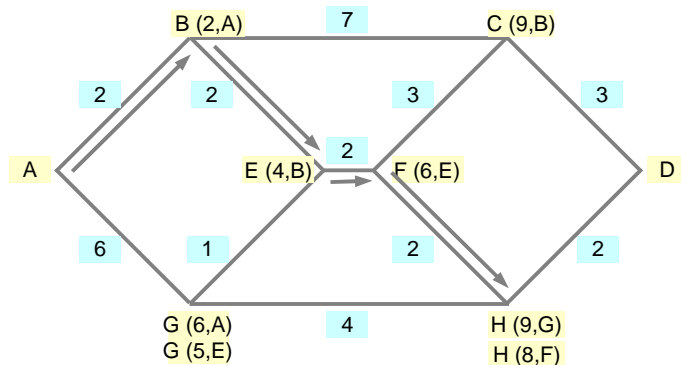


Abbildung 11.13: Situation nach dem 5. Schritt

- f) H übermittelt die Daten nach D. Der total zurückgelegte Weg beträgt 10.

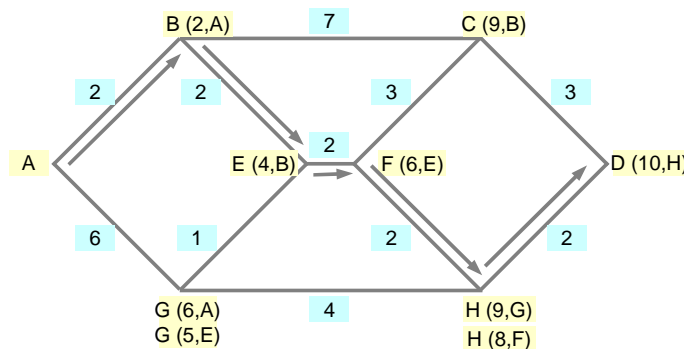


Abbildung 11.14: Der letzte Schritt

11.2.2 Verbindungslose Protokolle

Die Bestimmung von Leitwegen kann in grossen Netzen sehr aufwändig werden. Aus diesem Grund hat man Netze mit verbindungslosen Strategien entwickelt. In diesem Abschnitt werden die Nachrichteneinheiten in verbindungslosen Netzen, die Datagramme und ihre Vermittlung untersucht.

11.2.2.1 Datagramme

Datagramme heissen die unabhängigen Pakete der verbindungslosen Vermittlung. In einem Datagramm-Teilnetz werden keine Routen zum Voraus festgelegt, vielmehr leiten die Knotenrechner in diesen Netzen die Datagramme anhand ihrer Adressen auf dem kürzesten (günstigsten) Weg an ihren Bestimmungsort. So kann es geschehen, dass ein Datagramm von Zürich via Tokio und ein anderes der gleichen Nachricht via Boston nach Paris gelangt.

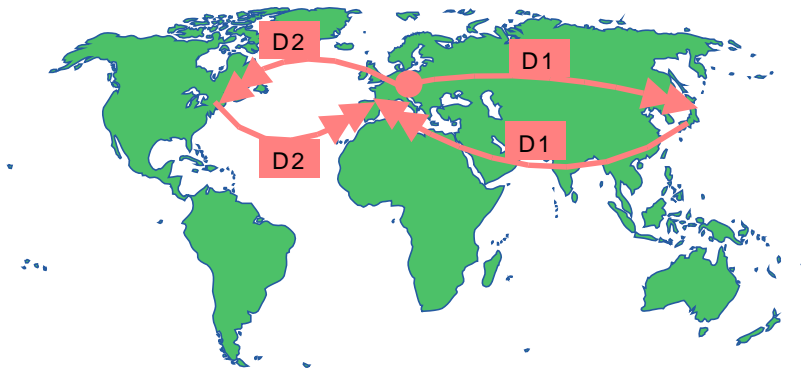


Abbildung 11.15: Vermittlung von Datagrammen im GAN

Eine typische Anwendung eines solchen Verfahrens wird im Internet und anderen TCP/IP-Netzen (UNIX-Welt) auf IP (Internet-Protokoll) basierend eingesetzt.

11.2.2.2 Die Vermittlung im IP (Internet-Protokoll)

Dieses verbindungsunabhängige Protokoll ist seit 1980 im Einsatz. Es beruht auf transparenten, wenn auch nicht immer zuverlässigen Internet-Datagrammen, die von der Quelle zum Host verschiedene Arten von Netzen durchqueren können. Abbildung 11.16 zeigt eine solche Situation:

Das IP-Protokoll arbeitet wie folgt. Die Teilnachricht (TN) wird von der darüberliegenden Transportschicht in 64-KByte-Datagrammen inklusive Transport-Nachrichtenkopf (TK) übernommen. Alle Datagramme werden in der Vermittlungsschicht mit einer IP-Nummer versehen und das Ganze wird als Paket 1 durch das Netz 1 übertragen. Paket 1 wird im Router 1 in Paket 2 umgepackt und über das Netz 2 zum Router 2 übermittelt. Dieser erstellt Paket 3 und sendet das Ganze an den Empfänger. Im Empfänger werden die Datagramme wieder vereinigt, die Teilnachrichten ausgepackt und an die Transportschicht (Layer 4) übergeben. Die Datagramme können unterwegs in kleinere Datagramme zerlegt werden.

Das Datagramm besteht aus einem IP-Kopf mit mindestens 20 Bytes, einem Transport-Nachrichtenkopf (TK) und einem Datenteil (TN). Als Besonderheit wird die Versionsnummer des Protokolls mitgesendet. Falls im Netz ein Rechner mit einem älteren oder neueren IP-Protokoll vorhanden ist, so kann das Datagramm trotzdem transportiert werden. Der Kopf enthält noch andere Angaben, wie z.B. die Angabe über die Länge des Kopfes und eine Identifikationsnummer für allfällige Fragmente eines Datagrammes.

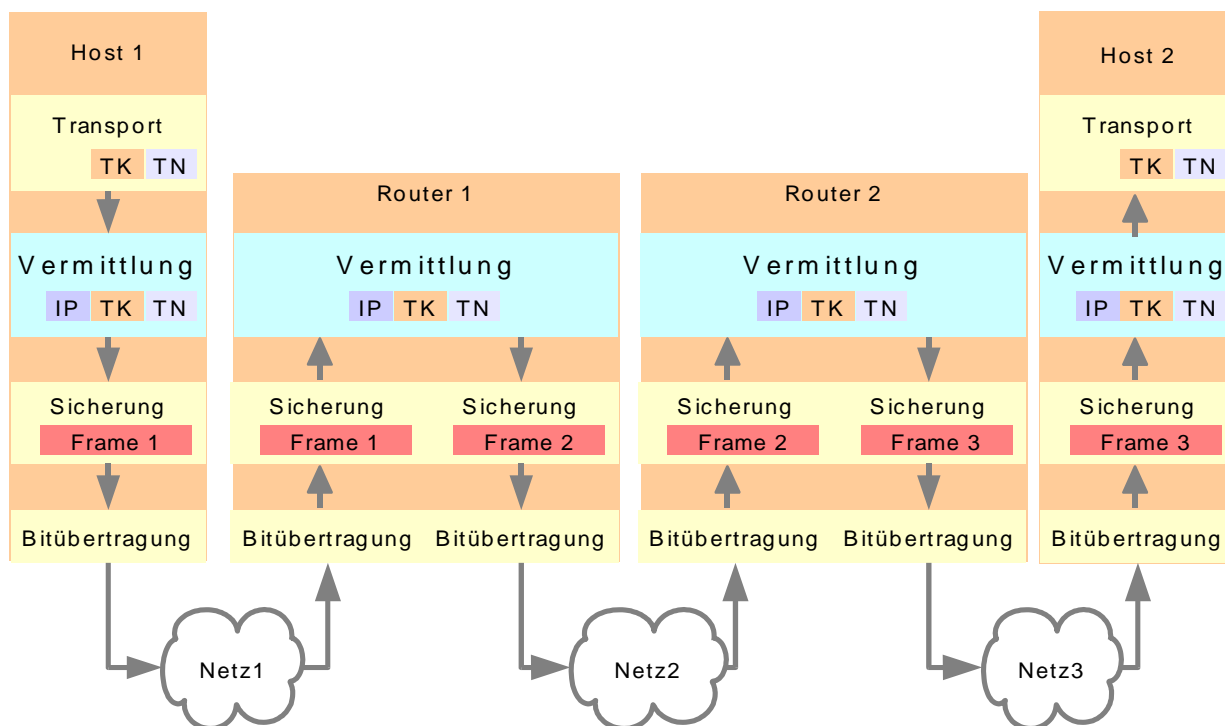


Abbildung 11.16: Vermittlung im IP-Netz

Die Datagramme werden von der Schicht 2 (Verbindungsschicht) mit den Sicherungsnachrichtenköpfen (SK1 für Netz 1, SK2 für Netz 2 und SK3 für Netz 3) und dem Sicherungsnachspann (SN1, SN2 und SN3) in netzabhängige Frames eingepackt. Dies erlaubt das Versenden von Frames über alle existierenden Netzarten (Ethernet, Token Ring, ATM, X.25 ...).

11.2.3 Vergleich zwischen Leitungsvermittlung, VC und Datagrammen

Abbildung 11.17 zeigt den Verbindungsaufbau der Leitungsvermittlung, der Paketvermittlung mit VCs und der Paketvermittlung mit Datagrammen im Vergleich. Der Sender (S) sendet in allen drei Fällen die gleichen Daten a, b und c über die Knoten (Kn.A) und (Kn.B) zum Empfänger (E).

Datagramme brauchen keinen Verbindungsaufbau und sind somit für einzelne kurze Meldungen am effizientesten.

Virtual Circuits erlauben eine einfachere Verteilung der Verkehrslast und halten die Paketsequenz ein.

Die Effizienz (Durchsatz, Throughput) hängt wesentlich von der Topologie und der Größe des Netzes sowie vom Verkehrsverhalten der Nachrichtenquellen ab.

Tabelle 11.1 vergleicht verschiedene Diskussionspunkte in Teilnetzen mit Datagrammen und solchen mit Vcs:

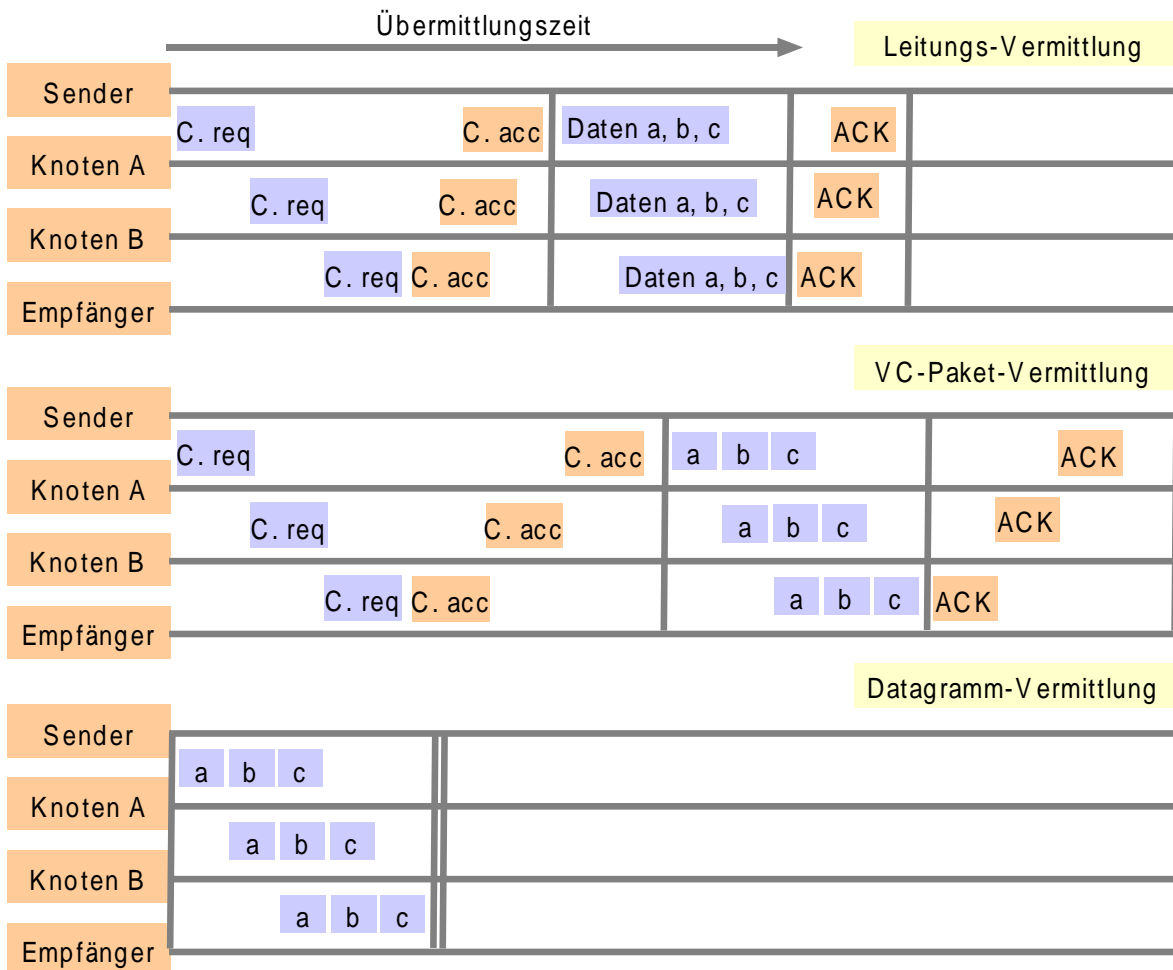


Abbildung 11.17: Vergleich der Vermittlungsarten

Diskussionspunkt	Datagramme	Virtuelle Verbindungen
Verbindungsaufbau	Nicht erforderlich	Erforderlich
Adressieren	Jedes Paket enthält die volle Quell- und Zieladresse	Jedes Paket enthält eine Nummer der virtuellen Verbindung
Statusinformationen	Das Teilnetz muss keine Statusinformationen führen	Für jede Verbindung ist ein Tabelleneintrag erforderlich
Routing	Jedes Paket wird unabhängig befördert	Die Route wird beim Aufbau der virtuellen Verbindung gewählt; alle Pakete folgen diesem Leitweg
Wirkung von Routerfehlern	Keine, ausser dass Pakete verloren gehen und noch einmal gesendet werden müssen	Alle virtuellen Verbindungen über den ausgefallenen Router werden beendet
Überlastungsüberwachung	Schwierig	Einfach, wenn im Voraus für jede virtuelle Verbindung ausreichend Puffer bereitgestellt werden.

Tabelle 11.1: Vergleich zwischen Datagrammen und VC

11.3 Aufgaben

1. Welches sind die Hauptfunktionen des 3. Layers im ISO/OSI-Modell?
2. Nach welcher Art werden bei folgenden Netzwerken die Verbindungen aufgebaut, verbindungslos oder verbindungsorientiert? ISDN, PSTN, ATM, IP, Frame Relay, X.25
3. Was verstehen Sie unter Routing?
4. Welches Routing-Verfahren wurde von Dijkstra entwickelt?
5. Erklären Sie das Routing-Verfahren von Dijkstra kurz.
6. Hat dieses Verfahren auch Nachteile?
7. Wie funktioniert der Verbindungsaufbau in einem X.25-Netz?
8. Wozu braucht ein IP-Paket Quell- und Ziel-Adresse? (Eigentlich würde doch die Ziel-Adresse ausreichen.)
9. Welches sind die unterschiedlichen Wirkungen von Router-Fehlern bei Datagrammen und bei virtuellen Verbindungen?
10. Warum braucht es bei Datagrammen keinen Verbindungsaufbau?

Lösungen unter www.sauerlaender.ch/downloads

12 Sicherung der Nachrichten

Leider gehen immer wieder Nachrichten verloren oder werden nicht empfangen. Um die Nachrichten trotzdem vollständig übertragen zu können, werden in den Netzen Sicherungsmethoden eingesetzt. Das vorliegende Kapitel beschreibt solche grundsätzlichen Sicherungsmethoden.

12.1 Dienste, Schnittstellen und Protokolle

Es wird Zeit, an dieser Stelle die drei Begriffe Dienst, Schnittstelle und Protokoll im ISO/OSI-Modell genauer zu erklären, bevor die Sicherung der Nachrichten auf der Schicht vier genauer erläutert wird. Ein *Dienst* wird einer im ISO/OSI-Modell weiter oben liegenden Schicht grundsätzlich von der unteren Schicht über eine *Schnittstelle* zur Verfügung gestellt.

Protokolle hingegen sind Abmachungen (Regelgefüge), auf deren Basis die Einheiten ihre Dienste definieren. Man kann die Protokolle beliebig ändern, so lange man nicht die für die Dienstanutzer sichtbaren Dienste ändert.

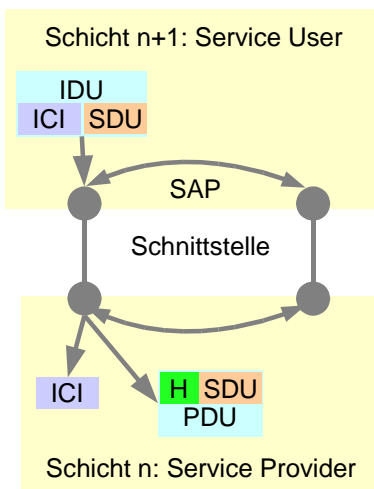


Abbildung 12.1: Zusammenhang zwischen Dienst, Schnittstelle und Protokoll

Eine Protokoll dateneinheit (Protocol Data Unit, PDU) in Schicht n beinhaltet neben einem Protokollkopf (Header, H) eine Dienst dateneinheit (Service Data Unit, SDU). Eine Schicht weiter oben benötigt einen bestimmten Dienst einer weiter unten liegenden Schicht. Schicht n+1 besitzt eine Schnittstellendateneinheit (Interface Data Unit, IDU), der bekannt ist, über welche Schnittstellensteuerdaten (Interface Control Information, ICI) sie die benötigten SDU von der unteren Schicht bekommt. Über einen Dienstzugriffspunkt (Service Ac-

Applikation

7 Anwendung

6 Darstellung

5 Sitzung

4 Transport

3 Vermittlung

2 Sicherung

1 Bitübertragung

Übertragungsmedien

cess Point, SAP) und der dazugehörenden Schnittstelle kann die Schicht n+1 den Dienst beantragen. Schicht n erkennt anhand der ICI und SDU, welcher Dienst angefordert wird und kann über den umgekehrten Weg den Dienst bereitstellen.

12.2 Aufbau, Betrieb und Abbau von Verbindungen

Die Transportschicht (Layer 4) ist die letzte der unteren Schichten im ISO/OSI-Modell, die noch direkt mit dem Datentransport auf dem Netz zu tun hat. Die höheren Schichten sind anwendungsorientierte Schichten. Die Schicht vier dient vor allem dem geordneten Aufbau, Betrieb und Abbau der Netzwerk-Verbindung. Dass vor allem der Abbau nicht immer gesichert werden kann, wird weiter hinten dargelegt.

12.2.1 Zweck

Für die Übertragung von Daten haben wir bis jetzt schon einiges bereitgestellt. Es ist uns bereits möglich, Daten in Rahmen zu verpacken und diese so abzusichern, dass Sender und Empfänger eine Chance haben, die Richtigkeit zu überprüfen und allenfalls zu korrigieren. Diese Rahmen können wir bereits in Pakete verpacken und auf verschiedene Arten einer Gegenstelle übermitteln.

Doch was geschieht, wenn Datenpakete verloren gehen oder ein Router defekt ist? Die Vermittlungsschicht würde solche Fehler je nach Protokollvariante herausfinden und die Verbindung unterbrechen. Doch wer soll diese Vorfälle den höheren Schichten mitteilen?

Die höheren Schichten des ISO/OSI-Modelles sind anwendungsorientiert und müssen sich auf einen zuverlässigen Transportdienst der Daten im Netz verlassen können (Transport Service User). Die Schichten 1 bis 3 des ISO/OSI-Modelles befassen sich mit der Bereitstellung der Daten für das Übertragungsnetz, der Fehlerkorrektur und dem Vermitteln der Daten auf den verschiedenen Netzen. Diese Schichten sind eng an die Kommunikationsteilnetze gebunden und somit abhängig von den Netzbetreibern (Beispiel WANs mit teilweise proprietären Protokollvarianten). Die Transportprotokolle sind zwischen den höheren, rein anwendungsorientierten Schichten und den tieferen, netzwerkorientierten Schichten angesiedelt und sind somit für den zuverlässigen Transport der Daten vom Quellrechner zum Zielrechner verantwortlich. Würde die Transportschicht nicht zur Verfügung stehen, hätten die Benutzer keine Möglichkeit, verlorene Pakete, auf Grund unzuverlässiger Verbindungen oder ausgefallenen Routern, festzustellen. Normalerweise bauen die Transportprotokolle für jede Transportverbindung eine eigene Netzwerkverbindung auf, mit einer eigenen Transportadresse, die es der Transportschicht des Empfängers erlaubt, verlorene Pakete und verstümmelte Daten für

Praxis-Hinweis:

Was geschieht, wenn Router defekt sind, oder Datenpakete aus anderen Gründen nicht weitergeleitet werden können? Die Schicht 4 stellt die notwendigen Dienste zur Verfügung, um die höheren Schichten zu alarmieren und sichert somit den Aufbau, den Betrieb und den Abbau der Netzwerkverbindung.

die höheren Schichten aufzubereiten. Die Schicht 4 ist somit die letzte Schicht, die diese Dienste bereitstellt und gehört somit zusammen mit den Schichten 1 bis 3 zu den Transport Service Providern.

12.2.2 Dienste, Dienstqualität und Dienstoperationen

Die Transportschicht stellt den höheren Schichten verbindungsorientierte oder verbindungslose Dienste für den Verbindungsaufbau, die Datenübertragung und den Verbindungsabbau bereit. Es werden dazu die Elemente „Adressierung“ (Transportadressen, Netzwerkadressen, NA), „Transport Protocol Data Units“ (TPDU), die „Flusssteuerung“, „Zwischenspeicher“ und „Multiplexing“ (siehe drei Pfeile beim Host 2) eingesetzt.

Damit das geschilderte Ziel erreicht werden kann, stützen die Dienste der Transportschicht auf den Diensten der Vermittlungsschicht ab. Die Hardware oder Software, die diese Aufgabe übernimmt, heisst Transportinstanz (TI). Diese Transportinstanz kann sich im Betriebssystem-Kernel (Kernstück des Betriebssystems), in einem Benutzerprozess, in einem Netzwerk-Bibliothekspaket (siehe Theorie der Betriebssysteme) oder auf der Netzwerk-Schnittstellenkarte befinden.

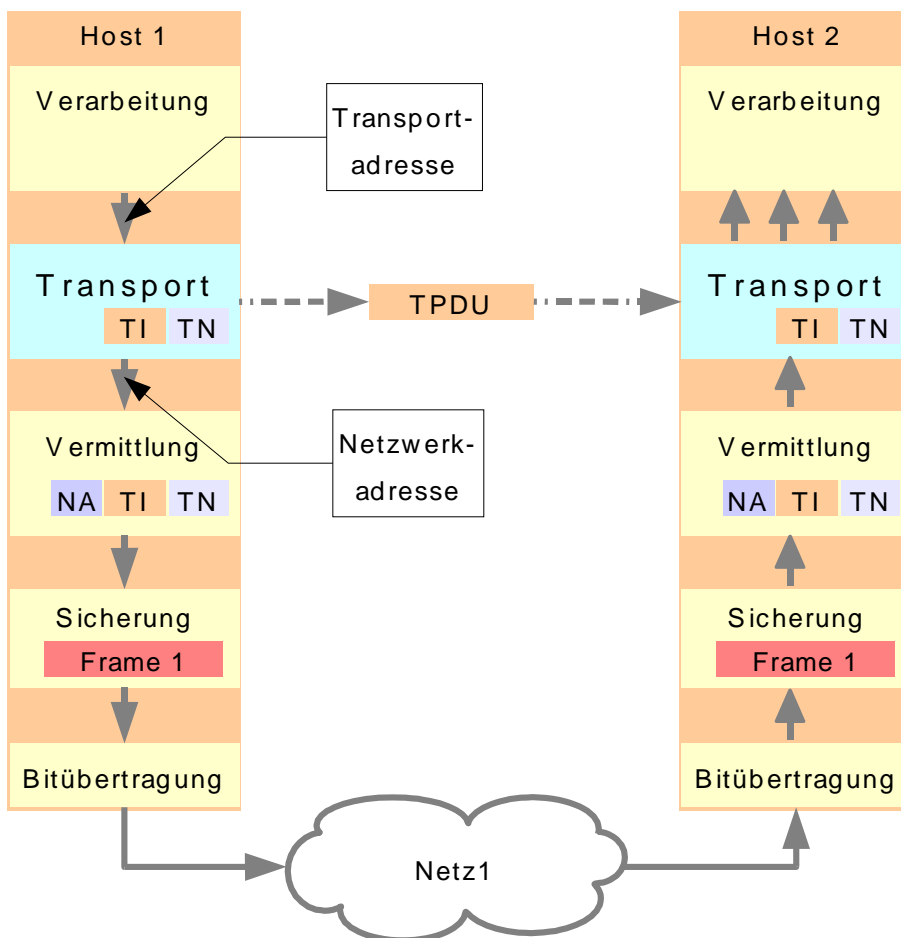


Abbildung 12.2: Die Funktion des Layers 4

12.2.3 Dienstqualitäten

Die Dienstqualitäten der Vermittlungsschicht (QoS, Quality of Service) werden durch die Transportschicht auf das vom Benutzer gewünschte Niveau angehoben. Die Transportschicht kann die Mängel von schlecht arbeitenden Vermittlungsschichten ausgleichen.

Die typischen Parameter für die Dienstqualität auf der Transportschicht sind:

- Dauer des Verbindungsaufbaues (Connection Establishment Delay). Je kürzer die Verzögerung beim Aufbau der Verbindung, desto besser der Dienst.
- Ausfallwahrscheinlichkeit beim Verbindungsaufbau (Connection Establishment Failure Probability). Bei einer Netzüberlastung kann eine Verbindung nicht innerhalb einer festgelegten Zeit aufgebaut werden.
- Durchsatz (Throughput). Es wird in kurzen Abständen in jede Richtung gemessen, wie viele Benutzerbytes pro Sekunde übertragen werden.
- Übertragungsverzögerung (Transit Delay). Es wird in jede Richtung gemessen, wie lange die Übertragung einer Nachricht von der Transportinstanz des Senders bis zur Transportinstanz des Empfängers dauert.
- Restfehlerrate (Residual Error Ratio). Diese misst die Anzahl zerstörter oder verlorener Nachrichten im Verhältnis zur gesamten Anzahl an versendeten Nachrichten.
- Schutz (Protection). Der Benutzer kann angeben, dass er seine Daten vor unerlaubtem Lesen oder Verändern durch Unbefugte (Hacker, Eindringlinge) schützen will.
- Priorität (Priority). Stuft die Verbindung als wichtig ein. Dies ist bei überlasteten Netzen von Bedeutung.
- Störungsausgleichsverhalten (Resilience). Dieser Parameter definiert, wie wahrscheinlich es ist, dass eine Transportschicht die Verbindung im Falle von Überlastung oder internen Problemen spontan beendet.

12.2.4 Dienstoperationen

Die Dienstoperationen ermöglichen den Benutzern dieser Schicht (z.B. Anwendungsprogrammen) den Zugriff auf den Transportdienst. Im Prinzip werden diese Operationen ähnlich genutzt wie die Steueranweisungen in den HDLC-Frames. Die Operatoren werden wiederum in den Nutzdaten eingekapselt und übertragen.

Abbildung 12.3 zeigt die Zusammenhänge des Einkapselns.

Ein Teil der Nutzdaten (TN) wird mit dem TPDU-L4-Header (L4 H) gekapselt und stellt somit die Paket-Nutzdaten in Layer 3 dar. Dort

werden die Frame-Nutzdaten zusammengestellt, indem die L3-Headerinformationen angehängt werden.

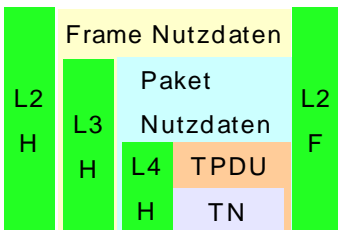


Abbildung 12.3: Das Bereitstellen der TPDU

Eine sehr einfache Transportschnittstelle kann zum Aufbau, Betrieb und Abbau beispielsweise die folgenden fünf Operatoren haben:

- LISTEN, zum Blockieren der Kommunikationseinrichtung, bis ein Prozess versucht, eine Verbindung aufzubauen.
- CONNECT, für den aktiven Versuch, eine Verbindung aufzubauen. Im Nutzdatenfeld der TPDU wird ein CONNECTION REQ. übertragen.
- SEND, um die Informationen zu senden. In der TPDU werden Daten übertragen mit dem Operator DATA.
- RECEIVE blockiert den Prozess, bis TPDU's ankommen mit den Operatoren DATA.
- DISCONNECT wird benutzt, um die Verbindung wieder abzubauen. In der TPDU wird der Operator DISCONNECT REQ. übertragen.

In UNIX heissen diese Operationen „Socket-Operationen“. Zu den oben genannten kommen noch einige zusätzliche hinzu, welche für die IP-basierenden TCP-Verbindungen wichtig sind. An dieser Stelle wollen wir diese Spezialitäten aber nicht abhandeln, da dies den Rahmen des vorliegenden Werkes deutlich sprengen würde.

12.2.5 Elemente der Transportschicht

Die Transportschicht benutzt in ihrem Protokoll ähnliche Elemente wie die Sicherungsschicht (Schicht 2). So werden auch hier Adressierung, Fehlerüberwachung, Folgesteuerung (von Paketen), Flusststeuerung, Zwischenspeicherung und Multiplexing eingesetzt.

Die Adressierung erfolgt über die Transport Service Access Points (TSAP), Transportadressen. Die analogen Adressen auf der Vermittlungsschicht heissen Network Service Access Points (NSAP), Netzadressen. TSAP und NSAP erscheinen in allen Netzen immer in Paaren. So sind im IP-Netz die IP-Adressen den NSAPs zuzuordnen und die TSAPs sind die lokalen „Ports“.

Abbildung 12.2 zeigt beim Host 2 bereits, wie eine Transportschicht der Verarbeitungsschicht mehrere TSAPs zur Verfügung stellen kann (Multiplexen). So können verschiedene Serverprozesse durch ein und dieselbe Transportinstanz aus nur einem Netz beliefert werden. Diese erklärt, weshalb die Transportinstanz eines Webserver die verschiedenen Dienste wie FTP, HTTP einem Benutzer aus dem Netz gleichzeitig zur Verfügung stellen und nebenbei noch virtuelle Verbindungen zu anderen Benutzern (durch Multiplexing) aufrecht erhalten kann.

Die Elemente Flusssteuerung und Zwischenspeicherung dienen einer geschickten Steuerung der Netzauslastung. Unzuverlässig arbeitende Vermittlungsschichten benötigen Pufferspeicher für die TPDU's, damit, ähnlich dem Mechanismus in der Sicherungsschicht, verloren gegangene Pakete aus den Puffern wieder angefordert werden können. Die Flusssteuerung und die Folgesteuerung sorgen dafür, dass die Speicher nie überlaufen und die TPDU's in der richtigen Reihenfolge an die Verarbeitungsschicht übergeben werden können.

12.3 Standards der Transportschicht

Für die in diesem Kapitel besprochenen Transportprotokolle existieren die folgenden Standards:

- ISO Transport Class 4 (ISO TOP Network)
- TCP und UDP (ARPA Network)
- SPX (Novell Netware)

12.3.1 ISO TOP

Dieses Protokoll geht davon aus, dass die Vermittlungsschicht (3) nicht ganz zuverlässig ist und übernimmt die gesamte Fehlerbehandlung und Flusssteuerung selbst. Mit diesem Protokoll können fast alle Netzwerkkarten verbunden werden. Der Aufwand dafür sind komplizierte Transportprotokolle, die einen unzuverlässigen Vermittlungsdienst kompensieren müssen.

12.3.2 TCP und UDP

Das TCP (Transmission Control Protocol) ist zusammen mit dem UDP (User Datagram Protocol) wie IP (Internet Protocol) integraler Bestandteil des ARPA-Netzwerkes. Im Gegensatz zu IP, das mit Host-Adressen arbeitet (d.h. mit System-Adressen), ermöglichen TCP und UDP eine Kommunikation zwischen Prozessen beteiligter Systeme (also zwischen Verarbeitungsinstanzen) über Ports. UDP erlaubt eine ungesicherte Übertragung ohne permanente Verbindung. TCP ist verbindungsorientiert und ermöglicht eine gesicherte Verbindung mit gepufferter Datenübertragung. Es ist duplex-fähig.

12.3.3 SPX

SPX (Sequenced Packet Exchange) ist das Pendant zum TCP von Novell. Dieses Protokoll ist proprietär.

12.3.4 Transmission Control und NetBEUI

Dies sind die Protokolle der IBM-SNA-Netze (Systems Network Architecture), wobei NetBEUI (NetBIOS Extended User Interface) zusammen mit dem Layer 6-Protokoll NetBIOS (Network Binary Input Output System) in kleinen LAN grosse Verbreitung gefunden hat. NetBEUI ist nicht ganz innerhalb Layer 4 anzusiedeln, da es noch Aufgaben aus der Sitzungsschicht übernimmt und eigentlich ein Subprotokoll des NetBIOS ist.

12.4 Abbau der Verbindung – eine heikle Sache

Der Abbau der Verbindung auf Layer 4 ist nicht ganz unproblematisch. Es besteht keine direkte Verbindung zwischen den Teilnehmern wie beim Layer 2 (Punkt-Punkt-Verbindung). Ob eine Verbindung in den dazwischenliegenden Netzen immer restlos abgebaut wurde, kann von den Teilnehmern der Kommunikation nicht immer mit Bestimmtheit garantiert werden. Das kann zu unabgebauten Verbindungsteilen und somit zum Verlust von Kapazitäten führen. Spezielle Mechanismen (z.B. Timeouts) verhindern dies jedoch.

12.5 Aufgaben

1. Problematik des Abbaus von Verbindungen.
Diskutieren Sie eine Lösung für das folgende Problem (machen Sie eine Skizze):
Zwei Armeen stehen sich vor einer entscheidenden Schlacht gegenüber. Die blaue Armee hat die weisse Armee in einem Tal eingekesselt und steht mit je 200 Mann auf den gegenüberliegenden Hügeln. Die weisse Armee im Tal hat 300 Krieger. Wenn nun die blaue Armee gleichzeitig angreift, dann ist die weisse Armee in der Unterzahl und verliert die Schlacht. Im anderen Fall ist die Hälfte der blauen Armee in der Unterzahl. Der Feldherr der blauen Armee möchte daher, dass sein Stellvertreter auf der anderen Seite des Tals zum gleichen Zeitpunkt angreift wie er und schickt einen Meldeläufer durch das feindliche Gebiet zu ihm. Kann er sich sicher sein, dass der Stellvertreter die Nachricht erhält und synchronisiert angreift? Zusatzaufgabe: Erklären Sie die Analogie mit dem Abbau von Verbindungen im Netzwerk.
2. In einem Netz gehen Datenpakete verloren. Die Vermittlungsschicht stellt diesen Verlust fest und fordert neue Pakete an. Weshalb wissen die weiter oben liegenden Schichten davon und wie wird verhindert, dass die Pakete doppelt verwendet werden? Bitte antworten Sie in wenigen Sätzen.
3. Erklären Sie bitte in wenigen Sätzen die Funktion der Transport Protocol Data Units (TPDU).
4. Weshalb ist das Multiplexing von Diensten im Layer 4 eine sinnvolle Einrichtung? Was kann damit erreicht werden? Antworten Sie in wenigen Sätzen.

Lösungen unter www.sauerlaender.ch/downloads

13 Anwendungsorientierte Funktionen

Bis jetzt haben Sie alle relevanten Dienste kennen gelernt, die eine gesicherte Nachrichten-Übertragung durch die unterschiedlichen Netze gewährleisten.

Die Nachrichten wurden von den Benutzern ursprünglich mithilfe bestimmter Applikationen erstellt. Die Empfänger sollten somit in der Lage sein, diese Nachrichten mit den entsprechenden Applikationen wieder zu lesen. Die Sender müssen somit der Nachricht Informationen zu den zu benutzenden Applikationen mitliefern.

13.1 Aufgaben der Sitzungsschicht (Schicht 5)

Bei der Besprechung der Transportprotokolle in Schicht 4 haben wir gesehen, dass die Schicht 4 bei einem Unterbruch oder einer Störung im Netz versucht, die Verbindung aufrecht zu erhalten. Ein korrekter Abbau der Verbindung zwischen den Endgeräten ist jedoch in einem solchen Fall nicht möglich, da eine unterbrochene Verbindung auch keine TPDU's mehr übertragen kann. Dies kann zu einer unerwünschten Netzüberlastung führen, wenn im Netz Kommunikationsreste von früheren Verbindungen „herumirren“ und die Kapazitäten der Netzknoten verschwenden. Damit dies verhindert werden kann, folgen nach den Transportprotokollen die Sitzungsschicht, welche die Aufgabe haben, einen logischen Kommunikationspfad (session connection) zwischen den Applikationen der beiden Endgeräte aufzubauen, zu unterhalten und wieder abzubauen.

In den folgenden Abschnitten werden die 5 Aufgaben der Sitzungsschicht genauer erläutert. Die 5 Aufgaben sind:

- Synchronisation
- Aufbau, Betrieb und geordneter Abbau beim Datenaustausch
- Dialogverwaltung
- Aktivitätsverwaltung
- Ausnahmeberichterstattung.

Während einer Datenübertragungssitzung sind diese Protokolle dafür verantwortlich, dass die zwischen den beiden Verarbeitungsinstanzen (des Senders und des Empfängers) vereinbarten Betriebsparameter eingehalten werden.

13.1.1 Synchronisationspunkte für Recovery markieren

Die Transportprotokolle stellen zwar einen gesicherten Übertragungsweg zur Verfügung. Damit kann aber ein darüberliegender Prozess nicht sicher sein, dass er oder sein Partnerprozess zu jedem Zeitpunkt die ankommende Information korrekt verarbeiten kann.

Applikation

7 Anwendung

6 Darstellung

5 Sitzung

4 Transport

3 Vermittlung

2 Sicherung

1 Bitübertragung

Übertragungsmedien

Typische Beispiele dafür sind Hardwarefehler beim Schreiben in einen Massenspeicher oder die Blockierung eines Druckers. Das Sitzungsprotokoll bringt am Informationsfluss Synchronisationsmarken an, damit das Anwendungsprogramm des Benutzers im Falle einer Störung eine entsprechende Meldung (Speicher defekt, Papier ausgegangen ...) erhält und der Fehler behoben werden kann (manuell oder automatisch). Mithilfe der Synchronisationsmarken kann die Wiederaufnahme (Recovery) der Kommunikation erreicht werden.

13.1.2 Aufbau, Betrieb und geordneter Abbau

Ähnlich wie bei den Transportprotokollen muss auch das Sitzungsprotokoll eine Sitzung aufbauen, betreiben und wieder abbauen. Der Unterschied zur Transportverbindung besteht darin, dass hier beim Abbau zuerst ein „REQUEST“ gesendet wird, um einen Datenverlust durch abruptes Trennen der Verbindung zu verhindern.

13.1.3 Die Dialogverwaltung

Viele Verbindungen arbeiten im Duplex-Betrieb. Nicht jede Software kann diesen Betrieb unterstützen.

Beispiel: Bei Anfragen auf einer Datenbank einer Fluggesellschaft macht es wenig Sinn, wenn von einem Terminal aus fünf Fragen gleichzeitig auf dem Host plaziert werden, ohne vorher die einzelnen Antworten vom System zu kennen. Damit ein Dialog mit der Datenbank aber trotz Halb-Duplex-Betrieb geordnet ablaufen kann, wird eine Dialogverwaltung benötigt.

Bei der Dialogverwaltung werden verschiedene Rechte durch Tokens (Marken) verwaltet. Nur der Partner, der das jeweilige Recht (Token) besitzt, kann entsprechende zugehörige Aktionen ausführen. Der Partner, der das Token nicht besitzt, kann es durch S-TOKEN-PLEASE.request anfordern. Tokens werden durch S-TOKEN-GIVE oder S-CONTROL-GIVE übergeben.

13.1.4 Aktivitätsverwaltung

Das Prinzip der Aktivitätsverwaltung besteht darin, dass der Benutzer den Datenstrom vorteilhafterweise in Einheiten einteilt und mithilfe dieser Einteilung den auf die Sitzungsschicht folgenden Protokollen die Möglichkeit gegeben wird, Transaktionen zu formen.

Die Einheiten werden zur Verwaltung der Transaktionen eingesetzt. Zu Beginn einer Einheit (S-ACTIVITY-START.request) setzt das Sitzungsprotokoll automatisch einen Major Synchronisation Request ab. Das Ende einer Activity wird durch ein S-ACTIVITY-END signalisiert. Eine Activity kann vorzeitig beendet, zeitweise angehalten und wieder aufgenommen werden. Nach dem Anhalten können andere Daten gesendet werden. Aktivitäten erhalten Identifikationen.

13.1.5 Ausnahmeberichterstattung

Eine weitere Funktion der Sitzungsschicht ist die Berichterstattung im Falle eines aufgetretenen Fehlers. Gerät ein Anwender aus irgendwelchen Gründen in Schwierigkeiten (z.B. kein Papier mehr im Drucker), kann er dies der Gegenstelle mit einem S-U-EXCEPTION-REPORT.request mitteilen.

13.1.6 Standards der Sitzungsschicht

Die Hersteller von Netzwerksoftware implementieren diese Kommunikationssteuerungsprotokolle auf verschiedene Weise in ihren Programmen.

- Die ARPANET-Vertreter (TCP/IP) setzen dafür den Remote Procedure Call (RPC) ein.
- Novell rüstet seine Software mit dem Service Advertising Protokoll (SAP) aus.
- IBM verwendet für PC-LANs einen Teil des NetBEUI/NetBIOS und für die Grosssysteme den Data Flow Control.

13.2 Aufgaben der Darstellungsprotokolle (Schicht 6)

Die Darstellungsprotokolle haben die Aufgabe, die folgenden Probleme der Schicht 7 auszugleichen und damit der Schicht 5 weitgehend herstellerunabhängige, bereinigte Daten zur Verfügung zu stellen:

1. Alle in der Anwendungsschicht (Layer 7) implementierten Standards sind mit unterschiedlichen Programmiersprachen umgesetzt (Notation).
2. Die Daten aus Layer 7 haben unterschiedliche Dateistrukturen und Parameter (lokale Syntax, Sprache).
3. Schliesslich gibt es einige Regeln für die bitweise Darstellung von Datenstrukturen und Parametern (lokale Codierung). Das bedeutet, dass unterschiedliche Rechner auch unterschiedliche Datenformate für Buchstaben und Zahlen verwenden.
4. Die Chiffrierung und Dechiffrierung von Daten ist eine weitere Aufgabe dieser Schicht.
5. Die Kompression und Dekompression von Daten findet ebenfalls in Layer 6 statt.

13.2.1 Beispiel zur Aufgabe der Codierung

Stellvertretend für die Aufgaben der Darstellungsschicht betrachten wir eine Auswahl aus der Vielfalt der Zeichencodes. Es ist auch ganz nützlich, wenn wir einige der Zeichencodes kennen.

Praxis-Hinweis:

Mithilfe dieser Aktivitätsverwaltung wird der Mangel der Transportinstanz, eine Verbindung auch im Falle unterbrochener Verbindungen sauber zu beenden, kompensiert.

<i>Internationales Alphabet Nr. 2</i>	<i>Internationales Alphabet Nr. 5</i>	<i>EBCDIC Alphabet</i>	<i>PC Alphabet</i>
Auch bekannt unter dem Namen IA2	Auch bekannt unter den Namen IA5 und vor allem ASCII.	Firmenstandard von IBM.	Standard in der PC-Welt.
Der Code besteht aus fünf Bit. Dementsprechend werden nur Grossbuchstaben, Ziffern und einige Spezialzeichen codiert. Verwendung vor allem im Bereich Fernschreiber/Telex. Wird immer weniger verwendet (Lochstreifen sind out).	Der Code besteht aus sieben Bit, wobei ein Bereich mit nationalen Sonderzeichen belegt werden kann. Es bestehen Versuche, den Code auf acht Bit zu erweitern. Verwendung im Bereich von Textterminals, zur Textübertragung etc. Sehr verbreitet.	Der Code besteht aus acht Bit. Er enthält ebenfalls nationale Sonderzeichen. Üblich in der IBM-Welt.	Der Code besteht aus acht Bit, wobei im 7-Bit-Bereich praktisch der ASCII-Zeichensatz übernommen wird. Enthält im oberen Bereich (über 128) unter anderem nationale Sonderzeichen, Grafikzeichen und einige griechische Buchstaben. Durch die Verwendung im PC-Bereich sehr verbreitet.

Tabelle 13.1: Zeichensätze⁶²

Praxis-Hinweis:

Bei Novell sind diese Dienste im Netware Core Protokoll (NCP) untergebracht. Bei IBM sind diese Protokolle für PC-Netze im Net BIOS und für grosse Netze in den Presentation Services untergebracht.

13.2.2 Standards der Darstellungsschicht

Auch hier haben die verschiedenen Netzwerksoftware-Anbieter wieder unterschiedliche Lösungen vorgeschlagen: ARPANET (TCP/IP) nennt diese Protokolle Lightweight-Presentation-Protokoll (LPP) und External Data Representation (XDR).

13.3 Aufgaben der Anwendungsprotokolle (Schicht 7)

Die Anwendungsprotokolle stellen die Verbindung zu den Anwenderprogrammen sicher. Mithilfe dieser Protokolle kann vermieden werden, dass sich ein Programmierer einer Applikation mit den Netzwerkfunktionen abgeben muss. Er muss lediglich die Schnittstelle zu den standardisierten Protokollen der Anwendungsschicht kennen und kann seine Programme darauf aufbauen.

Unterschiedliche Dateisysteme haben oft verschiedenartige Konventionen für Dateinamen oder zur Darstellung von Text; diese Inkompatibilitäten werden mittels der Anwendungsprotokolle behoben.

Beispielsweise gehört die Behandlung der elektronischen Post (E-Mail) in diese Protokollgruppe, aber auch die Funktion des Server Message Blocks (SMB) von IBM und andere vergleichbare Aufgaben.

⁶² EBCDIC: Extended Binary Coded Decimal Information Code
ASCII: American Standard Code for Information Interchange

Einige Protokolle sollen im Folgenden genannt werden. Die Liste kann nicht vollständig sein und wird auch laufend durch neue Protokoll-Varianten erweitert.

13.3.1 Beispiele von Protokollen (TCP/IP, ARPANET)

Die Protokolle des Application Layers sind von den Herstellern von Applikationssoftware oft in ganz spezieller Art und Weise implementiert worden. Dies hat zur Folge, dass eine ganze Vielzahl solcher Protokolle existiert. Einige der Protokolle wollen wir an dieser Stelle genauer betrachten.

Für TCP/IP-Netze gibt es folgende Protokolle:

13.3.1.1 FTP (File Transfer Protocol)

FTP ist dasjenige Protokoll, mit dessen Hilfe es möglich ist, Daten zwischen Rechnern im TCP/IP-Netz zu übertragen. Gleichnamige Programme von diversen Herstellern benutzen dieses Protokoll in ihren Applikationen.

Für das Herunterladen von Daten von einem Server über das TCP/IP-Netz müssen wir zuerst eine Verbindung vom Client zum Server herstellen. Die Applikation verlangt daher die IP-Adresse (oder den IP-Namen) des Servers. Normalerweise benötigt man ein Passwort, um auf die Dateien eines fremden Rechners (Server) zugreifen zu können. Einige Daten sind jedoch auf dem Internet frei erhältlich und es ist möglich, mit dem so genannten „anonymous FTP“ darauf zuzugreifen und herunterzuladen. Beim anonymen FTP müssen wir auf dem Client bei der Benutzeridentifikation das Wort „anonymous“ eingeben und bei der Passwortaufforderung geben wir unsere E-Mail-Adresse ein.

13.3.1.2 Telnet (Virtual Terminal Emulation)

Das Telnet-Protokoll wurde entwickelt, um von Remote-Rechnern aus auf Grossrechnern oder Unix-Servern Programme ausführen zu können. Das gleichnamige Programm benutzt dieses Protokoll. Auch hier müssen Sie eine Benutzeridentifikation und ein Passwort eingeben, um die Dienste des Hosts zu nutzen. Der Host muss über ein leistungsfähiges Multi-User-Betriebssystem verfügen.

13.3.1.3 SMTP (Simple Mail Transfer Protocol)

Dies ist das Protokoll, das von den E-Mail-Programmen benutzt wird, um unsere elektronische Post auf dem TCP/IP-Netz zu versenden. Dem Protokoll liegen die Mailprotokolle des UNIX zu Grunde. Weil die Mail-Protokolle des UNIX mit eigenen acht-Bit-Codes arbeiten, müssen alle anderen Codes umgewandelt werden. Wird diese Umwandlung nicht durchgeführt, ist es beispielsweise nicht möglich, deutsche Umlaute per E-Mail zu übermitteln.

Praxis-Hinweis:

Einige Firmen haben eigene Protokolle für ihre Netzwerkimplementationen geschrieben und es existieren daher eine ganze Vielzahl von Applikations-Protokollen. Neben den TCP/IP-ARPANET-Protokollen haben vor allem IBM, ISO, DEC, 3Com, Xerox, Apple und Banyan VINES solche Protokolle entwickelt.

Praxis-Hinweis:

Die Protokolle setzen zum grossen Teil direkt auf dem TCP (Transmission control Protocol) oder dem UDP (User Datagram Protocol) auf.

Praxis-Hinweis:

Der Aufruf von FTP in einem kommandozeilenorientierten Betriebssystem lautet:
ftp <Internetadresse>

Praxis-Hinweis:

Der Aufruf von Telnet in einem kommandozeilenorientierten Betriebssystem lautet:
telnet <Internetadresse>
oder
telnet hostname.ort.land

Praxis-Hinweis:

Der Client kann auf dem Server Seiten im HTML-Format aufrufen, wenn er den URL (Uniform Resource Locator) (z.B. www.acm.org) oder die Adresse (z.B. 194.235.16.2) der Seite kennt.

13.3.1.4 HTTP (HyperText Transfer Protocol)

HTTP ist das Kommunikationsprotokoll zwischen World Wide Web (WWW)-Servern und WWW-Clients.

Das HTTP übermittelt Hyper Text im HTML-Format (Hyper Text Markup Language).

Die Verbindung des Clients mit einem Server geschieht über eine TCP/IP-Verbindung.

13.3.1.5 Protokolle für UNIX-Systeme

Für UNIX-Systeme existieren die folgenden Protokolle:

X-Windows-Systems (Betriebssystem Oberflächen-Schnittstelle)

lpr Remote print

rcp remote copy

rex ec remote execution

login remote login

rsh remote shell

NFS Network File System

UUCP Unix to Unix Copy Program

Für die Steuerung oder die Verwaltung von TCP/IP-Netzen sind die folgenden Protokolle im Einsatz:

13.3.1.6 DNS (Domain Name System)

Für die Benutzer von TCP/IP-Netzen ist die IP-Adresse im Zahlenformat (xxx.xxx.xxx.xxx) nicht unbedingt brauchbar. Aus diesem Grund wurde das Domain Name System erfunden. Dieses System stellt einen Bezug her zwischen einer Adresse im Zahlenformat und dem besser verständlichen Namen im Textformat. So wird beispielsweise aus der IP-Adresse 194.235.xxx.xxx der Name im Textformat www.dus.ch.

13.3.1.7 SNMP (Simple Network Management Protocol)

Mit zunehmender Grösse der Netze stieg auch die Anfälligkeit auf Fehler. In TCP/IP-Netzen besteht eine Möglichkeit, nicht funktionierende Rechner von jedem Punkt aus im Netz ausfindig zu machen. Man sendet kleine Pakete an den fehlerhaften Rechner und wartet auf die Antwort. Gibt der Rechner Antwort, ist die Verbindung und der Rechner in Ordnung. Das Verfahren benötigt das Programm „Ping“. Bald einmal war diese Lösung nicht mehr angemessen und man entwickelte ein eigenes Protokoll für die Netzwerk-Überwachung und -Verwaltung, das SNMP. Dieses Protokoll wurde in vielen kommerziellen Netzwerk Management Systemen eingebaut und hat heute eine grosse Verbreitung.

13.3.1.8 DHCP (Dynamic Host Configuration Protocol)

Das DHCP ermöglicht die Verwaltung der Netzwerkadressen in einem Netzwerk. Ein DHCP-Server verteilt eine bestimmte Anzahl

Praxis-Hinweis:

Mit diesem Protokoll lassen sich alle Arten von Netzkomponenten überwachen. Hosts, Router, Bridges, Drucker und andere Geräte haben heute diese Protokolle in ihrer Firmware implementiert und können somit Statusinformationen an den Netzmanager senden.

von IP-Adressen innerhalb eines Netzwerksegmentes dynamisch an die Stationen. Die Adressen sind also nicht mehr fest einer Station zugeordnet, sondern werden nur im Bedarfsfall einer Station ausgeliehen. Benutzt die Station das TCP/IP-Netz nicht mehr, gibt sie die Adresse dem DHCP-Server zurück.

13.3.1.9 NTP (Network Time Protocol)

Das NTP wird zur Zeitsteuerung in einem globalen Netzwerk benutzt.

13.3.1.10 RPC (Remote Procedure Calls)

Das RPC ermöglicht die Fernsteuerung eines externen Rechners. Die Remote Procedure Operationen basieren auf dem Client-Server-Modell. Das heisst, dass ein Client eine Anfrage an den Server sendet, eine Aktion abwartet und kontrolliert, ob die Anfrage fehlerfrei erkannt wurde.

13.3.1.11 Weitere Protokolle im Überblick

TFTP Trivial File Transfer Protocol

FTAM File Transfer Access und Management

VTS Virtual Terminal Service etc.

Die Liste ist nicht abschliessend, weil hier, wie bereits gesagt, sehr viele Protokolle existieren.

13.3.1.12 Beispiele von Protokollen (Novell)

Novell hat eigene Protokolle entwickelt: die NDS (Netware Directory Services) und das Netware Lite.

13.3.1.13 Beispiele von Protokollen (IBM-PC und IBM-SNA)

Der PC-LAN-Manager und diverse Remote Network Program Loader von IBM sind in vielen modernen PC-Betriebssystemen implementiert. Diese Protokolle sind direkt mit dem NetBIOS in Layer 6 verbunden. Für Grossrechner existieren unter dem Begriff SNA Transaction Services einige Netzwerkprotokolle, die direkt auf dem Data Flow Control in Layer 5 aufsetzen.

13.4 Die vollständige Kommunikation

Abbildung 13.1 zeigt eine Kommunikation über mehrere Stationen.

In der Bitübertragungsschicht muss jedes Gerät mit dem Nachbargerät physisch mit der richtigen Schnittstelle und Übertragungstechnologie (z.B. Ethernet, Twisted Pair-Kabel, RJ45-Stecker) verbunden sein. Im Layer 2 sind die Geräte über eine virtuelle Verbindung gekoppelt. Geräte, wie Repeater oder Modems, die keinen Layer 2 haben, merken somit nichts von den virtuellen Verbindungen anderer Geräte. Router unter sich kommunizieren über die virtuelle Verbindung in Layer 3. Hier gilt ebenfalls, dass alle Geräte ohne Layer 3

oder Geräte aus „fremden“ Netzen (hier ATM) nichts „merken“ von dieser Verbindung. Weiter oben liegende Schichten (sofern vorhanden!) kommunizieren somit direkt mit den Endpunkten zwischen Client und Server. Die Geräte dazwischen, die lediglich einen Aspekt der Kommunikation bereitstellen, merken davon nichts.

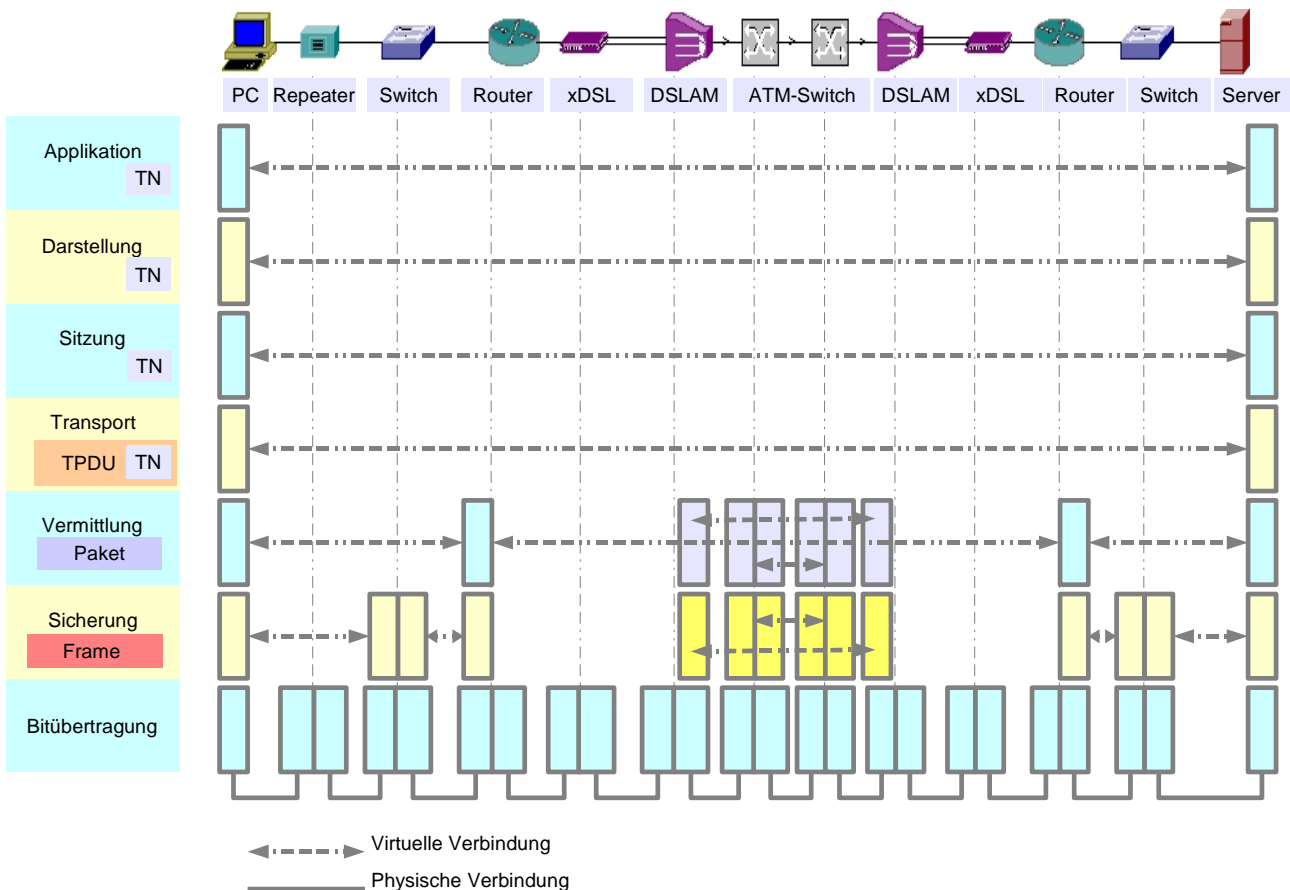


Abbildung 13.1: Eine Übertragungsstrecke (vereinfacht)

13.5 Aufgaben

1. Installieren Sie in einem Labornetzwerk einen Netzwerkscanner (z.B. Ethereal) und erfassen Sie einige Frames. Suchen Sie nach den bisher besprochenen Protokollnamen und Diensten. Erstellen Sie eine Liste mit den gefundenen Protokollnamen und geben Sie an, auf welcher Seite im Buch diese besprochen wurden. (Hinweis: Es ist verboten, ohne Erlaubnis ein produktives Netz zu scannen. Falls Ihr Labornetzwerk keine sinnvollen Frames aufweist, verwenden Sie die Unterlagen aus der Lösung.)
2. Wie stellen die Netzwerkscanner die Protokolle der verschiedenen Schichten dar? Zeigen Sie das anhand eines Beispielausdrucks und mit einem Satz als Erklärung.

Lösungen unter www.sauerlaender.ch/downloads