Prof. Dr.-Ing. Firoz Kaderali

Kommunikationsnetze und -protokolle

Offene Systeme, X.21 und X.25 Protokolle, LANs und MANs, ISDN, B-ISDN, GSM, WLAN, Internet Protokoll, Agenten im Internet

Das Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung des Autors reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Vorwort

Die Entwicklung der Informations- und Kommunikationstechnologie hat zu einer Vielzahl von Systemen, Netzen und Diensten geführt. Kommunikationsprotokolle bilden eine essentielle Grundlage für deren erfolgreiches Funktionieren. Im Kurs **Kommunikationstechnik I und II** wurden die Grundlagen der digitalen Übertragung und Vermittlung behandelt. Im folgenden zweisemestrigen Kurs **Kommunikationsnetze und -protokolle** werden einige wichtige digitale Kommunikationsnetze und deren Protokolle systematisch behandelt. Zwar ist dieser Kurs so angelegt, dass er unabhängig vom Kurs **KT I und II** studiert werden kann, Grundkenntnisse der digitalen Übertragungs- und Vermittlungstechnik sind jedoch sicher sehr hilfreich.

Im ersten Kapitel werden die Grundgedanken der offenen Systeme und deren Normung übermittelt. Es folgen Protokolle verschiedener Netze, im Einzelnen: X.21, X.25, LANs, MANs, ISDN, SSNr7, GSM, B-ISDN, Internetprotokolle und Agenten im Internet.

Für die Erstellung vieler Übungsaufgaben, die Durchsicht der Manuskripte und deren Überarbeitung (nunmehr in der dritten Auflage) danke ich besonders meinen Mitarbeitern Hagen Hagemann, Burkhard Heyber, Christian Grosch, Bernhard Löhlein, Jürgen Apfelbeck, Dirk Westhoff, Andreas Grebe, Thorsten Kisner und meinen Sekretärinnen Susanne Oetzel und Jessica Gabski.

Die Erstellung der Graphiken erfolgte durch die Mitarbeiter Barbara Wegner, Eva Otto, Hendrik Vitte und Ulrike Schmidt.

Hagen im Sommer 2005

Autorenvorstellung

Prof. Dr.-Ing. Firoz Kaderali

1963-69	Studium der Theoretischen Elektrotechnik an der Technischen Hochschule Darmstadt
1969-72	Assistent/Dozent für Statistische Signaltheorie an der Technischen Hochschule Darmstadt
1974	Promotion auf dem Gebiet der Netzwerktheorie an der Technischen Hochschule Darmstadt
1976-81	Projektleiter (Digitales Ortsnetz) im Forschungszentrum der Firma SEL (ITT)/Stuttgart
1981-86	Hauptabteilungsleiter (Systementwicklung Großsysteme) bei (Bosch) Telefonbau und Normalzeit/Frankfurt
Seit 1986	Professor für Kommunikationssysteme an der FernUniversität Hagen Arbeitsgebiete: Kommunikationssysteme, -netze und -protokolle; Datenschutz und Datensicherheit in Kommunikationsnetzen; Einsatz von neuen Medien in der Lehre
1989-94	Leiter der Projektträgerschaft TELETECH NRW
1990-96	Mitglied der ISDN Forschungskommission des Landes NRW
Seit 1992	Direktor des Forschungsinstituts für Telekommunikation (FTK)/Dortmund
1995 - 2001	Mitglied der Steuerungsgruppe der Landesinitiative media NRW
1999 - 2002	Sprecher des Forschungsverbundes Datensicherheit NRW
2000-2002	Vorsitzender des Beirates der Gesellschaft für IT-Sicherheit in Bochum
Seit 2002	Vorsitzender der Open Source Initiative CampusSource

Gliederung

Vorwort ..... **iii**

Autorenvorstellung ..... **iv**
Prof. Dr.-Ing. Firoz Kaderali iv

1 Offene Systeme **1**
1.1 Anwendungsbeispiele 1
1.2 Ziele beim Entwurf offener Systeme 4
1.3 OSI-Begriffe 6
1.4 Kommunikationstechnische Aufgaben 14
1.5 Einige Anmerkungen zum OSI-Modell 26
1.6 Spezifikation und Testen von Protokollen 27

2 Die X.21 und X.25 Protokolle **38**
2.1 Die X.21 - Empfehlung 38
2.2 Transparenz, Sicherung und Quittierung 41
2.3 Die HDLC-Prozedur 50
2.4 Das X.25 Protokoll 56

3 LANs und MANs **66**
3.1 Die Entwicklung von lokalen Netzen 66
3.2 Die CSMA-LANs 73
3.3 Die Token-LANs 91
3.4 WANs, MANs, HSLANs 101

4 ISDN - Das diensteintegrierende Kommunikationsnetz **126**
4.1 Das ISDN-Referenzmodell 126
4.2 Die Bitübertragungsschicht 129
4.2.1 Die S-Schnittstelle 129
4.2.2 Die U-Schnittstelle 139
4.3 Die Sicherungsschicht 146
4.4 Die Vermittlungschicht 152
4.5 Das Signalisierungssystem Nr. 7 160
4.6 Dienste und Leistungsmerkmale 175

5 B-ISDN: Die ATM-Technik **187**
5.1 ATM als Basis für B-ISDN 187
5.2 Anforderungen von Breitbandanwendungen 196
5.3 Die Referenzmodelle 199
5.4 Die physikalische Schicht 202
5.5 Die ATM-Schicht 214
5.6 Die ATM-Adaptionsschicht 224
5.7 ATM-Verkehrskontrolle 234

6 GSM **241**
6.1 Das Konzept 241

6.2 Die Dienste ... 244
6.3 Netz- und Protokollarchitektur ... 248
6.3.1 Netzkomponenten und Systemhierarchie ... 248
6.3.2 Schnittstellen und Schichtung der Protokolle ... 252
6.3.3 Transport und Signalisierung ... 256
6.4 Übertragungstechnik ... 258
6.4.1 Funktechnik ... 258
6.4.2 Die GSM-Luftschnittstelle ... 259
6.4.2.1 FDMA-Komponente und TDMA-Rahmenaufbau ... 259
6.4.2.2 Logische Kanäle ... 264
6.4.2.3 Abbildung Logischer auf Physikalische Kanäle ... 268
6.4.2.4 Fehlerschutzmechanismen (Codierung, Verwürfelung und Frequenzspringen) ... 271
6.5 Protokolle ... 274
6.5.1 Kennungen in GSM ... 274
6.5.2 Call Management ... 277
6.5.2.1 Mobile Originated Call und Mobile Terminated Call ... 277
6.5.2.2 Handover ... 279
6.5.3 Mobilität ... 282
6.5.3.1 Location Update und Paging ... 282
6.6 Netz-Management ... 284
6.7 GSM-Standard ... 286

7 Wireless Local Area Networks ... 287
7.1 Einleitung ... 287
7.1.1 Komponenten ... 288
7.2 Übertragungsprinzipien ... 289
7.2.1 Direct Sequence Spread Spectrum (DSSS) ... 289
7.2.2 Frequency Hopping Spread Spectrum (FHSS) ... 289
7.2.3 Frequency Division Muliplexing (FDM) ... 290
7.2.4 Orthogonal Frequency Division Multiplexing (OFDM) ... 290
7.2.5 Gaussian Frequency Shift Keying ... 291
7.2.6 Verwendung der Übertragungsverfahren ... 291
7.3 IEEE 802.11 ... 292
7.3.1 Funkübertragung ... 292
7.3.1.1 Frequency Hopping Spread Spectrum (FHSS) ... 293
7.3.1.2 Direct Sequence Spread Spectrum ... 295
7.3.2 Nachrichtenformate ... 297
7.3.3 Zugriffskontrollebene ... 300
7.3.3.1 Carrier Sense Multiple Access with Collission Avoidance ... 301
7.3.3.2 Contention Free Period ... 303

7.3.4 Netzaufbau ... 304
7.3.4.1 Ad-Hoc-Modus ... 305
7.3.4.2 Infrastruktur-Modus ... 305
7.3.4.3 Dienste ... 306
7.3.5 Übersicht über die Standards ... 306

8 Internet-Protokolle ... 308
8.1 Einleitung ... 308
8.2 Das TCP/IP-Modell ... 310
8.3 Die Standardisierung ... 312
8.4 TCP / IP-Zugriff auf die Netz-Interface-Schicht ... 313
8.4.1 Maximum Transfer Unit ... 313
8.4.2 Address Resolution Protocol ... 315
8.5 TCP/IP-Protokolle der Vermittlungsschicht ... 317
8.5.1 Internet Protocol Version 4 ... 317
8.5.2 IPv4-Adressierungsschema ... 322
8.5.3 Internet Control Message Protocol ... 326
8.5.4 Internet Group Management Protocol ... 327
8.5.5 Resource Reservation Protocol ... 329
8.5.6 Multiprotocol Label Switching ... 330
8.5.7 IP-Routing ... 330
8.5.8 Internet Protocol Version 6 ... 333
8.5.8.1 Entwicklung ... 333
8.5.8.2 Das Protokoll: IPv6 ... 334
8.5.8.3 Erweiterungs-Header ... 335
8.5.8.4 IPv6-Adressierung ... 341
8.5.8.5 Quality of Service ... 346
8.5.8.6 Übergangsstrategien ... 348
8.6 TCP/IP-Protokolle der Transportschicht ... 350
8.6.1 User Datagram Protocol (UDP) ... 350
8.6.2 Transmission Control Protocol (TCP) ... 352
8.6.2.1 Zuverlässiger Transport von Daten ... 352
8.6.2.2 Der TCP-Header ... 353
8.6.2.3 Virtual Circuits ... 356
8.6.2.4 TCP-Fenstermechanismus ... 359
8.6.2.5 TCP-Performance-Merkmale ... 363

9 Agenten im Internet ... 364
9.1 Eigenschaften von Agenten ... 364
9.2 Arbeitsumgebung mobiler Agenten ... 366
9.3 Basistechnologien ... 367
9.3.1 Kommunikationskonzepte ... 367
9.3.2 Remote Procedure Call ... 369
9.3.3 Remote Programming ... 370
9.3.4 Vor- und Nachteile von Mobilen Agenten ... 371
9.4 Mobilität und Kommunikation ... 372
9.4.1 Mobilität ... 372

9.4.2 Kommunikation 374
9.4.2.1 Direkte Kommunikation 374
9.4.2.2 Indirekte Kommunikation 375
9.4.3 Migration von Mobilen Agenten 376
9.5 Arbeiten auf dem Gebiet mobiler Agenten 377
9.5.1 Aglets 378
9.5.2 D'Agents 379
9.5.3 Mole 380
9.5.4 AAPI-Paket 381
9.5.5 JAE 381
9.6 Überblick 382

Lösungshinweise zu Selbsttestaufgaben 385

Übungsaufgaben 407

Lösungen zu Übungsaufgaben 431

Literaturverzeichnis 471

1 Offene Systeme

1.1 Anwendungsbeispiele

Wir wollen zunächst einige Beispiele, wie sie heute in Telekommunikationsanwendungen häufig auftreten im Hinblick auf die Kommunikationsabläufe näher ansehen.

In Abb. 1.1-1 wählt sich ein Anwender über das Fernsprechnetz auf einen Firmenserver ein. Hierzu verwendet er einen PC als Endgerät, über das er mit dem Firmenserver kommuniziert. Die Kommunikation wird über das analoge Fernsprechnetz unter Verwendung eines Modems abgewickelt. Auf der Serverseite wird wiederum ein entsprechendes Modem verwendet. Der Server bedient mehrere Applikationsprozesse gleichzeitig, d. h. er bedient mehrere Anwender quasi zur gleichen Zeit.

Abb. 1.1-1: Datenbankabfrage über Fernsprechnetz

Die lokale Kommunikation zwischen dem Anwender und dem PC wird über einen Bildschirm, eine Tastatur und eine Maus unter Verwendung einer graphischen Benutzeroberfläche abgewickelt. Die Kommunikation zwischen dem PC und dem Server wird je nach verwendetem Dienst über ein spezielles Protokoll abgewickelt. Der Zugang zum Netz erfolgt jeweils über Modems, die nach einem genormten Datenübermittlungsverfahren miteinander kommunizieren. Das Fernsprechnetz besteht aus zahlreichen übertragungs- und vermittlungstechnischen Einrichtungen, über die eine Verbindung zwischen den Modems unterhalten wird.

Abb. 1.1-2: *Client-Server* Anwendung über ein lokales Netz

In Abb. 1.1-2 ist eine *Client-Server* Anwendung über ein lokales Netz dargestellt. Typisch für lokale Netze ist, daß ein Übertragungsmedium (z. B. ein Twisted-Pair-Kabel oder Funknetzwerk) mehreren Teilnehmern gemeinsam zur Verfügung steht, und der jeweilige Zugriff des Teilnehmers zum Medium geregelt sein muß.

Abb. 1.1-3: Kommunikation über ISDN

PC-Karte
ISDN

In Abb. 1.1-3 ist ein Kommunikationsprozess zwischen zwei Teilnehmern dargestellt. Hier wird die Kommunikation unter Verwendung jeweils einer PC-Karte über das digitale Fernmeldenetz (ISDN - ***I**ntegrated **S**ervices **D**igital **N**etwork*) abgewickelt. Dem Leser dürften viele weitere Anwendungsbeispiele bekannt sein, unter anderem auch solche, bei denen keine Menschen beteiligt sind, sondern lediglich Rechner, Sensoren, Aktoren und andere Geräte miteinander kommunizieren.

Allen Beispielen ist gemeinsam, daß ein **Anwendungsprozeß** an beiden an dem Kommunikationsvorgang beteiligten Einrichtungen abläuft und verschiedene Geräte über**Schnittstellen** miteinander verbunden werden. Eine Schnittstelle hat stets einen lokalen Bezug und wird durch die über sie ausgetauschten physikalischen Signale (die Leitungen und über sie meßbare Ströme, Spannungen, Impulse, Impulskombinationen usw.) und deren Bedeutung definiert. Einige typische Beispiele solcher Schnittstellen sind V.24 zwischen Endgerät und Modem, PCI- oder PCI-E Busschnittstellen zwischen PC und ISDN PC-Karte und S_0 zwischen der PC-Karte und dem ISDN-Netz oder auch USB oder IEEE 1394 Firewire.

Anwendungsprozeß

Schnittstelle

Die von den einzelnen Kommunikationseinrichtungen (wie Endgerät, Host, Modem, Multiplexer, Konzentrator, Vermittlungsstelle usw.) erbrachten Funktionen werden in **anwendungsorientierte Funktionen** und **netzorientierte Funktionen** unterteilt (Abb. 1.1-4). Zu den anwendungsorientierten Funktionen gehören außer der Anwendung selbst (z. B. Filetransfer, Message Handling, Data Retrieval usw.) die Darstellung der ausgetauschten Nachrichten (z. B. am Bildschirm, als Text oder Dokument usw.) und die Abwicklung der Sitzung (z. B. zwischen dem Endgerät und dem Datenbankserver, zwischen dem *Client* und dem *Server*). ZU den netzortientierten Funktionen gehören die Bitübertragung über die Teilstrecken (Kabel), die Sicherung der Bitübertragung auf den Teilstrecken, die Wegesuche und der Verbindungsaufbau im Netz, der Ende-zu-Ende Transport von Nachrichten im Netz usw.

anwendungsorientierte und netzorientierte Funktionen

Abb. 1.1-4: Aufteilung der Kommunikationsaufgaben in anwendungsorientierte und netzorientierte Funktionen

1.2 Ziele beim Entwurf offener Systeme

Eine Detaillierung und Klassifizierung der Aufgaben eines Kommunikationssystems unter verschiedenen Gesichtspunkten führt zu Systemarchitekturen, die sich in den Implementierungen der Produkte einzelner Hersteller widerspiegeln. Typische Beispiele solcher Architekturen sind das SNA (*System Network Architecture*) von IBM und das DEC NET von Digital Equipment. Ein wesentlicher Aspekt bei der Realisierung von Produkten ist das Bestreben, gleichartige Kommunikationsaufgaben in einem Hardware- oder Softwaremodul zusammenzufassen. Solche **Module** können dann in unterschiedlichsten Kommunikationssystemen eingesetzt werden. Ein weiteres wichtiges Anliegen (vor allem der Anwender) ist die **Kompatibilität** zwischen verschiedenen Geräten und Modulen unterschiedlicher Hersteller, die letztendlich durch Einhaltung definierter Schnittstellen erreicht werden kann. Die Bildung von Schnittstellen zwischen Modulen ist stets mit Kosten verbunden. Häufig werden deshalb zwar Systeme aus logischen Modulen zusammengesetzt, die Schnittstellen jedoch nicht physikalisch zugänglich ausgebaut.

Module

Kompatibilität

Prinzipiell können die Funktionen eines Systems in gleichberechtigten Modulen zusammengefaßt werden, die miteinander interagieren. Dies führt zu einem recht komplexen System. In der Praxis hat sich bewährt, daß die Funktionen als Module (**Schicht** genannt) zusammengefaßt und hierarchisch (d. h. aufeinander aufbauend) gegliedert werden. Die beiden erwähnten Architekturen SNA und DEC NET nutzen eine solche hierarchische Schichten-Struktur.

Schicht

1978 begann eine Expertengruppe (ISO TC 97 SC 16 - International Standardisation Organisation, Technical Committee 97, Subcommittee 16) ein Schichten-Modell für die Strukturierung von Kommunikationssystemen zu erstellen. Bereits 1980 wurde ein Entwurf vorgestellt, der 1983 als ISO-Norm verabschiedet wurde. Das ISO-Modell, das aus sieben Schichten besteht, ist unter der Bezeichnung **OSI** (**O**pen **S**ystems **I**nterconnection) bekannt (Abb. 1.2-1).

OSI

Abb. 1.2-1: Struktur des OSI-Modells

Heute existieren bereits mehrere dieses Modell unterstützende Normen (Tabelle 1.2-1).

Tab. 1.2-1: *ISO - Normen zur Systemmodellierung*

ISO 7498	OSI Reference Model
ISO 8649/50	OSI Common Application
ISO 8517	OSI File Transfer
ISO 8831/32	OSI Job Transfer
ISO 9040/41	OSI Virtual Terminals
ISO 8822/23	OSI Presentation Service & Protocol
ISO 8824/25	OSI Transfer Syntax
ISO 8505/06	OSI Message Oriented Text
ISO 8613	Document Structure
ISO 646, 2022, 6937	Character Repertoire
ISO 8326/27	OSI Session Service & Protocol
ISO 8072/73	OSI Transport Service & Protocol
ISO 8602	OSI Connectionless Transport Service
ISO 8348	OSI Network Service
ISO 8473	OSI Connectionless Network Services
ISO 8878	X.25 Network Services
ISO 8808	X.25 Network Protocol
ISO 8880	LAN Network Services

Fortsetzung auf der naechsten Seite

Fortsetzung von der vorhergehenden Seite	
ISO 8886	OSI Data Link Services
ISO 7776	X.25 Link Layer
ISO 3309/ 4335/ 7809	High Level Data Link Control (HDLC)
ISO 8802/ 2	LAN Logical Link Control
ISO 8802/ 3, 4, 5, 6, 7,	LAN Media Access & Physical Layer

Durch die Verwendung des OSI-Modells besteht die Aussicht, daß Produkte verschiedener Hersteller zueinander kompatibel werden, d. h. freizügig miteinander kommunizieren können. Systeme, die nach dem OSI-Modell strukturiert sind, werden deshalb auch als "**offene Systeme**" (*Open Systems*) bezeichnet.

offene Systeme (*Open Systems*)

Bei der Zerlegung der Aufgaben eines Kommunikationssystems in logisch aufeinander aufbauende Schichten werden folgende Aspekte berücksichtigt:

- Gleiche Funktionen werden in einer Schicht zusammengefaßt.
- Jede Schicht hat nur direkte Interaktion mit den beiden benachbarten Schichten.
- Zwischen den einzelnen Schichten soll die Interaktion möglichst gering sein.
- Die von einer Schicht für die nächst höhere Schicht zu erbringenden Aufgaben werden in dieser Schicht gegebenenfalls unter Zuhilfenahme der jeweils niedrigeren Schicht realisiert.
- Kommunikationsmodelle sind logische Modelle. Sie legen keine Implementierungen fest, sondern beschreiben lediglich die jeweiligen Funktionen und deren logischen Zusammenhänge. Bei der Festlegung der einzelnen Funktionen in einer Schicht wird jedoch darauf geachtet, daß sie (hard- und softwaremäßig) jeweils einzeln implementiert werden können.

Das beschriebene Vorgehen impliziert, daß eine (hard- und softwaremäßige) Änderung in einer Schicht lediglich durch eine Modifizierung in dieser Schicht abgefangen wird - die anderen Schichten müssen nicht geändert werden. Das Vorgehen bei der Festlegung der einzelnen Aufgaben erwirkt zudem, daß jeweils höhere Schichten die logischen Funktionen des Systems auf einer jeweils höheren Abstraktionsebene darstellen.

1.3 OSI-Begriffe

In Abb. 1.3-1 sind zwei Endsysteme (z. B. die beiden PCs bei unserem Beispiel von Abb. 1.1-3) einschließlich einer Vermittlungsstelle, über die sie kommunizieren, dargestellt. Die beiden Systeme A und B sind entsprechend dem OSI-Modell jeweils in sieben Schichten strukturiert, die Vermittlungsstelle lediglich in drei Schichten, da sie nur die übertragungs- und vermittlungstechnischen Funktionen erbringt. Häufig wird zwischen den Funktionen, die die Vermittlung für das System A und für das System B erbringen, unterschieden, daher ergeben sich hier zwei Schichtenstapel. Durch den Begriff Schicht wird in der OSI-Nomenklatur einerseits die Schicht in

einem System bezeichnet, andererseits aber auch diese Schicht über alle Systeme hinweg - die jeweilige Bedeutung wird erst aus dem Kontext deutlich.

Abb. 1.3-1: Zwei Systeme A und B kommunizieren über eine Vermittlungseinrichtung (Strukturierung nach dem OSI-Modell)

Abb. 1.3-2: Meldungsaustausch in und zwischen den Systemen

In Abb. 1.3-1 ist angedeutet, daß die logischen Aufgaben in den verschiedenen Systemen jeweils in einer Schicht (also waagerecht im Bild) abgewickelt werden. So wird z. B. die Bitfehlersicherung auf der ersten Teilstrecke zwischen der Schicht 2 des Systems A und der Schicht 2 der Vermittlung durchgeführt. Abb. 1.3-2 zeigt

den tatsächlichen Meldungsaustausch, der im wesentlichen senkrecht zwischen den Schichten verläuft. So wird eine von der Schicht 7 des Systems A erzeugte Meldung für die Schicht 7 des Systems B zunächst die einzelnen Schichten des Systems A durchlaufen, über Kabel 1 die Vermittlung erreichen, hier die drei Schichten beider Stapel durchlaufen, über Kabel 2 System B erreichen und hier durch die einzelnen Schichten bis zur Schicht 7 hinaufgereicht.

In der OSI-Nomenklatur wird die N-te Schicht N-Schicht genannt, die Schicht darüber (N+1)-Schicht, die Schicht darunter (N-1)-Schicht. Die zwischen den benachbarten Schichten eines Systems ausgetauschten Meldungen werden

Primärmeldungen

Primärmeldungen (*Primitives*) genannt. Sie bestehen allgemein aus Nutzinformation und Steuerinformation. Primärmeldungen stellen Ereignisse (Aktionen) beim Kommunikationsablauf dar. Die Regeln für den logischen Meldungsaustausch (zeitliche Abwicklung einbezogen) zwischen zwei gleichen Schichten von

Protokoll

Systemen, die an der Kommunikation beteiligt sind, nennt man ein **Protokoll** (Abb. 1.3-3)

Abb. 1.3-3: Zu den Begriffen Schicht, Primärmeldung und Protokoll

Instanz

OSI-Dienste

Eine **Instanz** einer Schicht ist eine aktive Einheit dieser Schicht. Sie bietet der nächst höheren Schicht Kommunikationsfunktionen, **OSI-Dienste** (*Services*) genannt, an und/oder beteiligt sich an der Kommunikation mit einer anderen Instanz der gleichen Schicht eines anderen Systems über Protokolle. Eine Instanz kann auch lediglich eine Aufgabe innerhalb einer Schicht wahrnehmen, ohne einer höheren Schicht einen Dienst anzubieten (z. B. Fehler- oder Verwaltungsaufgaben erbringende Instanzen). In Implementierungen werden Instanzen mit hard- oder softwaremäßigen Modulen identifiziert, obwohl dies nicht zwingend ist.

Die Dienste, die eine Instanz der Schicht N (wir bezeichnen die Instanz als eine N-Instanz) einer (N+1)-Instanz anbietet, können Funktionen beinhalten, die die N-Instanz selbst mit Hilfe der nächst niedrigeren Schicht erbringt oder auch (über die Protokollabwicklung) mit Hilfe einer anderen N-Instanz der Schicht N.

Jede N-Schicht-Instanz hat einen über das gesamte Netz eindeutigen (N-Schicht) Namen. Eine (N+1)-Instanz kann einen Dienst, der von einer N-Instanz angeboten wird, über einen durch seine N-Adresse eindeutig gekennzeichneten Dienstzugangspunkt zwischen den beiden Schichten in Anspruch nehmen. Ein **Dienstzugangspunkt** kann jeweils nur von einer N-Instanz bedient werden. Eine N-Instanz kann mehrere N-Dienstzugangspunkte bedienen und eine (N+1)-Instanz mehrere N-Dienstzugangspunkte benutzen. Der Sachverhalt ist in Abb. 1.3-4 dargestellt. Sollen Nachrichten zwischen verschiedenen Instanzen der Schicht (N+1) ausgetauscht werden, so werden von der Schicht N logische Verknüpfungen zwischen den Dienstzugangspunkten (zur Schicht N), über die die Nachrichten ausgetauscht werden, erstellt. Besteht die Nachricht nur aus einzelnen Meldungen, die alle Adressierungs- und Sequenzierungsinformationen enthalten, so ist keine weiteren Kennzeichnung als die Verknüpfung der Dienstzugangspunkte erforderlich, und man spricht von **verbindungsloser Datenübermittlung**.

Dienstzugangspunkt

verbindungslose Datenübermittlung

Im anderen Falle wird eine (logische) Verbindung zwischen den Dienstzugangspunkten aufgebaut. Sie wird durch eine Zuordnung zwischen den **Verbindungsendpunkten**, die zu den jeweiligen Dienstzugangspunkten führen, identifiziert (Abb. 1.3-4). Verbindungen stellen somit eine logische Kommunikationsbeziehung zwischen den Verbindungsendpunkten, den Dienstzugangspunkten und letztlich, den diese Dienstzugangspunkte verwendenden Instanzen dar. Gewöhnlich werden Punkt-zu-Punkt Verbindungen verwendet, jedoch auch Mehrpunktverbindungen (z. B. für globale Mitteilungen) sind möglich. Zwischen zwei Dienstzugangspunkten können auch gleichzeitig mehrere Verbindungen bestehen.

Verbindungsendpunkte

N-I N-Instanz
N-DZP N-Dienstzugangspunkt

Abb. 1.3-4: Kennzeichnung der N-Instanzen durch N-Name und der N-Dienstzugangspunkte durch die N-Adresse

(N+1)-I (N+1)-Instanz
N-DZP N-Dienstzugangspunkt
N-VEP N-Verbindungspunkt

Abb. 1.3-5: N-Verbindungen zwischen N-Dienstzugangspunkten werden durch Verknüpfung zwischen eindeutig bezeichneten N-Verbindungsendpunkten gekennzeichnet.

verbindungsorientierte Datenübermittlung

Die **verbindungsorientierte Datenübermittlung** verläuft in drei Phasen:

1. Eine Verbindungsaufbauphase, in der die Verbindung erstellt wird
2. Eine Verbindungsphase, in der Nutzdaten übermittelt werden
3. Eine Verbindungsabbauphase, in der die Verbindung wieder abgebaut wird.

Beispiele der verbindungsorientierten Datenübermittlung sind: Durchschalteverbindungen (Leitungsvermittlung) und virtuelle Verbindungen bei der Datenpaketübermittlung. In Abb. 1.3-6 bis Abb. 1.3-8 ist der typische Ablauf einer solchen Verbindung für eine verbindungsorientierte Datenpaketübermittlung mit den dabei verwendeten Primärmeldungen dargestellt.

Connect Request	Verbindungswunsch
Connect Indicate	Anzeige des Verbindungswunsches
Connect Response	Annahme der Verbindung
Connect Confirm	Betätigung des Verbindungsaufbaus

Abb. 1.3-6: Typische Primärmeldungen bei verbindungsorientierter Schicht-N Kommunikation: Verbindungsaufbau

Data Request	Wunsch, Datenpaket zu übermitteln
Data Indicate	Anzeige des Datenpakets
Data Response	Annahme der Datenpackets
Data Confirm	Betätigung der Übermittlung des Datenpakets

Abb. 1.3-7: Typische Primärmeldungen bei verbindungsorientierter Schicht-N Kommunikation: Verbindungsphase. Der Datenaustausch wird mehrfach, ggf. auch verschachtelt durchgeführt.

Disconnect Request	Wunsch, Verbindung abbauen
Disconnect Indicate	Anzeige des Abbauwunsches
Disconnect Response	Annahme des Abbauwunsches
Disconnect Confirm	Betätigung des Verbindungsabbaus

Abb. 1.3-8: Typische Primärmeldungen bei verbindungsorientierter Schicht-N Kommunikation: Verbindungsabbau

verbindungslose Datenübermittlung

Ein wesentlicher Punkt bei der**verbindungslose Datenübermittlung** ist, daß die Zeitbedingungen gegenüber der verbindungsorientierten Datenübermittlung (insbesondere gegenüber der Durchschalteverbindung) gelockert werden. Die einzelnen Instanzen einer Schicht müssen nicht unmittelbar zur Verfügung stehen; lediglich eine Maximalzeit für den Übermittlungsvorgang sollte nicht überschritten werden. Obwohl eine zeitlich begrenzte Verbindung im Falle der verbindungslosen Datenübermittlung nicht existiert, wollen wir den Begriff "Route" auch für verbindungslose Datenübermittlung verwenden, um im folgenden einheitliche Formulierungen verwenden zu können. Als **Route** bezeichnen wir den Weg, den eine Meldung zwischen zwei Endsystemen durch das Kommunikationsnetz benutzt.

Route

Für die Adressierung im ISO-Modell gilt folgendes: Instanzen, Dienstzugangspunkte und Verbindungsendpunkte werden jeweils eindeutig gekennzeichnet (adressiert). Über die Zuordnung von N-Instanzen und den (N-1)-Dienstzugangspunkten, über die sie kommunizieren können, wird (in der N-Schicht) ein Verzeichnis geführt. Die N-Verbindungen (d. h. die Verknüpfungen zwischen den N-Verbindungsendpunkten) werden in der N-Schicht verwaltet. Ihr sind jeweils auch die Dienstzugangspunkte, die eine N-Instanz bedient und die (N-1)-Dienstzugangspunkte, deren Dienste sie hierfür in Anspruch nimmt, bekannt. Die jeweiligen Zuordnungen können einfach (eins zu eins oder hierarchisch) oder aber auch recht kompliziert sein. Wesentlich ist, daß die (N+1)-Schicht diese Zuordnungen nicht kennt und nicht zu verwalten braucht. Für die (N+1)-Schicht sind lediglich die Dienste, die ihr von der Schicht-N angeboten werden, relevant. Wie sie erbracht, d. h. auch implementiert werden, ist für sie irrelevant und bleibt ihr verborgen.

In Abb. 1.3-9 sind typische Formate von Meldungen, wie sie bei der Kommunikation zwischen zwei Systemen verwendet werden, dargestellt. Charakteristisch für einen solchen Meldungsaustausch ist, daß jede Schicht eine Meldung von der jeweils höheren Schicht übernimmt, Steuerinformationen der eigenen Schicht hinzufügt und die neue Meldung an die jeweils niedrigere Schicht weitergibt. Auf der Empfangsseite werden dann jeweils die Steuerinformationen von der Meldung abgenommen, verwertet und die Meldung an die jeweils höhere Schicht weiterge-

geben. Die Steuerinformationen einer Schicht werden somit nur in dieser Schicht verwendet. Sie sind für die jeweils höhere und niedrigere Schicht irrelevant.

Abb. 1.3-9: Typische Formate von Meldungen bei der Kommunikation zwischen zwei Systemen

Selbsttestaufgabe 1.3-1:

1. *Was versteht man unter einem Protokoll ?*
2. *Wieviele Schichten im OSI-Modell durchlaufen die Nachrichten bei einem Schicht-4 Protokoll zwischen zwei Endgeräten insgesamt, wobei die Kommunikation über 2 Vermittlungsstellen läuft ?*

1.4 Kommunikationstechnische Aufgaben

Die Aufgaben, die von Kommunikationssystemen wahrgenommen werden, sind recht umfangreich. In den folgenden Kapiteln werden wir uns die verschiedenen Netze im einzelnen ansehen. Hier wollen wir zunächst einen kurzen, zusammenfassenden Überblick über die wichtigsten Aufgaben geben.

Übertragungsmedium

Der Kommunikationsvorgang zwischen zwei Systemen wird stets über ein **Übertragungsmedium** abgewickelt. Man unterscheidet zwischen Freiraumübertragung (wie beim zellularen Mobilfunk oder Richtfunk) und leitergebundener Übertragung (wie über Kupferdoppeladern, Koax-Kabeln oder Glasfasern).

Schicht 1

Bitübertragungsschicht (*Physical Layer*)

Leistungscodes

Eine der Hauptaufgaben, die der **Schicht 1** des OSI-Modells zugeordnet wird, ist die Übertragung der meist als Binärfolge vorliegenden Nachrichten über die Teilstrecke zwischen zwei Systemen (z. B. Endgeräte und Vermittlungsanlage). Die Schicht 1 wird deshalb auch als die **Bitübertragungsschicht** bezeichnet (*Physical Layer*). Bei der digitalen Übertragung wird die vorliegende binäre Zahlenfolge in eine physikalische Impulsfolge umgewandelt, die sich auf der Teilstrecke ausbreitet. Hierzu werden binäre oder ternäre **Leitungscodes** (wie Binär RZ, Top Hat, AMI, MMS43 usw.) verwendet (s. Abb. 1.4-1 bis Abb. 1.4-4) .

Abb. 1.4-1: Bildungsgesetz einiger Binärcodes

Abb. 1.4-2: Beispiele für Binärcodes

AMI, Pseudo-Ternär		0 ⟶ 0 1 ⟶ +, - +, - alternierend
B6ZS	Substitution von 6 Nullen im AMI	.. | 000000 .. + | -0-+0+.. - | +0+-0-..
HDB3	Substitution von 2 Nullen im AMI	.. | 0000 + | -00- > Falls Polarität des letzten substituierten Bit < + - + | 00+ - | +00+ > Falls Polarität des letzten substituierten Bit < + - - | 000-
PST		00 ⟶ -+ 01 ⟶ 0+/0- > Moduswechsel nach Jeder 10 oder 01 10 ⟶ +0/-0 11 ⟶ +-

Abb. 1.4-3: Bildungsgesetz einiger Ternärcodes

Abb. 1.4-4: Beispiele für Ternärcodes

Abb. 1.4-5: Leistungsdichtespektren einiger Binärcodes
Die Amplitude der Binärsignale ist $A = \frac{1}{\sqrt{T}}$, mit T als Schrittdauer genormt.

Von den Leitungscodes wird meist gefordert :

- daß sie effizient sind, d. h. daß sie den verfügbaren Frequenzbereich optimal ausnutzen (Abb. 1.4-5).
- daß sie Bittransparenz aufweisen, d. h. daß sie jede Bitkombination ohne Einschränkung übertragen und in der richtigen Reihenfolge an den Empfänger weitergeben können,
- daß sie einen hohen Taktgehalt aufweisen, damit am Empfänger der richtige Abtastzeitpunkt leicht bestimmt werden kann und
- daß sie Gleichstromfreiheit aufweisen, damit das Übersprechen minimiert wird und ggf. eine Gleichstromversorgung über das gleiche Medium möglich ist.

Abb. 1.4-6: Bitübertragung auf einer Kupferdoppeladernstrecke

Die übertragene Impulsfolge wird auf der Strecke gedämpft, verzerrt und ggf. durch Störungen überlagert (Abb. 1.4-6). Sie wird am Empfänger (meist unter Verwendung adaptiver Verfahren) entzerrt bzw. rekonstruiert und abgetastet, um die Binärfolge möglichst richtig wiederzugeben. Häufig werden auch fehlerminimierende Schätzverfahren (wie z. B. Maximum Likelihood Verfahren oder Mean Square Estimation) verwendet.

In einigen Systemen werden asynchrone (Start-Stop) Verfahren angewandt, wir werden jedoch im folgenden lediglich die modernen synchronen, d. h. mit einer nominell festen Bitrate arbeitenden Bitübertragunsprotokolle betrachten.

Stromversorgung

Zu den Hauptaufgaben, die der Schicht 1 zugeordnet werden, gehört auch die **Stromversorgung**. Sie wird meist lokal, gelegentlich (wie z. B. im Fernsprechnetz) auch über die Anschlußleitung als Gleichstrom Fernspeisung ausgelegt. Wir werden das Prinzip der Fernspeisung im ISDN näher kennenlernen. Weitere Funktionen der Schicht 1 sind die Aktivierung und Deaktivierung der physikalischen Strecken, die Verwaltung der physikalischen Verbindungen (z. B. Zuordnung nach Güteparametern) und die Fehlerbehandlung und -verwaltung (z. B. die Resynchronisation oder die Notspeisung).

Schicht 2
Sicherungsschicht
(*Data Link Layer*)

Die **Schicht 2** wird als die **Sicherungsschicht** (*Data Link Layer*) bezeichnet. Sie verwendet die von der Schicht 1 angebotenen Dienste (im wesentlichen die Bitübertragungsstrecken), strukturiert sie und ermöglicht eine gesicherte Datenübertragung auf den Teilstrecken. Ferner ermöglicht sie auch eine Flußsteuerung auf der Teilstrecke.

Der synchrone Bitstrom wird entweder periodisch strukturiert (wie bei PCM-30 oder der S-Schnittstelle im ISDN) oder sporadisch verwendet (wie bei paketori-

entierten Verfahren nach X.25 oder SS Nr. 7). Bei der periodischen Strukturierung (**Multiplexbildung**) wird das Rahmensynchronisationswort (z. B. `0011011` bei PCM 30) verwendet, das periodisch gesendet wird. Hierdurch wird der Beginn des Rahmens identifizierbar und durch einfache Abzählung einzelne Kanäle bzw. Wörter wieder auffindbar (Abb. 1.4-8). Das Rahmenkennungswort wird häufig am Anfang des Rahmens komplett gesendet (wie z. B. bei PCM 30); es kann aber auch über den Rahmen verteilt gesendet werden (wie bei T1-Systemen in den USA). Das Synchronwort wird so gewählt, daß es im laufenden Bitstrom leicht auffindbar ist. Da das Synchronwort im gleichen Abstand stets wiederholt wird, stört es nicht, wenn es gelegentlich auch in dem zu übermittelnden Nutzdatenstrom vorkommt.

Multiplexbildung

Abb. 1.4-7: Periodische Bitstrukturierung (z. B. PCM 30)

Abb. 1.4-8: Sporadische Bitstrukturierung (z. B. X.25 unter Verwendung der HDLC-Prozedur)
Strukturierung des synchronen Bitstromes

Beispiel 1.4-1: Fehler bei der Synchronisation
Wir betrachten die Synchronisierung eines Rahmens der Länge 256 Bit mit dem 7-Bit-Rahmenkennungswort `0011011`.

Abb. 1.4-9:

Die Wahrscheinlichkeit für das Eintreten von "1" und die Wahrscheinlichkeit für das Eintreten von "0" im Nutzbitstrom seien gleich, beide 0,5. Die Wahrscheinlichkeit, daß das Rahmenkennungswort an einer bestimmten Stelle im Nutzdatenstrom zufällig auftritt, ist

$$P = (0,5)^7 \approx 7,8 \cdot 10^{-3}$$

Im Mittel tritt das Rahmenkennungswort zufällig annähernd

$$(256 - 7 - 6) \cdot 7,8 \cdot 10^{-3} \approx 1,9$$

mal in einem Rahmen auf. Die Wahrscheinlichkeit, daß das Rahmenkennungswort an zwei bestimmten Stellen im Nutzstrom zufällig auftritt, (z. B. an einer bestimmten Stelle im Nutzstrom und dann genau einen Rahmen später wieder) ist

$$(0,5)^{14} \approx 6,1 \cdot 10^{-5}$$

Im Mittel passiert dies

$$243 \cdot 6,1 \cdot 10^{-5} \approx 1,5 \cdot 10^{-2}$$

mal pro Rahmen, d. h. 1,5mal alle hundert Rahmen. In der Praxis sind die Wahrscheinlichkeiten wesentlich geringer, da Rahmenerkennungsworte so gewählt werden, daß sie bei Nutzdaten selten oder gar nicht vorkommen.

Paket, Flag

Wird der synchrone Bitstrom sporadisch verwendet, so wird der Beginn des Rahmens (**Paket** genannt) durch ein Synchronwort (**Flag**) wie z. B. `01111110` bei X.25) gekennzeichnet. Es muß nun sichergestellt werden, daß die für das Flag verwendete Bitfolge in dem Nutzbitstrom entsprechenden Teil des Gesamtbitstromes nicht vorkommt. Im nächsten Kapitel werden wir das zero-insertion-Verfahren kennenlernen, das dies gewährleistet und gleichzeitig die Bit- bzw. Worttransparenz für den Nutzbitstrom gewährt. Das Ende eines Pakets wird häufig auch mit dem gleichen Flag gekennzeichnet. Liegen keine Daten zur Übermittlung vor, wird ent-

weder wiederholt Flags oder eine Nullfolge gesendet. Häufig wird im Steuerfeld die Paketlänge angegeben.

Bei manchen Systemen werden kleine Pakete fester Länge (**Zellen**) verwendet. Häufig werden alle auf der Teilstrecke übertragenen Pakete durchnummeriert (**sequenziert**) und die Sequenznummer im Steuerfeld mitübertragen. Dies ermöglicht fehlerfrei empfangene Pakete zu **quittieren** oder fehlerhaft empfangene Pakete neue anzufordern. Um die Wartezeiten für die Quittierung nicht unverhältnismäßig zu vergrößern, werden häufig mehrere Pakete gleichzeitig quittiert bzw. es wird verabredet, mehrere unquittierte Pakete absenden zu dürfen. Man bezeichnet dies als Quittierungsfenster oder einfach **Fenster**. Der Empfänger hat häufig die Möglichkeit den Übertragungsfluß durch das Nichtquittieren von Meldungen zu unterbrechen. Man bezeichnet dies als **Flußkontrolle**. Wir werden in den folgenden Kapiteln sowohl Quittierungs- als auch Flußkontrollmechanismen an konkreten Beispielen kennenlernen.

Zellen

Sequenz

Quittierung

Fenster

Flußkontrolle

Gewöhnlich werden übertragene Wörter mit Paritätsbits gegen Bitverfälschung gesichert. Ganze Pakete werden durch Verwendung von linearen Codes gegen Bitfehler gesichert. Diese ermöglichen Bitfehler zu detektieren (die Pakete werden dann gewöhnlich wieder angefordert) oder aber auch Bitfehler zu korrigieren. Häufig werden lineare **zyklische Codes** (CRC-8, CRC-16 und CRC-32) eingesetzt. Sie sind recht einfach zu implementieren und bieten einen hohen Schutz vor allem gegen Bündelfehler.

zyklische Codes

Polynome für CRC-8, CRC-16 und CRC-32 sind:

CRC-8: $x^8 + x^4 + x^3 + x^2 + 1$

CRC-16: $x^{16} + x^{15} + x^2 + 1$

CRC-32: $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x^1 + 1$

Beispiel 1.4-2: Zyklische Fehlerkorrektur (Cyclic Redundancy Check, CRC)

Zur Sicherung eines zu übertragenden Blocks M mit einer Länge von k Bit bildet der Sender ein Kontrollwort F (Frame Check Sequence, FCS) mit einer Länge von n Bit, das vor dem „Ende-Flag“ eingefügt wird. Errechnet wird dieses Kontrollwort vor der Durchführung der Zero-Insertion aus dem gesamten Block bestehend aus Adress-, Steuer- und Informationsfeld.

Das Kontrollwort soll vom Sender so gewählt werden, daß der gesamte Block bestehend aus dem zu übertragenden Block M und dem angefügten Kontrollwort F durch eine vorgegebene Binärzahl exakt ohne Rest teilbar ist.

Der Empfänger dividiert den ankommenden Rahmen durch dieselbe Binärzahl. Behält er keinen Rest, kann er annehmen, daß der Rahmen fehlerfrei übertragen wurde. Zunächst wird das Produkt $2^n \cdot M$ gebildet und dadurch der Block M um n Stellen nach links geschoben und rechts mit n Nullen aufgefüllt. Der gesicherte

Block T soll nun aus dem ursprünglichen Block M und dem Kontrollwort F bestehen:

$$T = 2^n \cdot M + F$$

Das korrekte Kontrollwort F liegt vor, wenn der Block T exakt durch die Binärzahl G dividierbar ist. Um diese Bedingung erfüllen zu können, wird das Kontrollwort wie folgt errechnet:

$$\frac{2^n \cdot M}{G} = Q + \frac{R}{G}$$

Diese Division liefert einen Quotienten Q und einen Rest R. Letzterer wird als Kontrollwort F verwendet, also an den zu übertragenden Block M angehängt. Somit lautet der gesicherte Block T:

$$T = 2^n \cdot M + R$$

Der Empfänger überprüft die Richtigkeit dieses Blocks, indem er den gesamten Block T durch die ihm bekannte Binärzahl G dividiert. Ist dies exakt ohne Rest möglich, wurde der Block fehlerfrei übertragen.

$$\frac{T}{G} = \frac{2^n \cdot M + R}{G}$$

Die erneute Division des Terms $2^n \cdot M$ *durch G liefert wiederum den Quotienten Q und den Rest R:*

$$\frac{T}{G} = Q + \frac{R}{G} + \frac{R}{G}$$

Nach der modulo2-Arithmetik ergibt die Additon einer Binärzahl mit sich selbst Null.
→ *Der Term T ist exakt ohne Rest durch G dividierbar.*
→*Der Block T wurde fehlerfrei übertragen.*

Die Binärzahlen, mit deren Hilfe das Kontrollwort gebildet wird, werden auch als Generatorpolynom bezeichnet. Für zyklische Fehlerkorrekturen von Bitwörtern bestimmter Länge wurden jeweils Generatorpolynome genormt.

Zugriff zum Übertragungsmedium

Wie wir bei den lokalen Netzen sehen werden, wird auch der **Zugriff zum Übertragungsmedium**(z. B. durch CSMA- oder Tokenverfahren) der Schicht 2 zugeordnet.

Schicht 3 Vermittlungsschicht (Network Layer)

Die **Schicht 3** wird als die **Vermittlungsschicht** (*Network Layer*) bezeichnet. Zu ihren Aufgaben gehört es, über die gesicherten Teilstrecken, die von der Schicht 2 bereitgestellt werden, eine Verbindung aufzubauen oder Datenpakete auszutauschen. Je nach Vermittlungsart (Durchschaltevermittlung, Paketvermittlung mit vir-

tuellen Verbindungen, Datagramm-Vermittlung, Vermittlungsfunktionen bei lokalen Netzen) fallen unterschiedliche Aufgaben in der Schicht 3 an.

Durchschaltevermittlung

Bei der **Durchschaltevermittlung** wird die Wählinformation ausgewertet und die gewünschte Verbindung über Raum- und Zeitkoppelfelder aufgebaut. Die Wegesuche und die Leitweglenkung gehören zu den wichtigsten Aufgaben.

Paketvermittlung

Bei der **Paketvermittlung mit virtuellenVerbindungen** wird in der Verbindungsaufbauphase ein Weg durch das Kommunikationsnetz ausgesucht und vorbereitet. Alle zu der Verbindung gehörenden Pakete werden auf diesem Wege durch das Netz übermittelt. Die Reihenfolge der Pakete wird dabei erhalten. Zu den wichtigen Aufgaben gehören die Wegesuche, der Aufbau der virtuellen Verbindung, die Speicherung der Pakete an den Netzknoten, die Auswertung der (Schicht 3) Adressen, die Übermittlung der Pakete und der Abbau der Verbindung am Ende.

Datagramme

Bei **Datagrammen** können die einzelnen Pakete unterschiedliche Routen durch das Netz nehmen und sich überholen. Eine Sequenzierung beim Sender und Resequenzierung beim Empfänger ist deshalb erforderlich. Die anderen Aufgaben entsprechen denen bei der Paketvermittlung mit virtueller Verbindung. Dabei sind Datagramme eine Untermenge der Paketvermittlung.

lokale Netze

Wesentliche Vermittlungsaufgaben bei **lokalen Netzen** sind wiederum die Adressauswertung mit der entsprechenden Vermittlung der Pakete und die Routingfunktionen (insbesondere bei Netzen, die aus mehreren LANs bestehen).

Zur Schicht 3 gehören auch Optimierungsaufgaben wie Kostenminimierung, Verzögerungsminimierung, Überlastbehandlung und Blockierungsauflösung.

Schicht 4
Transportschicht
(*Transport Layer*)
Ende-zu Ende
Funktionen

Die **Schicht 4** wird als die **Transportschicht** (*Transport Layer*) bezeichnet. Sie übernimmt die Dienste der Schicht 3 (z. B. zeitmultiplex Verbindungen, virtuelle Verbindungen oder Datagrammübermittlung) und ergänzt diese durch weitere **Ende-zu Ende** (Peer-to-Peer) **Funktionen** .

Hierzu können Ende-zu-Ende Authentifikation, Verschlüsselung, Codierung, Multiplexbildung (Zusammenfassung mehrerer Datenströmer) oder Inversmultiplexbildung (Unterteilung eines Datenstromes), Sicherung sowie Flußkontrolle gehören.

Schicht 5
Kommunikationssteuerungsschicht
Sitzungsschicht
(*Session Layer*)

Die **Schicht 5** wird als die **Kommunikationssteuerungsschicht** oder auch **Sitzungsschicht** (*Session Layer*) bezeichnet. Ihre Aufgabe ist es, unter Verwendung von Diensten der Transportschicht den Betrieb und die Verwaltung von Sitzungen zwischen Anwenderinstanzen zu organisieren. Zur Durchführung von Sitzungen gehört die Dialogverwaltung (also z. B. die Verwaltung des Senderechtes oder die Abgrenzung von Aktivitäten innerhalb eines Dialoges), die Synchronisation des Dialoges (z. B. das Setzen von Synchronisationspunkten, die Rücksetzung des Dialoges auf einen Synchronisationspunkt beim Verlust der Synchronisation) und die Verwaltung von Sitzungen (nach Güteparametern, Prioritäten etc.).

Schicht 6
Darstellungsschicht
(*Presentation Layer*)

Die **Schicht 6** wird als die **Darstellungsschicht** (*Presentation Layer*) bezeichnet. Ihre Aufgabe ist es, für eine einheitliche Darstellung von Informationen der Anwen-

dungsinstanzen zu sorgen, um die Kommunikation zwischen verschiedenen Endsystemen während einer Sitzung zu ermöglichen.

Es ist durchaus möglich, daß die an der Kommunikation beteiligten Endsysteme unterschiedliche lokale Darstellungen verwenden, und die globale, im Netz verwendete Darstellung wiederum hiervon abweicht. Zu den Aufgaben der Schicht 6 gehört es in diesem Fall z. B. Vereinbarungen sowohl für lokale Darstellungen von Informationen für eine Sitzung (wie die Zeichencodierung oder die Darstellung auf dem Bildschirm) als auch für die globale, neutrale Darstellung (z. B. Codierung der Anwenderinformationen oder Codierung der Darstellungsinformationen) zu treffen. Sie überprüft ggf. die Einhaltung der lokalen Darstellungsvereinbarungen und übernimmt die Umsetzung zwischen lokaler und globaler Darstellung während der Sitzung.

Schicht 7
Anwendungsschicht
(*Application Layer*)

Die **Schicht 7** wird als die **Anwendungsschicht** (*Application Layer*) bezeichnet. Ihre Aufgabe ist die Wahrnehmung der kommunikationsrelevanten Aspekte des Anwendungsprozesses. Sie bildet die Quelle und die Senke für die Kommunikation. Ihre Funktionen können je nach Anwendung sehr verschieden sein. Einige häufig auftretende Beispiele sind:

- File Transfer
- Remote Job Entry
- Message Handling
- Directory Services
- Databank Access.

Es gibt viele Normen, in denen unterschiedliche Protokolle und Schnittstellen zur Realisierung der verschiedenen kommmunikationstechnischen Aufgaben festgelegt werden. In Abb. 1.4-10 sind einige der Normen im Sinne des OSI-Modells hierarchisch gegliedert dargestellt.

	ISO	CCITT		
Layer 7 (Application)	Remote Database Access (ISO 9579) Distributed Transaction Processing (ISO 10026) FTAM (ISO 8571) JTAM (ISO 8831/2) VT (ISO 9040) Common Application Service Elements (ISO 8649/50) Directory System / Access Protocol (ISO 9554) Common Management Information Protocol (ISO 9595/6)	Interpersonal Messaging (X.420) Message Handling (X.400) Teletex (TTX) Videotex (T.100) Facsimile (T.0/4/5) Directory Service (X.500/520)		
Layer 6 (Presentation)	ISO 8822/3/4/5	X.216/226	T.50/ 51/61	
Layer 5 (Session)	ISO 8326/7	X.215/225	T.62	
Layer 4 (Transport)	ISO 8072/3	X.214/224	T.70	
Layer 3 (Network)	ISO 8880/8473/9542/10589/8208/8881	X.213 X.25 Paketverm.	T.30	I.450 Verm.
Layer 2 (Data Link)	ISO 8802.2	X.212/222	T.71	I.440 Sicherung
Layer 1 (Physical)	ISO 802.3 CSMA/CD ISO 802.4 Token Ring ISO 802.5 Token Bus ISO 802.6 DQDB	X.21 / X.21 bis Leistungs-vermittelt	V.24 Modem	I.430 Bitübertragung
	LANs	Daten-netze	Fesp.-netze	ISDN

Abb. 1.4-10: Normen nach OSI-Modell gegliedert

Selbsttestaufgabe 1.4-1:

Gegeben ist eine zufällige Binärfolge $a_n = 1110\ 0001\ 0101$, *die am Eingang eines Leitungscodierers als NRZ-Signal vorliegt. Stellen Sie ausgehend von der Eingangsfolge die Ausgangsfolgen graphisch für die beiden Codierer dar:*

1. *AMI-Codierer*
2. *CMI-Codierer*

Selbsttestaufgabe 1.4-2:

Gegeben ist der zu übertragende Block M und das Generatorpolynom G.

$$M = x^9 + x^7 + x^3 + x^2 + 1 = \texttt{1010001101} \qquad 1.4\text{-}1$$
$$G = x^5 + x^4 + x^2 + 1 = \texttt{110101}$$

Berechnen sie das Kontrollwort FCS

1. in der modulo2-Arithmetik,

2. in der Polynomschreibweise

3. und stellen Sie die Bitfolge des zu übertragenden gesicherten Blocks T auf.

1.5 Einige Anmerkungen zum OSI-Modell

Das OSI-Modell wurde seinerzeit im Kontext mit der Paketvermittlung nach X.25 entworfen. Es berücksichtigt sowohl die Paketvermittlung nach virtuellen Verbindungen als auch Datagramme (also verbindungslose Datenübermittlung). Beiden ist gemeinsam, daß Steuerinformationen auf der selben Weise, nämlich im Steuerfeld der Nutzpakete oder der reinen Steuerpakete, übertragen werden. Bei der Durchschaltevermittlung werden häufig Steuerinformationen auf getrennten Signalisierkanälen (also Außerband) übertragen. Dies tritt beispielsweise im ISDN auf und führt dazu, daß das OSI-Modell einer entsprechenden Modifizierung bedarf. Man baut sowohl für die Nutzinformation als auch für die Steuerinformation die Schichtenstapel auf. Die Bitübertragungsschicht wird sowohl für den Nutzkanal (im ISDN für zwei Nutzkanäle) als auch für den Signalisierkanal gemeinsam verwendet. Die Schicht 2 wird für die beiden Kanäle getrennt ausgelegt (im ISDN für B-Kanal nicht genormt, für D-Kanal LAPD). Die Schicht 3 für den Nutzkanal wird im Steuerkanal realisiert (Abb. 1.5-1), während diese im Nutzkanal leer bleibt. Wie im Kapitel 4 dargelegt wird, kann das OSI-Modell für die Außenbandsignalisierung zwischen den Vermittlungsstellen fast vollständig übernommen werden, obwohl das hier verwendete Protokoll SS Nr. 7 gegenüber X.25 einige Unterschiede aufweist.

	B-Kanal	D-Kanal
Schicht 7	Anwendung	
Schicht 6	Nutzkanal	
Schicht 5	Nutzkanal	
Schicht 4	Nutzkanal	
Schicht 3	Leer	Nutz- und Steuerkanal
Schicht 2	Nutzkanal	Steuerkanal
Schicht 1	Gemeinsam für Nutz - und Steuerkanäle	

Abb. 1.5-1: Das Schichtenmodell bei Außerband Signalisierung wie im ISDN.

Das ursprünglich für Individualkommunikation (Punkt-zu-Punkt Verbindungen) ausgelegte OSI-Modell konnte leicht für Punkt-zu-Mehrpunkt Verbindungen ausgebaut werden, denn globale Mitteilungen für die Steuerung waren bereits vorgesehen.

Der wesentliche Nachteil des OSI-Modells ist der Aufwand, der für die Implementierung der einzelnen Schichten erforderlich ist. Deshalb wird ohnehin die Implementierung des OSI-Modells bis zur Schnittstellenebene kaum vorgenommen. Aber auch die Implementierung auf logischer Ebene verursacht erhebliche Aufwendungen.

Bei der Modellierung der lokalen Netzen durch IEEE ging man pragmatisch vor. Das dort verwendete Modell für Lokale Netze wurde von ISO übernommen und unterscheidet sich vom OSI-Modell erheblich. Wir werden es im Kapitel 3 näher kennenlernen.

Mit dem Aufkommen der schnellen paketvermittelten Netzen bzw. zellorientierten Netzen (wie Frame Relay und ATM) setzte sich ein neuer Gedanke durch.

Die Protokolle werden so strukturiert, daß alle zur Vermittlung erforderlichen Funktionen an einem Knoten in einer Schicht (der Schicht 2) zusammengefaßt werden. Somit hat man lediglich drei Schichten Tabelle 1.5-1:

- die Bitübertragungsschicht wie bisher
- die Vermittlungsschicht (z. B. ATM-Schicht) in der alle Übermittlungsfunktionen wahrgenommen werden und
- eine Adaptionsschicht, die alle Anwendungen an die Zellstruktur der Vermittlung adaptiert.

Wir werden ein solches Protokoll, das in ATM-Netzen angewendet wird, im Kapitel 5 näher kennenlernen. Obwohl das OSI-Modell durch diese Entwicklung überholt worden ist, kommt ihm für das Verständnis von Kommunikationsprotokollen eine erhebliche Bedeutung zu. Wir haben es deshalb an den Anfang unseres Kurses gestellt.

Tab. 1.5-1: *Schichten-Modell Zellorientierte Vermittlung (z. B. ATM)*

Anwendungen
Schicht 3 : Adaption der Anwendungen auf Zellstruktur
Schicht 2 : Zellvermittlung
Schicht 1 : Bitübertragung

1.6 Spezifikation und Testen von Protokollen

Wie wir gesehen haben, können Kommunikationsprotokolle recht komplex sein. Das Schichtenmodell ermöglicht deren modulare Betrachtung. Jede Schicht eines Systems bietet der nächst höheren Schicht gewisse Dienste an und wickelt mit der Partnerinstanz ein Protokoll ab. Aus einer genauen Dienstespezifikation können häufig Protokollspezifikationen semiautomatisch erzeugt werden. Protokollspezifikationen werden teilweise formal verifiziert. Aus ihnen werden meist automatisch effiziente Implementierungen erstellt. Diese werden wiederum getestet und teilweise formal validiert. Implementierungen von Standardprotokollen werden nach

ihrer Konformität zur Standardprotokollspezifikation getestet. Sie werden immer häufiger von berechtigten Instituten zertifiziert. Mittlerweile haben sich eine Reihe von formalen und semiformalen Methoden und entsprechende Werkzeuge für Protokolle entwickelt. Zunächst kommen wir zu den verwendeten Begriffen.

Protokollverifikation

Protokollverifikation ist die Überprüfung der Richtigkeit eines Protokolls an sich. Hauptbestandteil der Protokollverifikation ist der Nachweis der Korrektheit, der Vollständigkeit und der Konsistenz des Protokolls.

Korrektheit

Die **Korrektheit** des Protokolls bedeutet seine funktionelle Korrektheit, d. h. die Erfüllung der Dienstespezifikation durch die Protokollaktionen und ihre zeitrichtige Erbringung.

Vollständigkeit

Die **Vollständigkeit** eines Protokolls bedeutet, daß es alle möglichen Eingaben akzeptiert, bzw. auf alle möglichen Ereignisse reagiert. Es enthält also keine nichtspezifizierten Eingänge.

Konsistenz

Die **Konsistenz** eines Protokolls bedeutet, daß es widerspruchsfrei ist. Die Widerspruchsfreiheit bezieht sich sowohl auf innere Widersprüche (d. h. zwischen seinen verschiedenen Teilen) als auch solche zu den angrenzenden Schichten.

Zur Protokollverifikation gehört die Überprüfung weiterer Protokolleigenschaften. Hierzu zählen unter anderem:

- **Ausführbarkeit**, d. h. das Protokoll enthält keine Aktionen, die nie zur Ausführung kommen.
- **Erreichbarkeit** oder **Lebendigkeit**, d. h. jeder Zustand des Systems ist vom Ausgangszustand erreichbar.

Deadlock Freiheit

- **Deadlock-Freiheit**, d. h. das Protokoll gelangt niemals in einen Systemzustand, der keine Wechselwirkung mit der Umgebung gestattet (Tote Schleife oder Sackgasse).

Livelock Freiheit

- **Livelock-Freiheit**, d. h. das Protokoll enthält keine unproduktiven Zyklen (z. B. ungewollte Wiederholungen).

Fehlertoleranz
Resynchronisation

- **Fehlertoleranz** und **Resynchronisation**, d. h. das Protokoll kehrt nach einem Fehler oder einem abnormalen Zustand in einen geregelten Zustand zurück.

Sicherheit

- **Sicherheit**, d. h. das Protokoll erreicht niemals einen unvorgesehenen Zustand.

Termination

- **Termination**, d. h. das Protokolle erreicht immer einen Endzustand (bei zyklischen Protokollen den Anfangszustand).

Protokollvalidation

Häufig wird der Begriff Protokollvalidation mit dem Begriff Protokollverifikation verwechselt. **Protokollvalidation** ist der Nachweis der Richtigkeit des Protokolls bezüglich der Nutzanforderungen.

Konformität

Es sei darauf hingewiesen, daß man stets zwischen der Protokollspezifikation und der Protokollimplementierung unterscheidet. Die definierten Begriffe sind auf beide anwendbar. Man testet (vor allem bei Standardprotokollen) die Implementierung auf die **Konformität** mit der Protokollspezifikation oder mit einer anerkannten (*Master*) Realisierung.

Häufig werden Protokolle modelliert oder unter Belastung getestet, um ihre **Leistungsfähigkeit** (Durchsatz, Wartezeit usw.) zu bewerten.

Leistungsfähigkeit

Um Protokolle zu spezifizieren und zu verifizieren werden verschiedene Verfahren angewandt, die auf unterschiedlichen Ansätzen beruhen. Hierzu gehören:

- erweiterte, endliche Zustandsautomaten
- Petrinetze
- höhere Programmiersprachen
- algebraische Methoden und
- deskriptive Methoden.

Wir wollen diese Ansätze kurz betrachten.

Ein **erweiterter endlicher Zustandsautomat** (*Extended Finite State Machine*) M ist ein 5-Tupel

erweiterter endlicher Zustandsautomat (*Extended Finite State Machine*)

M = (I, O, S, T, F) wobei

- I (**I***nput*) die Menge der Protokolleingänge darstellt,
- O (**O***utput*) die Mengen der Protokollausgänge darstellt,
- S (**S***tates*) die Zustände des Automaten darstellt,
- T (**T***ransition*) die Zustandsübergänge darstellt, die durch die Abbildung $T : I \cdot S \rightarrow S$ definiert sind und
- F (**O***utput***F***unction*) die Ausgangsfunktionen darstellt, die durch die Abbildung $F : I \cdot S \rightarrow O$ definiert sind.

Aus dieser Definition des erweiterten endlichen Zustandsautomaten werden Zustandstabellen oder Zustandsdiagramme für Protokolle aufgestellt. Diese am häufigsten verwendete Methode ermöglicht ein konstruktives Vorgehen bei der Spezifikation und der Verifikation von Protokollen.

Beispiel 1.6-1: Alternating Bit Protocol
Wir betrachten folgendes vereinfachtes System:

Abb. 1.6-1:

Ein Nutzer A sendet Daten zum Nutzer B. Der Sender markiert die gesendeten Meldungen alternierend mit 0 und 1 (Data 0 bzw. Data 1). Entsprechend werden die Quittungen mit 0 und 1 gekennzeichnet (`ACK 0` bzw. `ACK 1`). Nach jeder gesendeten Meldung wird auf die Quittung gewartet. Gewöhnlich wird nach einer bestimmten Wartezeit eine nicht quittierte Meldung wiederholt. Der Empfänger merkt an dem alternierenden Bit, ob es sich um die Wiederholung oder um die nächste Meldung handelt.

Wir betrachten nun den Sender als Zustandsautomaten.

Es sind drei Eingänge

- `Send_req` vom Nutzer,
- `ACK 0` vom Empfänger und
- `ACK 1` vom Empfänger

und drei Ausgänge

- `Send_cnf` zum Nutzer,
- `Data 0` zum Empfänger und
- `Data 1` zum Empfänger

möglich. Der Sender kann sich in folgenden Zuständen befinden:

Bereit :	`Send_req` anzunehmen nach `ACK 0`
Bereit :	`Send_req` anzunehmen nach `ACK 1`
Nicht bereit :	warten auf `ACK 0`
Nicht bereit :	warten auf `ACK 1`

Wir wollen diese Zustände als `Bereit 1`, `Bereit 2` und `Warten 0`, `Warten 1` bezeichnen.

Für das Beispiel erhalten wir die im Bild dargestellte Zustandstabelle. Die Eintragungen in dem mittleren Teil der Tabelle geben jeweils den Ausgang und den

neuen Zustand an. Die Striche zeigen, daß das Protokoll nicht vollständig definiert ist. Das entsprechende Zustandsdiagramm ist im unteren Bild abgebildet.

Eingang \ Zustand	**Bereit 1**	**Warten 0**	**Bereit 2**	**Warten 1**
Send_req	Data 0 Warten 0	- - -	Data 1 Warten 1	- - -
ACK 0	- - -	Send_cnf Bereit 2	- - -	Data 1 Warten 1
ACK 1	- - -	Data 0 Warten 0	- - -	Send_cnf Bereit 1

Petrinetze arbeiten transitionsorientiert. Sie erlauben Interaktionen zwischen parallelen Prozessen auf einem hohen Abstraktionsniveau darzustellen. Da sie auf Rechnern abgebildet werden können, eignen sie sich gut, Protokollabläufe zu studieren. Sie eignen sich besonders zum Analysieren von Flußabläufen. Sie unterliegen denselben Einschränkungen wie Zustandsautomaten, nämlich daß sie schnell sehr komplex und somit nicht handhabbar werden.

Petrinetze

Petrinetze bestehen aus einer Anzahl von Stellen (die durch Kreise dargestellt werden) und einer Anzahl von Transitionen (die durch Striche dargestellt werden). Die

Stellen und Transitionen werden durch gestrichelte Kanten zu einem Graph verbunden. Die Kanten, die von einer Stelle ausgehen, enden immer an einer Transition und umgekehrt. An jeder Stelle (d. h. in jedem Kreis) können sich ein oder mehrere Marken befinden. Eine Marke an einer Stelle bedeutet, daß die Bedingung für die Transition, die mit der Stelle geknüpft wird, erfüllt ist. Sind alle Stellen die zu einer Transition gehören mit mindestens einer Marke belegt, so kann die Transition durchgeführt werden - man sagt die Transition kann zünden. Bei der Zündung einer Transition wird jeder Stelle, von der eine Kante zu der Transition führt, eine Marke entnommen und jeder Stelle, zu der eine Kante von der Transition ausgeht, eine Marke hinzugefügt.

Abb. 1.6-2: Beispiel eines Petrinetzes: Zünden einer Transition

Beispiel 1.6-2: Quittierung mit Fenstermechanismus

Animation 1.6-3: Fenstermechanismus

Das obige Bild stellt die im letzten Abschnitt (Schicht 2) kurz besprochene Quittierung mit Fenstermechanismus dar. Der Sender darf maximal drei unquittierte Meldungen absenden. Der Sender wird durch eine Stelle, in der pro Nachricht die zum Senden vorliegt eine Marke eingelegt wird und eine Stelle, zu der für

jede quittierte Nachricht eine Marke ankommt, modelliert. Der Empfänger wird durch eine Stelle, in die für jede Quittung eine Marke eingelegt wird und eine Stelle, an der jede ankommende Nachricht durch eine ankommende Marke dargestellt wird, modelliert. Zwei Transitionen `N` (Nachricht wird gesendet) und `Q` (Quittung wird gesendet) und zwei Stellen `O` (Offenes Fenster) und `M` (Empfangene Meldungen) modellieren das Protokoll. Am Anfang der Übertragung werden drei Marken in `O` gelegt. Dies bedeutet, daß das Fenster ganz offen ist; es dürfen maximal drei Nachrichten abgesendet werden, ohne daß quittiert werden muß. Liegt eine Nachricht vor, so wird eine Marke in die Nachrichtenstelle im Sender gelegt. Die Transition `N` kann nun zünden. In `O` bleiben 2 Marken übrig (es können noch zwei Nachrichten ohne Quittung abgesendet werden), in `M` befindet sich nun eine Marke (eine Nachricht wurde empfangen). Sendet der Empfänger eine Quittung und liegt in M eine Marke, so kann `Q` zünden, eine Marke wird `M` (und dem Empfänger) entnommen und eine Marke (Quittung) wird dem Sender gegeben. Ein solches Modell läßt sich leicht auf einem Rechner implementieren und erlaubt verschiedene Abläufe durchzuspielen.

Selbsttestaufgabe 1.6-1:

Stellen Sie das Petrinetz aus Beispiel 1.6-2 in einem Zustandsdiagramm dar.

Höhere Programmiersprachen wie Pascal, C und Chill werden häufig verwendet, um Protokolle zu spezifizieren. Sie liefern unmittelbar auch eine Implementierung des Protokolls. Wir wollen diesen Ansatz nicht weiter verfolgen, um den Rahmen des Kurses nicht zu sprengen.

höhere Programmiersprachen

Algebraische Methoden basieren auf algebraischen Strukturen und deren abstrakten Darstellungen. Die für Kommunikationsprotokolle verwendeten Methoden orientieren sich an Algebren zur Beschreibung kommunizierender paralleler Prozesse. Die bekannteste ist *Calculus of Communicating Systems* - CCC von Milner, die die Grundlage für die ISO-Sprache LOTOS bildet. Auch die von CCITT entwickelte Abstract Syntax Notation ASN.1 und ACT.1 basieren auf der algebraischen Theorie der abstrakten Datentypen. Algebraische Verfahren werden insbesondere zur Modellierung der Darstellungsschicht angewandt. Wir wollen diese Verfahren hier nicht weiter vorstellen, sondern lediglich auf die Literatur verweisen.

algerbaische Methoden

Zu den **deskriptiven Methoden** zählen Methoden, die auf Theorien der mathematischen Logik basieren. Sie erlauben eine exakte und unmittelbare Verifizierbarkeit der Spezifikationen. Temporale Logik, Intervall-Logik und Prädikationslogik sind Beispiele solcher Verfahren. Sie sind alle recht aufwendig in der Anwendung und werden daher wenig eingesetzt.

deskriptive Methoden

Drei Spezifikationssprachen, die genormt worden sind und in denen verschiedene Ansätze integriert wurden, sind:

Estell (Von ESTL - Extended State Transition Language) wurde von einer ISO

Estell

Untergruppe entwickelt und genormt (ISO 9074), Sie basiert auf dem erweiterten Zustandsautomatenmodell.

LOTOS Auch **LOTOS** (***L***anguage f***o***r ***T***emporal ***O***rdering ***S***pecifications) wurde von einer ISO Untergruppe entwickelt und genormt (ISO 8807) und basiert auf algebraischen Methoden.

SDL **SDL** (***S***pecification and ***D***escription ***L***anguage) basiert wie Estell auf dem erweiterten Zustandsautomatenmodell. Sie wurde ursprünglich von CCITT für die Spezifikation von Vermittlungssoftware entwickelt (CCITT Rec. Z 100) und wird nun auch für OSI-Systeme verwendet. Da viele vermittlungstechnische Protokolle in SDL spezifiert wurden, wollen wir uns die Grundnotationen von SDL kurz ansehen.

SDL basiert auf vier Grundkonzepten:

- Ein System wird hierarchisch durch folgende Elemente beschrieben: System, Blöcke, Prozesse und Prozeduren.
- Das Systemverhalten wird durch einen erweiterten endlichen Zustandsautomaten beschrieben.
- Daten werden unter Verwendung von abstrakten Datenstrukturen und variablen Definitionen angegeben.
- Kommunikation wird asynchron über Kanäle, die als unendliche Warteschlangen modelliert werden, abgewickelt.

Die Basiselemente von SDL sind in Abb. 1.6-4 als grafische Symbole dargestellt.

Abb. 1.6-4: Basiselemente in SDL

Die Systemebene bildet den Rahmen der Beschreibung. Sie besteht aus einem oder mehreren Blöcken, die jeweils ein Subsystem beschreiben. Blöcke sind mit Kanälen miteinander verbunden. Ein Block besteht aus mehreren Prozessen, die miteinander kommunizieren. Ein Prozeß ist ein erweiterter endlicher Zustandsautomat. Eine Prozedur ist wiederum ein erweiterter Zustandsautomat innerhalb eines Prozesses.

Durch den Konstrukt `CREATE` können beliebige Exemplare eines Prozesses erzeugt werden, die nebeneinander existieren. Der Startpunkt einer Prozeßausführung wird durch `START` gekennzeichnet und durch `STOP` bzw. `RUHE` beendet.

Wir wollen beispielhaft die SDL-Spezifikation eines einfachen Fernsprechapparates ansehen:

Beispiel 1.6-3: SDL-Spezifikation eines einfachen Fernsprechapparates
Ein einfaches Beispiel eines SDL-Prozesses ist in der folgenden Abbildung zu sehen. Es handelt sich um eine Vermittlungsstelle, die ein Telefon bedient. Nach dem Start des Prozesses befindet sich das System im Zustand `On_Hook`, d. h. der Hörer liegt auf der Gabel. Durch das Signal Lift, das erzeugt wird, sobald der Hörer abgenommen wird, wird ein Zustandswechsel in den Zustand `Off_Hook` durchgeführt. Beim Wechseln des Zustandes werden einige Variablen initialisiert, außerdem wird ein „Tone-On“ an den Anschluß versandt, um den üblichen Bereitschaftston zu erzeugen.

Da es sich um einen einfachen Prozeß handelt, wird lediglich eine zweistellige Nummer erwartet, die durch zwei Signale vom Typ Digit (Number) vom Teilnehmer versandt werden. Alternativ kehrt der Prozeß beim Auflegen des Hörers wieder in den Zustand `On_Hook` zurück.

Sobald beide Ziffern gewählt sind, wird ein Signal `Number_Dialed` (Dialed_Number) an eine weitere Vermittlungsstelle gesandt, um das eigentliche Telefonat im Zustand `Wait_For_Answer` durchzuführen. Durch das Signal `Hang_Up` wird wieder der Ausgangszustand erreicht.

Abb. 1.6-5: Spezifikation in SDL (Teil 1)

Abb. 1.6-6: Spezifikation in SDL (Teil 2)

2 Die X.21 und X.25 Protokolle

2.1 Die X.21 - Empfehlung

X.21 - Empfehlung

Die **X.21 - Empfehlung**, die von CCITT bereits 1976 verabschiedet wurde, beschreibt, wie eine Datenendeinrichtung (DEE) des Benutzers über die Datenübertragungseinrichtung (DÜE) des Netzbetreibers eine synchrone Datenverbindung unter Verwendung der Zeichengabe auf- und abbaut. Die Verbindung führt über das Netz des Betreibers zum B-Teilnehmer. Die in der Empfehlung beschriebenen prozeduralen Abläufe werden durch zahlreiche Normen wie X.26, X.27 (elektrische Eigenschaften), X.20, X.21/bis (asynchrone Verbindung, Anpassung an Modemübertragung) usw. ergänzt. Da somit sowohl logische als auch physikalische Signale und Abläufe an dem Referenzpunkt (zwischen der DEE und DÜE) spezifiziert werden, spricht man auch von der X.21-Schnittstelle. In Abb. 2.1-1 sind die verwendeten Signale dargestellt.

T	Transmit	*B*	Byte Synchronisation
C	Control	*G*	Ground
R	Receive	*Ga*	Ground (DTE common return)
I	Indicate	*DEE*	Dateneinrichtung
S	Synchronisation	DÜE	Datenübertragungseinrichtung

Abb. 2.1-1: Signale an der X.21 Schnittstelle

Man unterscheidet bei der verbindungsorientierten Datenübermittlung wie X.21 drei aufeinanderfolgende Phasen: die Verbindungsaufbauphase, die Verbindungsphase (Datenübertragungsphase) und die Verbindungsabbauphase. In der jeweiligen Phase kann die Schnittstelle verschiedene Zustände annehmen. In der Empfehlung sind alle Zustände (ca. 25), Zustandsübergänge und Testzustände und -abläufe beschrieben. In Abb. 2.1-2 ist ein typischer Ablauf in neun wesentlichen Schritten skizziert:

Zustand	DEE sendet T	DEE sendet C	DÜE sendet R	DÜE sendet I
1. DEE, DÜE unbelegt, bereit	1	Aus	1	Aus
2. Verbindungsanforderung	0	Ein		
3. Wahlauffordung			Syn Syn + … + Syn Syn	
4. Wählzeichen	Syn Syn IA5 Syn Syn … IA5 Syn Syn			
5. Verbindung im Aufbau	1	Ein	1	Aus
6. B-Tln empfangsbereit			1	Ein
7. Datenübertragung	Daten	Ein	Daten	Ein
8. Auslöseanforderung	0	Aus		
9. Auslösebestätigung			0	Aus
1. DEE, DÜE unbelegt, bereit	1	Aus	1	Aus

Abb. 2.1-2: Ein typischer X.21 Verbindungsablauf

Im Ruhezustand (Zustand 1, Abb. 2.1-2) sind alle vier Signale T, C, R und I auf Dauer-1 gesetzt. Wir folgen der üblichen Bezeichnung und nennen Dauer-1 (bzw. Dauer-0) bei C und I „Aus“ (bzw. „Ein“). Die Verbindungsanforderung (Zustand 2) wird von der DEE durch Setzen von T auf 0 und Einschalten von C angezeigt. Die DÜE antwortet innerhalb von 3 Sekunden mit der Wahlaufforderung (Zustand 3) durch R. Diese besteht aus einer Folge von +-Zeichen des **I**nternationalen **A**lphabets **IA Nr. 5** (Zeichen 2/11, CCITT/T.50, KT I, Bild 6.2) unbestimmter Länge, eingebettet in mindestens je zwei Synchronisationszeichen des IA Nr. 5 (Zeichen 1/6, CCITT/T.50). Die DEE muß ihrerseits nun innerhalb von 5 Sekunden mit dem Senden der Wählzeichen (bestehend aus IA Nr. 5 Zeichen, getrennt durch mindestens zwei Syn-Zeichen) beginnen (Zustand 4). Es sei hier darauf hingewiesen, daß sich die im Beispiel betrachteten Schritte überlappen oder es auch Wartezeiten entstehen können. Es ergeben sich entsprechend weitere Zustände, die hier nicht betrachtet werden sollen. Im nächsten Zustand wird der Verbindungsaufbau durch

IA Nr. 5

den Wartezustand der DEE (T=1, C=Ein) und den Bereitzustand der DÜE (R=1, I=Aus) angezeigt (Zustand 5); die Empfangsbereitschaft des B-Teilnehmers wird als nächstes durch das Einschalten von I durch die DÜE angezeigt (Zustand 6). Nun beginnt die Datenübertragungsphase (gekennzeichnet durch C=EIN, I=EIN). In dieser Phase (Zustand 7) stehen T und R für die Übertragung von Nutzdaten transparent zur Verfügung. Die Verbindungsabbauphase wird in unserem Beispiel durch das Setzen von T=0, C=Aus von der DEE eingeleitet (Zustand 8) und durch das Setzen von R=0 und I=Aus von der DÜE bestätigt (Zustand 9). Anschließend kehren sowohl die DEE als auch die DÜE in den Ruhezustand zurück (Zustand 1). Zur Einhaltung von zeitlichen Anforderungen an das Protokoll werden mehrere *Timer* (Zeitüberwachungen) eingesetzt.

Das X.21 Protokoll wurde bereits vor der Entwicklung des OSI-Modells spezifiziert und regelt, wie wir gesehen haben, den kompletten verbindungsorientierten synchronen Datenaustausch. Es wird jedoch auch häufig lediglich als Schicht 1 Protokoll, d. h. für die Bitübertragung auf einem physikalischen Medium zwischen zwei benachbarten Systemen eingesetzt.

Beispiel 2.1-1: Signale bei X.21

Wir betrachten zwei Datenendeinrichtungen, die nach dem X.21 - Protokoll Daten austauschen. Die Signale T, C, R, I werden dann, wie im folgendem Bild skizziert, umgesetzt:

Abb. 2.1-3: Signale bei X.21

Selbsttestaufgabe 2.1-1:

1. *Nennen Sie die Unterschiede zwischen verbindungsorientierter und verbindungsloser Datenübertragung.*
2. *Welchen Verfahren lassen sich X.25 und X.21 zuordnen?*
3. *In welche Phasen wird die Datenübertragung nach X.21 aufgeteilt?*
4. *Ordnen Sie die Zustände der Tabelle 2.1 diesen Phasen zu.*

2.2 Transparenz, Sicherung und Quittierung

In diesem Abschnitt wollen wir einige Grundprozeduren als Einführung für die in späteren Abschnitten behandelten Protokolle kennenlernen. Analog zur Transparenz eines Codes (s. KT II, Abschnitt 8.1) sprechen wir von der **Transparenz eines Netzes**, wenn das Netz es erlaubt, jede beliebige Kombination von Bits, Symbolen oder Wörtern zu übertragen. Man spricht dann auch von Bit-, Symbol- oder Wort-Transparenz. Bei der Multiplexbildung werden häufig Rahmenkennungsworte verwendet, um eine Wort- und Rahmensynchronisation abzuleiten (s. KT II, Abschnitt 9.2 und 9.3). Das Synchronwort tritt hierbei im laufenden Bitstrom immer wieder periodisch auf. Durch das Mehrfachsuchen des Synchronwortes wird die Wahrscheinlichkeit einer Verwechslung mit Nutzdaten herabgesetzt (s. KT II, Beispiel 9.2). Es wird dadurch möglich, die Zeitschlitze mit den Nutzdaten jeweils wiederzufinden, und diese können als transparente Kanäle genutzt werden.

Transparenz eines Netzes

Hat man sporadisch auftretende Daten, so verwendet man auch oft ein Rahmenkennungswort **(Flag)**, um den Beginn einer Meldung zu kennzeichnen. Manchmal werden Flags auch zum Auffüllen eines Kanals verwendet, wenn keine Daten zur Übertragung vorliegen. In vielen Datenübertragungsprotokollen wird das Flag bestehend aus der binären Zeichenfolge `01111110` verwendet. Um nun die Transparenz zu gewährleisten, wird das als ***Zero Insertion*** bekannte Verfahren eingesetzt. Stets, wenn in dem zu übertragenden Datenstrom fünf Einsen hintereinander auftreten, fügt der Sender eine Null hinzu. Hierdurch wird vermieden, daß das Flag im übertragenen Nutzdatenstrom auftritt. Der Empfänger nimmt stets eine Null, die nach fünf Einsen auftritt, wieder heraus.

Flag

Zero Insertion

Beispiel 2.2-1: : Zero Insertion

Abb. 2.2-1: Zero Insertion

Bei der Kanalcodierung (s. KT II, Kapitel 7) lernten wir verschiedene Möglichkeiten, eine Fehlererkennung oder eine Fehlerkorrektur durchzuführen, kennen. Unser Bestreben dort war es, die Wahrscheinlichkeit möglichst herabzudrücken, daß Fehler nicht erkannt oder nicht korrigiert werden. Im Einzelfall kann es jedoch vorkommen, daß die Sicherungsmaßnahmen nicht greifen und Fehler unerkannt bleiben, so z. B. wenn bei linearen Codes durch Fehler ein Codewort in ein anderes Codewort umgewandelt wird. Es ist in der Praxis durchaus üblich, weitere, die Kanalcodierung

ergänzende Sicherungsmaßnahmen einzusetzen, um die Fehlerwahrscheinlichkeit weiter zu verringern.

Plausibilitätsprüfung

Die **Plausibilitätsprüfung** ist eine solche Maßnahme, die in Kommunikationsanlagen häufig angewandt wird. Bei vermittlungstechnischen Abläufen wird z. B. überprüft, ob eine Meldung, die von einem Teilnehmer kommt, mit dem Zustand, in dem sich der Teilnehmer befindet, kompatibel ist. Falls dies nicht der Fall ist, wird entweder die Meldung ignoriert oder eine Fehlerprozedur eingeleitet. Solche Plausibilitätstabellen, die alle zulässigen Meldungen für einzelne Teilnehmerzustände angeben, werden in Vermittlungsanlagen häufig verwendet.

Quittierung

positive Quittierung

time out

negative Quittierung

positiv-negativ Quittierung

Die **Quittierung** (**ARQ** - ***a****utomatic* ***r****epeat re****q****uest*) ist eine weitere Maßnahme, die bei Datenübermittlungsprotokollen angewendet wird. Man unterscheidet zwischen drei Quittierungsarten. Bei der **positiven Quittierung** bestätigt der Empfänger eine richtig empfangene Nachricht, indem er eine Meldung (die Quittung **ACK** - ***acknowledge***) an den Sender schickt. Hierbei wird angenommen, daß die Nachricht richtig empfangen wird, wenn die Sicherungsverfahren (z. B. Paritätsprüfung, CRC oder Plausibilitätsprüfung) keinen Fehler erkennen. Erhält der Sender innerhalb einer gewissen Zeit (***time out*** genannt) keine Quittung (ACK), so wiederholt er die Nachricht. Bei der **negativen Quittierung**, fordert der Empfänger eine als falsch erkannte Nachricht vom Sender durch eine Meldung (**NAK** - ***negative acknowledge***) wieder an. Bei der **positiv-negativ Quittierung**, wird jede empfangene Nachricht mit ACK oder NAK quittiert.

***Stop-and-Wait* ARQ**

Wir wollen nun einige Quittierungsverfahren näher ansehen und fangen mit dem einfachsten Verfahren, dem ***Stop-and-Wait*** **ARQ**-Verfahren an. Hierbei schickt die Sendestation eine Nachricht ab und wartet bis sie eine positive oder negative Quittierung von dem Empfänger erhält (Abb. 2.2-2).

Abb. 2.2-2: Stop-and-Wait ARQ-Verfahren für die Quittierung

time out

Nun kann es aber vorkommen, daß die Nachricht oder die Quittung verloren geht. Der Sender würde in diesem Fall im Wartezustand verharren, ohne je wieder eine Nachricht senden zu dürfen. Es wird deshalb vereinbart, daß nach einer Zeit t_0 (***time***

out genannt) der Sender die letzte Nachricht wiederholt. Wir nehmen nun vereinfachend an, daß der Sender stets Nachrichten gleicher Länge (d. h. der gleichen Dauer t_p) zur Übertragung vorliegen hat, die Quittung die Dauer t_q aufweist und die Signallaufzeit (in einer Richtung) *l* beträgt. Die Reaktionszeiten des Senders und Empfängers sollen vernachlässigbar, alternativ je zur Hälfte in *l* enthalten sein. Ferner werde t_0 (die Wartezeit *time out* für A) so gewählt, daß

$$t_0 = t_c - t_p \qquad 2.2\text{-}1$$

ist, wobei t_c die Zykluszeit (Zeit zwischen zwei Nachrichten) ist. Für t_c gilt

$$t_c = t_p + 2l + t_q \qquad 2.2\text{-}2$$

q sei die Wahrscheinlichkeit, daß die Nachricht *N* oder Quittung *Q* verfälscht (und als solche erkannt) wird oder nicht ankommt.

Wir betrachten nun den Erwartungswert der Wiederholungen einer Nachricht und erhalten

$$E\{n\} = 0 \cdot P\{0\} + 1 \cdot P\{1\} + \ldots + i \cdot P\{i\} + \ldots, \qquad 2.2\text{-}3$$

wobei *P{i}* die Wahrscheinlichkeit ist, daß eine Nachricht genau *i*-mal wiederholt wird. Es gilt somit

$$\begin{aligned} P\{0\} &= (1-q) \qquad 2.2\text{-}4 \\ P\{1\} &= q \cdot (1-q) \\ P\{2\} &= q^2 \cdot (1-q) \\ &\vdots \\ P\{i\} &= q^i \cdot (1-q). \end{aligned}$$

Somit haben wir

$$E\{n\} = (1-q) \cdot \sum_{i=1}^{\infty} i \cdot q^i.$$

Wegen

$$\sum_{i=1}^{\infty} i \cdot q^i = \frac{q}{(1-q)^2}$$

haben wir schließlich

$$E\{n\} = \frac{q}{1-q}. \qquad 2.2\text{-}5$$

Es sei **t** die Zeit, die erforderlich ist, eine Nachricht erfolgreich abzusenden; im Mittel benötigt man pro Nachricht die Zeit

$$E\{t\} = t_c + t_c \cdot E\{n\}$$

$$E\{t\} = t_c + \frac{q}{1-q} t_c = \frac{1}{1-q} \cdot t_c. \qquad 2.2\text{-}6$$

Beispiel 2.2-2: Leistungsfähigkeit des Stop-and-Wait ARQ-Verfahren
Die Nachricht N beim Stop-and-Wait ARQ-Verfahren besteht aus 256 Byte Nutzinformation und 6 Byte Steuerinformation. Die Quittungen ACK und NAK bestehen jeweils aus 6 Byte. Die Einwegsignallaufzeit beträgt 10 ms, und die Stationen haben je eine Reaktionszeit von 1ms. Die Übertragungsgeschwindigkeit sei 64 kbit/s. Die Sendestation habe stets Nachrichten zu übertragen.

Es gilt somit:

$$t_p = \frac{(256+6)\cdot 8Bit}{64kbit/s} = 32,75ms \qquad 2.2\text{-}7$$
$$t_q = \frac{48Bit}{64kbit/s} = 0,75ms$$
$$l = 10ms + \frac{1}{2}ms + \frac{1}{2}ms = 11ms.$$

Aus Gl. 2.2-2 erhalten wir

$$t_c = 32,75ms + 22ms + 0,75ms = 55,5ms$$

Beim idealen Kanal ist $q = 0$ und aus Gl. 2.2-6

$$E\{t\} = t_c = 55,5ms.$$

Der Durchsatz in diesem Fall ist

$$D = \frac{256\cdot 8Bit}{55,5ms} = 36,9kbit/s$$

oder genormt auf die Übertragungsgeschwindigkeit

$$\overline{D} = 0,5766.$$

Bei einem realen Kanal mit $q = 2\cdot 10^{-2}$ (dies entspricht etwa einer Bitfehlerrate von 10^{-5}) erhalten wir

$$E\{t\} = \frac{t_c}{1-2\cdot 10^{-2}} = 1,0204\cdot t_c = 56,63ms$$

und

$$D = 36,2kbit/s$$

oder

$$\overline{D} = 0,565.$$

Für $q = 0,5$ erhalten wir entsprechend

$$E\{t\} = 111ms, \qquad 2.2\text{-}8$$
$$D = 18,45kbit/s$$

oder

$$\overline{D} = 0,288.$$

Wenn für die Datenübertragung je Richtung ein Kanal zur Verfügung steht, ist das *Stop-and-Wait* Verfahren besonders ungünstig, denn es verursacht erhebliche Wartezeiten. Es ist dann üblich, mehrere Nachrichten abzusenden, ohne auf die Quittierung der einzelnen Nachrichten zu warten. Die einzelnen Nachrichten werden nun durchnumeriert, und der Empfänger bestätigt je nach Verfahren die richtig oder falsch empfangenen Nachrichten. Da auch Quittungen verloren gehen und Nachrichten wiederholt werden können, ist es erforderlich, auch die Quittungen durchzunumerieren. Da für die Durchzählung nicht beliebig lange Bitfolgen verwendet werden können, bedient man sich der zyklischen Zählung (**Sequenzierung**), auch **Modulo** *m*-**Zählung** genannt. Häufig wird $m = 8$ oder $m = 256$ gewählt. Im ersten Fall zählt man also von 0 bis 7 durch und fängt dann wieder bei 0 an, so daß hierfür bei Binärzählung 3 Bit ausreichen. Für Modulo 256 sind 8 Bit erforderlich. Wird nun nur positive Quittierung verwendet, so wird es erforderlich, daß ein Kriterium festgelegt wird, wann die nicht quittierte Nachricht wiederholt wird. Hierzu kann eine maximale Wartezeit (*time out*) verwendet werden. Üblich ist es auch festzulegen, wieviele Nachrichten maximal abgesendet werden dürfen, bevor eine nicht quittierte Nachricht wiederholt werden muß. Man nennt dies die **Fenstergröße** *n* (***window size***). Diese muß kleiner als *m* sein ($n < m$), um die Zählung eindeutig zur Identifizierung der Nachrichten verwenden zu können.

Sequenzierung

Modulo Zählung

Fenstergröße (window size)

Beispiel 2.2-3: Zählung und Fenstergröße
Gegeben sei ein Go-back-n-Verfahren mit Modulo 4-Zählung. Wählt man nun auch die Fenstergröße zu $n = 4$, so kann dies dazu führen, daß die Wiederholung einer Nachricht für eine neue Nachricht gehalten wird. Bei der Modulo 4-Zählung und Fenstergröße 4 sendet der Sender die Nachrichten 0, 1, 2, 3 ab. Der Empfänger quittiert die Nachricht Null, die Quittung geht jedoch verloren. Der Sender wiederholt deshalb die Nachricht 0. Der Empfänger hält jedoch die wiederholte Nachricht 0 für eine neue Nachricht und merkt dies nicht, wenn nicht andere Prüfungsmechanismen (z. B. Plausibilitätsprüfung) greifen.

Abb. 2.2-3: *Go-back-n* Verfahren: Fehlerfreier Fall

Abb. 2.2-4: *Go-back-n* Verfahren: Wiederholung bei ausgeschöpftem Fenster (n=4)

In Abb. 2.2-3 ist ein typischer Ablauf für ein *Go-back-n*-Verfahren mit der Fenstergröße $n = 4$ wiedergegeben. Die gesendeten Nachrichten werden Modulo 8 durchgezählt. Man bezeichnet diese vom Sender in der Nachricht mitgesendete Zahl S als die **Sendenummer** (*send number*). Die Quittungen enthalten eine Zahl R, **Empfangsnummer** (*receive number*) genannt. Diese gibt an, welche Nachricht als

Sendenummer

Empfangsnummer

nächstes erwartet wird. Dies bedeutet, daß alle Nachrichten $R-1$ bis vom Empfänger als richtig empfangen eingestuft wurden. Hat der Sender die mit der Quittung gesendete Zahl *R* richtig empfangen, so darf er weitere Nachrichten bis zur Nummer $R+n-1$ senden, bevor er die Nachricht mit der Sendenummer *R* wiederholen muß, falls keine weitere Quittierungen ankommen.

In Abb. 2.2-3 haben wir vereinfacht dargestellt, daß Nachrichten von A gesendet und von B quittiert werden. Häufig haben wir eine symmetrische Duplexübertragung und die Quittungen werden in den Nutzdatenstrom eingebettet und kommen sporadisch. In der Grundversion des *Go-back-n*-Verfahrens werden die falsch empfangenen Nachrichten vom Empfänger verworfen, während die richtig empfangenen Nachrichten quittiert werden, soweit dies dem Empfänger möglich ist. Im Abb. 2.2-4 ist der Fall dargestellt, daß das Fenster ($n = 4$) ausgeschöpft wird, keine Quittung vorliegt, und deshalb eine Wiederholung durch den Sender eingeleitet wird. Das Fehlen der Quittung kann an einer Überlastung des Empfängers liegen (dient hier also als Flußsteuerung, indem es A zum Wiederholen zwingt) oder daran, daß der Empfänger die Meldung verfälscht erhielt und sie verwarf; es kann allerdings auch sein, daß die Quittung schlicht verloren ging. Durch die Wiederholung der Nachrichten entstehen auch beim *Go-back-n*-Verfahren Wartezeiten.

Um eine unnötige Vergeudung der Zeit und somit der Übertragungskapazität zu unterbinden, kann nun vereinbart werden, daß der Empfänger, wenn er einen Fehler feststellt, dies dem Sender mitteilt und die Wiederholung ab der nächsten erwarteten Meldung einleitet. Man unterscheidet dabei zwischen Fehlern, die über die Sequenzierung festgestellt werden, und Fehlern, die über die zyklische Codierung erkannt werden.

Beispiel 2.2-4: Unerwartete Quittung bei Go-back-n-Verfahren
Im folgendem Bild ist der Fall dargestellt, daß der Sender das Fenster ($n = 4$) ausschöpft und deshalb die Sendung ab N5 wiederholt.

Abb. 2.2-5: Ausschöpfung der Fenstergröße n=4

Der Empfänger, der bisher überlastet war und daher die korrekt empfangenen Meldungen bisher nicht quittieren konnte, holt dies nach. Der weitere Verlauf könnte sich damit wie folgt darstellen:

Abb. 2.2-6: Quittierung der korrekt empfangenen Meldungen

Beispiel 2.2-5: Go-back-n-Verfahren mit Fehlerbehandlung

Im folgenden betrachten wir den Fall, daß der Empfänger Blöcke mit CRC-Fehler verwirft und Sequenzfehler dem Sender anzeigt. Der Sender wiederholt daraufhin ab der fehlerbehafteten Meldung (siehe Bild).

Abb. 2.2-7: Wiederholung der Sendung ab der fehlerhaften Meldung

Bei der **selektiven Wiederholung** (*selective repeat* ARQ-Verfahren) werden die Nachrichten auch durchnumeriert und nur die falsch angekommenen Nachrichten vom Empfänger gezielt wieder angefordert. Es ist nun ein höherer Verarbeitungsaufwand erforderlich, dafür kann in der Regel ein besserer Durchsatz erzielt werden, da unnötige Wiederholungen vermieden werden.

selektive Wiederholung

Selbsttestaufgabe 2.2-1:

1. *Erklären Sie den Begriff der Transparenz eines Netzes anhand eines Beispiels.*
2. *Wo liegt in diesem Zusammenhang ein Problem bei der Übertragung sproadisch auftretender Daten und wie wird es gelöst?*
3. *Wir betrachten das Stop-and-Wait ARQ-Verfahren (Abb. 2.2-2). In welchem Zusammenhang stehen die Länge des Nachrichtenpakets und der Gesamtdatendurchsatz der Verbindung? Welche Probleme sind bei der Vergrößerung der Paketlänge zu erwarten?*

2.3 Die HDLC-Prozedur

SDLC

HDLC

LAP

LAPB

In den frühen Siebziger Jahren entwickelte IBM ein bitorientiertes Protokoll, **SDLC** (***S**ynchronous **D**ata **L**ink **C**ontrol*) genannt, für seine SNA-Netze (siehe Abschnitt 1.1 KT I). 1976 wurde eine modifizierte Version des Protokolls, **HDLC** (***H**igh Level **D**ata **L**ink **C**ontrol*) genannt, als ISO-Norm verabschiedet. Im gleichen Jahr adaptierte CCITT die Prozedur zu **LAP** (***L**ink **A**ccess **P**rotocol*) für den Einsatz in Paketdatennetzen als Teil (Schicht 2) des X.25 Protokolls. 1980 wurde das Protokoll von CCITT zu **LAPB** (***L**ink **A**ccess **P**rotocol in **B**alanced Mode*) erweitert.

Normal Response Mode

Asynchronous Balanced Mode

Im normalen Betriebsmodus (***Normal Response Mode***) wird die HDLC-Prozedur unsymmetrisch betrieben (*Unbalanced Mode*), d. h. die Stationen sind nicht gleichberechtigt. Der Sendebetrieb kann nur von der Leitstation veranlaßt werden. Die Folgestationen reagieren nur auf Anforderung der Leitstation (siehe Abschnitt 12.1 KT II, *Sequential Polling*). Das Protokoll ermöglicht sowohl eine Verbindung (als *Single Link Protocol*) als auch mehrere Verbindungen (als *Multiple Link Protocol*) über eine Schnittstelle zu unterhalten. Im folgenden wollen wir die HDLC-Prozedur im wesentlichen in der LAPB Version näher ansehen. Das LAPB Protokoll ist ein Duplex-Protokoll, d. h. es ermöglicht, Steuersignale und Daten in beiden Richtungen zwischen gleichberechtigten Stationen auszutauschen. Die Steuerung wird von beiden an der Verbindung beteiligten Systemen symmetrisch wahrgenommen, deshalb die Bezeichnung *Balanced Mode*. Da ferner die Steuersignale von beiden Seiten sporadisch kommen können, wird auch vom ***Asynchronous Balanced Mode*** gesprochen.

CRS Cyclic Redundancy Check
FCS Frame Check Sequence

Abb. 2.3-1: Rahmenformat der HDLC Meldungen

Abb. 2.3-2: HDLC Format der Steuermeldung (Control Field)

Abb. 2.3-1 zeigt, wie Bitströme zwischen Stationen, die die HDLC-Prozedur verwenden, formatiert werden. Durch Verwendung von *Flags* (Blockbegrenzer die aus der Bitfolge `01111110` bestehen) werden Rahmen gebildet, die jeweils die auszutauschenden Nutz- und/oder Steuerinformationen enthalten.

Das in Abschnitt 2.2 erläuterte *Zero Insertion* Verfahren wird verwendet, um die Bittransparenz zu gewährleisten. Die gegebenenfalls zwischen den Rahmen entstehenden Wartezeiten werden durch die Wiederholung von *Flags* oder durch Senden der Dauer-1 Folge überbrückt. Verschickt der Sender anstatt des Ende-*Flags* die Folge `01111111`, so wird der vorangegangene Rahmen vom Empfänger verworfen (*Abort Frame*). Um die Effizienz des Protokolls bei hoher Belastung zu erhöhen, ist es gestattet, das Ende-*Flag* eines Rahmens als das Anfangs-*Flag* des nächsten Rahmens zu verwenden (d. h. das Senden eines *Flags* zu sparen).

Man unterscheidet zwischen Rahmen, die nur Steuerinformationen enthalten (siehe Abb. 2.3-1) und eine feste Länge haben (*Control Frames*) und Rahmen, die auch Nutzinformationen (genauer Schicht 3 Meldungen) enthalten und eine variable Länge aufweisen (*Information Frames*). Die variable Länge kommt dadurch zustande, daß es erlaubt ist, Nutzinformationen beliebiger Länge (eine maximal zulässige Länge für die Nutzinformationen - z. B. 1024 Byte - wird vorab vereinbart) zu versenden. Wegen der Eigenschaft der variablen Länge kann die Nutzinformation praktisch an jeder Stelle beginnen und dann Bit für Bit übertragen werden; deshalb bezeichnet man das Protokoll auch als bitsynchron.

Adresse

DTE

DCE

Dem 8 Bit *Flag* folgt eine 8 Bit **Adresse**. Es wird zwischen der Adresse des Datenendgerätes (**DTE** - ***D**ata **T**erminal **E**quipment*) und der Adresse der Datenübertragungseinrichtung (**DCE** - ***D**ata **C**ircuit Terminating **E**quipment*) unterschieden. Ferner ermöglicht die Adresse, zwischen Befehlen und Meldungen/Antworten (siehe

unten) zu unterscheiden. Befehle enthalten die Adresse des Empfängers, während in Meldungen die Adresse des Absenders angegeben wird.

Steuerfeld

Das **Steuerfeld** (*Control Field* , siehe Abb. 2.3-2) besteht in der Regel (bei Modulo 8 Zählung) aus 8 Bit und im erweiterten Modus (bei Modulo 128 Zählung) aus 16 Bit. Man unterscheidet zwischen Meldungen mit Nutzinformationen (*Information Frames*, charakterisiert durch 0 am Anfang des Steuerfeldes) und reinen Steuermeldungen (s. Abb. 2.3-3). Diese werden wiederum unterteilt in Meldungen für die Flußkontrolle (*Supervisory Frames* mit 10 am Anfang) und nichtnumerierte Steuersignale (*Unnumbered Frames* mit 11 am Anfang).

Im übrigen werden *Supervisory Frames* auch nicht durchnumeriert; lediglich die Meldungen mit Nutzinformation (*Information Frames*) werden durchnumeriert und quittiert wie beim *Go-back-n*-Verfahren. *Supervisory Frames* werden allerdings zur Quittierung mitverwendet. Die Felder N(S) dienen der Durchzählung der gesendeten Nachrichten, die Felder N(R) der Quittierung der empfangenen Nachrichten. Durch N(R) wird eine empfangene Meldung mit der Nummer N(S)=N(R)-1 quittiert.

Die zu verwendende Fenstergröße muß vor der Verbindungsaufnahme fest vereinbart werden. Das in jeder Meldung als P/F (*Poll/Final*) bezeichnete Bit dient im *Normal Response Mode* dazu, eine Richtungsumkehr einzuleiten; im *Asynchronous Balanced Mode* erzwingt das gesetzte P/F in einem Befehl eine Antwort der Gegenseite.

Anwendung	Format	Befehle	Meldungen	Codierung 8 7 6 5 4 3 2 1
Unquittierte und quittierte Informationsübermittlung	Supervisory	RR (Receive Ready)	RR (Receive Ready)	N(R) P/F 0 0 0 1
		RNR (Receive Not Ready)	RNR (Recieve Not Ready)	N(R) P/F 0 1 0 1
		REJ (Reject)	REJ (Reject)	N(R) P/F 1 0 0 1
		SREJ (Selective Reject)		N(R) P/F 1 1 0 1
	Unnumbered	SNRM (Set Normal Res-ponse Mode		1 0 0 P 0 0 1 1
		SABM (Set Asynchronous Balanced Mode		0 0 1 P 1 1 1 1
		SABME (Set Asynchronous Balanced Mode Extended		0 1 1 P 1 1 1 1
		DISC (Disconnect)		0 1 0 P 0 0 1 1
		UI (Unnumbered Information)	UI (Unnumbered Information)	0 0 0 P/F 0 0 1 1
			UA (Unnumbered Acknowledge-ment)	0 1 1 F 0 0 1 1
			FRMR (Frame Reject)	1 0 0 F 0 1 1 1
			DM (Disconnect Mode)	0 0 0 F 1 1 1 1
Verbindungs - steuerung		XID (Exchange Identification)	XID (Exchange Identification)	0 1 0 P/F 1 1 1 1

Abb. 2.3-3: HDLC Format - Beispiel zur Codierung der Steuerfelder

In Abb. 2.3-3 sind die wesentlichen Steuermeldungen und ihre Codierung angegeben. Ferner ist gekennzeichnet, ob es sich bei der Meldung um einen Befehl (*Command*), eine Meldung/Antwort (*Response*) oder beides handeln kann. Wir wollen kurz die Bedeutung einiger Meldungen ansehen:

Tab. 2.3-1: *Meldungen*

RR	RECEIVE READY RR zeigt die Empfangsbereitschaft der sendenden Seite, mit gesetztem P/F-Bit (=1) kann RR als Befehl zur Abfrage des Status der Gegenseite verwendet werden.

Fortsetzung auf der naechsten Seite

Fortsetzung von der vorhergehenden Seite	
RNR	RECEIVE NOT READY RNR zeigt an, daß von der sendenden Seite vorübergehend keine *Information Frames* empfangen werden können. Die erneute Empfangsbereitschaft kann durch Senden des RR angezeigt werden.
REJ	REJECT REJ fordert eine Wiederholung von *Information Frames* ab N(R).
SNRM	SET NORMAL RESPONSE MODE Mit SNRM wird die Gegenseite aufgefordert, nach der NRM-Betriebsweise zu arbeiten (siehe Anfang dieses Abschnittes). Alle Zähler werden initialisiert.
SABM	SET ASYNCHRONOUS BALANCED MODE Mit SABM wird die Gegenseite aufgefordert, nach der ABM-Betriebsweise zu arbeiten (siehe Anfang dieses Abschnittes). Alle Zähler werden initialisiert.
SABME	SET ASYNCHRONOUS BALANCED MODE EXTENDED Wie SABM im erweiterten Modus, d. h. mit Modulo 128 Zählung.
DISC	DISCONNECT DISC hebt den Betriebszustand, der zuletzt aufgenommen wurde, wieder auf. Es wird ein Wartezustand eingenommen, bei dem keine *Information Frames* gesendet oder empfangen werden.
UA	UNNUMBERED ACKNOWLEDGE Mit UA bestätigt die sendende Station den Empfang und die Ausführung einer UI (*Unnumbered Information*) Meldung.
FRMR	FRAME REJECT Mit der FRMR-Meldung wird eine empfangene Meldung zurückgewiesen. Die FRMR-Meldung enthält ein Datenfeld, welches das Steuerfeld des zurückgewiesenen Befehls und den Grund der Zurückweisung beinhaltet. Typische Gründe für die Auslösung von FRMR können sein: undefinierte Steuermeldung, zu langes Informationsfeld, ungültiges N(R).

Dem Steuerfeld folgen die Nutzdaten, falls es sich um *Information Frames* handelt. Ein 16-Bit Feld für die zyklische Redundanzprüfung (**CRC - *C*yclic *R*edundancy *C*heck**) schließt den Rahmen ab. Für die Prüfung wird das von CCITT empfohlene Generatorpolynom $x^{16} + x^{12} + x^5 + 1$ verwendet.

Beispiel 2.3-1: Abläufe bei HDLC
In Abb. 2.3-5 bis Abb. 2.3-7 sehen wir einige typische Abläufe in der Schicht 2 bei folgender Konfiguration:

Abb. 2.3-4: Abläufe bei HDLC

Abb. 2.3-5: Abläufe bei HDLC

Abb. 2.3-6: Abläufe bei HDLC

Abb. 2.3-7: Abläufe bei HDLC

2.4 Das X.25 Protokoll

X.25 Protokoll

Die CCITT Empfehlung **X.25** spezifiziert die ersten drei Schichten des ISO-Modells für die Datenpaketübermittlung. Wie bereits erwähnt, wird das X.21 Protokoll, alternativ X.21bis Protokoll (V.24/RS 232 für den Anschluß mit Modem über das Fernsprechnetz), für die Schicht 1 empfohlen. Hierüber wird die Schicht 2 unter Verwendung des HDLC LAPB Protokolls aufgebaut. Im folgenden wollen wir uns den Kern der X.25 Empfehlung - die Behandlung der Schicht 3 - näher ansehen. Es handelt sich dabei um den Aufbau von virtuellen Verbindungen zwischen zwei Endgeräten über das Paketvermittlungsnetz, den Austausch von Daten über diese Verbindungen und die Wartung und Verwaltung der Verbindungen. Die X.25 Empfehlung beschreibt im wesentlichen das Verhalten an der DTE/DCE-Schnittstelle (siehe Abb. 2.4-1und Abb. 2.4-2).

Abb. 2.4-1: Eine virtuelle Verbindung nach X.25

Abb. 2.4-2: X.25 Empfehlung für Paketvermittlung

Abb. 2.4-3: Struktur der Schicht-3-Meldungen

In Abb. 2.4-3 ist das Rahmenformat der X.25 Schicht-3-Meldungen und deren Einbettung in die Schicht 2 dargestellt. Die ersten vier Bit des Rahmens, **GFI** (***G****eneral*

GFI (*General Format Identifier*)

***F**ormat **I**dentifier*) genannt, werden verwendet, um das Grundformat für das Paket, so z. B. ob die Modulo 8 oder Modulo 128 Zählung verwendet wird, festzulegen. Das GFI wird auch verwendet, um zwischen Nutz- und Steuerpaketen zu unterscheiden (s. CCITT X.29) und eine explizite Bestätigung bestimmter Pakete (*Delivery Confirmation*) vorzunehmen. Die nächsten 12 Bit (**LCI** - ***L**ogical **C**hannel **I**dentifier*) dienen der Adressierung der logischen Verbindung (*Logical Channel*); hierbei wird zwischen einer Bündeladresse (*Logical Channel Group Number*, die ersten vier Bit) und einer Verbindungsadresse (*Logical Channel Number*, 8 weitere Bit) unterschieden.

LCI (*Logical Channel Identifier*)

Pakettyp

Der Adresse folgt ein Byte, das den **Pakettyp** (*packet type*) kennzeichnet. Je nach Pakettyp folgen dann ggf. weitere Elemente, die als Rest (*Remainder*) bezeichnet werden. In Abb. 2.4-4 sind einige Pakettypen mit deren Codierung und Bezeichnung angegeben.

Pakettyp		Byte 3 Bit Nr.							
DCE → DTE	DTE → DCE	8	7	6	5	4	3	2	1
Verbindungsauf - und Abbau									
Incoming Call	Call Request	0	0	0	0	1	0	1	1
Call Connected	Call Accepted	0	0	0	0	1	1	1	1
Clear Indication	Clear Request	0	0	0	1	0	0	1	1
DCE Clear Confirm	DTE Clear Confirm	0	0	0	1	0	1	1	1
Datenübertragung									
DCE Data	DTE Data	x	x	x	x	x	x	x	0
DCE Interrupt	DTE Interrupt	0	0	1	0	0	0	1	1
DCE Interrupt Confirm	DTE Interrupt Confirm	0	0	1	0	0	1	1	1
Flusskontrolle									
DCE RR (Modulo 8)	DTE RR (Modulo 8)	x	x	x	0	0	0	0	1
DCE RR (Modulo 128)	DTE RR (Modulo 128)	0	0	0	0	0	0	0	1
DCE RNR (Modulo 8)	DTE RNR (Modulo 8)	x	x	x	0	0	1	0	1
DCE RNR (Modulo 128)	DTE RNR (Modulo 128)	0	0	0	0	0	1	0	1
	DTE REJ (Modulo 8)	x	x	x	0	1	0	0	1
	DTE REJ (Modulo 128)	0	0	0	0	1	0	0	1
Weitere Meldungen									
Reset Indication	Reset Request	0	0	0	1	1	0	1	1
DCE Reset Confirm	DTE Reset Confirm	0	0	0	1	1	1	1	1
	Registration Request	1	1	1	1	0	0	1	1
Registration Confirm		1	1	1	1	1	0	1	1

Abb. 2.4-4: Einige Pakettypen und deren Codierung

Die Schicht-3-Pakettypen werden in Meldungen für Verbindungsaufbau und -abbau, Datenübertragung, Flußkontrolle und weitere Meldungen unterteilt. Die Bedeutungen sind ähnlich wie bei Schicht-2-Meldungen. Wir wollen uns nun einige Abläufe näher ansehen.

virtuelle Verbindung

In Abb. 2.4-5 ist der Ablauf beim Aufbau einer **virtuellen Verbindung** (*Virtual Circuit*) skizziert. Die Datenendeinrichtung A (DTE A) sendet die Meldung *Call Request* an seine Datenübermittlungseinrichtung (DCE A). Innerhalb des Paketvermittlungsnetzes wird diese Meldung (als *call* angedeutet) bis zur DCE B geleitet und von dort aus als *Incoming call* an DTE B weitergegeben. Bei Annahme des Rufes wird die Meldung *Call Accepted* von DTE B zu DCE B, weiter durch das Netz als *Accept* zu DCE A und von dort zu DTE A als *Call Connected* weitergeleitet. Man beachte, daß die LCN (*Logical Channel Number*) lediglich lokale Bedeutung haben und deshalb auf beiden Seiten unterschiedlich sein können (in Abb. 2.4-5: LCN=6 und LCN=2).

In Abb. 2.4-6 ist die Meldung *Call Request* bzw. *Incoming Call* genauer aufgeschlüsselt. Der aus 2 Byte bestehende Kopf (*Header*) und der aus einem Byte bestehende Pakettyp müssen stets angegeben werden. Im GFI wird übrigens angezeigt, daß es sich um das lange Adressenformat handelt. Das vierte Byte gibt die Längen der anrufenden und der angerufenen DTE Adressen an; hierfür sind je 4 Bit vorgesehen, also maximal die Länge 15 angebbar - die maximale Adressenlänge beträgt somit 15 Dezimalziffern.

Abb. 2.4-5: Ablauf des Verbindungsaufbaus (LCN: *Logical Channel Number*)

Abb. 2.4-6: Aufbau der Meldungen *Call Request* und *Incoming Call* (Long Address Format)

Abb. 2.4-7: Aufbau der Meldungen *Call Accepted* und *Call Connected* (Long Address Format)

Im Adressenfeld wird dann die Adresse (je 4 Bit pro Dezimalziffer) binär angegeben. Bei einer ungeraden Anzahl von Ziffern im Adreßfeld wird das letzte halbe Oktett mit Nullen aufgefüllt. Die Angabe der Adressenlänge (4. Byte) ist stets zwingend vorgeschrieben. Während in der Meldung *Incoming Call* beide Adressen (des Gerufenen und des Rufenden) zwingend vorgeschrieben sind, ist in der Meldung *Call Request* nur die Adresse des angerufenen Teilnehmers zwingend vorgeschrieben - die Adresse des rufenden Teilnehmers wird gegebenenfalls von seiner Vermittlungseinrichtung weitergeleitet. Das erste Byte nach der Adresse gibt die Länge der nächsten Nachricht an; diese betrifft die gewünschten Leistungsmerkmale, die wiederum ausgehandelt werden können. Die Angabe der Länge dieser *Facility* - Meldung ist zwingend, die Meldung selbst optional. Typische Leistungsmerkmale, die ausgehandelt werden, sind z. B. Paketlänge, Fenstergröße, Gebührenübernahme und die beschleunigte Übermittlung von Datenpaketen. Das Datenpaket am Schluß der Meldung (*User Data*) beträgt maximal 16 Oktette. Es gibt die Möglichkeit, unter Verwendung eines Leistungsmerkmales (*Fast Select* genannt) das Feld optional bis zu 128 Byte zu erweitern. Dies ist besonders günstig, wenn kurze Meldungen (Telemetriedaten oder *Point of Sales* Daten) übermittelt werden sollen, denn es wird damit möglich, kurze Nutzdatenblöcke bis 128 Byte mit Steuermeldungen direkt zu übermitteln.

In Abb. 2.4-7 ist die Meldung *Call Accepted* bzw. *Call Connected* genauer aufgeschlüsselt. Die Bedeutung der einzelnen Elemente der Meldung sind denen der ausführlich besprochenen Meldung *Call Request* identisch.

Nach dem erfolgreichen Aufbau der virtuellen Verbindung können nun Datenpakete und weitere Steuerpakete, die die virtuelle Verbindung betreffen, übermittelt werden. Die Adressen der Endeinrichtungen werden nicht mehr in den Paketen angegeben, die logische Adresse mit jeweils lokaler Bedeutung reicht nun aus.

In Abb. 2.4-8 und Abb. 2.4-9 sind die Meldungen DATA, RR, RNR und REJ für Modulo 8 Zählung genauer angegeben. Die Meldungen DATA (genauer DCE DATA, DTE DATA) werden verwendet, um die Daten zwischen den Endgeräten (DTE A und DTE B) auszutauschen. Wie bei den *Information*-Meldungen der Schicht 2 werden die DATA Meldungen der Schicht 3 durchnumeriert. Hierzu dient das Feld P(S) (*Packet Sent*), das aus 3 Bit besteht (Abb. 2.4-8). Das Feld P(R) (*Packet Received*) dient der Quittierung der empfangenen Pakete. Die Fenstergröße kann maximal 7 gewählt werden, der voreingestellte Wert (*Default Value*) ist 2. Das mit M gekennzeichnete Bit (*More Data*) zeigt an, ob noch weitere, zu der Nachricht gehörende Pakete folgen. Die Meldungen RR, RNR und REJ werden wie bei der Schicht 2 verwendet.

Abb. 2.4-8: DATA Meldung und RR (*Receive Ready*) Meldung

Abb. 2.4-9: *Receive Not Ready* (RNR) und *Reject* (REJ) Nachricht

Die Quittierung der Schicht-2-Meldungen *Information* und der Schicht-3-Meldungen *Data* sind voneinander unabhängig. In der Schicht 3 besteht die Möglichkeit, durch Setzen eines Bits im GFI-Feld die Quittierung vom nächsten Knoten im Netz oder vom fernen Endgerät durchführen zu lassen. Durch die Quittierung vom nächsten Knoten kann ein höherer Durchsatz erreicht werden, die Datenpakete müssen jedoch zeitweise im Netz gespeichert werden. In der

Schicht 3 wird für jede aufgebaute virtuelle Verbindung die Zählung (und somit der Quittierungsmechanismus) gesondert durchgeführt.

Das X.25 Protokoll sieht vor, einzelne Pakete bevorzugt abzufertigen (*Expedited Data*). Dies wird mit der *Interrupt*-Meldung vorgenommen. *Interrupt*-Meldungen unterliegen nicht der Zählung, werden bevorzugt behandelt und können hierdurch andere Meldungen überholen. Sie werden durch *Interrupt Confirmation* unmittelbar von fernen Endgeräten quittiert. Abb. 2.4-10 zeigt einen typischen Ablauf der bevorzugten Abfertigung.

Abb. 2.4-10: Ein typischer Ablauf der bevorzugten Datenabfertigung (*Expedited Data*)

Der Verbindungsabbau einer virtuellen Verbindung kann von einem Endgerät oder vom Netz vorgenommen werden. Beide Abläufe und der Aufbau der Meldungen sind in Abb. 2.4-11 bis Abb. 2.4-13 gezeigt.

Abb. 2.4-11: Abbau einer virtuellen Verbindung durch den A-Teilnehmer

Abb. 2.4-12: Abbau einer virtuellen Verbindung durch das Netz

Abb. 2.4-13: Meldungen für den Abbau einer virtuellen Verbindung

Selbsttestaufgabe 2.4-1:

1. *Welchen Ebenen des Schichtenmodells werden HDLC, X.25 und X.21 zugeordnet?*

2. *Weshalb haben die LCN lediglich lokale Bedeutung ? Welcher Vorteil ergibt sich daraus, bzw. welche Folgen würde eine netzweit eindeutige Numerierung haben ?*

3 LANs und MANs

3.1 Die Entwicklung von lokalen Netzen

Lokale Netze

Lokale Netze (**LAN**s - ***L**okal **A**rea **N**etworks*) werden für die Datenkommunikation im lokalen Bereich eingesetzt. Hauptmerkmale von lokalen Netzen, durch die sie sich auch von Nebenstellenanlagen unterscheiden, sind:

- eine verteilte Struktur (kein Koppelfeld oder zentrales System, autarke Stationen)
- die gemeinsame Nutzung des Übertragungsmediums durch alle Stationen unter Verwendung eines Zugriffsverfahrens
- burstartige Übertragung; die maximale Übertragungskapazität steht für eine kurze Zeit einer zugreifenden Station exklusiv zur Verfügung.

Weitere typische Merkmale sind:

- die räumliche Begrenzung auf ein privates Grundstück oder Firmengelände
- die maximale Ausdehnung bis ca. 10 km
- die Übertragungsraten zwischen ca. 0,1 und 16 Mbit/s und
- die begrenzte Anzahl von Stationen bis einigen 100.

Klassifizierung von lokalen Netzen

Für die **Klassifizierung von lokalen Netzen** werden folgende drei Kriterien häufig herangezogen:

- das Übertragungsmedium (meist verdrillte Kupferdoppeladern, Koaxkabel oder Lichtwellenleiter)
- die Topologie (meist Stern, Bus oder Ring) und
- das Zugriffsverfahren (stochastisch oder deterministisch).

stochastische Verfahren

Bei **stochastischen Verfahren** handelt es sich um Konkurrenzverfahren, bei denen die angeschlossenen Stationen zufällig gleichzeitig auf das Medium zugreifen und dadurch Kollisionen auslösen können. Deshalb müssen Kollisionsauflösungs- oder Vermeidungsstrategien eingesetzt werden.

deterministische Verfahren

Bei **deterministischen Verfahren** ist der Zugriff deterministisch geregelt. Geeignete Mechanismen verhindern die direkte Kollision. Die Sendeberechtigung wird jeweils einer Station explizit gewährt.

Entwicklung von lokalen Netzen

Die **Entwicklung von lokalen Netzen** geht auf das Jahr 1970 zurück, in dem das ALOHA-System von der University of Hawaii implementiert wurde. Es diente der Kommunikation zwischen verschiedenen, auf Inseln verteilten Rechnern über eine Funkschnittstelle mit zentralen Kontrollrechnern und zählt heute zu den klassischen stochastischen Verfahren. Das Verfahren wurde bei Xerox Research Labs in Paloalto zunächst für interne Anwendungen weiterentwickelt und 1976 der Öffentlichkeit vorgestellt. Heute ist das **ETHERNET**, das auf dieser Entwicklung basiert, mit seinen verschiedenen Varianten das am meisten verbreitete lokale Netz. 1980 begann die Standardisierung des Verfahrens beim IEEE (***I**nstitut of **E**lectrical and*

ETHERNET

E**lectronic **E**ngineers*) unter der Bezeichnung CSMA/CD (C**arrier **S**ense **M**ultiple **A**ccess with **C**ollision **D**etection*) . Der Standard wurde 1985 in seiner Grundvariante für Koaxkabel als IEEE 802.3 Standard verabschiedet. Heute gibt es mehrere standardisierte Varianten für unterschiedliche Übertragungsmedien.

Die Ansätze zur Entwicklung von lokalen Netzen mit deterministischen Verfahren, bei denen eine Sendeberechtigung (*Token*) von Station zu Station gereicht wird, gehen auch bis zum Jahre 1970 zurück. Bei IBM wurde in den frühen siebziger Jahren ein *Token Passing* Verfahren entwickelt, aber noch nicht favorisiert. Der Anreiz für die Entwicklung lag darin, daß die Verwendung von Token Verfahren mit einer Ringstruktur es ermöglicht, die Signallaufzeitabhängigkeit der *Carrier Sense* Protokolle zu überwinden. Man erkannte auch, daß die Taktung, wie sie bei Verfahren mit festen Zeitschlitzen (*Slotted Structure*) eingeführt wird, den Durchsatz der Netze erheblich erhöht. Mitte der siebziger Jahre wurden zahlreiche lokale Netze konzipiert und untersucht. Darunter auch solche, bei denen der Zugriff zum Medium über einen Einfügepuffer geregelt wird. Beispiele für *Slotted Ring* und *Buffer Insertion Ring* Netze sind Cambridge Ring und Silk.

Abb. 3.1-1: Frontend LANs

Im Gegensatz zu CSMA-Netzen, die für Meldungen keine Antwortzeiten garantieren können, sind *Token Passing* Netze, die eine maximale Antwortzeit einhalten, für Echtzeit-Anwendungen besonders geeignet. 1982 begann deshalb die Standardisierung der Token Verfahren bei IEEE. 1985 wurden der Token Bus Standard IEEE 802.4 und der Token Ring Standard 802.5 verabschiedet. Der Token Bus wird heute überwiegend in der Fabrikautomation, der Token Ring in der Büroumgebung eingesetzt.

Die Entwicklung der Glasfasertechnologie führte zu lokalen Hochgeschwindigkeitsnetzen, den *High Speed Lans* (*HSLANs*). Bereits 1982 wurde die Normung eines Hochgeschwindigkeitsnetzes auf Basis des *Token Passing* Verfahrens bei der Standardisierung bei ANSI (*American National Standards Institute*) aufgenommen. 1987 wurde ein erster Teil dieses Standards unter der Bezeichnung FDDI (***F**iber **D**istributed **D**ata **I**nterface*) verabschiedet. Heute ist FDDI das am häufigsten eingesetzte HSLAN.

HSLANs werden einerseits als *Frontend LANs* zu direkten Verbindung von Arbeitsplätzen mit hohem Bitratenbedarf und Hochgeschwindigkeitsrechnern (Abb. 3.1-1), andererseits als *Backbone LANs* also als zweite Hierarchiestufe von lokalen Netzen zur Vernetzung von einfachen LANs (Abb. 3.1-2) verwendet.

Abb. 3.1-2: Backbone LANs

Während FDDI noch für asynchrone Datenübermittlung entwickelt wurde, begannen Mitte der achtziger Jahre die Weiterentwicklung der HSLANs für Echtzeit-Daten wie Sprache und Video. Außer der Weiterentwicklung der bekannten Verfahren (wie beim FDDI II) wurden auch gänzlich neue Konzepte entwickelt. Hierzu gehören DQDB (***D**istributed **Q**ueue **D**ual **B**us*) mit einer verteilten Warteschlangenverwaltungsstrategie und einer dualen Busstruktur, CRMA (***C**yclic **R**eservation*

M**ultiple **A**ccess*) mit einem Reservierungsverfahren und ATM-Ring mit einem Kreditverfahren. Diese neuen Konzepte zeichnen sich aus durch hohe Geschwindigkeit (mehrere hundert Mbit/s bis einige Gbit/s), eine Annäherung an die Zellstruktur, die sich ab 1988 in öffentlichen Netzen durchsetzte (ATM-Zellen im Breitband-ISDN) und die Möglichkeit der Trennung zwischen dem privaten und dem öffentlichen Netzteil oder zwischen dem lokalen und dem regionalen Netzteil. Ferner unterstützen sie die Übertragung isochroner Daten, d. h. Echtzeit-Anwendungen. Hochgeschwindigkeitsnetze (HSLANs) mit diesen Eigenschaften werden auch als **MANs** (M**etropolitan **A**rea **N**etworks*) bezeichnet (Abb. 3.1-3).

MANs

AA	Application Adapter	IR	Inter-MSS-Router
CGW	Customer Gateway	MSS	MAN Switching System
EGW	Edge Gateway	SAN	Subscriber Access Network
I-MSS	MSS-Interconnection Network		

Abb. 3.1-3: Zweistufiges MAN

Wie besprochen, werden MAC-Verfahren (***M**edia **A**ccess **C**ontrol*) unterteilt in Verfahren mit stochastischem, Zugriff und solche mit deterministischem Zugriff.

Die Verfahren mit **stochastischem Zugriff**werden weiter klassifziert nach der verwendeten Kollisionsauflösungs- oder Kollisionsvermeidungsstrategie. Beispiele sind:

stochastischer Zugriff

- Kollisionsauflösung durch Warten der Stationen mit einer festen Wartewahrscheinlichkeit wie beim ALOHA-System.
- Kollisionsauflösung durch Warten der Stationen mit einer dynamischen Wartewahrscheinlichkeit wie bei CSMA/CD-Systemen.
- Kollisionsauflösung bzw. -vermeidung durch Prioritätssteuerung wie beim ISDN D-Kanal oder Hyperchannel.

Wir werden die einzelnen Strategien im folgenden kennenlernen.

deterministischer Zugriff

Die Verfahren mit **deterministischem Zugriff** werden nach dem verwendeten Konzept unterschieden. Beispiele sind:

- Statische Medienzugriffsverfahren wie Zeitmultiplexbildung (TDMA - ***T**ime **D**ivision **M**ultiple **A**ccess*), Frequenzmultiplexbildung (FDMA- ***F**requency **D**ivision **M**ultiple **A**ccess*), Codemultiplex (CDMLA - ***C**ode **D**ivision **M**ultiple **A**ccess*).
- Token Passing Verfahren wie bei Token Ring, Token Bus und FDDI.
- Verfahren mit Einfügepuffer (*Register Insertion*) wie Silk und Pierce Ring.
- Getaktete Verfahren, d. h. Verfahren mit Zeitschlitzen (*Slotted Structure*) wie Cambridge Ring.
- Kredit-Verfahren wie ATM-Ring, MetaRing.
- Reservierungsverfahren wie DQDB, CRMA.

Wir werden uns die wichtigsten Verfahren näher ansehen.

Die Standardisierung von LANs und MANs wurde einerseits im IEEE (***I**nstitut of **E**lectrical and **E**lectronic **E**ngineers*), vor allem in der Working Group 802.X (siehe Abb. 3.1-4), andererseits im ANSI (***A**merican **N**ational **S**tandards **I**nstitute*), im T*echnical Committe X3T9* (z. B. FDDI), durchgeführt. Diese Standards wurden später von der ISO (***I**nternational **S**tandardisations **O**rganisation*) übernommen (siehe Tabelle 3.1-1).

Abb. 3.1-4: IEEE-Standards 802.X

Tab. 3.1-1: *IEEE- und ISO-Normen für LANs und HSLANs*

	IEEE - Norm	ISO - Norm
LLC Sublayer	802.2	8802/2
CSMA/CD (MAC Sublayer)	802.3	8802/3
Token-Bus (MAC Sublayer)	802.4	8802/4
Token-Ring (MAC Sublayer)	802.5	8802/5
DQDB (MAC Sublayer)	802.6	8802/6

IEEE LAN Reference Model

Da sich das ***IEEE LAN Reference Model***, das zeitlich parallel zum OSI-Modell entstand, heute weitgehend durchgesetzt hat, haben wir beide in Abb. 3.1-5 gegenüber gestellt. Das LAN-Referenz Modell besteht aus 2 Schichten, die zusammen etwas mehr als die ersten beiden Schichten des OSI-Modells abdecken. Darüber liegen die nicht weiter spezifizierten Schichten, die als Anwendungen angesehen werden.

<table>
<tr><td>OSI

Reference Model</td><td colspan="2"></td><td>IEEE

Local Area Network

Reference Model</td></tr>
<tr><td>Application Layer</td><td colspan="2">Schicht 7</td><td rowspan="5">Higher-level Protocols</td></tr>
<tr><td>Presetation Layer</td><td colspan="2">Schicht 6</td></tr>
<tr><td>Session Layer</td><td colspan="2">Schicht 5</td></tr>
<tr><td>Transport Layer</td><td colspan="2">Schicht 4</td></tr>
<tr><td>Network Layer</td><td colspan="2">Schicht 3</td></tr>
<tr><td rowspan="3">Data Link Layer</td><td rowspan="3">Schicht 2</td><td></td><td>⋮</td></tr>
<tr><td>2b</td><td>Logical Link Control</td></tr>
<tr><td>2a</td><td>Media Access Control</td></tr>
<tr><td>Physical Layer</td><td colspan="2">Schicht 1</td><td>Physical Layer</td></tr>
</table>

Abb. 3.1-5: Gegenüberstellung: OSI-Modell und das LAN Referenzmodell nach IEEE

Die Schicht 1 wird als die physikalische Schicht (*Physical Layer*) bezeichnet und deckt die Bitübertragung für verschiedene Medien ab. Sie ist für LANs meist standardisiert, für MANs wird eine Adaption als *Physical Convergence Function* spezifiziert.

Die Schicht 2 wird als die Sicherungsschicht (*Data Link Layer*) bezeichnet und besteht aus zwei Teilschichten (s. Abb. 3.1-4):

MAC • Schicht 2a: Medienzugriffsschicht (***M**edia **A**ccess **C**ontrol* - **MAC**) und

LLC • Schicht 2b:Verbindungssteuerungsschicht (***L**ogical **L**ink **C**ontrol* - **LLC**).

Hinzu kommen weitere Aspekte wie Datensicherheit (*Data Security*) und Schicht-2-Adaptionen (*Bridging*).

In der MAC-Schicht (Schicht 2 a) wird die Steuerung des verteilten Zugriffs aller Stationen auf das gemeinsame Medium geregelt. Ziele dabei sind ein hoher Durchsatz, geringe Wartezeiten und eine faire Behandlung aller Stationen. Die lokalen Netze werden meist nach dem Zugriffsprotokoll bezeichnet, z. B. CSMA/CD, Token Bus, Token Ring, DQDB usw..

In der LLC-Schicht (Schicht 2 b) wird die Verbindungssteuerung geregelt. Hierzu gehören:

- Datagrammdienst ohne Bestätigung (LLC-Typ 1)
- verbindungsorientierter Datendienst (LLC-Typ 2)

- Datagrammdienst mit Bestätigung (LLC-Typ 3).

Bei MANs kommen weitere Funktionen wie z. B. Erweiterungen für synchrone Dienste, oder Konvergenzfunktionen (*Convergence Functions*) hinzu, um die unterstützen Dienste vom MAC-Protokoll unabhängig zu machen.

In den folgenden Abschnitten werden wir zunächst die einfachen lokalen Netze, die auf CSMA/CD und Token-Verfahren basieren, ansehen und anschließend uns den Weitverkehrsnetzen (***W**ide **A**rea **N**etworks* - **WANs**) und einigen HSLANs zuwenden.

WAN

Selbsttestaufgabe 3.1-1:

Zählen Sie die Eigenschaften von HSLANs und MANs auf.

3.2 Die CSMA-LANs

Das erste stochastische Zugriffsverfahren (*random access*) wurde 1970 an der University of Hawaii implementiert und ist als **Aloha System** bekannt. In der Grundversion (**Pure Aloha**) greifen die Stationen auf das Übertragungsmedium zu, sobald eine zu übertragende Nachricht vorliegt. Es entsteht eine Situation, wie sie in Abb. 3.2-1 dargestellt ist, wobei wir vereinfachend angenommen haben, daß alle Nachrichten die gleiche Länge von P Bits aufweisen. Betrachtet man die i-te Nachricht, so sieht man (Abb. 3.2-2), daß eine Kollision genau dann eintritt, wenn innerhalb des Intervalls $2P\tau$ (τ ist dabei die Dauer einer Bitübertragung) eine weitere Meldung ankommt.

Aloha System (Pure Aloha)

Abb. 3.2-1: Pure Aloha: Zufällig ankommende Nachrichten gleicher Länge

Abb. 3.2-2: Pure Aloha: Kollisionsbereich: $2P\tau$

Nimmt man nun an, daß der Ankunftsprozeß der Nachrichten ein Poisson-Prozeß mit der Ankunftsrate λ ist, so sind die einzelnen Ankünfte voneinander unabhängig. Die Wahrscheinlichkeit, daß keine Ankunft in dem Intervall $2P\tau$ liegt, ist (vgl. KT II) ?

$$1 - P\{t \leq 2P\tau\} = e^{-2\lambda P\tau}. \qquad 3.2\text{-}1$$

Dies ist auch die Wahrscheinlichkeit, daß keine Kollision auftritt. Da die Ankunftsrate λ ist, ist der Durchsatz

$$D = \text{Ankunftsrate x Wahrscheinlichkeit, dass keine Kollision auftritt}$$
$$D = \lambda e^{-2\lambda P\tau} \qquad 3.2\text{-}2$$

In Abb. 3.2-3 ist *D* in Abhängigkeit von λ (für$P\tau = 1$) dargestellt. Das Maximum ergibt sich für $\lambda = \frac{1}{2}$, d. h. für $E\{T_B\} = \frac{1}{\lambda} = 2$, wie zu erwarten ist. Ist die Ankunftsrate niedriger als $\lambda = \frac{1}{2}$, so wird der Durchsatz geringer, weil im wesentlichen nicht genügend Verkehr angeboten wird. Ist die Ankunftsrate größer als $\lambda = \frac{1}{2}$, so wird der Durchsatz wieder geringer, weil vermehrt Kollisionen auftreten. Der maximale Durchsatz von 18,4 % ist recht gering. Wir werden im folgenden noch sehen, wie dieser erhöht werden kann.

Abb. 3.2-3: Durchsatz D in Abhängigkeit der Ankunftsrate λ bei Aloha

In unseren bisherigen Betrachtungen haben wir angenommen, daß Kollisionen keine Rückwirkung auf die Ankunftsrate der Meldungen haben. Wir nehmen nun an, daß Kollisionen lediglich die Ankunftsrate auf λ' erhöhen. Wir haben dabei nicht betrachtet, wie diese Rückkopplung tatsächlich zustande kommt. Die Annahme ist erfüllt, wenn die Wiederholungen nach Kollisionen auch einen Poisson-Prozeß bilden, der von dem ursprünglichen Ankunftsprozeß unabhängig ist, und die Wiederholrate nun größer als λ ist. Für den Fall, daß recht viele Stationen vorhanden sind und Kollisionen häufig vorkommen, dürfte unsere Annahme annähernd erfüllt sein. Wir erhalten für diesen Fall

$$D = \lambda' e^{-2\lambda' P\tau} \text{ mit } \lambda' > \lambda \qquad 3.2\text{-}3$$

Wir betrachten als nächstes den Fall, daß das Übertragungsmedium synchron betrieben wird, d. h., daß ein Takt vorhanden ist, zu dem die Stationen eine Meldung absetzen können. Dieser Takt sei so gewählt, daß die Nachrichten jeweils gerade in einer Taktperiode gesendet werden können, d. h.

$$T = P \cdot \tau \qquad 3.2\text{-}4$$

Die Ankünfte seien wieder poissonverteilt. Die ankommenden Nachrichten werden nun an den Stationen gespeichert, bis die nächste Taktperiode beginnt. Da wir wieder von vielen Stationen ausgehen, können wir in unserem Modell stets annehmen, daß alle in einer Taktperiode ankommenden Nachrichten an verschiedenen Stationen vorliegen. Dieses **getaktete Aloha System** ist als **Slotted Aloha** bekannt. Durch die Taktung können Kollisionen jeweils nur am Anfang einer Taktperiode auftreten (Abb. 3.2-4) und nicht mehr über der ganzen Nachricht, wie beim Pure-Aloha-Verfahren. Eine Kollision mit einer vorliegenden Nachricht tritt somit nicht auf, wenn in der vorangegangenen Taktperiode keine (weitere) Nachricht ankam, d. h. mit der Wahrscheinlichkeit

getaktetes Aloha System

Slotted Aloha

$$1 - P\{t \leq P\tau\} = e^{-\lambda P\tau} \qquad 3.2\text{-}5$$

Für den Durchsatz gilt entsprechend

$$D = \lambda e^{-\lambda P\tau} \qquad 3.2\text{-}6$$

Abb. 3.2-4: Slotted Aloha ($P\tau = T$): Zufällig ankommende Nachrichten gleicher Länge

Abb. 3.2-5: Slotted Aloha ($P\tau = T$): Kollisionsbereich: $P\tau$

In Abb. 3.2-6 ist der Durchsatz des Slotted-Aloha-Verfahrens im Vergleich zum Pure-Aloha-Verfahren aufgezeichnet. Man sieht, daß durch die Taktung der maximale Durchsatz verdoppelt werden kann.

Abb. 3.2-6: Durchsatz D in Abhängigkeit von der Ankunftsrate λ bei Pure und Slotted Aloha

Unsere bisherigen Betrachtungen zeigen, daß beide Aloha Systeme instabil sind, da eine Erhöhung der Ankunftsrate zu einer erheblichen Verringerung des Durchsatzes führen kann. Insbesondere kann das Wissen über den Zustand des Systems zur Optimierung des Durchsatzes verwendet werden, indem versucht wird, das Verkehrsangebot so zu gestalten, daß der maximale Durchsatz erreicht wird bzw. erhalten bleibt. Da die Sendestationen jedoch dezentral angeordnet sind, ist eine solche Optimierung nicht ohne weiteres möglich. Treten Kollisionen auf, so werden diese an den Empfangsstationen gewöhnlich über Sicherungsverfahren (meist CRC) erkannt, und die Nachrichten werden wieder angefordert. Dieses führt zu erhöhtem Verkehrsangebot und so wiederum zu mehr Kollisionen. Häufig ist es so, daß die sendende Station auch das Medium abhört, so daß sie beim Auftreten einer Kollision diese erkennt, falls die gesendete Nachricht verfälscht wird. Sie kann dieses Wissen zur Optimierung des Durchsatzes verwenden, wobei sie allerdings meist nur einen kleinen Anteil am gesamten Verkehrsaufkommen hat. Es gibt auch Varianten, bei denen eine Station, die eine Kollision erkennt, die Übertragung der Nachricht abbricht und eine Rundsendung (*Jam*) an alle schickt, daß eine Kollision aufgetreten ist. Somit können alle Stationen ihr Verkehrsangebot an die Häufigkeit der Kollisionen anpassen und so dynamisch eine Optimierung des Durchsatzes anstreben. Die räumliche Entfernung spielt hierbei eine gewisse Rolle, wie das folgende Beispiel zeigt.

Beispiel 3.2-1: Kollisionserkennung und Kollisionsauflösung
Wir betrachten drei Stationen, die über ein Koaxkabel kommunizieren und wie folgt angeschlossen sind:

Abb. 3.2-7: Verbindung von drei Stationen über ein Koaxkabel

Nimmt man eine Signallaufzeit von $5\mu s$ pro km für das Koaxkabel an, so benötigt ein Signal $5 \cdot 10^{-7}$ Sekunden, um 100 Meter zu überbrücken. Wird das System mit 10 Mbit/s betrieben, so benötigt man pro Bit die Übertragungszeit von 10^{-7} Sekunden. Nimmt A eine Kollision wahr und sendet eine JAM-Meldung ab, so könnte B bereits 5 Bit und C bereits 15 Bit abgesendet haben, bevor sie diese Meldung erreicht. Sind die Entfernungen 1 km (statt 100 m) und 2 km (statt 200 m), so könnte B bereits 50 Bit und C bereits 150 Bit in der Zwischenzeit abgesendet haben.

Wir betrachten nun das getaktete Aloha Verfahren (*Slotted Aloha*) für den Fall, daß eine Kollision von den an der Kollision beteiligten Stationen erkannt wird. Tritt nun eine Kollision auf, so würden die an ihr beteiligten Stationen in der nächsten Taktperiode versuchen, die Meldung wieder abzusetzen. Dies würde erneut zu einer Kollision führen. Um solche Kollisionswiederholungen zu vermeiden, ist es erforderlich, eine Strategie über das Verhalten der Stationen, die an einer Kollision beteiligt sind, zu vereinbaren. Solche Strategien werden ***Contention Resolution*** **oder Kollisionsauflösungs-Strategien** genannt.

Contention Resolution
Kollisionsauflösungs-Strategien

Die Kollisionsauflösungs-Strategien können wiederum stochastisch oder deterministisch sein. Wir wollen einige dieser Strategien kennenlernen. Die einfachste stochastische Strategie (**gleichverteilte Wiederholung zur Kollisionsauflösung**) besteht darin, daß jede an einer Kollision beteiligte Station eine ganze Zahl *i* mit $1 \leq i \leq n$ auswürfelt und die Nachricht in der *i*-ten Taktperiode sendet. Diese Strategie hat den Effekt, daß die kollidierten Nachrichten über *n* Taktperioden annähernd gleichmäßig verteilt werden.

gleichverteilte Wiederholung zur Kollisionsauflösung

Bei der **Wiederholung mit fester Wahrscheinlichkeit zur Kollisionsauflösung** wird eine kollidierte Nachricht jeweils in der nächsten Taktperiode mit einer Wahrscheinlichkeit p_w wiederholt, bis sie erfolgreich abgesendet wird. Dies ist gleichwertig damit, daß eine Station nach einer Kollision eine Zahl *i* auswürfelt, die angibt, in welcher Taktperiode die Nachricht wieder gesendet wird. Dabei ist die Wahrscheinlichkeit, daß eine bestimmte Zahl *i* ausgewürfelt wird,

Wiederholung mit fester Wahrscheinlichkeit zur Kollisionsauflösung

$$P\{i = i\} = p_w(1 - p_w)^{i-1} \text{ für } i = \{1, 2, 3, \ldots\}. \qquad 3.2\text{-}7$$

Wir können die Methoden, die wir im Kapitel 11, KT II kennengelernt haben, verwenden, um solche Verfahren zu analysieren. Wir wollen dies kurz skizzieren. Dazu betrachten wir ein **Slotted Aloha mit einer endlichen Anzahl *q* von Quellen** (hier Stationen). Es sei β die Ankunftsrate einer freien Quelle. Wir nehmen an, daß eine Quelle, deren Nachricht an einer Kollision beteiligt war und die die Wiederholungs-

Slotted Aloha mit endlicher Anzahl *q* von Quellen

strategie eingeleitet hat, keine neue Nachricht generiert, solange die vorliegende Nachricht nicht abgesendet werden konnte. Wir nennen sie eine wartende Quelle. Wir betrachten das getaktete Aloha System jeweils unmittelbar vor dem Beginn eines neuen Taktes. Wir sagen, daß das System sich im Zustand *k* befindet, wenn genau *k* der *q* Quellen warten. Die Wahrscheinlichkeit, daß eine freie Quelle in einem Taktintervall eine Nachricht generiert, ist

$$p_f = 1 - e^{-\beta P \tau}, \qquad 3.2\text{-}8$$

wenn wir Markoff-Ankünfte voraussetzen.

Wir betrachten nun die Situation unmittelbar vor dem Beginn einer Taktperiode. Das System befinde sich im Zustand *k*. In der bevorstehenden Taktperiode werden *k* Stationen versuchen, jeweils eine Nachricht mit der Wahrscheinlichkeit p_w abzusetzen. Sei $Q_w(i,k)$ die Wahrscheinlichkeit, daß *i* der *k* Stationen eine Nachricht absenden, so gilt

$$Q_w(i,k) = \binom{k}{i} \cdot (1-p_w)^{k-i} \cdot p_w^i. \qquad 3.2\text{-}9$$

Von den (*q* - *k*) freien Stationen senden diejenigen eine Nachricht ab, die in der vorangegangenen Periode eine Nachricht generierten. $Q_f(i,k)$ sei die Wahrscheinlichkeit, daß *i* der (*q* - *k*) freien Stationen eine Nachricht absenden, so gilt entsprechend

$$Q_f(i,k) = \binom{q-k}{i} \cdot (1-p_f)^{q-k-i} \cdot p_f^i. \qquad 3.2\text{-}10$$

Wir können das Zustandsdiagramm wie in Abb. 3.2-8 angeben. Wir können auch die Übergangswahrscheinlichkeiten in Abhängigkeit von $Q_w(i,k)$ und $Q_f(i,k)$ angeben, wenn wir folgendes beachten. Von einer Takt-Periode zur nächsten erhöht sich die Anzahl der wartenden Stationen genau um die Anzahl der Ankünfte der freien Quellen weniger eins, falls eine Nachricht erfolgreich abgesendet werden konnte. Eine Nachricht kann allerdings nur erfolgreich abgesendet werden, wenn versucht wird, genau eine Nachricht abzusenden. Für die Übergangswahrscheinlichkeit vom Zustand *k* in den Zustand *k* + *i* haben wir somit

$$P_{k,k+i} = \begin{cases} Q_f(0,k) \cdot Q_w(1,k) & \text{für } i = -1 \\ Q_f(1,k) \cdot Q_w(0,k) + Q_f(0,k)[1 - Q_w(1,k)] & \text{für } i = 0 \\ Q_f(1,k)[1 - Q_w(0,k)] & \text{für } i = 1 \\ Q_f(i,k) & \text{für } 2 \le i \le (q-k) \end{cases} \qquad 3.2\text{-}11$$

Man sieht, daß die Zustandsänderung nach unten jeweils nur um eins möglich ist. Hierdurch können die Zustandsgleichungen iterativ gelöst werden, wie wir dies in KT II für Geburts- und Sterbeprozesse bereits getan haben. Mit dem Gesetz von Little (s. KT II) kann man die mittlere Wartezeit berechnen.

Abb. 3.2-8: Zustandsdiagramm für Slotted Aloha mit einer endlichen Anzahl von Quellen q

Adressenpriorität Kollisionsauflösung

Wir wenden uns nun einer deterministischen Kollisionsauflösungs-Strategie zu. Hierbei wird die **Adressenpriorität**für die **Kollisionsauflösung** verwendet. Wir betrachten wieder das Slotted Aloha Verfahren und betrachten den Fall, daß das erste gesendete Wort eine Adresse ist (die eigene oder die des Empfängers). Tritt nun eine Kollision auf, so werden die Bits auf dem Übertragungsmedium so verfälscht, daß sich bei binärer Übertragung eine physikalische Eins (Pegel auf der Leitung) gegenüber einer physikalischen Null (kein Pegel) durchsetzt (Animation 3.2-9). Kann diese Verfälschung von der betroffenen Station vor dem Senden des nächsten Bits erkannt werden und gibt die Station das Senden sofort auf, so kann die andere Station ihre Nachricht ungestört weiter senden. Dieses Verfahren setzt voraus, daß eine unmittelbare Rückkopplung für die Stationen möglich ist, und die Signallaufzeiten so klein sind, daß vor dem Senden des nächsten Bits der Nachricht eine Kollisionserkennung möglich ist. Dieses Verfahren wird im ISDN für den Zugriff auf den Signalisierkanal (D-Kanal) des Basisanschlusses angewandt.

Animation 3.2-9: Kollisionsauflösung über Adressenpriorität

Splitting Algorithms Spaltungsalgorithmen für die Kollisionsauflösung

Es gibt eine Reihe von weiteren Verfahren, die als ***Splitting Algorithms***, d. h. **Spaltungsalgorithmen für die Kollisionsauflösung** bezeichnet werden. Bei diesen

Algorithmen teilen sich die an einer Kollision beteiligten Stationen in zwei Gruppen auf, wobei jeweils eine Gruppe wartet, während die andere Gruppe senden darf. Tritt wieder eine Kollision auf, so wird die sendende Gruppe weiter aufgespalten. Jede Station führt ein Bild über auftretende Kollisionen und ermittelt daraus, wann sie wieder senden darf.

In Abb. 3.2-10 ist der Ablauf in einem konkreten Fall wiedergegeben. An einer Kollision sind fünf Stationen *(A, B, C, D, E)* beteiligt. Es wird angenommen, daß alle Stationen unmittelbar erfahren, ob in einer Taktperiode eine Nachricht erfolgreich abgesendet wurde *(e)*, eine Kollision auftrat *(k)* oder gar keine Nachricht gesendet wurde *(l)*. Die Stationen würfeln, *A, B, C*, daß sie senden und *D* und *E*, daß sie warten. In der zweiten Taktperiode tritt wieder eine Kollision auf, da *A, B* und *C* zu senden versuchen. Sie würfeln wieder. Unglücklicherweise würfeln alle drei, daß sie senden dürfen - in der wartenden Gruppe befindet sich daher keine Station. Den einzelnen Stationen wird lediglich die Rückkopplung *e, k* oder *l* gegeben, so daß ihnen die Zusammensetzung der Gruppen nicht unmittelbar bekannt ist. In der dritten Taktperiode tritt nun wieder eine Kollision auf. Dieses Mal würfelt *A*, daß sie senden darf, *B* und *C*, daß sie warten. In der vierten Taktperiode sendet somit nun *A* erfolgreich. Es darf jetzt die nächste wartende Gruppe, in unserem Beispiel *BC*, den Kanal verwenden, und so tritt wieder eine Kollision in der fünften Taktperiode auf. Erneutes Würfeln führt dazu, daß *B* senden und *C* warten darf. In der siebten Taktperiode darf dann *C* senden. Die nächste Taktperiode läuft leer, da nicht bekannt war, daß diese Gruppe leer war. Als nächstes darf die Gruppe *DE* senden, was wiederum zu einer Kollision führt. Auswürfeln führt dazu, daß *D* in der zehnten Taktperiode senden darf. *E* sendet in der elften Taktperiode, und die ursprüngliche Kollision ist somit aufgelöst.

Wir haben in unserem Beispiel in der Tabelle die einzelnen wartenden Gruppen explizit aufgeführt. Den Stationen ist jedoch nur die Rückkopplung verfügbar, so daß sie lediglich annehmen, daß bei jeder Kollision sich zwei Gruppen bilden, die sie mit 1 (sendeberechtigte Gruppe) und 0 (wartende Gruppe) bezeichnen. Somit können sie den Baum mit den binären Bezeichnungen der Zwischenknoten bzw. Blätter aufstellen (Abb. 3.2-10). Es genügt für die Stationen, jeweils lediglich einen Zähler zu haben, der bei der ersten Kollision auf Null gesetzt wird, falls sie senden darf, und auf Eins, falls sie warten muß. Zeigt die Rückkopplung eine Kollision, so wird der Zähler um Eins erhöht; zeigt sie *e* oder *l*, so wird er um Eins herabgesetzt. Ist der Zähler bei Null, so darf die Station wieder senden.

Beispiel 3.2-2: Kollisionsauflösung mittels Spaltungsalgorithmus
Wir betrachten Station E im Beispiel der Abb. 3.2-10. Nach der ersten Kollision wird ihr Zähler auf 1 gesetzt, da sie das Warten würfelt. Der weitere Verlauf ist:

nach der 2. Taktperiode ist der Zählerstand 2,
nach der 3. Taktperiode ist der Zählerstand 3,
nach der 4. Taktperiode ist der Zählerstand 2,
nach der 5. Taktperiode ist der Zählerstand 3,

nach der 6. Taktperiode ist der Zählerstand 2,
nach der 7. Taktperiode ist der Zählerstand 1,
nach der 8. Taktperiode ist der Zählerstand 0.
In der 9. Taktperiode darf die Station E senden. Es tritt eine Kollision auf. Das Würfeln leitet wieder das Warten ein und der Zählerstand wird nach der 9. Taktperiode auf 1 gesetzt. Der weitere Verlauf ist somit:
nach der 9. Taktperiode ist der Zählerstand 1,
nach der 10. Taktperiode ist der Zählerstand 0.
In der 11. Taktperiode sendet E nun erfolgreich ihre Nachricht ab.

Abb. 3.2-10: Spaltungsbaum für Spaltungsalgorithmus bei der Kollisionsauflösung

Tab. 3.2-1: *Beispiel einer Spaltungstabelle für Spaltungsalgorithmus bei der Kollisionsauflösung*

Taktperiode	Gruppe S (Senden)	Gruppen W (Warten)	Rückkopplung *k* = Kollision *e* = erfolgreiches Senden *l* = Leerlauf
1	A B C D E	-	*k*
2	A B C	DE	*k*
3	A B C	∅, DE	*k*
4	A	BC, ∅, DE	*e*
5	B C	∅, DE	*k*
6	B	C, ∅, DE	*e*
7	C	∅, DE	*e*
8	∅	DE	*l*
9	D E	-	*k*
10	D	E	*e*
11	E	-	*e*

Am vorangegangenen Beispiel dürfte deutlich geworden sein, daß viele Varianten zur Optimierung des vorgestellten Verfahrens denkbar sind. Tritt z. B. ein Leerlauf nach einer Kollision auf, so war die Spaltung nicht optimal, und eine weitere Kollision ist vorprogrammiert, wenn die entsprechende Gruppe an die Reihe kommt. Es ist deshalb sinnvoll, eine weitere Spaltung vorab vorzunehmen. Im Prinzip ist stets eine Strategie anzustreben, die es ermöglicht, eine Kollision in möglichst wenig Schritten aufzulösen. Die minimale Anzahl der erforderlichen Schritte ist gleich der Anzahl der Stationen, die an der Kollision beteiligt sind. In diesem Fall sendet in jeder Taktperiode eine Station ihre Nachricht erfolgreich ab. Mit dieser Überlegung können wir auch die Lösung eines weiteren Problems angehen. Während die Strategie zur Auflösung der Kollision angewandt wird (in unserem Beispiel 11 Taktperioden), dürften die an der Kollision nicht beteiligten Stationen weitere Ankünfte zu verzeichnen haben. Nimmt man an, daß diese sich - so lange die Kollision nicht aufgelöst ist - zurückhalten, so versuchen alle Stationen, die eine Meldung vorliegen haben, nun diese abzusetzen. Eine Kollision ist also wiederum vorprogrammiert. Hier kann folgende Strategie weiterhelfen. Ist die Ankunftsrate bekannt, so kann der Erwartungswert der Anzahl der neuen Ankünfte errechnet werden. Es werden so viele Teilgruppen aus den wartenden Stationen gebildet, daß pro Teilgruppe eine Ankunft zu erwarten ist. Die folgenden Taktperioden werden jeweils einer Teilgruppe zur Verfügung gestellt. Auch hierfür sind unterschiedliche Strategien denkbar.

Häufig sind die an einer Kollision nicht beteiligten Stationen nicht empfangsbereit. Sie haben dann auch keine Rückkopplung erhalten. Es ist deshalb auch üblich, diese gleich in die Gruppe der sendeberechtigten Stationen aufzunehmen. Sie setzen also ihre Zähler zunächst auf Null und können sich unmittelbar in den Spaltungsalgorithmus einbinden.

Heute werden häufig Zugriffsverfahren implementiert, die voraussetzen, daß bevor eine Station auf das Übertragungsmedium zugreift, sie das Medium abhört, um festzustellen, ob es frei ist. Solche Verfahren werden als **CSMA-Verfahren** bezeichnet. CSMA ist die Abkürzung für ***Carrier Sense Multiple Access***, d. h. Mehrfachzugriff mit dem Abhören des Trägers; das Übertragungsmedium wird dabei als Träger bezeichnet, und das Abhören bezieht sich auf das Feststellen, ob das Medium frei ist. Man unterscheidet manchmal zwischen Verfahren, bei denen nur vor dem Senden abgehört wird (LBT - ***l****isten* ***b****efore* ***t****alking*) und Verfahren, bei denen während des Sendens abgehört wird, um Kollisionen zu erkennen (LWT - ***l****isten* ***w****hile* ***t****alking*). Verfahren, bei denen abgehört wird und Kollisionen erkannt werden, werden auch als **CSMA/CD - Verfahren** bezeichnet (***c****arrier* ***s****ense* ***m****ultiple* ***a****ccess with* ***c****ollision* ***d****etection*). Manchmal wird auch von **CSMA/CR - Verfahren** gesprochen (***c****arrier* ***s****ense* ***m****ultiple* ***a****ccess with* ***c****ollision* ***r****esolution*), um zu betonen, daß eine Strategie zur Kollisionsauflösung angewendet wird. Ein Verfahren, bei dem die ständig abhörende Station senden darf, sobald das Medium frei ist, wird als ***persistent*** **CSMA** bezeichnet. Bei ***non-persistent*** **CSMA** hört eine Station das Medium unmittelbar vor dem Senden einer Nachricht ab; ist das Medium besetzt, so leitet sie (für sich) die Kollisionsauflösungsstrategie (wie wir sie bereits

CSMA-Verfahren

Carrier Sense Multiple Access

CSMA/CD - Verfahren

CSMA/CR - Verfahren

persistent/non-persistent **CSMA**

kennengelernt haben, z. B. würfeln, wann sie wieder senden darf) ein. Beim persistent CSMA-Verfahren besteht die Gefahr, daß beim Freiwerden des Mediums mehrere wartende Stationen gleichzeitig das Senden beginnen und somit eine Kollision verursachen. Beim non-persistent CSMA-Verfahren besteht die Gefahr, daß das Medium, obwohl es frei ist, nicht verwendet wird. Beim ***p-persistent* CSMA** Verfahren hört eine Station, die eine Nachricht übertragen möchte, das Medium ab und sendet die Nachricht mit der Wahrscheinlichkeit *p* ab, wenn das Medium frei wird. Mit der Wahrscheinlichkeit (1 - *p*) sendet sie also die Nachricht nicht ab, sondern wartet bis zum nächsten Zeitschlitz und wiederholt die Prozedur.

***p-persistent* CSMA**

Wir wollen im folgenden den Durchsatz eines synchronisierten non-persistent CSMA-Verfahrens (slotted CSMA) unter vereinfachenden Bedingungen berechnen. Wir nehmen wieder an, daß es sich bei den Ankünften der Nachrichten um einen Poisson-Prozeß mit der Gesamtankunftsrate (d. h. einschließlich der Wiederholversuche) λ handelt. Alle Pakete haben die Länge $P \cdot \tau$, wobei τ der Zeittakt ist. Das Abhören soll unmittelbar möglich sein, d. h. wir vernachlässigen Laufzeiten und Verarbeitungszeiten. Wenn eine Nachricht ankommt, sind folgende Fälle möglich (Abb. 3.2-11):

1. Ist das Medium besetzt, so wird die Nachricht zurückgestellt und entsprechend der non-persistent Strategie der Versuch wiederholt. Dies ist in der Ankunftsrate λ in unserem Modell enthalten.

2. Ist das Medium frei, so wird bis zum Anfang der nächsten Taktperiode gewartet und dann die Nachricht abgesandt. War dies die einzige Nachricht, die während der letzten Taktperiode ankam, wird sie erfolgreich gesendet und belegt den Kanal für die nächsten *P* Taktperioden. Waren mehrere Nachrichten während der letzten Taktperiode angekommen, so entsteht eine Kollision. Wir wollen annehmen, daß die gestörten Nachrichten bis zum Ende gesendet werden und dann entsprechend der Kollisionsauflösungsstrategie wieder gesendet werden. Auch diese Wiederholungen sind in unserem Modell in der Ankunftsrate λ enthalten.

Abb. 3.2-11: Slotted-CSMA (Nachrichten gleicher Länge $P\tau$), wobei τ = Taktdauer und P=Paketlänge

Die Vereinfachungen, die wir angenommen haben, entsprechen in etwa den Annahmen, die wir bei Ableitungen der Gleichungen Gl.3.2-2 und Gl. 3.2-3 bzw. Gl. 3.2-6 machten. Die Ergebnisse sind prinzipiell vergleichbar und spiegeln die Eigenschaften der Verfahren wider.

Zunächst sei bemerkt, daß jede gesendete Nachricht einen Zyklus (dessen Länge t_c eine Zufallsvariable ist) einleitet, der beim Beginn der nächsten gesendeten Nachricht endet. Die ersten *P* Zeittakte dieses Zyklus werden für die Übertragung der ungestörten oder der gestörten Nachricht verwendet. Weitere Nachrichten, die während dieser Zeit ankommen, werden abgewiesen. Der nächste Zeittakt (Hörtakt genannt) enthält stets keine Nachricht, denn alle in der vorangegangenen Taktperioder angekommenen Nachrichten wurden abgewiesen. Es folgen nun **L** Taktperioden, in denen keine Nachrichten gesendet werden (wobei **L** = 0 sein kann). In der letzten Taktperiode des Zyklus kam mindestens eine Nachricht an, denn in der nächsten Taktperiode wird wieder eine gestörte oder ungestörte Nachricht gesendet.

Wir wollen nun den Erwartungswert der Zykluslänge t_c errechnen. Es gilt

$$\begin{aligned} E\{\mathbf{t}_c\} &= E\{P\tau + \tau + \mathbf{L}\tau\} \\ &= P\tau + \tau + \tau E\{\mathbf{L}\} \end{aligned} \qquad 3.2\text{-}12$$

Ferner gilt

$$E\{\mathbf{L}\} = 0 \cdot P\{0\} + 1 \cdot P\{1\} + 2 \cdot P\{2\} + \ldots + i \cdot P\{i\} + \ldots \qquad 3.2\text{-}13$$

wobei $P\{i\}$ die Wahrscheinlichkeit, daß $\mathbf{L} = \mathbf{i}$ ist darstellt. Wir können nun die einzelnen Wahrscheinlichkeiten wie folgt angeben:

$$\begin{aligned} P\{0\} &= P\{\text{Mindestens 1 Ankunft in der Taktperiode } \tau\} \\ &= 1 - e^{-\lambda\tau} \end{aligned} \qquad 3.2\text{-}14$$

Wir wollen diese Wahrscheinlichkeit mit q abkürzen

$$q = P\{0\} = 1 - e^{-\lambda\tau} \tag{3.2-15}$$

$$\begin{aligned} P\{1\} &= P\{\text{Keine Ankunft in der Periode } \tau\} \cdot P\{0\} \\ &= e^{-\lambda\tau} \cdot P\{0\} = (1-q) \cdot q \\ P\{2\} &= P\{\text{Keine Ankunft in der Periode } \tau\} \cdot P\{1\} \\ &= (1-q)^2 \cdot q \\ &\vdots \\ P\{i\} &= (1-q)^i \cdot q \end{aligned} \tag{3.2-16}$$

Somit bildet $E\{\mathbf{L}\}$ eine geometrische Reihe mit dem Quotienten $\rho = (1-q)$ und der Summe $S = \frac{\rho}{1-\rho}$, d. h.

$$E\{\mathbf{L}\} = S = \frac{\rho}{1-\rho} = \frac{1-q}{q} = \frac{e^{-\lambda\tau}}{1-e^{-\lambda\tau}} \tag{3.2-17}$$

In Gl. 3.2-12 eingesetzt ergibt dies

$$E\{\mathbf{t}_c\} = P\tau + \tau + \frac{\tau e^{-\lambda\tau}}{1-e^{-\lambda\tau}} \tag{3.2-18}$$

Wir betrachten nun die Zyklen, insbesondere das Verhältnis der Erwartungswerte der Zeit, in der erfolgreich gesendet wird (im folgenden als ”erfolgreiche Zeit” bezeichnet), zur Gesamtzeit. Hierfür gilt:

$$\begin{aligned} &\frac{E\{\text{erfolgreiche Zeit}\}}{E\{Gesamtzeit\}} = \frac{E\{\text{erfolgreiche Zeit}\}}{P\tau} \cdot \frac{P\tau}{E\{Gesamtzeit\}} \\ &= \frac{E\{\text{Anzahl d. erfolgr. Ankünfte}\}}{E\{Gesamtzeit\}} \cdot P\tau \\ &= D \cdot P\tau, \end{aligned} \tag{3.2-19}$$

wobei D wie bisher der Durchsatz ist. Da wir Zyklen betrachten, ist

$$E\{Gesamtzeit\} = E\{\mathbf{t}_c\} \tag{3.2-20}$$

die mittlere Zyklusdauer, während für $E\{\text{erfolgreiche Zeit}\}$ gilt:

$$\begin{aligned} &E\{\text{erfolgreiche Zeit}\} \\ &\quad = P \cdot \tau \cdot P\{\text{eine Ankunft während } \tau \text{ ist erfolgreich}\} \\ &\quad = P \cdot \tau \cdot P\{\text{eine Ankunft in } \tau | \text{überhaupt Ankünfte in } \tau\} \end{aligned} \tag{3.2-21}$$

Mit der Definition der bedingten Wahrscheinlichkeit (s. KT I) erhalten wir

$$\begin{aligned} E\{\text{erfolgreiche Zeit}\} &= P\tau \cdot \frac{P\{\text{eine Ankunft in } \tau \cap \text{ Ankünfte in } \tau\}}{P\{\text{Ankünfte in } \tau\}} \\ &= P\tau \cdot \frac{P\{\text{eine Ankunft in } \tau\}}{P\{\text{Ankünfte in } \tau\}} \end{aligned}$$

Mit $P\{\text{eine Ankunft in } \tau\} = \lambda\tau e^{-\lambda\tau}$ und $P\{\text{Ankünfte in } \tau\} = 1 - e^{-\lambda\tau}$ haben wir

$$E\{\text{erfolgreiche Zeit}\} = \frac{P\tau \cdot \lambda\tau e^{-\lambda\tau}}{1-e^{-\lambda\tau}}. \tag{3.2-22}$$

Gleichungen Gl. 3.2-20, Gl. 3.2-22 und Gl. 3.2-18 eingesetzt in Gl. 3.2-19 ergeben

$$D = \frac{\lambda e^{-\lambda\tau}}{1 + P - Pe^{-\lambda\tau}}. \tag{3.2-23}$$

Wie zu erwarten, ist der Durchsatz abhängig von der Ankunftsrate, der Paketlänge und der Taktdauer. Setzen wir das Verhältnis der Taktdauer zu Paketdauer gleich *a*, d. h.

$$a = \frac{\tau}{P\tau} = \frac{1}{P} \qquad \text{3.2-24}$$

und normieren $P\tau$ auf 1, d. h. $P\tau = 1$, so erhalten wir

$$D = \frac{a\lambda e^{-a\lambda}}{a + 1 - e^{-a\lambda}}. \qquad \text{3.2-25}$$

In Abb. 3.2-12 ist der Durchsatz für verschiedene Werte von *a* in Abhängigkeit von λ aufgezeichnet. Die Extremwerte liegen bei

$$D = \frac{\lambda}{1 + \lambda} \quad \text{für a=0}$$

und

$$D = \frac{\lambda e^{-\lambda}}{2 - e^{-\lambda}} \quad \text{für a=1}$$

Abb. 3.2-12: Durchsatz D in Abhängigkeit der Ankunftsrate λ bei nonpersistent slotted CSMA mit $a = 1/P$ als Parameter und P=Paketlänge

In Abb. 3.2-13 sind die Durchsätze der drei betrachteten Verfahren (Pure Aloha, Slotted Aloha und non-persistent slotted CSMA) im Vergleich dargestellt.

Abb. 3.2-13: Durchsatz D in Abhängigkeit der Ankunftsrate λ bei Pure Aloha, Slotted Aloha und nonpersistent slotted CSMA ($a = 1/P = 0$), P=Paketlänge

Die erste Realisierung des CSMA/CD Verfahrens wurde für interne Anwendung bei Xerox Paloalto Research Labs in Kalifornien entwickelt und 1976 vorgestellt. DEC, Intel und Xerox entwickelten das Verfahren weiter zum heute meistverwendeten lokalen Netz, dem **Ethernet**. Der IEEE (***I****nstitute of* ***E****lectrical and* ***E****lectronics* ***E****ngineers*)-Standard 802.3 für CSMA/CD basiert auf dieser Entwicklung.

Ethernet

In Abb. 3.2-14 ist das Meldungsformat dieses Standards wiedergegeben. Das Trennzeichen (***Starting Frame Delimiter***) zeigt den Beginn der Meldung an. Es besteht aus einer 1 Byte langen 10-Folge in der die letzte 0 auf 1 gesetzt wird - die beiden 1 hintereinander zeigen somit den Beginn an.

Starting Frame Delimiter

Es folgen die Ziel- und die Absenderadresse. Diese können Individual- oder Gruppenadressen sein und aus 2 oder 6 Byte bestehen. Sie müssen in einem Netzwerk einheitlich sein. Das erste Bit der Adresse zeigt an, ob es sich um eine Individual- oder Gruppenadresse handelt. Die Gruppenadressen können lokal verwaltet werden oder global angelegt sein. Das zweite Bit der Gruppen-Adresse gibt hierüber Auskunft. Die Globaladressen werden für standardisierte lokale Netze von IEEE verwaltet. Die restlichen 46 Bit ermöglichen die Vergabe von $2^{46} \approx 10^{14}$Adressen - ausreichend für die eindeutige weltweite Adressierung aller LAN-Teilnehmer. Die Absenderadresse ist stets eine Individualadresse. Das I/G-Bit kann deshalb für andere Zwecke (z. B. Hinweis auf Routing) verwendet werden. Die nächsten beiden Bytes geben die Länge des nachfolgenden Datenfeldes an. Dieses darf eine maximale Länge von 1500 Byte und eine minimale Länge von 46 Byte haben. Liegt die Nutzdatenlänge unter 46 Byte, so muß gestopft werden; die Angabe im Längenfeld ermöglicht es, die Stopfbits wieder zu entfernen. Die minimale Länge ist erforderlich, um die einwandfreie Kollisionserkennung und -auflösung zu gewährleisten. Die maximale Länge erwirkt, daß ein Anwender das lokale Netz nicht zu lange blockiert. Die maximale Länge ist jedoch so gewählt, daß sie gewöhnlich für eine ganze Nachricht ausreicht.

Abb. 3.2-14: IEEE 802.3 CSMA/CD Meldungsformat

Das letzte Feld aus 4 Byte (FCS) wird für die zyklische Fehlererkennung verwendet. Das bei Ethernet eingesetzte Verfahren wurde in KT II behandelt. Am Abschluß der Meldung folgt eine Schutzzeit von mindestens $9,6\ \mu s$.

Der IEEE 802.3 Standard schreibt die persistent Variante des CSMA/CD vor. Das Kollisionsauflösungsverfahren wird als ***truncated binary exponential backoff*** bezeichnet. Nach einer Kollision wartet eine Station die Zeit **W** ab, wobei

truncated binary exponential backoff

$\mathbf{W} = \mathbf{i} \times \mathbf{T}$,
$i =$ Zufallszahl aus $\{0 \leq i \leq 2^k - 1\}$ mit
k=min(n,10) und $n =$ Anzahl der Wiederholungen der gleichen Meldung.
$T =$Slot time (*round trip delay*) = 2 x max. Signallaufzeit.

Nach 10 vergeblichen Versuchen steigt die Wartezeit im Mittel nicht weiter an, nach 16 Versuchen wird abgebrochen und eine Fehlermeldung erzeugt.

Das **Backoff-Verfahren** ist so angelegt, daß die mittlere Wartezeit exponentiell mit der Anzahl der erlittenen Kollisionen ansteigt. Dies bedeutet eine Benachteiligung aller mehrfach kollidierter Meldungen.

Backoff -Verfahren

Das Ethernet hat eine Busstruktur (Abb. 3.2-15) und wird am häufigsten auf Koaxkabel mit einer Impedanz von $50 \pm 2\Omega$ realisiert. Die Dämpfung darf bei 10 MHz, der Frequenz mit der das Netz betrieben wird, 17 dB/km nicht überschreiten. Die Signalausbreitungsgeschwindigkeit muß mindestens 77 % der Lichtgeschwindigkeit betragen. Die maximale Länge eines Segments beträgt 500 m (d. h. die Dämpfung pro Segment $\leq$ 8,5 dB), die maximale einfache Laufzeit pro Segment ist 2,165 μs. An ein Kabelsegment können bis zu 100 Anschlußsätze, die Transceiver genannt werden, angeschlossen werden [KAD]. Mehrere Segmente können über Signalregeneratoren (Repeater) zusammengeschlossen werden, um die Reichweite zu vergrößern.

Abb. 3.2-15: Ethernet Struktur

Die klassische CSMA/CD Realisierung auf Koaxkabel wird auch als 10 Base 5 bezeichnet (10 Mbit/s Datenübertragungsrate, Basisbandübertragung, 5 x 100 Meter maximale Segmentlänge). Es gibt zahlreiche Varianten des Ethernet, die auch standardisiert sind. **Cheapernet** ist eine billige Variante auf dünnerem, flexiblerem Koaxialkabel und trägt die Bezeichnung 10 Base 2 (10 Mbit/s, Basisband, 2 x 100 Meter maximale Segmentlänge). **StarLan** ist eine noch billigere Variante für verdrillte Kupferadern (1 Base 5). Eine Breitbandvariante auf 75 Ω CATV-Kabeln mit einer Baumstruktur und Frequenzmodulation wird als 10 Broad 36 bezeichnet.

Cheapernet

Starlan

Selbsttestaufgabe 3.2-1:

Betrachtet wird das Aloha System in der Grundversion (Pure Aloha) und das getaktete Aloha System mit folgenden Daten:

- *Übertragungsrate* $R = \frac{1}{\tau} = 64kbit/s$
- *Die Paketlänge umfaßt 96 Byte (d. h. eine Bildschirmzeile und Steuerinformation)*
- *100 Terminals*

- *Die Ankunftsrate beträgt 2 Pakete pro Minute pro Terminal*

Berechnen Sie die Ankunftsrate λ des gesamten Systems, die Übertragungsdauer für ein Paket t_p und den normierten Durchsatz $\overline{D} = t_p \cdot D$ für

a) Aloha System in der Grundversion (Pure Aloha)
b) das getaktete Aloha System (Slotted Aloha)

3.3 Die Token-LANs

Wir betrachten nun deterministische Zugriffsverfahren, d. h. Zugriffsverfahren mit Sendeberechtigung (***Token Access***). Wie bereits erwähnt, sind diese Verfahren dem *Hub Polling* verwandt (KT II, Kap. 12.1) - der Unterschied besteht lediglich darin, daß die Meldungen zwischen beliebigen Stationen ausgetauscht werden, und der Zugriff dezentral verwaltet wird. Wir wollen die Grundzüge der Token Access Verfahren an Abb. 3.3-1 illustrieren.

Token Access

Abb. 3.3-1: Token Access Verfahren
Die Reihenfolge der Tokenübergabe ist stets geschlossen

n Stationen sind an ein Übertragungsmedium (Netz) angeschlossen. Die Sendeberechtigung (*Token*) wird nach der Initialisierung in der Reihenfolge $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow F \rightarrow A$ übergeben. Die letzte Station übergibt also die Berechtigung wieder an die erste Station - man spricht deshalb auch von einem logischen Ring (nicht zu verwechseln mit der physikalischen Struktur des Netzes). Die Station, die die Sendeberechtigung (*Token*) besitzt, darf für eine befristete Zeit (*token holding time*) das Medium für die Datenübertragung verwenden. Hat die Station

keine Daten zu übertragen, oder ist die Zeit abgelaufen, so reicht die Station die Sendeberechtigung weiter. Systeme mit Token-Verfahren unterscheiden sich unter anderem in ihrer physikalischen Struktur (z. B. Bus oder Ring), in dem Verfahren zur Festlegung der logischen Reihenfolge und in der Implementierung der Verwaltungsaufgaben (z. B. Überwachung der Sendeberechtigung, Aufnahme neuer Stationen, Verhalten im Fehlerfall). In Abb. 3.3-2 ist die typische Rahmenstruktur einer Meldung eines Systems mit Token Access Verfahren dargestellt. Die Rahmensteuerung (*Frame Control* FC) enthält u. a. die Information, ob es sich um eine Sendeberechtigung (*Token*) oder um einen Informationsrahmen handelt. Als Ziel können gewöhnlich einzelne oder alle Stationen, manchmal auch Gruppen von Stationen adressiert werden.

Abb. 3.3-2: Rahmenstruktur einer Meldung eines Systems mit Token Access Verfahren

Token-Verfahren werden als deterministisch bezeichnet, weil die Reihenfolge, in der die Stationen die Sendeberechtigung erhalten, vorab festliegt. Sie sind insofern fair, daß jede Station pro Zyklus die Sendeberechtigung erhält und die Zyklusdauer (*token rotation time*) in der Regel begrenzt ist, da auch die Sendedauer (*token holding time*) der Stationen in der Regel begrenzt ist. Es ist recht einfach bei Token-Verfahren den Sendern und/oder den einzelnen Meldungen Prioritäten einzuräumen. Es besteht auch die Möglichkeit die Sendedauer abhängig von der Priorität der Meldung und/oder der tatsächlich beim letzten Umlauf aufgetretenen Zyklusdauer (und damit der Auslastung des Systems) zu machen.

Für die Analyse der Zyklusdauer (d. h. die Dauer, in der die Sendeberechtigung den logischen Ring einmal durchläuft) des *Token Access*-Verfahrens erhalten wir (analog zur Kapitel 12, KT II) für *Hub Polling*

$$t_c = t_z + n \cdot t_p + \sum_{i=1}^{n} l_i + \sum_{i=1}^{n} \mathbf{t}_{d_i} + \sum_{i=1}^{n} \mathbf{t}_{n_i} \ . \qquad 3.3\text{-}1$$

Wir haben dabei angenommen, daß eine der Stationen gewisse Verwaltungsaufgaben übernimmt und hierfür pro Zyklus die Zeit t_z benötigt. t_p ist wie bisher die Reaktionszeit der Peripherie. l_i ist die Laufzeit der Meldung der i-ten Station (bei mehreren Meldungen an einer Station nehmen wir an, daß diese unmittelbar hintereinander gesendet werden). $\mathbf{t}_{d_i}$ist die Zeit, die für die Übertragung der Steuerinformationen benötigt wird. Sie ist nun abhängig von der Anzahl der Meldungen pro Station und somit eine Zufallsvariable. $\mathbf{t}_{n_i}$ ist die Zeit für die Übertragung der Nutzinformationen und somit auch eine Zufallsvariable. Für den Erwartungswert der Zyklusdauer erhalten wir für eine physikalische Busstruktur

$$E\{\mathbf{t}_c\} = t_z + n \cdot t_p + \frac{nl}{3} + \sum_{i=1}^{n} E\{\mathbf{t}_{d_i} + \mathbf{t}_{n_i}\} \qquad 3.3\text{-}2$$

und für eine physikalische Ringstruktur

$$E\{\mathbf{t}_c\} = t_z + n \cdot t_p + \frac{nl}{4} + \sum_{i=1}^{n} E\{\mathbf{t}_{d_i} + \mathbf{t}_{n_i}\} \qquad 3.3\text{-}3$$

Beispiel 3.3-1: Mittlerer Abstand zweier Stationen an einem Bus und einem Ring

Wir leiten nun Gl. 3.3-2 ab, indem wir zeigen, daß bei einer zufälligen Verteilung der Stationen an einem lokalen Netz der Länge l der mittlere Abstand zwischen zwei Stationen $E\{\mathbf{l}_i\}$ für eine Busstruktur $\frac{l}{3}$ und für eine Ringstruktur $\frac{l}{4}$ beträgt.

Zunächst betrachten wir zwei Stationen x_1 und x_2 an einem Bus.

Abb. 3.3-3: Zwei Stationen an einem Bus

Für ein festes x_1 gilt

$$\begin{aligned} E\{|\mathbf{x}_1 - \mathbf{x}_2|\} &= \int_0^l |x_1 - x_2| \cdot \tfrac{1}{l} \cdot dx_2 \\ &= \int_0^{x_1} (x_1 - x_2) \cdot \tfrac{1}{l} \cdot dx_2 + \int_{x_1}^{l} (x_1 - x_2) \cdot \tfrac{1}{l} \cdot dx_2 \\ &= \tfrac{1}{l}(x_1 x_2 - \tfrac{x_2^2}{2})|_0^{x_1} + \tfrac{1}{l}(\tfrac{x_2^2}{2} - x_1 x_2)|_{x_1}^{l} \\ &= \tfrac{1}{l}(\tfrac{l^2}{2} - x_1 l + x_1^2) \end{aligned}$$

Da der Erwartungswert von x_1 abhängig ist, mitteln wir über alle x_1 und erhalten

$$\begin{aligned} E\{|\mathbf{x}_1 - \mathbf{x}_2|\} &= \int_0^l \tfrac{1}{l}(\tfrac{l^2}{2} - x_1 l + x_1^2)\tfrac{1}{l} \cdot dx_1 \\ &= \tfrac{1}{l^2}(\tfrac{l^2 x_1}{2} - \tfrac{x_1^2 l}{2} + \tfrac{x_1^3}{3})_0^1 \\ &= \tfrac{l}{3} \end{aligned}$$

Für zwei Stationen an einem Ring der Länge l gilt für ein festes x_1

$$E\{|\mathbf{x}_2 - \mathbf{x}_1|\} = \frac{\pi r}{2} = \frac{l}{4}$$

Der Erwartungswert ist unabhängig von x_1 und gilt somit allgemein.

Abb. 3.3-4: Erwartungswert unabhängig von x_1

Token-Verfahren werden physikalisch als Bus oder Ring implementiert. Beide Realisierungen sind von IEEE standardisiert.

Token Bus

Der **Token Bus** wird im IEEE 802.4 Standard spezifiziert. Für die Realisierung können verschiedene Medien, Geschwindigkeiten und Modulationsverfahren verwendet werden. Beispiele sind 1,5 oder 10 Mbit/s Einkanal-Bus mit FSK-Modulation (***F**requency **S**hift **K**eying*) oder gerichteter Breitband-Bus mit kombinierter Amplituden- und Phasen-Modulation. Das Meldungsformat für den Token Bus nach IEEE 802.4 ist in Abb. 3.3-5 wiedergegeben.

Die Preambel hat eine Mindestdauer von $2\mu s$ und beträgt eine ganze Anzahl von Oktetts. Die Preambel und das Trennzeichen (***S**tart **D**elimiter*) werden abhängig vom Modulationsverfahren festgelegt.

Die ersten beiden Bits des Kontrollfeldes (***F**rame **C**ontrol*) zeigen an, ob es sich bei der Meldung um eine Datenmeldung oder eine Steuermeldung (wozu auch der Token gehört) handelt. Im Falle der Datenmeldung können noch Prioritäten angezeigt werden. Einige Kontrollmeldungen werden im folgenden besprochen.

Die Adressierung entspricht der beim CSMA/CD-Verfahren nach IEEE 802.3. Auch das für die zyklische Fehlerprüfung vorgeschriebene Polynom CRC-32 ist identisch

mit dem Polynom beim Ethernet (s. KT II, Abschnitt 7.4). Ein Trennzeichen (***E**nd **D**elimiter*) zeigt das Meldungsende an.

Das Token Bus Protokoll ist wesentlich komplexer als das CSMA/CD Protokoll, da aufwendige Verwaltungsaufgaben durchgeführt werden. Hierzu gehören die Tokenverwaltung, das Ein- und Ausgliedern der Stationen und die Initialisierung.

Abb. 3.3-5: IEEE 802.4 Token-Bus Meldungsformat

Die Token-Verwaltung ist dezentral ausgelegt und wird gemeinsam von allen Stationen wahrgenommen. Jeder am Bus angeschlossenen Station wird eine Adresse zugeordnet. Die Vergabe der Adressen am Bus ist frei, so daß Adressen fehlen können und eine beliebige Reihenfolge am physikalischen Bus möglich ist. Der logische Ring ist nach fallenden Adressen geordnet, d. h. der Token wird in dieser Reihenfolge weitergereicht; die Station mit der niedrigsten Adresse reicht den Token an die Station mit der höchsten Adresse weiter, um so den Ring zu schließen. Jede Station merkt sich ihren Vorgänger und ihren Nachfolger am logischen Ring. Den Vorgänger erfährt sie über die Quelladresse der Token-Meldung. Den Nachfolger kann sie mit der Meldung *who follows next* erfragen. Mögliche Konfliktfälle, z. B. wenn mehrere Nachfolger sich melden, werden gesondert aufgelöst. Erhält eine Station den Token, so darf sie für eine begrenzte Dauer (*token holding time*) Daten senden. Anschließend gibt sie den Token an die Folgestation weiter und überwacht, daß diese aktiv wird. Ist dies innerhalb der Überwachungsdauer (*lost token time*) nicht der Fall, so versucht sie erneut ihr den Token zu übermitteln. Gelingt dies auch nicht, so versucht sie die nächste Folgestation (ggf. über den ganzen Adressraum) zu ermitteln und so den Token weiter zu reichen. Stellt eine Station, die einen Token besitzt, fest, daß eine andere Station sendet, so kehrt sie in den Abhörmodus zurück. Dies sichert letztlich, daß sich nur ein Token im Netz befindet.

Wird eine neue Station an den Bus angeschlossen, so muß sie erst in den Ring aufgenommen werden, bevor sie sich an dem Nutzdatenaustausch beteiligen kann. Jede aktive Station sendet hierzu regelmäßig eine Aufforderung (*solicit successor*), die an alle Stationen, deren Adresse zwischen der Sendenden und ihrer Nachfolger liegt, gerichtet ist, und wartet ein Antwortzeit-Fenster ab. Antwortet keine Station, so bleibt es wie zuvor. Die bisherige Nachfolgestation erhält dann den Token. Meldet genau eine Station sich zurück, so wird diese in den Ring aufgenommen. Sie merkt sich die Adressen ihrer Vorgänger- und Nachfolge-Station, die in der *solicit successor* Meldung enthalten waren. Die Nachfolgestation merkt sich wiederum die Adresse ihrer neuen Vorgängerstation, sobald sie ihrerseits den Token erhält. Antworten mehrere Stationen auf die Aufforderung (*solicit successor*), so liegt ein Konfliktfall vor, und eine Konfliktauflösungsstrategie wird eingeleitet.

Wenn eine Station den Ring verlassen möchte, so sendet sie, wenn sie den Token erhält, eine entsprechende Meldung (*set successor*) an ihre Vorgängerstation und teilt ihr die Adresse ihres Nachfolgers mit. Außer diesem geordneten Ausgliedern kann sie auch auf den Tokenempfang hin gar nicht agieren und somit den bereits besprochenen Fehlerfall durch Überschreitung der Überwachungszeit (*lost token time*) auslösen.

Die Initialisierung ist eine besondere Variante der Eingliederung einer Station. Stellt eine neu angeschlossene Station fest, daß binnen der Überwachungszeit (*lost token time*) der Bus inaktiv bleibt, so ergreift sie die Initiative und sendet eine *claim token* Meldung. Auch hier kann der Konfliktfall auftreten, daß mehrere Stationen die fehlende Aktivität beobachten und gleichzeitig die Initiative ergreifen. In diesem Fall wird eine Konfliktauflösungsstrategie angewandt, um die Station zu ermitteln, die den Token erhält. Der Ring wird dann durch die Eingliederungsstrategie weiter aufgebaut, bis alle Stationen daran beteiligt sind.

Wir wollen nun den Konfliktfall, daß mehrere Stationen auf eine Eingliederungsaufforderung (*solicit success* Meldung) antworten, näher betrachten. Die initiierende Station merkt an dem nicht identifizierbaren Signal im Antwortfenster, daß der Konfliktfall vorliegt, sendet daraufhin eine *resolve contention* Meldung und wartet vier Antwortzeitfenster. Gemäß den ersten beiden Bits ihrer Adresse ordnen sich die antwortenden Stationen den Zeitfenstern zu. In dem ihnen zugeordneten Zeitfenster dürfen die Stationen wieder antworten, wenn vorher keine andere Station geantwortet hat. Nun kann wieder der Fall auftreten, daß in einem verwendeten Fenster genau eine Station sendet, dann erhält diese als nächstes den Token. Es kann allerdings auch der Fall auftreten, daß wieder mehrere Stationen versuchen in einem Fenster zu senden. In diesem Fall wird die Prozedur wiederholt, wobei nun die nächsten beiden Adressbits den Ausschlag geben. In dem Fehlerfall, daß zwei Stationen die gleiche Adresse haben, wird die Prozedur bis zu den letzten beiden Adressbits durchgeführt. Anschließend würfeln die noch beteiligten Stationen zwei Bits zufällig aus und beteiligen sich mit diesen an der Konfliktauflösungsprozedur. Auf diese Weise werden die Stationen differenziert und die Station, die, ggf. nach wiederholter Verwürfelung, den Token erhält, wird eingegliedert. Die andere

Station stellt nun fest, daß ihre Adresse bereits vergeben ist, erzeugt eine Fehlermeldung und versucht nicht mehr sich einzugliedern.

In Abb. 3.3-6 sind die wesentlichen Zustände, in denen sich eine Station an einem Token-Bus LAN befinden kann und die möglichen Zustandsübergänge dargestellt. Der Abbildung ist zu entnehmen, daß das Protokoll recht komplex ist. Für weitere Details wird dem Leser die IEEE 802.4 Spezifikation empfohlen.

Abb. 3.3-6: Zustandsdiagramm einer Token-Bus Station nach IEEE 802.4

Token Ring

Der **Token Ring** wird im IEEE 802.5 Standard spezifiziert. Er wurde vor allem von IBM für die Vernetzung ihrer Rechner konzipiert und weiterentwickelt.

Die physikalische Struktur des Ringes ist in Abb. 3.3-7 dargestellt. Wesentlich ist, daß es sich um eine gerichtete Übertragung handelt und die Stationen aktiv in dem Ring eingebunden sind, d. h. daß sie die Signale regenerieren. Dies hat den Vorteil, daß im Vergleich zu einer passiven Struktur wesentlich größere Entfernungen überbrückt werden können. Auch der Einsatz von Lichtwellenleitern für die Verbindung zwischen den Stationen ist einfach möglich. Nachteil dieser Struktur ist, daß der Ausfall einer Station den Ring unterbricht. Deshalb wird eine Anschlußtechnik verwendet, die es ermöglicht, eine Station im Störungsfall zu umgehen bzw. den Ring kurz zu schließen und somit ihn zu heilen (*bypass technique*). Als Übertragungsmedium werden verdrillte Doppeladern verwendet. Die Übertragungsgeschwindigkeit beträgt 1, 4, und gelegentlich auch 16 Mbit/s. Bei 4 Mbit/s können auf verdrillten, geschirmten Doppeladern bis zu 260 Stationen an einem Ring angeschlossen werden, dessen Gesamtlänge dann etwa 50 km betragen kann.

Abb. 3.3-7: Physikalische Struktur des Token Rings

Typisch für den Token Ring ist, daß die Stationen die Meldungen Bit für Bit erhalten, die einzelnen Bits lesen, ggf. diese verändern und an die nächste Station weiterreichen. Eine Meldung wandert auf diese Weise vom Sender zum Empfänger über den ganzen Ring zurück zum Sender, der die Meldung vom Ring entfernt; der Empfänger kopiert lediglich die an ihn adressierte Meldung. Bei dieser Arbeitsweise haben die einzelnen Stationen zu einem Zeitpunkt einen unterschiedlichen Informationsstand. Es ist deshalb nicht möglich, eine gemeinsame Überwachung des Ringes durchzuführen. Eine ausgewählte Station führt deshalb die Überwachung (*monitoring*) durch. Die Stationen sind so ausgelegt, daß jede Station diese Funktion übernehmen kann. Die überwachende Station verzögert auch den Token-Rahmen um 24 Bit, so daß die sendende Station auch bei einem sehr kleinen Ring den vollständigen Token absenden kann, bevor sie ihn wieder empfängt. Wir werden weitere Funktionen der überwachenden Station im folgenden noch kennenlernen.

J, K	Violation Bits	F	Frame Type	C	Frame Copied
P	Priority	Z	Control		
T	Token	I	Intermediate Frame		(Alle numerischen
M	Monitor	E	Error Detected		nichtbezeichneten Werte
R	Reserved	A	Address Recognized		tragen die Einheit : Byte)

Abb. 3.3-8: Token Ring Meldungsformat

In Abb. 3.3-8 ist das Meldungsformat für den Token Ring nach IEEE 802.5 wiedergegeben. Als Leitungscode wird der *Differential Manchester Code*, den wir bereits im Kapitel 8.2, KT II kennenlernten, verwendet. Kennzeichnend für diesen Code ist, daß in der Bitmitte stets ein Nulldurchgang stattfindet. Ein Sprung am Bitanfang zeigt eine Null, kein Sprung eine Eins an. Beim Token Ring wird die Verletzung der Coderegel, daß in der Bitmitte stets ein Übergang stattfindet, verwendet, um zwei weitere Zeichen zu codieren. Findet kein Sprung am Bitanfang des regelverletzenden Bits statt, so wird es mit `J`, andernfalls mit `K` bezeichnet. Die Symbole `J` und `K` werden nur in den Trennzeichen (*Delimiter*) verwendet. Dies ermöglicht, die Trennzeichen im laufenden Bitstrom leicht aufzufinden. Das Beginn-Trennzeichen (*Start Delimiter*) besteht in dieser Bezeichnung aus der Folge `JK0JK00`.

Start Delimiter

Der Token besteht aus einem verkürzten Rahmen aus drei Bytes: dem Beginn-Trennzeichen (*Start Delimiter*), dem Zugangssteuerfeld (*Access Control*) und dem Ende-Trennzeichen (*End Delimiter*). Das Zugangssteuerfeld (*Access Control*) besteht aus drei Prioritätsbits, dem Token-Bit, dem Monitor Bit und drei Reservierungsbits. In den Prioritätsbits und Reservierungsbits können jeweils 8 Prioritäten angegeben werden. Eine Station, die einen Token mit T = 0 erhält (*free token*), darf Meldungen mit Prioritäten höher oder gleich der in den drei Prioritätsbits eingetragenen Priorität senden. Sie verändert das Token-Bit T zu 1 und ergänzt den Rahmen zu einem vollständigen Informationsrahmen.

Eine Station, die einen Rahmen mit T = 1 erhält (*Busy Token*) und Meldungen zu senden hat, überprüft die Reservierungsbits. Ist die hier eingetragene Reservie-

rungspriorität niedriger als die der zu sendenden Meldungen, so ändert die Station die Reservierungsbits auf die höhere Priorität. Wenn die sendende Station einen freien Token generiert, setzt sie die Prioritätsbits auf den Wert der Reservierungsbits. Somit ermöglicht sie es einer Station mit der hohen reservierten Priorität, den Token zu nutzen. Die Station, die Reservierungsbits gesetzt hat, setzt diese wieder herunter, wenn sie ihre Meldungen dieser Priorität abgesetzt hat. Ein neugenerierter Rahmen enthält das Monitorbit M = 0. Die überwachende Station (*Active Monitor*) setzt stets M = 1. Wenn ein Rahmen mit M = 1 an der überwachenden Station vorbeikommt, nimmt sie den Rahmen vom Ring und generiert einen freien Token. Dies verhindert, daß eine Meldung mehrfach im Ring kreist und ihn blockiert.

Das Rahmensteuerfeld (*Frame Control*) gibt den Rahmentyp an. Die ersten beiden Bits FF geben an, ob es sich um einen Steuerrahmen (FF = 00) handelt. Die weiteren Bits sind für künftige Anwendung reserviert (RR) oder übermitteln Steuerinformationen (ZZZZ).

Die Adressierung und die zyklische Fehlerprüfung entsprechen der beim CSMA/CD-Verfahren nach IEEE 802.3. Die zyklische Fehlerprüfung erstreckt sich von *Frame Control* bis einschließlich der *Frame Check Sequence*. Dies läßt zu, daß dem Protokoll entsprechend einzelne Bits (des Access-Control Feldes) von Zwischenstationen verändert werden können, ohne daß FCS einen Fehler anzeigt.

Das Ende-Trennzeichen (*End Delimiter*) zeigt das Ende eines Rahmens an. Das I-Bit (*Intermediale Frame Bit*), wenn auf 0 gesetzt, zeigt, daß es sich um den letzten Rahmen einer logischen Gruppe von mehreren zu übertragenden Rahmen oder um einen Einzelrahmen handelt.

Das E-Bit (*Error Detected*) zeigt normalerweise 0 an. Findet eine Station eine Fehlerbedingung im Rahmen (z. B. FCS zeigt einen Fehler an, Codeverletzungen treten außerhalb der Trennzeichen auf, Rahmenlänge ist nicht ganzzahliges Vielfaches eines Bytes), so setzt sie E = 1. Die Sendestation erkennt die Fehleranzeige, wenn der Rahmen zu ihr zurückkommt und kann den Rahmen wiederholen.

Das letzte Byte des Rahmens gibt den Rahmenstatus an (*Frame Status*) und wird somit für besondere Steuerungsaufgaben verwendet. Da das Byte durch FCS nicht geschützt ist, werden die ersten vier Bits in der zweiten Bytehälfte wiederholt und dienen somit zur Fehlererkennung. Das A Bit wird von der Sendestation auf 0 gesetzt. Erkennt eine Station, daß der Rahmen an sie adressiert war, setzt sie A auf 1. Die Sendestation erfährt, daß eine Station, die von ihr adressiert wurde, existiert, sobald der Rahmen mit A = 1 zu ihr zurückkommt. Diese Prozedur kann auch von einer Station verwendet werden, um festzustellen, ob eine andere Station mit ihrer Adresse bereits im Ring existiert. Hierzu braucht sie lediglich eine Meldung an sich selbst zu adressieren. Das C-Bit (*Frame Copied*) wird vom Sender auf 0 gesetzt. Sobald die Empfangsstation die an sie gerichtete Meldung kopiert hat, setzt sie C auf 1. Der zurückkehrende Rahmen mit C = 1 (und A = 1) bestätigt dem Sender, daß der Empfänger die Meldung erhalten hat.

Wie wir gesehen haben spielt die überwachende Station (*Active Monitor*) im Token Ring eine besondere Rolle. Insbesondere ist sie zuständig für das Erzeugen des

Ringtaktes, das Verzögern des Tokens um 24 Bit, das Unterbinden von kreisenden Meldungen im Ring und für das Räumen des Ringes bei undefinierten Zuständen und Erzeugen des neuen Tokens.

In regelmäßigen Abständen sendet die aktive Überwachungsstation die Meldung *Active Monitor Present*, und zeigt damit den anderen Stationen, daß sie aktiv ist. Im Fehlerfall wird eine *Token Claiming* Prozedur eingeleitet. Am Ende dieser Prozedur erhält die beteiligte Station mit der höchsten Adresse die Monitorfunktion - die vorherige Monitorstation wird von der Prozedur ausgeschlossen.

Das Eingliedern einer Station wird in mehreren Schritten durchgeführt. Zunächst führt sie bei kurzgeschlossenem Adapter interne Tests durch, dann hört sie den Ring ab, ggf. leitet sie dann die *Token Claiming* Prozedur ein. Sobald sie einen freien Token erhält, überprüft sie, wie bereits beschrieben, die Eindeutigkeit ihrer Adresse, ggf. koppelt sie sich wieder ab. Falls die Adresse eindeutig ist, erfragt sie die Adresse der Station vor ihr und identifiziert sich gegenüber der Folgestation. Gewöhnlich werden die Adressen der Nachbarn nicht benötigt. Im Fehlerfall einer Station kann die Folgestation die Ausgliederung der Station erwirken. Als letzten Schritt überprüft die Station ihre Kompatibilität mit dem Ring und setzt ggf. die erforderlichen Parameter auf zulässige Werte.

Die Komplexität des Token Rings ist vergleichbar mit der Komplexität des Token Busses. Während der Bus überwiegend in Fabrikumgebung eingesetzt wird, findet der Ring überwiegend in der Büroumgebung Einsatz.

Selbsttestaufgabe 3.3-1:

Ein Token Ring wird mit 4 Mbit/s betrieben und ist 10 km lang. Die Signallaufzeit beträgt $4,2\mu s$ *pro km. Es sind 100 Stationen am Ring angeschaltet, die jeweils eine Verzögerung von 1 Bit verursachen. Zur Adressierung der Stationen werden Adressen der Länge 6 Byte verwendet.*

a) Wie groß ist die Token-Umlaufzeit beim Ring ohne Verkehr?

b) Wie lange verweilt eine Nachricht mit 1024 Byte Daten auf dem Ring?

3.4 WANs, MANs, HSLANs

Mit wachsendem Einsatz von Rechnern steigt auch der Bedarf nach ihrer Vernetzung. Viele Leistungen werden vermehrt von einzelnen Rechnern als Server zur Verfügung gestellt. Auch viele Ressourcen wie hochwertige Drucker, Massenspeicher, Datenbanken usw. werden zentral aufgebaut. Als Folge nehmen sowohl die Anzahl der Lokalen Netze als auch ihre Ausdehnung zu.

Repeater

Wir haben bereits gesehen, daß **Repeater** als Signalregeneratoren eingesetzt werden, um die geographische Ausdehnung eines Lokalen Netzes zu vergrößern. Ihre Aufgabe wird nach der OSI-Modellierung der Schicht 1 zugeordnet. Repeater verlängern lediglich die Reichweite eines Lokalen Netzes; das Segment, das durch den Repeater bedient wird, ist ein interner Teil des Netzes.

Bridge

Eine Einrichtung, die zwei Lokale Netze unter Einhaltung der Vielfachzugriffsregeln beider Netze (d. h. auf MAC - ***M**edium **A**ccess **C**ontrol*-Basis) miteinander verbindet, wird als Brücke bzw. **Bridge**bezeichnet. Im einfachsten Fall verhält sich eine Brücke als ein Teilnehmer an beiden Netzen. Tritt eine Meldung in einem Netz für einen Teilnehmer des zweiten Netzes auf, so übernimmt die Brücke die Meldung und speist diese entsprechend den MAC-Regeln in das zweite Netz ein. Hierbei werden ggf. auch Umwandlungen des Rahmenformats vorgenommen. Die Aufgaben einer Brücke werden nach der OSI-Modellierung der Schicht 2 zugeordnet.

In einer etwas weiter entwickelten Form beobachtet die Brücke die auftretenden Meldungen in beiden Netzen und lernt daraus, welche Teilnehmer an welchem Netz angeschlossen sind (*self learning bridge*). Änderungen im Netz werden also von der Brücke automatisch berücksichtigt - sie verhält sich in diesem Sinne adaptiv.

Werden mehrere Lokale Netze über Brücken miteinander verbunden, so entsteht das Problem, daß es zwischen zwei Teilnehmern am Netz mehrere Wege geben kann. Gewöhnlich wird dann ein kürzester Weg verwendet - wobei das Kriterium die Anzahl der überquerten Netze oder aber auch die erforderliche Übertragungszeit (die ja vom Verkehr abhängig ist) sein kann.

In einer Variante wird ein Baum (d. h. ein Graph ohne Maschen) über das gesamte Netz gespannt und die Zweige des Baumes als Routen verwendet (*spanning tree algorithm*). Die Routing-Tabelle wird auf diese Weise eindeutig festgelegt. Meist handelt es sich beim gewählten Baum um einen in einer definierten Weise optimalen Graphen. Dieses Verfahren wird häufig verwendet, um CSMA/CD-Netze miteinander zu vernetzen.

Source Routing

In einer anderen Variante werden zunächst Probemeldungen (*Explorer*) vom Sender an den Empfänger gesendet. Diese werden von den Brücken an die angeschlossenen Netze weitergeleitet, wobei die Brücken die Pakete mit der jeweiligen Brückenkennung markieren. Auf diese Weise gelangen mehrere Probemeldungen über verschiedene Wege zum Empfänger und enthalten Informationen über die jeweils verwendete Route. Vom Empfänger werden die Meldungen an den Sender zurückgeschickt. Dieser wertet die Information aus, legt die für die Datenmeldungen zu verwendende Route fest und trägt die Routinginformation in die Paketköpfe der Datenmeldungen ein. Diese Information wird von den Brücken ausgewertet und die Pakete werden entsprechend geroutet. Das Verfahren wird als **Source Routing** bezeichnet und wird für Token Ring Netze verwendet.

Brouter
Router

Um zu kennzeichnen, daß eine Bridge auch Routing-Information (OSI-Schicht 3) verarbeitet, wird sie gelegentlich als **Brouter** (Bridge-Router), der Routing-Information verarbeitende Teil als **Router**, bezeichnet. Leider werden die Bezeichnungen in der Literatur nicht einheitlich verwendet.

Häufig werden auch öffentliche Netze verwendet, um Verbindungen von einem Lokalen Netz zum Rechner an einem öffentlichen Netz oder zwischen zwei Lokalen Netzen herzustellen. Man spricht dann auch von **WANs** - ***W**ide **A**rea **N**etworks*.

WANs

Einrichtungen, die es ermöglichen Lokale Netze an andere Netze zu koppeln, werden als **Gateway** bezeichnet - hierbei wird unterstellt, daß die drei untersten Schichten des OSI-Modells (die Bitübertragungs-, die Sicherungs- und die Vermittlungsschicht) angepaßt werden. Typische Beispiele sind Ethernet-X.25 und Token Ring-ISDN Gateways.

Gateway

Der technologische Fortschritt, vor allem in der Mikroelektronik und in der Glasfasertechnik, hat dazu geführt, daß schnelle Lokale Netze - HSLANs (***H**igh **S**peed **LANs***) - mit Bitraten von 100 Mbit/s und darüber realisiert werden können. Sie werden eingesetzt, um schnelle Rechner und/oder Lokale Netze zu verbinden. Häufig bilden sie die zweite Hierarchie (***Backbone Network***) in großen privaten Netzen. Wir wollen im folgenden vier HSLANs - FDDI, DQDB, CRMA und ATMR - kennenlernen.

HSLANs

Backbone Network

FDDI (***F**iber **D**istributed **D**ata **I**nterface*) wurde ursprünglich von ANSI (***A**merican **N**ational **S**tandard **I**nstitute*) Task Group X3T9.5 spezifiziert und ist inzwischen als ISO IS 9314 weltweit genormt. FDDI ist ein HSLAN, das auf dem Token Ring Protokoll basiert und auf Glasfaser bei 100 Mbit/s realisiert wird. Es wird häufig zur Kopplung von Ethernet und Token Ring LANs verwendet (s. Abb. 3.4-1), eignet sich aber auch für die schnelle Back-End (Mainframe-to-Mainframe) und Front-End (Workstation-to-Workstation) Kopplung. Die große Ausdehnung (über 100 km) ermöglicht es auch, regionale Netze (**MANs** - ***M**etropolitan **A**rea **N**etworks*) zu bilden.

FDDI

MANs

Abb. 3.4-1: FDDI als Backbonenetz

Im FDDI sind zwei Arten von Stationen vorgesehen (s. Abb. 3.4-2): zweiseitig angeschlossene Stationen (**DAS** - ***D****ual* ***A****ttached* ***S****tations*) und einseitig angeschlossene Stationen (**SAS** - ***S****ingle* ***A****ttached* ***S****tations*). Entsprechend werden auch zwei Arten von Konzentratoren eingesetzt. Es ist möglich, mit diesen vier Elementen die unterschiedlichsten Strukturen zu realisieren, wobei zur Zeit der Dual Ring die häufigste Realisierung ist. Der Dual Ring wird so ausgelegt, daß im Störungsfall die Störstelle durch Kurzschluß ausgeschlossen wird und das Lokale Netz als einfacher Ring weiter betrieben werden kann (Abb. 3.4-3).

Abb. 3.4-2: FDDI-Stationstypen und mögliche Topologien

Abb. 3.4-3: Heilung beim dualen Ring

Im Gegensatz zum Token Ring ist beim FDDI die Dauer der Sendeberechtigung (**THT** - ***T**oken **H**olding **T**ime*) lastabhängig. Das Protokoll wird deshalb auch als *timed token rotation protocol* bezeichnet. In der Initialisierungsphase wird die kleinste unter den Stationen gewünschte Zyklusdauer (**TTRT** - ***T**arget **T**oken **R**otation **T**ime*) festgelegt. Jede Station mißt die aktuelle Zyklusdauer (TRT - ***T**oken **R**otation **T**ime*) und errechnet daraus die Dauer der Sendeberechtigung (THT) wie folgt

$$THT = TTRT - TRT \ . \qquad 3.4\text{-}1$$

Der Nachrichtenverkehr im Netz wird in zwei Kategorien aufgeteilt: synchron und asynchron[1]. Erhält eine Station einen freien Token, so sendet sie grundsätzlich die vorliegenden, zum Synchronverkehr gehörenden Meldungen ab. Die maximale Rahmenlänge beträgt stets 4500 Byte. Liegen auch Meldungen des asynchronen Verkehrs vor, so können sie abgesendet werden, solange die gesamte Sendedauer geringer als die berechnete THT ist. Ist die berechnete THT jedoch ≤ 0, so wird kein asynchroner Verkehr bedient, sondern die Sendeberechtigung weitergegeben. Das Protokoll garantiert auf diese Weise, daß die Verzögerungszeit für synchrone Meldungen unter einem festgelegten Wert bleibt.

Ferner wird beim FDDI mit der Tokenweitergabe nicht gewartet, bis die Meldung beim Sender wieder ankommt. Zwar ist wie beim Token Ring der Sender zuständig für die Abnahme des Rahmens vom Ring nach einem Umlauf, der Token wird jedoch unmittelbar am Ende des Rahmens weitergereicht. Da der Verzug im FDDI gewöhnlich mehrere tausend Bits beträgt, können zu einem festen Zeitpunkt auch mehrere Rahmen auf dem Ring sein, wenn dieser lang genug ist.

Die anderen FDDI-Abläufe, wie Prioritäten beim asynchronen Verkehr, der *Token Claim* Prozess, die Ein- und Ausgliederung usw., sind identisch oder ähnlich wie beim Token Ring. Auch eine gemischte Adressierung (2 und 6 Byte) ist möglich.

FDDI-II Die Weiterentwicklung von FDDI, als **FDDI-II** bezeichnet, sieht die Erweiterung für plesiochronen Verkehr vor. Es werden 16 Kanäle vom 6,144 Mbit/s (d. h. 3xPCM 30 bzw. 4xPCM 24) hierfür vorgesehen - etwa 1 Mbit/s verbleiben dann für das Tokenverfahren. Die Zuordnung des plesiochronen Verkehrs wird über ein dynamisches Reservierungsverfahren vorgenommen; die verbleibende Kapazität ist für das Tokenverfahren verfügbar.

FFOL Ein darüberhinausgehender Vorschlag wird als **FFOL** - ***F**DDI* - ***F**ollow-**o**n **L**AN* bezeichnet. FFOL zielt auf die Standardisierung eines neuen HSLANs der 3. Generation im Gbit/s-Bereich ab. Die Arbeiten werden im gleichen Gremium X3T9.5 (ANSI) wie zuvor für FDDI/FDDI-II durchgeführt [ROS 2].

DQDB Der ***D**istributed **Q**ueue **D**ual **B**us* **(DQDB)** wurde von der University of Western Australia und der Australian Telecom Mitte der 80er Jahre als HSLAN unter der

1 Die Bezeichnung synchron hier weist lediglich darauf hin, daß eine Echtzeitanforderung erhoben wird. Der Verkehr ist tatsächlich nicht synchron im Sinne der Definition im Abschnitt 9.2 des Kurses KT II

Bezeichnung QPSX (*Queued Packet and Synchronous Exchange*) konzipiert. Es wurde von der gleichnamigen neugegründeten Firma QPSX in Australien weiterentwickelt, 1987 zur Standardisierung vorgeschlagen und im Juli 1991 als IEEE 802.6 DQDB MAN Standard verabschiedet.

Wie die Bezeichnung DQDB bereits impliziert, handelt es sich um ein System mit einer verteilten Warteschlangenverwaltung, die wir noch kennenlernen werden, und einer dualen Busstruktur. Als HSLAN kann es mit 155 Mbit/s auf Glasfaser betrieben werden und sich z. B. über einige 100 km erstrecken. Die Norm sieht unterschiedliche Geschwindigkeiten (34, 140, 155 Mbit/s) und eine Unabhängigkeit vom Übertragungsmedium vor. Die Busse sind gegenläufig ausgelegt (Abb. 3.4-4) und jede angeschlossene Station hat Lese- und Schreibzugriff auf beide Busse. Der Schreibzugriff basiert auf einer logischen ODER-Verknüpfung, die so ausgelegt ist, daß Stationsausfälle den Betrieb nicht stören bzw. Stationen im laufenden Betrieb an- und ausgekoppelt werden können. Jeweils am Kopfende eines Busses erzeugt ein Generator die Rahmen in einem Takt von $125\mu s$. Die Rahmen sind mit ATM kompatibel - ein DQDB *Slot* entspricht einer ATM-Zelle (siehe Abb. 3.4-5). Die Rahmengenerierung kann von jeder Station übernommen werden. Wird der duale Bus als Ring ausgelegt (siehe Abb. 3.4-6), so kann deswegen im Störungsfall die Störungsstelle durch Kurzschluß ausgeschlossen und das System weiter betrieben werden.

Abb. 3.4-4: DQDB-Busarchitektur

Um die verteilte Warteschlangenverwaltung kennenzulernen, betrachten wir zunächst ein System, in dem keine Prioritäten vorgesehen sind. Jeder DQDB-*Slot* hat in diesem Fall ein *Busy* Bit das anzeigt, ob der *Slot* besetzt oder frei ist und eins, das aufzeigt, ob ein Sendewunsch vorliegt. Hat eine Station A eine Meldung für eine Station B (Abb. 3.4-4), so verwendet sie den Bus in Richtung B zur Übertragung der Meldung. Dieser Bus wird nachfolgend als "eigener Bus" bezeichnet. Zunächst meldet sie ihren Sendewunsch auf dem Bus in der entgegengesetzten Richtung. Hierzu wartet sie, bis ein nichtgesetztes *Request* Bit auf diesem Bus vorbeikommt und setzt dieses. Die verteilte Warteschlangenverwaltung besteht nun darin, daß jede Station (jeweils für jeden Bus) einen *Request*-Zähler hat. Dieser wird um Eins hochgesetzt, wenn ein Sendewunsch (auf dem entgegengesetzten Bus) ankommt und um Eins herabgesetzt, wenn ein freier *Slot* (d. h. *Slot* mit nicht gesetztem *Busy* Bit) auf dem eigenen Bus vorbeigeht. Der *Request*-Zähler gibt somit an, wieviele Sendewünsche in Abwärtsrichtung an einem Bus noch warten (= angemeldete Sendewünsche - freie *Slots*, die zur Verfügung standen). Sobald eine Station eine Meldung abzuschicken hat, kopiert sie den Zählerstand in einen zweiten (*Countdown*-) Zähler. Nun wird für jeden leeren *Slot*, der auf dem eigenen Bus vorbeigeht, auch der *Countdown*-Zähler um Eins herabgesetzt. Wenn der *Countdown*-Zähler den Wert Null erreicht, wird der nächste leere *Slot* als *busy* gekennzeichnet und verwendet, um die Meldung abzusetzen. Der *Request*-Zähler wird während dieser Zeit weitergeführt. Sein Wert dient entsprechend für die nächste Meldung. Die beschriebene Zählerführung weicht geringfügig von der Normung ab, vereinfacht jedoch das Verfahren etwas.

ACF	Access Control Field	REQ	Request
CHn	Channel n	RES	Reserved
HCS	Header Check Sequence	SPR	Segment Priority
PLT	Pay Load Type	VCI	Virtual Channel Identifier
PSR	Previous Slot Received		

Abb. 3.4-5: DQDB-Rahmenstruktur nach IEEE 802.6

Abb. 3.4-6: Heilungsvorgang beim geschleiften Dualen Bus

Es ist nicht erforderlich, daß alle *Request*-Zähler eines Busses den gleichen Wert anzeigen - eine relative Zählung an jeder Station genügt um einen geregelten Betrieb zu gewährleisten. Deshalb ist auch eine Ein- und Ausgliederung von Stationen ohne weiteres mit einem beliebigem Initialwert des Zählers möglich.

Das Verfahren ist besonders effizient bei hoher Last, denn dann kann praktisch jeder *Slot* besetzt, die Buskapazität also voll ausgeschöpft werden. Das Protokoll in der bisher betrachteten Form hat den Nachteil, daß es nicht fair ist - d. h. es werden nicht alle Stationen gleich behandelt. Einerseits sind die Stationen in der Busmitte insofern bevorzugt, als sie im Mittel einen geringen Abstand zu anderen Stationen haben als Stationen am Busrand. Andererseits sind Stationen am Busrand insoweit im Vorteil, als sie sich unmittelbar am Rahmengenerator befinden und entsprechend unverhältnismäßig viele leere *Slots* erhalten. Bei entsprechend großer Entfernung und niedriger Belastung überwiegt der erste Einfluß. Um Fairneß zwischen den Stationen zu erzwingen, wird ein sogenannter Bandwidth Balancing Algorithmus angewandt. In Abhängigkeit der erfolgreich abgesetzten Meldungen werden die Stationen (durch Erhöhung der *Request*- und *Countdown-Zähler*) jeweils angehalten, freie *Slots* passieren zu lassen.

Die Verwendung von Prioritäten bei DQDB erfordert, daß für jede Prioritätsklasse gesonderte *Request*- und *Countdown-Zähler* geführt werden. Der *Request-Zähler*

einer Priorität wird um Eins heraufgesetzt, wenn ein Sendewunsch der gleichen oder höheren Priorität gemeldet wird; er wird beim Passieren eines leeren *Slots* jeweils um Eins herabgesetzt. Auch der *Countdown*-Zähler einer Prioritätsklasse wird um Eins heraufgesetzt, wenn ein Sendewunsch mit höherer Priorität ankommt. Dies ermöglicht, die höher priorisierte Meldung vor der wartenden Meldung der niedrigen Priorität zu bedienen.

Wie bei ATM müssen Nachrichten, die länger als 48 Byte sind, segmentiert werden. Die Struktur der DQDB-*Slots* (Abb. 3.4-5) entspricht der ATM-Zelle, so daß wir die einzelnen Elemente hier nicht betrachten wollen, sondern erst im Kapitel 1 in Kommunikationsnetze und Protokolle II.

Das DQDB-Protokoll sieht vor, außer dem bisher betrachteten asynchronen (Paket-) Verkehr auch plesiochronen Verkehr zu bedienen. Hierzu wird eine Reservierungsstrategie angewandt. Die jeweils mit $125\mu s$ zyklisch auftretenden 8 Bit (d. h. 1 Byte) des Segment-Payloads werden zu einem 64 kbit/s Kanal zusammengefaßt und für eine plesiochrone Verbindung verwendet. Die Reservierung wird über VCI gesteuert. Die entsprechenden *Slots* werden vom Rahmengenerator als besetzt gekennzeichnet und stehen dem DQDB-Zugriffsprotokoll nicht zur Verfügung. Im folgenden betrachten wir ein weiteres **Reservierungsverfahren**bei einem HSLAN (dem CRMA) im Detail.

Reservierungsverfahren

Beispiel 3.4-1: Station an einem DQDB-Bus
Wir betrachten eine Station eines DQDB-Busses. Die momentanen Zählerstände in einer Station sind in der folgenden Skizze dargelegt:

Abb. 3.4-7: Station eines DQDB-Busses

Folgende Meldungen (R = Reserve, B = Busy) kommen in der angegebenen Reihenfolge an den Bussen (A und B) vorbei.

ankommende Meldungen in der Station	Bus A B=	Bus B R=
	0	0
	0	1
	0	0
	1	1
	1	1
Meldung liegt zur Übertragung an der Station an		
	0	1
	1	0
	0	1
	0	1
	1	1

Wir betrachten die Request- und Countdown- Counter und geben an, wann die Meldung abgeschickt werden kann.

Meldung/Aktion in Station	Bus A B=	Bus B R=	Request Counter	Countdown Counter
			2	0
	0	0	1	0
	0	1	1	0
	0	0	0	0
	1	1	1	0
	1	1	2	0
Meldung kommt an			2 →	2
	0	1	2	1
	1	0	2	1
Request wird abgesetzt		R=1		
	0	1	2	0
	0	1	2	0
Meldung wird abgesetzt	B=1			
	1	1	3	0

***Cyclic Reservation Multiple Access* CRMA**

Das ***Cyclic Reservation Multiple Access*** (**CRMA**)-Verfahren wurde im IBM Zürich Research Laboratory in Rüschlikon in der Schweiz entwickelt und 1990 vorgestellt [NAS]. Die zyklische Reservierung ermöglicht, daß ganze Nachrichten ohne Sequenzierung und Reassemblierung übermittelt werden können. Hierdurch wer-

den Wartezeiten vermieden, die wie bei DQDB dadurch entstehen, daß in den Stationen nur eine Zelle zu einer Zeit behandelt wird. Das Verfahren ist für HSLANs, die sich über mehrere 100 km erstrecken und im Gbit/s-Bereich betrieben werden, konzipiert.

Abb. 3.4-8: CRMA-Busstruktur: Gefalteter Bus mit Reservierungsaufruf
Reservierungen bei k:4, c:11, b:0, a:2

Abb. 3.4-9: CRMA-Rahmenstruktur

Es wird, wie bei DQDB, die duale Busstruktur verwendet, die in der vereinfachten Version und an einem Ende als gefalteter Bus ausgeführt werden kann (Abb. 3.4-8). Der Bitstrom wird in *Slots* fester Länge unterteilt. Diese bestehen aus einem Kopf (**ACF** - ***A**ccess **C**ontrol **F**ield*) und einem Segment ähnlich wie bei DQDB (s. Abb. 3.4-9). Das *Command*-Feld im ACF dient der Signalisierung. Hierüber werden

die Reservierungen vorgenommen und bestätigt. Das *Busy* Bit zeigt an, ob ein *Slot* besetzt oder frei ist. Die Kopfstation übernimmt zentral die Verwaltung der Warteschlange der angemeldeten Reservierungen und erzeugt die *Slots* und die Rahmen, die Zyklen genannt werden und eine variable Länge haben. Sie kennzeichnet den Zyklus-*n*-Beginn durch Setzen des *Start*-Zeichens im Kommandofeld und numeriert den Zyklus (Modulo $2^8 - 1$). In bestimmten Abständen sendet sie einen Reservierungsaufruf, der am Busende zurückgeschleift wird. Dieser enthält die Nummer *n* des Zyklus, für den die Reservierung vorgenommen wird und eine Zahl, die angibt, wieviele *Slots* für den Zyklus reserviert werden.

Wenn eine Station den Reservierungsaufruf auf dem Rückweg erhält, erhöht sie diese Zahl um die Anzahl der *Slots*, die sie benötigt. Die Kopfstation erfährt auf diese Weise, wieviele *Slots* für den Zyklus n insgesamt reserviert werden. Sie verwaltet eine Warteschlange, in der angemeldete Zyklen nach dem FIFO-Prinzip abgefertigt werden. Hat diese Warteschlange eine kritische Obergrenze (gemessen an der Zahl der angemeldeten *Slots*) nicht überschritten, so bestätigt die Kopfstation die Reservierung und übernimmt den Zyklus in die Warteschlange. Ist der kritische Wert erreicht, schickt sie eine *Reject*-Meldung für den Zyklus *n*; die Stationen erfahren dadurch, daß eine Überlast herrscht und eine Reservierung für sie nicht durchgeführt wurde. Erhält eine Station in einem Zyklus *Slots*, die nicht als *Busy* gekennzeichnet sind, so darf sie soviele *Slots* belegen, wie sie für den Zyklus angemeldet hat. Da der Zyklus am Kopfende generiert wird und die Stationen eine feste Reihenfolge haben, erhält auch jede Station mindestens die angemeldete Anzahl von leeren *Slots* hintereinander weg. Eine größere Nachricht kann also auf mehrere, hintereinander liegende *Slots* einfach aufgeteilt werden. Dies ist ein wesentlicher Vorteil des CRMA-Protokolls.

Das Verfahren hat den Nachteil, daß jede Nachricht mindestens die Phase Reservierung und deren Bestätigung einschließlich der Signallaufzeit abwarten muß, bis sie gesendet werden kann. Um Nachrichten mit hohen Echtzeitanforderungen diese Wartezeit zu ersparen, werden auch *Slots* erzeugt, die nicht reserviert wurden. Die reservierten *Slots* werden durch das Allocation Bit 1 (*Reserved*), die nicht reservierten durch 0 (*Gratis*) gekennzeichnet. Die nicht reservierten *Slots* werden stets erzeugt, wenn am Kopfende keine Reservierungen vorliegen. Eine Station, die eine Nachricht in nicht reservierten *Slots* übertragen möchte, versucht die ganze Nachricht in eine Folge von nicht reservierten, freien *Slots* zu übertragen. Gelingt dies nicht, muß die Nachricht erneut übertragen werden. Hierdurch wird erzielt, daß kein Aufwand für Segmentierung und Reassemblierung erforderlich ist.

Das CRMA Verfahren ermöglicht leicht, Prioritäten einzuführen. Hierzu werden in der Kopfstation getrennte Warteschlangen für unterschiedliche Prioritäten gebildet. Die reservierten Zellen erhalten entsprechende Prioritätsangaben im ACF; 2 Bits sind hierfür vorgesehen. Am Kopfende werden die Warteschlangen nach den Prioritäten abgearbeitet. Es ist also zulässig, daß *Slots* mit hoher Priorität gegenüber solchen mit niedriger Priorität und niedriger Zyklusnummer vorgezogen werden. Die Realisierung der verdrängenden Prioritätsbehandlung führt dazu, daß Nachrichten einer Priorität durch die einer höheren Priorität unterbrochen werden. Nachdem die

höher priorisierte Nachricht abgesendet wurde, wird die verdrängte Nachricht an der unterbrochenen Stelle weiter gesendet. Da beim Empfänger getrennte Speicher für unterschiedliche Prioritäten vorgesehen werden, ist auch in diesem Falle eine Sequenzierung und Reassemblierung nicht erforderlich.

Der Betrieb eines CRMA HSLAN kann über mehrere Parameter optimiert werden. Hierzu gehören die Abstände zwischen den Reservierungsanforderungen, die kritische Warteschlangenlänge am Kopfende (bei der *Reject*-Meldungen ausgelöst werden), wie häufig nicht reservierte *Slots* erzeugt werden und wie sich die Stationen im Überlastfall verhalten.

CRMA-II

Eine Erweiterung des Verfahrens, **CRMA-II** genannt, wurde 1991 vorgestellt [AS]. Es unterstützt vier Topologien: einfacher Ring, dualer Ring, gefalteter Bus und dualer Bus. Eine Station, *Scheduler* genannt, übernimmt die Reservierung der *Slots*. Jede Station kann diese Funktion übernehmen. Die Empfänger löschen die an sie gerichteten Nachrichten und geben die belegten *Slots* wieder frei. Diese werden als nicht reservierte *Slots* (*gratis Slots*) weitergegeben und können durch Folgestationen wiederverwendet werden. Hierdurch wird die Nutzung der verfügbaren Kanalkapazität erheblich gesteigert.

Auf dem HSLAN kann man einerseits zwischen freien (F) und besetzten (B) *Slots*, andererseits zwischen *gratis* (G) und reservierten (R) *Slots* unterscheiden. Man hat die Unterscheidung FG, FR, BG, BR. Der *Scheduler* teilt jeder Station eine bestimmte Anzahl von *Slots* zu. Die Anzahl wird für jede Station individuell nach einer festgelegten Zuteilungsstrategie bestimmt. Verschiedene Zuteilungsstrategien können realisiert werden. Jede Station führt zwei Zähler: *Confirm Counter c* und *Gratis Counter g*. Die einer Station zugeteilte Anzahl von *Slots* wird in c eingetragen. Hat eine Station Nachrichten zu übertragen und ist $c > 0$, so darf sie sowohl gratis als auch reservierte *Slots*, die frei sind, verwenden. Ist $c = 0$, so darf sie nur freie gratis *Slots* verwenden. Bei der Verwendung eines reservierten *Slots* wird FR zu BG gesetzt und c um eins erniedrigt. Bei der Verwendung eines *gratis Slots* wird FG zu BG gesetzt und g wird um Eins erhöht. Hat eine Station keine Nachrichten zu übertragen, so wird FR zu FG gesetzt und c um Eins erniedrigt; dies dient zum Abbau der zugeteilten *Slots*, die nicht benötigt werden. Beginnt eine Station eine Nachricht zu senden, so sendet sie diese bis zu Ende; gegebenenfalls erzeugt sie neue gratis *Slots* und verzögert die ankommenden besetzten *Slots* entsprechend. Bei nächster Gelegenheit entfernt sie freie *Slots*, um die Verzögerung wieder auszugleichen. Wenn ein Reservierungsaufruf vom *Scheduler* kommt, melden die Stationen die Anzahl der im letzten Zyklus übertragenen und der noch wartenden *Slots* an den *Scheduler*. Der *Scheduler* verwendet diese Werte, um die nächste Reservierung vorzunehmen, d. h. einen neuen c-Wert für die Stationen festzulegen. g wird nach der Übertragung durch die Stationen zu Null gesetzt.

Wie bereits erwähnt, löscht der Empfänger die an ihn gerichtete Nachricht und gibt die besetzten *Slots* wieder frei, indem er BR zu FR bzw. BG zu FG setzt. Der *Scheduler* führt einen Zähler (*mark Counter* m) über die noch offenen Reservierungen. Ist $m > 0$, so setzt der *Scheduler* FG zu FR und BG zu BR und setzt m um Eins

herab. Ist $m = 0$, so werden die *Slots* unverändert weiter gereicht. Im Abb. 3.4-10 ist das erläuterte Zugriffsverfahren zusammengefaßt.

Abb. 3.4-10: CRMA-II Zugriffsprotokoll

In Abb. 3.4-11 ist die CRMA-II Rahmenstruktur dargelegt. Die *Slots* werden aus 32-Bit Einheiten (**ADU** - ***A****tomic* ***D****ata* ***U****nit*) aufgebaut, die stets als Einheit verarbeitet werden. Dies ermöglicht eine schnelle Abfertigung. Die *Start ADU* und *End ADU* bilden ein Paar. *Command Slots* werden verwendet, um Signalisierinformationen (Reservierung, Zuteilung, Zählerstände) auszutauschen; sie haben eine variable Länge. Mehrere *Slots* können zusammengesetzt werden, um eine lange Nachricht zu übertragen. In CRMA-II wird auch die Übertragung von ATM-Zellen unterstützt. Hierfür werden 17 (3 (*Start*, *Address*, *End*) + 14 *Payload*) ADUs benötigt - die Zellenlänge wird entsprechend festgesetzt.

Abb. 3.4-11: CRMA-II Rahmenstruktur

Beispiel 3.4-2: Station an einem CRMA-II-Ring

Wir betrachten eine Station, die an einem CRMA-II-Ring angeschlossen ist. Am Zyklusbeginn werden ihr 4 Slots zugeteilt; in der Warteschlange befinden sich 6 Slots zur Übertragung. Folgende Slots kommen an der Station in der gegebenen Reihenfolge vorbei: FR, FG, FG. 3 weitere einzelne Meldungen zum Übertragen kommen an. Weiter geht es mit FG, BR, FR, FR, FR, FR, BR, FG. Es kommen wieder 4 einzelne Meldungen und dann BG. Anschließend kommt ein Request an. Wir wollen uns die Zähler näher ansehen.

Zyklusbeginn, Zugeteilt c = 4, Gratis g = 0, Warteschlange w = 6.

	FR → BG	c = 3	g = 0		w = 5
	FG → BG	c = 3	g = 1		w = 4
	FG → BG	c = 3	g = 2		w = 3
→ 3 Meldungen					w = 6
	FG → BG	c = 3	g = 3		w = 5
	BR → BR	c = 3	g = 3		w = 5
	FR → BG	c = 2	g = 3		w = 4
	FR → BG	c = 1	g = 3		w = 3
	FR → BG	c = 0	g = 3		w = 2
	FR → FR	c = 0	g = 3		w = 2
	BR → BR	c = 0	g = 3		w = 2
	FG → BG	c = 0	g = 4		w = 1
→ 4 Meldungen					w = 5
	BG → BG	c = 0	g = 4		w = 5
1 Request → Verschickt g = 4, w = 5; es wird reinitialisiert g = 0.					

Abb. 3.4-12: Station an einem CRMA-II-Ring

Der Scheduler erhält für die Station g = 4, w = 5 und könnte z. B. die Vorgabe haben, daß pro Station und Zyklus ein Durchsatz von 5 vorgesehen ist. 4 gratis Slots wurden allerdings genutzt, so daß sie nur noch einen Slot reserviert. Der nächste Zyklus für die Station beginnt somit mit c = 1, g = 0 und w = 5.

ATM Ring

Der **ATM Ring** wurde als HSLAN zur Bildung von ATM Metropolitan Area Networks bei NTT in Japan konzipiert und im Oktober 1990 als Spezifikation zur Normung vorgelegt [JTC]. Das ATM-Zellenformat wird konsequent sowohl für den Medienzugriff als auch für die Übermittlung verwendet.

Hierdurch werden Formatumsetzungen vermieden. Wie bei der ATM-Technik wird sowohl verbindungsorientierte, als auch verbindungslose Datenübermittlung unterstützt. Das ATM-Ring-Zellenformat ist in Abb. 3.4-13 erstellt. Das ACF wird für den Medienzugriff verwendet. Der RVCI (***R**ing **V**irtual **C**hannel **I**dentifier*) zeigt die Zieladresse an und entspricht dem VPI/VCI im ATM-Format. Er besteht aus der Stationsadresse (**AUA** - ***A**ccess **U**nit **A**ddress*) und der Kanaladresse (**LCA** - ***L**ogical **C**hannel **A**ddress*). Die Adresse Null zeigt eine leere Zelle an. Die restlichen Felder sind identisch mit denen beim ATM-Format. Der Empfänger einer Zelle löscht ihren Inhalt der Zelle und gibt sie somit wieder frei. Das Monitorbit wird wie beim Token Ring verwendet, um Mehrfachkreisen von Zellen zu unterbinden.

ACF for 3 Properties:

```
M 0 0 0 0 0 0 0  Busy Address
M 0 0 0 0 0 0 1  Busy Address
M 0 0 0 0 0 1 0  Busy Address
...   ...   ...   ...
M 1 1 1 1 0 1 1  Busy Address
M 1 1 1 1 1 0 0  Reset Level 4
M 1 1 1 1 1 0 1  Reset Level 3
M 1 1 1 1 1 1 0  Reset Level 2
M 1 1 1 1 1 1 1  Reset Level 1

RVCI  0 0 0 ... 0 0 0 Empty Cell
```

CLP	Cell Loss Priority
RVC	Ring Virtual Channel
M	Monitor Bit
R	Reserved

Abb. 3.4-13: ATM-Ring-Zellenformat

In der Grundversion wird der ATM-Ring als Doppel-Ring ausgeführt; die Ringe werden gegenläufig nach dem gleichen Protokoll betrieben. Wir betrachten deshalb nur einen Ring, um das Zugriffsprotokoll kennenzulernen. Dieses basiert auf einem Kredit-Prinzip. Jede Station erhält eine bestimmte Anzahl von Zellen als Guthaben am Anfang eines Zyklus, der durch eine Initialisierungsmeldung ausgelöst wird. Jede Station darf leere Zellen, die bei ihr vorbeikommen, verwenden bis ihr Guthaben ausgeschöpft ist. Danach geht sie in den Ruhezustand. Das Guthaben wird auch als Fenster (*Window*) bezeichnet. Der Anfangswert für dieses Fenster kann für verschiedene Stationen unterschiedlich sein - je nachdem wieviel Kapazität sie benötigen bzw. welchen Dienst sie benutzen. Eine Station, die keine Daten zu übertragen hat, ist ebenfalls im Ruhezustand. Sind nun alle Stationen im Ruhezustand, so wird der nächste Zyklus eingeleitet. Das ACF wird verwendet, um den Ruhezustand am Ring zu ermitteln. Jede Station, die nicht in Ruhe ist, überschreibt das ACF aller Zellen (außer *Reset*-Meldungen), die sie passieren, mit ihrer Identität. Erhält eine Station eine Zelle mit ihrer eigenen Identität im ACF zurück, so stellt sie fest, daß keine Station außer ihr aktiv ist (Abb. 3.4-14). Sie veranlaßt die Reinitialisierung

(*Reset*) durch Senden einer Adresse im ACF, die hierfür reserviert ist. Alle Stationen erhalten diese Meldung, nacheinander und setzen ihr Guthaben wieder auf den Initialwert. Die Station, die den *Reset* ausgelöst hat, erfährt dies als erste und kann weitersenden. Sie nimmt eine *Reset*-Meldung wieder vom Bus herunter, wenn diese bei ihr vorbeikommt.

Abb. 3.4-14: Feststellung der Aktivität auf dem Ring
C stellt über die Adresse fest, daß nur noch sie aktiv ist.

Um im ATM-Ring Prioritäten zu realisieren wird an jeder Station pro Priorität ein Fensterzähler (*Window Counter*) und eine Zeitüberwachung (*Queue Timer*) eingeführt. Der Fensterzähler zeigt das Restguthaben der Station für eine Priorität in Zellen an; die Zeitüberwachung die Restzeit in Zellen bis zur Überschreitung der zulässigen Verzögerung für die Priorität. Ferner wird pro Priorität ein weiterer *Reset* definiert. Hierfür werden entsprechend viele Adressen im AFC reserviert (Abb. 3.4-13). Wir wollen im folgenden drei Prioritäten betrachten, um das Verfahren kennenzulernen. Bei drei Prioritäten hat man pro Station drei Fensterzähler, drei Zeitüberwachungen und vier *Resets* (der *Reset*, den wir zuerst kennenlernten, der gesetzt wird, wenn nur noch eine einzige Station aktiv ist, wird als *Reset* Nr. (höchste Priorität + 1), d. h. in unserem Falle *Reset* 4 bezeichnet). Nach einem *Reset* der Priorität *i*, den eine Station beobachtet bzw. selbst auslöst, befindet sich der Ring von ihr aus gesehen im Zustand *i*. Im Zustand 1 dürfen nur Meldungen der Priorität 1, im Zustand 2 die der Priorität 1 und 2, in Zustand 3 und Zustand 4 die aller Prioritäten gesendet werden. Beim Erhalt bzw. Auslösen des *Reset* 4 werden die Fensterzähler der Prioritäten 1, 2, 3 und Timer der Prioritäten 1, 2, 3 auf ihren Initialwert zurückgesetzt. Beim Erhalt bzw. Auslösen des *Reset* 3 werden die Zähler 1, 2 und Timer 1, 2 zurückgesetzt. Beim Reset 2 werden Zähler 1 und Timer 1 zurückgesetzt. Der Sachverhalt ist in Tabelle 3.4-1 tabellarisch zusammengefaßt.

Tab. 3.4-1: *Zustände und Auswirkungen von Resets beim ATM-Ring*

Im Zustand i dürfen	Meldungen der Priorität k gesendet werden
$i = 1$	$k = 1$
$i = 2$	$k = 1, 2$
$i = 3$	$k = 1, 2, 3$
$i = 4$	$k = 1, 2, 3$

Reset	Erwirkt Übergang zum Zustand	Fensterzähler, die neu initialisiert werden	Timer, die neu initialisiert werden
Reset 1	Zustand 1	keine	keine
Reset 2	Zustand 2	Fenster 1	Fenster 1
Reset 3	Zustand 3	Fenster 1 und 2	Fenster 1 und 2
Reset 4	Zustand 4	Fenster 1, 2 und 3	Fenster 1, 2 und 3

Eine Station, deren *Timer* der Priorität *i* ausläuft, löst den *Reset* der Priorität *i* aus. Laufen mehrere *Timer* gleichzeitig aus, wird der Reset der höchsten Priorität verwendet. Der Initialwert eines *Timers* für eine Priorität wird so gewählt, daß er beim verwendeten Dienst bzw. erwarteter Bitrate ausläuft, wenn die maximal zulässige Verzögerung erreicht wird. Das Auslösen des *Resets* einer Priorität löst Maßnahmen aus, die darauf hinwirken, daß weitere Verzögerungen für diese und höhere Priorität vermieden werden. Bei wartenden Meldungen und Auslaufen des *Timers* einer Priorität wird für diese Meldungen allerdings die maximal zulässige Verzögerung trotzdem überschritten - die Maßnahmen reichen also nicht aus. Deshalb wird in

jeder Station ein Schätzwert gebildet, ob die zulässige Verzögerung überschritten wird. Ist für eine Priorität dieser Schätzwert

$$s = \frac{\text{min(Restguthaben, Wartende Zellen)}}{\text{Initialguthaben + Sicherheitsfaktor}} - \frac{\text{Restzeit im Timer}}{\text{Initialwert Timer}} > 0,$$

so wird bereits der *Reset* für diese Priorität ausgelöst.

Wie bereits erwähnt, existieren beim ATM-Ring mehrere Parameter (z. B. Initialwerte), die eingestellt werden und eine Möglichkeit zur Optimierung des HSLAN-Betriebs bieten. Zur Zeit wird das Lastverhalten des ATM-Rings unter verschiedenen Bedingungen untersucht.

Die Entwicklung von HSLANs geht von den deterministischen Tokenverfahren über Reservierungsverfahren hin zu Guthabenverfahren mit Last- und Verzögerungsschätzung. Es ist abzusehen, daß diese Verfahren erfolgreich weiterentwickelt werden.

Beispiel 3.4-3: Station an einem ATM-Ring
Wir betrachten eine Station an einem ATM-Ring mit drei Prioritäten. Die Initialwerte der Fenster und Timer sind $W_1 = 1$, $T_1 = 3$, $W_2 = 2$, $T_2 = 6$, $W_3 = 3$, $T_3 = 9$, $S_1 = S_2 = S_3 = 1$. Nach einem *Reset* 4 zum Zeitpunkt $T = 0$ stehen folgende Zellen in den Warteschlangen: $Q_1 = 1$, $Q_2 = 2$, $Q_3 = 3$. Zum Zeitpunkt $T = 3$ wird eine weitere Meldung der Priorität 1 erzeugt. Wir betrachten den Ablauf in der folgenden Tabelle.

T=0	Reset 4	$Q_1 = 1$	$W_1 = 1$	$T_1 = 3$	$S_1 = -1/2$
		$Q_2 = 2$	$W_2 = 2$	$T_2 = 6$	$S_2 = -1/3$
		$Q_3 = 3$	$W_3 = 3$	$T_3 = 9$	$S_3 = -1/4$
T=1		$Q_1 = 0$	$W_1 = 0$	$T_1 = 2$	$S_1 = -2/3$
		$Q_2 = 2$	$W_2 = 2$	$T_2 = 5$	$S_2 = -1/6$
		$Q_3 = 3$	$W_3 = 3$	$T_3 = 8$	$S_3 = -5/36$
T=2		$Q_1 = 0$	$W_1 = 0$	$T_1 = 1$	$S_1 = -1/3$
		$Q_2 = 1$	$W_2 = 1$	$T_2 = 4$	$S_2 = -1/3$
		$Q_3 = 3$	$W_3 = 3$	$T_3 = 7$	$S_3 = -1/36$
T=3		$Q_1 = 0$ oder +1	$W_1 = 0$	**$T_1 = 0$**	$S_1 = 0$
		$Q_2 = 0$	$W_2 = 0$	$T_2 = 3$	$S_2 = -1/2$
		$Q_3 = 3$	$W_3 = 3$	$T_3 = 6$	**$S_3 = 3/36 > 0$**
T=4	Reset 1	$Q_1 = 1$	$W_1 = 0$	**$T_1 = 0$**	$S_1 = 0$
		$Q_2 = 0$	$W_2 = 0$	$T_2 = 2$	$S_2 = -1/3$
		$Q_3 = 3$	$W_3 = 3$	$T_3 = 5$	**$S_3 = 7/36 > 0$**
T=5	Reset 4	$Q_1 = 0$	$W_1 = 0$	$T_1 = 2$	$S_1 = -2/3$
		$Q_2 = 0$	$W_2 = 2$	$T_2 = 5$	$S_2 = -5/6$
		$Q_3 = 3$	$W_3 = 3$	$T_3 = 8$	$S_3 = -5/36$
T=6		$Q_1 = 0$	$W_1 = 0$	$T_1 = 1$	$S_1 = -1/3$
		$Q_2 = 0$	$W_2 = 2$	$T_2 = 4$	$S_2 = -4/6$
		$Q_3 = 2$	$W_3 = 2$	$T_3 = 7$	$S_3 = -10/36$
T=7		$Q_1 = 0$	$W_1 = 0$	**$T_1 = 0$**	$S_1 = 0$
		$Q_2 = 0$	$W_2 = 2$	$T_2 = 3$	$S_2 = -3/6$
		$Q_3 = 1$	$W_3 = 1$	$T_3 = 6$	$S_3 = -15/36$
T=8	Reset 1	$Q_1 = 0$	$W_1 = 0$	**$T_1 = 0$**	$S_1 = 0$
		$Q_2 = 0$	$W_2 = 2$	$T_2 = 2$	$S_2 = -2/6$
		$Q_3 = 1$	$W_3 = 1$	$T_3 = 5$	$S_3 = -11/36$
	Reset 4				

Abb. 3.4-15: Station an einem CRMA-II-Ring

T = 0	Reset 4	$Q_1 = 1$	$W_1 = 1$	$T_1 = 3$	$S_1 = -1/2$
		$Q_2 = 2$	$W_2 = 2$	$T_2 = 6$	$S_2 = -1/3$
		$Q_3 = 3$	$W_3 = 3$	$T_3 = 9$	$S_3 = -1/4$
T = 1		$Q_1 = 0$	$W_1 = 0$	$T_1 = 2$	$S_1 = -2/3$
		$Q_2 = 2$	$W_2 = 2$	$T_2 = 5$	$S_2 = -1/6$
		$Q_3 = 3$	$W_3 = 3$	$T_3 = 8$	$S_3 = -5/36$
T = 2		$Q_1 = 0$	$W_1 = 0$	$T_1 = 1$	$S_1 = -1/3$
		$Q_2 = 1$	$W_2 = 1$	$T_2 = 4$	$S_2 = -1/3$
		$Q_3 = 3$	$W_3 = 3$	$T_3 = 7$	$S_3 = -1/36$
T = 3		$Q_1 = 0$ oder +1	$W_1 = 0$	$T_1 = 0$	$S_1 = 0$
		$Q_2 = 0$	$W_2 = 0$	$T_2 = 3$	$S_2 = -1/2$
		$Q_3 = 3$	$W_3 = 3$	$T_3 = 6$	$S_3 = 3/36 > 0$
T = 4	Reset 1	$Q_1 = 1$	$W_1 = 0$	$T_1 = 0$	$S_1 = 0$
		$Q_2 = 0$	$W_2 = 0$	$T_2 = 2$	$S_2 = -1/3$
		$Q_3 = 3$	$W_3 = 3$	$T_3 = 5$	$S_3 = 7/36 > 0$
T = 5	Reset 4	$Q_1 = 0$	$W_1 = 0$	$T_1 = 2$	$S_1 = -2/3$
		$Q_2 = 0$	$W_2 = 2$	$T_2 = 5$	$S_2 = -5/6$
		$Q_3 = 3$	$W_3 = 3$	$T_3 = 8$	$S_3 = -5/36$
T = 6		$Q_1 = 0$	$W_1 = 0$	$T_1 = 1$	$S_1 = -1/3$
		$Q_2 = 0$	$W_2 = 2$	$T_2 = 4$	$S_2 = -4/6$
		$Q_3 = 2$	$W_3 = 2$	$T_3 = 7$	$S_3 = -10/36$
T = 7		$Q_1 = 0$	$W_1 = 0$	$T_1 = 0$	$S_1 = 0$
		$Q_2 = 0$	$W_2 = 2$	$T_2 = 3$	$S_2 = -3/6$
		$Q_3 = 1$	$W_3 = 1$	$T_3 = 6$	$S_3 = -15/36$
T = 8	Reset 1	$Q_1 = 0$	$W_1 = 0$	$T_1 = 0$	$S_1 = 0$
		$Q_2 = 0$	$W_2 = 2$	$T_2 = 2$	$S_2 = -2/6$
		$Q_3 = 1$	$W_3 = 1$	$T_3 = 5$	$S_3 = -11/36$
	Reset 4				

Zum Zeitpunkt T = 3 ist sowohl $T_1 = 0$ als auch $S_3 > 0$. Reset 1, der höhere Priorität hat, wird deshalb ausgelöst. Da jedoch das Fenster $W_1 = 0$ bleibt, wird ein allgemeiner Reset ausgelöst. Die erste Station kann nun ihre Meldung bei $T = 5$ unmittelbar absetzen. Da am Anfang in der Priorität 2 eine hohe Last vorlag, erleiden die Meldungen der Priorität 3 eine hohe Verzögerung (zwischen T = 6 ... 8).

Selbsttestaufgabe 3.4-1:

a) Sowohl der ATM-Ring als auch CRMA-II können auf der Basis einer Doppelring-Struktur, in der beide Ringe in gegenläufiger Senderichtung betrieben werden, arbeiten. Beide Protokolle ermöglichen es, nach dem Empfang eines Slots diesen für die nachfolgenden Stationen wieder freizugeben (sog. Slot Reuse). Zum Senden einer Nachricht wird jeweils der Ring gewählt, auf dem der kürzere Weg zur Zielstation erreicht wird (Shortest Path Routing).

Wie oft kann ein Slot in dieser Anordnung während eines Ringumlaufs im Mittel wiederverwendet werden, wenn man annimmt, daß die Stationen auf dem Ring gleichverteilt an alle anderen Stationen senden ?

Wie groß ist dann die theoretisch maximal verfügbare Kanalkapazität einer solchen Anordnung ?

b) Was ist der wesentliche Unterschied zwischen der Übertragung ganzer Nachrichten im CRMA-II und im ATM-Ring ?

Welcher prinzipielle Unterschied beim Zugriff auf die im Ring kreisenden Slots im CRMA-II bzw. im ATM-Ring ermöglicht dieses unterschiedliche Übertragungsverhalten ?

c) Wie groß ist der zusätzliche Overhead beim Transport von ATM-Zellen in beiden Verfahren ?

4 ISDN - Das diensteintegrierende Kommunikationsnetz

4.1 Das ISDN-Referenzmodell

Wir haben bereits im Abschnitt 2.2.1 in Kurs KT I die Ansätze zur Digitalisierung der Netze und Integration der Dienste kennengelernt. Das auf diesen Ansätzen basierende Konzept des **ISDN** (***I**ntegrated **S**ervices **D**igital **N**etwork*) wurde erstmals 1984 als CCITT-Empfehlungen der I Serie verabschiedet. Es folgten zahlreiche nationale Empfehlungen (so z. B. 1TR6 der DBP), europäische Empfehlungen der **ETSI** (***E**uropean **T**elecommunication **S**tandards **I**nstitute*) und überarbeitete CCITT- Empfehlungen (1988 und 1992). Allein die CCITT-Empfehlungen von 1992 haben einen Umfang von mehreren tausend Seiten; die Grundstruktur dieser Empfehlungen ist in Abb. 4.1-1 dargelegt.

ISDN

ETSI

1.100	General ISDN Concept
	Structure of Recommendations
	Terminology , General Methods
1.200	Service Aspects of ISDN
	Principles
	Bearer Services , Teleservices
1.300	Network Aspects
	Functional Principles
	Reference Model , Numbering Addressing
	Routing, Performance
1.400	ISDN User-Network Interface
	Configurations , Structures and Capabilities
	Basic, Primary and Higher Rate Interfaces
	Layer 1, Layer 2, Layer 3 Specifications
	Support of Data Terminals
	Multiplexing
1.500	Internetwork Interfaces
1.600	Maintenance Principles

Abb. 4.1-1: Struktur der CCITT[2] ISDN-Empfehlungen

Basis des ISDN bildet das digitalisierte Fernsprechnetz; die Verbindungen im ISDN werden durchgehend digital geführt. Endgeräte werden über die beiden standardisierten Benutzer-Netz-Schnittstellen

- S_0 am **Basisanschluß** (**BA** - ***B**asic **A**ccess*) mit 144 kbit/s und

Primärmultiplexanschluß

- S_m am **Primärmultiplexanschluß** (**PRA** - ***P**rimary **R**ate **A**ccess*) mit 1984 kbit/s (31 B-Kanäle)

angeschlossen. Im B-ISDN (Breitband ISDN) sind weitere Anschlüsse mit 150 Mbit/s und 600 Mbit/s vorgesehen. Jeder Anschluß ist über eine Rufnummer, unabhängig von der Anzahl und Art der angeschlossenen Endgeräte, erreichbar. Geeignete Geräte werden nach einer Kompatibilitätsprüfung ausgewählt; einzelne Geräte können durch die Wahl einer weiteren Ziffer angesprochen werden.

Basisanschluß
B-Kanäle, D-Kanal

Am **Basisanschluß** (**BA** - ***B****asic* ***A****ccess*) sind zwei Kanäle mit je 64 kbit/s (**B-Kanäle**) und ein Kanal mit 16 kbit/s (**D-Kanal**) verfügbar. Alle drei Kanäle sind als vollduplex Kanäle ausgelegt. Die B-Kanäle stehen für Nutzinformationen (d. h. für die Abwicklung der Sprach-, Text- und Datendienste) zur Verfügung. Der D-Kanal steht in erster Linie für die Signalisierung zur Verfügung. Er ist für die Übermittlung paketierter Signalisiernachrichten ausgelegt und kann auch für Nutzsignale mit niedrigem Datenaufkommen (wie Telemetrie, Alarm- und Steuersignale) verwendet werden. In Abb. 4.1-2 ist eine typische Ausführung des Basisanschlusses dargestellt. Am Netzabschluß NT endet die Zuständigkeit des Netzbetreibers. Hier beginnt auch die vieradrige, busmäßig ausgelegte Haus- bzw. Büroverdrahtung (S-Bus). An diesem Bus können bis zu 8 Endgeräte an einheitlichen Steckdosen angeschaltet werden.

Primärmultiplexanschluß

Am **Primärmultiplexanschluß** (**PRA** - ***P****rimary* ***R****ate* ***A****ccess*) sind 30 B-Kanäle und ein D-Kanal verfügbar. Alle sind als vollduplex Kanäle mit 64 kbit/s ausgelegt. Die B-Kanäle sind wiederum als Nutzkanäle, der D-Kanal als Signalisierkanal konzipiert. Das Protokoll des D-Kanals für den Primärmultiplexanschluß ist im wesentlichen identisch mit dem Protokoll des Basisanschlusses. Er dient hier als zentraler Zeichengabekanal für 30 Nutzkanäle (anstatt für 2 wie beim Basisanschluß). Beim Primärmultiplexanschluß handelt es sich vom Prinzip her um eine 2 Mbit/s PCM 30 Strecke, in der der 16. Zeitschlitz für die D-Kanal Signalisierung verwendet wird. Er wird ausschließlich als vierdrähtige Punkt-zu-Punkt Verbindung ausgelegt und zur Zeit hauptsächlich für die Anschaltung von Nebenstellenanlagen eingesetzt.

Abb. 4.1-2: Beispiel für die Auslegung des Basisanschlusses (NT: Network Termination)

In Abb. 4.1-3 ist das CCITT ISDN-Referenzmodell für den Teilnehmeranschlußbereich wiedergegeben. Wie bei der ISO-Modellierung werden hier verschiedene logische Funktionsgruppen zusammengefaßt und durch Referenzpunkte getrennt. Referenzpunkte, an denen logische und physikalische Eigenschaften spezifiziert sind, bilden Schnittstellen.

ET	Vermittlungsabschluß (Exchange Termination)	TE 1	ISDN-Endeinrichtung (Terminal Equipment)
LT	Leistungsabschluß (Line Termination)		
NT 1, NT 2	Netzabschluß (Network Termination)	TE 2	herkömmliche Endeinrichtung
TA	Anpassungseinheit (Terminaladapter)		
R, S, T, U, V	Referenzpunkte		

Abb. 4.1-3: ISDN-Referenzkonfiguration

ISDN-Endeinrichtungen (**TE** - ***Terminal Equipment***) können direkt an die genannten Netzschnittstellen (S_0 für den Basisanschluß bzw. S_m für den Primärmultiplexanschluß) angeschlossen werden. Sie sind so ausgelegt, daß sie die entsprechende Schnittstelle zum Netz bedienen und die Protokolle sowohl zum Netz als auch

zum Kommunikationspartner hin behandeln können. Ferner sind in ihnen zahlreiche Wartungsfunktionen, die vom Netz aus bedient werden, implementiert (so z. B. das Schließen von Testschleifen, um einzelne Kanäle zu überprüfen).

Herkömmliche Endeinrichtungen (mit X.- und V.-Schnittstellen) können über Anpassungseinheiten (**TA** - ***T**erminal **A**dapter*) an das ISDN angeschlossen werden. Zwar können sie für die Abwicklung herkömmlicher Dienste über ISDN verwendet werden - ISDN-typische Eigenschaften können durch sie jedoch kaum genutzt werden. Terminal Adaptoren dienen also lediglich der Erhaltung der Kompatibilität für herkömmliche Endeinrichtungen.

Im Netzabschluß (**NT** - ***N**etwork **T**ermination*) sind zwei Funktionen vereinigt. Einerseits sind es die übertragungstechnischen Funktionen (NT 1) wie Leitungscodierung, Leitungsabschluß, Synchronisation, Schicht-1-Multiplexbildung, Stromversorgung, Schicht-1-Wartung usw. Andererseits sind es die vermittlungstechnischen Funktionen (NT 2) wie Behandlung der Schicht 2 und Schicht 3 Protokolle, Multiplexbildung auf Schicht 2 und Schicht 3, Verkehrskonzentration, Schicht-2- und Schicht-3-Wartungsfunktionen usw. Häufig werden die vermittlungstechnischen Funktionen nicht im Netzabschluß realisiert; die Referenzpunkte T und S fallen dann zusammen. Der Netzabschluß befindet sich beim Teilnehmer und wird über eine Leitung (Referenzpunkt U) mit der Vermittlungsstelle verbunden. Auf der Vermittlungsseite sind die übertragungs-technischen Funktionen entsprechend den NT-1-Funktionen im **Leitungsabschluß** (**LT** - ***L**ine **T**ermination*) zusammengefaßt. Die vermittlungstechnischen Funktionen sind im **Vermittlungsabschluß** (**ET** - ***E**xchange **T**ermination*) zusammengefaßt.

Leitungsabschluß

Vermittlungsabschluß

4.2 Die Bitübertragungsschicht

Im folgenden betrachten wir die Bitübertragungsschicht des Basisanschlusses und zwar sowohl auf der S-Schnittstelle (Hausverkabelung ab NT) als auch auf der U-Schnittstelle (zwischen NT und Vermittlungsstelle, siehe Abb. 4.1-2)

4.2.1 Die S-Schnittstelle

Für die Hausverkabelung werden gewöhnlich ungeschirmte, bündelverseilte Kupferkabel mit 0.6 mm Aderdurchmesser verwendet. Die Installation wird entweder als Punkt-zu-Punkt Konfiguration für ein Endgerät oder Punkt-zu-Mehrpunkt Konfiguration (passiver Bus) für mehrere Endgeräte (maximal 8) ausgelegt und reflektionsfrei abgeschlossen (siehe Abb. 4.2-1 und Abb. 4.2-2). Die Reichweite der Punkt-zu-Punkt Konfiguration ist durch die Kabeldämpfung begrenzt und kann bis zu 1000 Meter erreichen. Die Reichweite der Buskonfiguration ist durch die Umlaufverzögerung (vom weitest entfernten Endgerät bis zum NT) begrenzt und liegt meist unter 150 Meter.

Da alle Endgeräte bitsynchron senden, können Laufzeitunterschiede zu Bitverfälschungen führen. Eine direkte Kommunikation der Endgeräte untereinander über den Bus ist unabhängig von der Vermittlungsstelle (ggf. Vermittlungsfunktion im NT) nicht möglich.

Abb. 4.2-1: Die Auslegung der Hausverkabelung im ISDN: Punkt-zu-Punkt-Konfiguration

Abb. 4.2-2: Die Auslegung der Hausverkabelung im ISDN: Punkt-zu-Mehrpunkt-Konfiguration (passive Busstruktur)

Die S-Schnittstelle ist international genormt und stellt den angeschlossenen Endgeräten die beiden B-Kanäle mit 64 kbit/s und den D-Kanal mit 16 kbit/s für die Übermittlung der Nutz- bzw. Signalisierdaten zur Verfügung. Für die Bereitstellung dieser 144 kbit/s wird auf der Übertragungsstrecke eine Bitrate von 192 kbit/s erforderlich. Eine Bitfehlerrate besser als 10^{-5} wird zugesichert. Da für jede Übertragungsrichtung eine Doppelader zur Verfügung steht, ist eine einfache Übermittlung des anliegenden Bitstromes mit einem ternären Leitungscode möglich. Es wird der AMI-Code (siehe KT II, Abschnitt 8.3) mit einer einfachen Modifikation verwendet. Sie besteht darin, daß eine logische 1 in eine physikalische 0 und eine logische 0 alternierend in eine physikalische +1 bzw. -1 umgesetzt wird. Diese Modifikation bewirkt, daß im Ruhezustand, wenn eine logische Null anliegt, auf der Leitung physikalisch eine alternierende ± 1 Folge gesendet wird - damit bleibt der Bittakt auf der Leitung erhalten. Die Signalamplitude beträgt 750 mV (Null-Spitze).

Beispiel 4.2-1: Modifizierte AMI-Codierung

Abb. 4.2-3: Modifizierte AMI-Codierung

In Abb. 10 bis Abb. 4.2-7 ist die Rahmenstruktur der Schicht 1 an der S-Schnittstelle dargelegt. Der Rahmen besteht aus 48 Bit, die in $250\mu s$ übertragen werden. Der Rahmen in Richtung Endgerät zu NT weist einen Versatz von 2 Bit gegenüber dem Rahmen in Richtung NT zum Endgerät auf.

Die Lage der beiden B-Kanäle und des D-Kanals ist aus Abb. 10 ersichtlich. Pro Rahmen liegen zwei 8-Bit Wörter pro B-Kanal an. Dies entspricht der PCM Abtastrate von 8 kHz mit 8-Bit Quantisierung (= 64 kbit/s).

Die Synchronisation wird nach dem **Master-Slave**-Prinzip abgewickelt - die Vermittlungsstelle gibt den Takt an. Die Bitsynchronisation beim Endgerät wird über den modifizierten AMI-Code abgeleitet. Die Rahmensynchronisation wird durch das Herbeiführen einer Codeverletzung erzielt. Das Rahmenbit F wird so gesetzt, daß die AMI-Coderegel, daß ± 1 stets alternieren, verletzt wird (Abb. 4.2-5). Eine weitere Codeverletzung wird durch die Codierung der ersten logischen Null, die dem ersten L-Bit des Rahmens folgt, erzeugt. Diese wird spätestens durch das Setzen des F_A-Bits erwirkt. Diese weitere Codeverletzung sichert die Rahmensynchronisation ab; sie dient auch dazu, die laufende digitale Summe RDS, die durch die erste Codeverletzung erhöht wurde, wieder herabzusetzen. Die F_A, N- und S- Bits können auch zur Überrahmenbildung verwendet werden - in den Netzen der europäischen Verwaltungen wird dies in der Regel nicht vorgenommen.

Die folgenden Abbildungen zeigen die Rahmenstruktur der Schicht 1 an der S-Schnittstelle.

In beiden Richtungen
2x8 Bit B_1-Kanal = 64 kbits/s
2x8 Bit B_2-Kanal = 64 kbit/s
4 Bit D-Kanal = 16 kbit/s

2 Bit Rahmenversatz (Verzögerung im TE)

Abb. 4.2-4: Die Lage der B- und D-Kanäle

F-Bit: Rahmenkennungsbit
Codeverletzung zeigt den Beginn eines Rahmens an .
Nächste "0" wird in jedem Fall als negativer Impuls übertragen .
F_A -Bit: Zusätzliche Codeverletzung sichert Rahmenkennung ab .
N-Bit: Festgelegt auf "1" .
S-Bit: Werden zur Zeit nicht verwendet .

Abb. 4.2-5: Rahmensynchronisation

NT→TE ein Ausgleichsbit L für alle Kanäle.

TE→NT ein Ausgleichsbit L je Kanal.

TEs belegen nur Teile des Rahmens.

Abb. 4.2-6: Die Sicherstellung der Gleichstromfreiheit

D-Kanalbit des TE wird im E-Kanal des NT reflektiert.
E-Kanalbit wird vor nächstem D-Kanalbit empfangen.
A-Bit kennzeichnet Abschluß der Aktivierungsprozedur.
Nur NT→TE

Abb. 4.2-7: Der D-Echokanal und das Aktivierungsbit

Beispiel 4.2-2: Codeverletzung zur Synchronisierung
Im folgenden betrachten wir zwei Beispiele, um die Codeverletzung durch ein Bit des Nutzkanals B bzw. durch das Zusatzbit F_A kennenzulernen.

1. Codeverletzung durch ein B-Bit:

   Ein Rahmen endet mit der AMI-Folge `0-0+-+0`. Das `F`-Bit `+` ergibt die Codeverletzung am Rahmenanfang, das darauffolgende `L`-Bit – sorgt für die Gleichstromfreiheit. Die vorliegende binäre Nutzfolge `11010100` wird als `00-0+0-+` codiert, um die zweite Codeverletzung bei der ersten `0` zu ergeben.

Abb. 4.2-8: Codeverletzung durch B-Bit

2. Codeverletzung durch F_A-Bit.

   Ein Rahmen endet mit `0-0+000`. Das `F`-Bit `+` ergibt wieder die Codeverletzung am Rahmenanfang, das `L`-Bit `-` sorgt für die Gleichstromfreiheit. Die vorliegende Nutzbitfolge besteht aus einer Einsfolge; auch die `E`-, `D`- und `A`- Bits sind Eins. Die zusätzliche Codeverletzung wird in diesem Fall durch F_A erzeugt.

Abb. 4.2-9: Codeverletzung durch ein F_A-Bit.

Wir haben bei der Leitungscodierung (KT II,Abschnitt 8.1) gesehen, daß eine Gleichstromfreiheit des laufenden Bitstromes gewünscht ist. Dies wird durch das Setzen des `L`-Bits, so daß die laufende digitale Summe RDS Null wird, erreicht. In Richtung NT zu TE wird ein `L`-Bit zum Ausgleich des `F`-Bits und ein weiteres `L`-Bit zum Ausgleich des restlichen Rahmens erforderlich. Da die unterschiedlichen Kanäle in Richtung TE zu NT von unterschiedlichen Endeinrichtungen (TEs) genutzt werden können, ist ein Gleichstromausgleich pro Kanal erforderlich. Die L-Bits werden von den jeweiligen Endgeräten gesetzt. Die gleichstromfreien Teile des Rahmens sind in Abb. 4.2-10 durch Punkte (`L.`) angedeutet.

Beispiel 4.2-3: Gleichstromfreiheit durch Ausgleichbits

Abb. 4.2-10: Gleichstromfreiheit durch Ausgleichbits

Power Down Mode

Besteht kein Kommunikationsbedarf, so wird der Basisanschluß in den Ruhezustand (*Power Down Mode*) versetzt. In diesem Zustand werden keine Signale auf den Leitungen gesendet. Alle Schaltungen, bis auf die Schaltungen, die zur Erkennung von Wecksignalen erforderlich sind, werden abgeschaltet. Besteht der Wunsch seitens eines Endgerätes eine Verbindung aufzubauen, so wird ein Dauersignal bestehend aus `+-000000` gesendet. Die Weckschaltung auf der Vermittlungsseite erkennt das Signal und aktiviert alle Schaltungen. Nun beginnt die Vermittlungsseite, den kompletten Rahmen zu senden. Die Kanäle B, D, E und A werden dabei auf logische Null gesetzt. Durch den ankommenden Rahmen werden alle angeschlossenen Endgeräte aktiviert und leiten die Synchronisation ein. Beim Erreichen der Synchronisation senden sie den kompletten Rahmen (mit transparenten B- und D-Kanälen) um zwei Bit versetzt an die Vermittlung. Die Vermittlung quittiert durch Setzen des Aktivierungsbits A auf 1 und schaltet die B- und D-Kanäle transparent durch. Im E-Kanal wird der bei der Vermittlung ankommende D-Kanal gespiegelt. Den Grund hierfür werden wir gleich besprechen.

Besteht auf der Vermittlungsseite der Wunsch, eine Verbindung aufzubauen, so wird mit dem Senden des Rahmens von der Vermittlungsseite her begonnen. Der weitere Verlauf ist wie oben besprochen. Die Deaktivierung des Basisanschlusses wird von der Vermittlungsseite eingeleitet, wenn alle Schicht-2-Verbindungen abgebaut sind und eine bestimmte Schutzzeit ohne Aktivitäten abläuft. Die Vermittlungsseite schaltet dann den Rahmen ab und geht in den Ruhestand. Die Endgeräte folgen ihr entsprechend. Die Aktivierungs- und Deaktivierungsprozeduren sind in Abb. 4.2-11 dargestellt.

Abb. 4.2-11: Aktivierungs- und Deaktivierungsprozedur

Wie wir bereits gesehen haben, können mehrere Endgeräte an einem Basisanschluß angeschlossen sein. Für die Übertragung der Nutz- bzw. Signalisierungsnachrichten stehen ihnen die beiden B-Kanäle und der D-Kanal zur Verfügung. Die B-Kanäle werden von der Vermittlung verwaltet (in der Schicht 3) und nach Bedarf den Endgeräten für Verbindungen zugeteilt. Demgegenüber wird der **Zugriff auf den D-Kanal** auf der Bitebene (d. h. in der Schicht 1) unter Verwendung eines CSMA/CR-Verfahrens mit Adressenpriorität zur Kollisionsauflösung (siehe Abschnitt 3.2) geregelt. Für die Zuteilung des D-Kanals wird ferner eine **Prioritätsregelung** angewandt. Man unterscheidet zwischen zwei Prioritätsklassen - die Signalisierungsinformationen gehören zur Prioritätsklasse 1, die restlichen Informationen, wie z. B. paketierte Nutzdaten im D-Kanal, fallen in die Prioritätsklasse 2. Bevor ein Endgerät eine Meldung im D-Kanal absetzen kann, muß festgestellt werden, ob der Kanal (in Richtung TE → NT) frei ist. Da jedoch die Endgeräte nicht ausgerüstet sind, den Bitstrom in dieser Richtung abzuhören, wird der D-Kanal am NT in den E-Kanal (**Echo-Kanal**) gespiegelt und der E-Kanal dann am Endgerät abgehört. Für

Zugriff auf den D-Kanal

Prioritätsregelung

Echo-Kanal

Meldungen in der Prioritätsklasse 1 wird der Kanal als frei angesehen, wenn 8 Bit hintereinander als frei erkannt werden (d. h. auf logisch Null stehen). In der Prioritätsklasse 2 wird der Kanal als frei angesehen, wenn 10 Bit hintereinander als frei angesehen werden. Tritt nun trotzdem eine Kollision auf, so setzen sich physikalische Impulse (im D-Kanal in Richtung TE → NT ist dies stets eine -1) durch.

Die Meldung mit der niedrigsten Adresse setzt sich auf diese Weise durch. Betrachtet man Abb. 4.2-7, so sieht man, daß dem Endgerät, dessen D-Kanal Bit überschrieben wird, die Zeit, die für die Übertragung von 2 Bit erforderlich ist (zwischen dem E und dem nächsten D Bit), verbleibt, um sich vom Sendevorgang zurückzuziehen. Dies entspricht genau dem 2-Bit Versatz zwischen den beiden Rahmen. Das bisher beschriebene Verfahren hat den Nachteil, daß innerhalb einer Prioritätsklasse immer Meldungen mit niedriger Adresse bevorzugt werden. Um hier einen Ausgleich zu erzielen, wird eine **Fairneß-Strategie** angewandt. Diese besteht darin, daß nach einer erfolgreichen Übertragung einer Meldung der Prioritätsklasse 1 der Schwellwert (d. h. die Anzahl der freien Bits, die das erfolgreiche Endgerät abwarten muß) von 8 auf 9 erhöht wird. Entsprechend wird der Schwellwert für die Prioritätsklasse 2 von 10 auf 11 erhöht. Zählt ein Endgerät die höhere Anzahl von freien Bits (auch ohne daß eine Meldung vorlag), so wird der Schwellwert wieder herabgesetzt.

Fairneß-Strategie

Beispiel 4.2-4: Kollisionsauflösung beim Zugriff auf den D-Kanal
Drei Teilnehmer beginnen gleichzeitig auf dem D-Kanal zu senden. Sie haben jeweils folgende physikalische Folgen zu übertragen:

Sender A:	0	1	0	1	0	0
Sender B:	0	1	1	0	1	1
Sender C:	0	1	1	0	1	0

Die Skizze zeigt das Summensignal im E-Kanal und das Verhalten der Teilnehmer entsprechend dem D-Kanal-Zugangsprotokoll.

Animation 4.2-12: Summensignal im D-Kanal

In Abb. 4.2-13 ist die Schaltung für die Stromversorgung des S-Busses dargestellt. Es handelt sich um eine Phantomschaltung, bei der die Speiseleistung über die gleichen vier Adern übertragen wird, über die auch die Übertragung der Daten erfolgt. Die Sende- bzw. Empfangsdaten werden durch Übertrager ein- bzw. ausgekoppelt. Man unterscheidet zwischen dem Normalbetrieb und dem Notbetrieb. Im Normalbetrieb wird der S-Bus vom lokalen 230 V~ Netzteil des NT gespeist. Die maximale Speiseleistung beträgt 4 Watt bei 40 V (+5 % , -15 %) Speisespannung. Alle angeschlossenen Fernsprechgeräte werden hierüber versorgt; andere Endgeräte (Faxgeräte, PCs, usw.) werden wie üblich lokal gespeist. Im Aktivzustand stehen den Fernsprechgeräten jeweils maximal 900 mW zur Verfügung. Es können also 4 Fernsprechgeräte gleichzeitig versorgt werden; 400 mW verbleiben dann als Verlustleistung auf der Schnittstelle. Im Ruhezustand (*Power Down Mode*) dürfen insgesamt maximal 100 mW verbraucht werden. Bei einer Störung der lokalen Speisung des NT erfolgt die Umschaltung auf den Notbetrieb; die Versorgung erfolgt nun über die Vermittlungsstelle. Die Endgeräte erkennen den Notbetrieb an der Umpolung der Speisespannung. Mindestens ein Fernsprechapparat ist notspeiseberechtigt. Im aktiven Zustand steht einem notgespeisten Fernsprechapparat 400 mW, im passivem Zustand 25 mW zur Verfügung.

1) Polarität im Normalbetrieb(Speisung)
2) Strombegrenzung bei Überlast am SBus
3) Stromversorgungskopplung
4) Polaritätsumschalter für Notstromversorgung
5) Umschaltung für Not- und Normalversorgung
6) Stromversorgung für die VSt
7) Polarität des positiven Impulses

Abb. 4.2-13: Stromversorgung des S-Busses im Netz der DBP-Telekom

Beispiel 4.2-5: Leistungsverbrauch
Die Digitaltechnik hat den Nachteil, daß Sie einen erheblichen Stromverbrauch aufweist. Das ISDN D-Kanal-Protokoll ist diesbezüglich auch nicht sparsam ausgelegt, wie eine Überschlagsrechnung belegt.

Wir nehmen an, daß im Durchschnitt an einem S-Bus 2 Fernsprechapparate, ein Faxgerät und zwei PCs angeschlossen werden. Die Fernsprechapparate werden von der S-Schnittstelle, die anderen Geräte lokal gespeist. Entsprechend dem D-Kanal-Protokoll schalten sich alle Endgeräte ein, wenn ein Gerät aktiv ist (d. h. eine Verbindung unterhält). Nehmen wir nun an, daß, wenn der S-Bus aktiv ist, im Schnitt 1,2 Apparate eine Verbindung unterhalten (300 mW pro Apparat verbrauchen) und die restlichen 3,8 Geräte sich lediglich aufsynchronisieren (und 140 mW pro Apparat hierfür verbrauchen), so sieht die Bilanz wie folgt aus:

Verbrauch 1,2 aktive Geräte á 300 mW	= 360 mW
Verbrauch der 3,8 aufgeschalteten Geräte á 140mW	= 532 mW
Summe	= 892 mW

Der Anteil von 532 mW d. h. ca. 60 % der gesamten Leistung wird lediglich zum Aufschalten der Geräte, die nicht benutzt werden, aufgebraucht!

Im Ruhezustand wird pro angeschlossenem Gerät der Leistungsverbrauch von ca. 20 mW (für die Aktivierungsschaltung) angenommen. Für fünf Geräte zusammen sind dies ca. 100 mW.

4.2.2 Die U-Schnittstelle

Wie bereits erwähnt, basiert das ISDN-Konzept auf der Digitalisierung des vorhandenen Fernsprechnetzes. Die Investitionen im Teilnehmeranschlußnetz sind sehr hoch und es gilt diese weiterhin zu nutzen. Nun herrschen in den verschiedenen Ländern sehr unterschiedliche Verhältnisse im Teilnehmeranschlußbereich; so werden unterschiedliche Anschlußlängen, Aderdurchmesser und Kabelzerstückelungen verwendet. Sogar die Anschlußleistungen im öffentlichen und privaten Bereich in einem Land unterscheiden sich in dieser Hinsicht erheblich. Dies führt dazu, daß eine internationale Standardisierung der U-Schnittstelle, d. h. der Schnittstelle auf der Anschlußleitung, schwierig ist, und zahlreiche nationale Varianten zum Einsatz kommen.

Bei den Anschlußleitungen handelt es sich um herkömmliche zweidrahtige Kupferkabel. Dies bedeutet, daß in NT und LT eine 2-Draht - 4-Draht Umwandlung (Richtungstrennung) vorgenommen werden muß. Hierzu werden die beiden Verfahren, **Zeitgetrenntlage** und **Echokompensation** verwendet.

Zeitgetrenntlage

Echokompensation

Das **Zeitgetrenntlageverfahren** wird für die Richtungstrennung, insbesondere im Teilnehmeranschlußbereich von Nebenstellenanlagen eingesetzt. Es wird auch als **Zeitmultiplexverfahren**, Ping-Pong-Technik oder Zeitgabel bezeichnet (Abb. 4.2-14). Signale in beiden Richtungen werden im gleichen Frequenzbereich, jedoch

Zeitgetrenntlageverfahren

Zeitmultiplexverfahren

zeitlich nacheinander, als Datenpakete (Bursts) übertragen. Zwischen den Datenpaketen wird eine genügend große Pause für den Übertragungsvorgang (Signallaufzeit) und eine Schutzzeit eingelegt. Meist übernimmt eine der Stationen die Taktsteuerung (Master-Funktion) und die Gegenstation (Slave) schickt ihr Datenpaket nach einer kurzen Schutzzeit nach dem Empfang des ankommenden Datenpakets ab. Das Zeitgetrenntlageverfahren ist relativ einfach zu realisieren, denn es wird lediglich eine Speicherung und eine Steuerung mit Taktung erforderlich. Beide können digital ausgelegt werden. Der Hauptnachteil des Verfahrens ist, daß eine sehr hohe Bitrate (größer als die doppelte Bitrate des ursprünglichen Signals) erforderlich wird. Die Reichweite des Verfahrens ist im wesentlichen durch die Signallaufzeit begrenzt.

Abb. 4.2-14: Prinzip des Zeitgetrenntlageverfahrens

Abb. 4.2-15: Blockschaltbild des Zeitgetrenntlageverfahrens

Beispiel 4.2-6: Zeitgetrenntlageverfahren
Für die Duplexübertragung auf einer Teilnehmeranschlußleitung mit dem Zeitgetrenntlageverfahren werden Datenpakete von 38 Bits gebildet. Diese bestehen aus zwei Startbits für Synchronisation und Wartungsfunktionen und 36 Nutzbits. Die Stationen liefern diese 36 Bits alle $250\mu s$, d. h. die Nutzdatenrate pro Station beträgt $36/250\mu s = 144\,kbit/s$.

Mit den Startbits stehen somit pro Station $(38/36) \cdot 144\,kbit/s = 152\,kbit/s$ für die Übertragung in jede Richtung an. Für die Übertragung auf der Anschlußleitung wird eine Bitrate von $384\,kbit/s$ gewählt. Für die Übertragung eines Datenpakets aus 38 Bit benötigt man $(38Bit/384\,kbit/s) = 99\mu s$. Als Schutzzeit werden $5\mu s$ nach jedem Datenpaket angesetzt.

Abb. 4.2-16: Zeitgetrenntlageverfahren

Somit verbleiben für die Signallaufzeit $(125-104)\mu s = 21\mu s$. Bei einer Signallaufzeit von $6\mu s/km$ können somit 3,5 km überbrückt werden.

Das Zeitgetrenntlageverfahren ist kostengünstig, für Punkt-zu-Punkt-Verbindungen und Leitungslängen bis 2 km besonders gut geeignet. Es wird deshalb überwiegend im privaten Bereich für den Anschluß von Nebenstellen angewandt. Die Hersteller der Telekommunikationsanlagen in der BRD haben sich auf eine Spezifikation (U_{p0}-Schnittstelle des ZVEI - Zentralverband der Elektrotechnischen Industrie e.V.) geeinigt. Nach dieser Spezifikation werden pro Datenpaket (*Burst*) 36 Nutzbits, 1 Synchronisierbit und 1 Bit für Meldefunktionen gebildet (siehe Beispiel 4.2-6). Die Nutzbits bestehen aus 2 x (2 x 8 Bit für B-Kanäle + 2 Bit für D-Kanal). Die Nutzbitrate von 144 kbit/s wird unter Verwendung des AMI-Codes (siehe KT I, Abschnitt 8.3) mit einer Amplitude von 2 Volt (Null-Spitze) mit einer Schrittgeschwindigkeit von 384 kbit/s auf der Anschlußleitung übertragen.

1. Nahecho
2. Echo durch Reflexion an der Stoßstelle
3. Fernecho durch Reflexion an der Empfängergabel

Abb. 4.2-17: Echos bei Verwendung einer Gabel zur Richtungstrennung

Echokompensations-verfahren adaptive Gabel

Beim **Echokompensationsverfahren**, auch **adaptive Gabel** genannt, wird für die Ein- und Auskopplung der Sende- und Empfangssignale eine Gabelschaltung, wie sie auch beim analogen Telefon angewandt wird, eingesetzt. Die Gabel ist auf die Übertragungsfrequenz der digitalen Bitströme abgestimmt, reicht aber für die erforderliche Entkopplung gewöhnlich nicht aus. In Abb. 4.2-17 sind die Echos, die bei der Verwendung einer Gabel auftreten, aufgezeigt. Es ist einmal die unzureichende Entkopplung zwischen der Sende- und Empfangsrichtung in der Gabel, dann sind es die Reflektionen an Stoßstellen in der Anschlußleitung und letztlich auch das Fernecho an der Empfängergabel, die die Störungen verursachen. Alle diese Störungen korrelieren direkt mit dem entsprechenden Sendesignal, man kann deshalb unter Verwendung von Korrelationsverfahren die Störungen auf der Empfangsseite kompensieren.

In Abb. 4.2-18 ist das Prinzip des Verfahrens dargestellt. Das empfangene Signal $\tilde{S}$ an der Station *A* besteht aus dem gedämpften Signal S_B der Station *B* und den Echos *E*, die mit dem eigenen Sendesignal S_A korrelieren. Ein Regelalgorithmus stellt die Koeffizienten des Transversalfilters so ein, daß die Signale S_A und S_E möglichst nicht korrelieren bzw. $\hat{E}$ möglichst gleich *E* und somit E_A gleich S_B wird. In der einfachsten Version wird am Anfang der Übertragung eine Trainingsphase eingelegt. In dieser Phase sendet die Station *B* kein Signal, so daß $\tilde{S}$ gleich *E* wird. Die Filterkoeffizienten werden nun so gewählt, daß $\hat{E}$ gleich *E* wird und am Regler E_A gleich Null anliegt. Da die Echoeigenschaften auf Anschlußleitungen über längere Zeit konstant bleiben, ist meist eine Trainingsphase nicht erforderlich - die Adaption wird im laufenden Betrieb durchgeführt. Das Verfahren hängt entscheidend davon ab, daß die Folgen S_A und S_B statistisch unabhängig sind. Da dies bei Nutzdaten, insbesondere auch bei Sprache, häufig nicht der Fall ist, werden bei beiden Stationen unterschiedliche Verwürfler eingesetzt, um die statistische Unabhängigkeit zu gewährleisten (Abb. 4.2-19). Das Echokompensationsverfahren wird sowohl bei der Modemübertragung (meist in der Trainingsfolgeversion) als auch (in der adaptiven Version) bei der digitalen Übertragung im öffentlichen Netz (ISDN-Anschluß)

angewandt. Der Realisierungsaufwand ist wegen der erforderlichen Signalverarbeitung erheblich. Die Güte des Verfahrens hängt außer vom Regelalgorithmus entscheidend von der Genauigkeit des Transversalfilters und dem Zeitfenster, in dem eine Kompensation durchgeführt wird, ab.

Abb. 4.2-18: Prinzip des Echokompensationsverfahrens

Im Gegensatz zum Zeitgetrenntlageverfahren gibt es hier keine harte physikalische Grenze für die Reichweite des Verfahrens. Theoretisch kann bei einem entsprechend hohen Aufwand für die Kompensation die Reichweite einer Vierdrahtübertragung erreicht werden. Der wesentliche Vorteil des Verfahrens liegt darin, daß die Übertragungsrate auf der Leitung nicht erhöht wird und daher auch die Störeigenschaften des Systems nicht verschlechtert werden.

Abb. 4.2-19: Blockschaltbild des Echokompensationsverfahrens

Das Echokompensationsverfahren ermöglicht größere Reichweiten (bis 8 km) und wird deshalb im öffentlichen Netz bevorzugt eingesetzt. Das von der DBP-Telekom spezifizierte Verfahren (U_{k0}-Schnittstelle) verwendet den MMS43-Code (siehe KT I, Abschnitt 8.3) mit einer Amplitude von 2 Volt (Null-Spitze). Die Nutzbitrate

von 144 kbit/s wird durch die Codierung auf 120 kBaud herabgesetzt. International (vor allem in den USA) wird der 2B1Q-Code, der keine Redundanz enthält und die Schrittgeschwindigkeit auf 80 kBaud weiter herabsetzt, bevorzugt. In Abb. 4.2-21 ist die Rahmenstruktur der Daten auf der U_{k0}-Schnittstelle dargestellt. Die verwürfelten Nutzdaten (B+B+D-Kanäle) werden in vier Gruppen (T_1, T_2, T_3, T_4) zu je 27 ternären Schritten zusammengefaßt (Abb. 4.2-20). Zu diesen 108 ternären Schritten kommen ein Synchronwort (11 Schritte) und ein Meldewort (1 Schritt) hinzu, um den Rahmen aus 120 Schritten zu bilden. Dieser wird in 1 ms übertragen. Als Synchronwort wird das Signal $+ + + - - - + - - + -$ (SW1 VST → NT) bzw. $- + - - + - - - + + +$ (SW2 NT → VST), das einer **Barkerfolge** der Länge 11 mit einer charakteristischen Autokorrelationsfunktion (siehe Abb. 4.2-22) entspricht, verwendet.

Barkerfolge

Abb. 4.2-20: Bilden der ternären Gruppen T_i aus den Nutzdaten

Abb. 4.2-21: Die Rahmenstruktur der Schicht 1 an der U_{k0}-Schnittstelle

Abb. 4.2-22: Die Autokorrelationsfunktion der Barker-Folge $+ + + - - - + - - + -$

Entsprechend der S-Schnittstelle wird auch die U_{k0} -Schnittstelle aktiviert und deaktiviert. Das Wecksignal besteht aus einer Impulsfolge aus acht positiven und acht negativen Impulsen ($+ + + + + + + + - - - - - - - -$) in 2,133 ms, die 16 mal wiederholt werden; es wird mit dem gleichen Signal von der geweckten Seite quittiert. Die Abläufe bei der Aktivierung und Deaktivierung sind denen der S-Schnittstelle ähnlich. Die gesamte Aktivierung der U_{k0} -Schnittstelle wird in der Regel binnen 170 ms abgeschlossen.

Selbsttestaufgabe 4.2-1:

Wir betrachten den Einsatz des Ping-Pong-Verfahrens bei der PCM-Sprachübertragung. Die Pakete bestehen aus einem Startbit, n x (8 Sprachbits + 2 Signalisierbits) und einem Ausgleichsbit, d. h. insgesamt aus ($2 + 10 \cdot n$) Bits, die jeweils in eine Richtung pro $n \cdot 125\mu s$ anfallen. Die Schutzzeit zwischen den Paketen wird auf $5\mu s$ festgelegt. Die Signallaufzeit beträgt $6\mu s/km$, und die Pakete werden mit 256 kbit/s übertragen.

Geben Sie die überbrückbare Leitungslänge l in Abhängigkeit von der Anzahl der Abtastwerte n an, und zeichnen Sie den Zusammenhang für $n \leq 8$. Berechnen Sie die maximal überbrückbare Länge, falls eine Verzögerung von 0,5 ms bei der Sprachübertragung zulässig ist.

Selbsttestaufgabe 4.2-2:

Berechnen Sie die Autokorrelationsfunktionen der verschiedenen Synchronworte, die bei PCM 30 (Rahmenkennung), PCM 120 (Rahmenkennung), Token Ring (Starting Frame Delimiter), HDLC (Flag) und im ISDN (U_{k0}) verwendet werden.

Welches ist das sicherste Synchronwort? Begründen Sie dies. Warum wird beim Token Ring für die Synchronisierung der Startdelimiter mitverwendet?

4.3 Die Sicherungsschicht

Die Schicht 1 des Basisanschlusses (bestehend aus den S- und U-Schnittstellen) bietet zwischen der Vermittlungsstelle und den Endgeräten drei transparente physikalische Kanäle 2 B + D_{16} (der Subindex wird verwendet, um anzudeuten, daß es sich um einen D-Kanal mit 16 kbit/s handelt). Entsprechend bietet die Schicht 1 des Primärmultiplexanschlusses die transparenten, physikalischen Kanäle 30B + D_{64} (hier handelt es sich um einen D-Kanal mit 64 kbit/s). Auf beiden Kanälen D_{16} und D_{64} wird das gleiche Schicht-2-Protokoll **LAPD** (***L**ink **A**ccess **P**rocedure on the **D**-Channel*) verwendet, allerdings mit geringfügig anderen Protokollparametern (wie z. B. Fenstergröße). Die Aufgabe dieses Schicht-2-Protokolls ist es, gesicherte Informationsübermittlung für die Schicht 3 zu gewährleisten; das Protokoll ist deshalb im wesentlichen identisch mit der HDLC - Prozedur. Es unterstützt Mehrgerätekonfigurationen am Teilnehmeranschluß (z. B. auf der Schnittstelle) und auch die Bildung von mehreren Schicht-3-Verbindungen auf einer Schicht-2-Verbindung (Multiplexbildung in der Schicht 2). Das Protokoll ermöglicht, sowohl quittierte als auch unquittierte Nachrichten zu übermitteln und eine Flußregelung vorzunehmen. Für den Einsatz des Protokolls wird ein duplexer, transparenter D-Kanal mit einer beliebigen Übertragungsrate benötigt. In Abb. 4.3-1 sind die beiden Formate der verwendeten Rahmen in der Schicht 2 dargestellt. Wir wollen diese im folgenden näher ansehen.

LAPD

Abb. 4.3-1: Rahmenstruktur der LAPD-Meldungen

Wie bei der HDLC-Prozedur handelt es sich beim LAPD um ein bitorientiertes Protokoll. Der Rahmenbeginn wird durch ein Flag (`01111110`) angezeigt. Um die Bittransparenz zu gewährleisten wird das *zero insertion* Verfahren angewandt (siehe Abschnitt 2.2). Stehen keine Daten zur Übertragung an, so werden im Gegensatz zur HDLC-Prozedur keine Flags, sondern eine Folge von logischen 1 (d. h. physikalische 0) übertragen. Dies erfordert das unter Abschnitt 4.2.1 erläuterte Zugriffsverfahren. Die Schicht-2-Adresse besteht aus 2 Bytes (siehe Abb. 4.3-2). Das erste Byte enthält den **SAPI** (***S**ervice **A**ccess **P**oint **I**dentifier*) Wert, das zweite Byte den **TEI** (***T**erminal **E**ndpoint **I**dentifier*) Wert. Durch diese beiden Werte ist ein Verbin-

SAPI
TEI

dungsendpunkt der Schicht 2 eindeutig festgelegt (Abschnitt 1.3) siehe Abschnitt 1.3 im Kurs KT I).

C/R Command/Response SAPI Service Access Point Identifier
EA Address Field Extension TEI Terminal Endpoint Identifier

Abb. 4.3-2: Aufbau der LAPD-Adresse

Das **SAPI**-Feld besteht aus 6 Bit, so daß 64 Werte möglich sind. Bisher sind folgende vier Werte von CCITT festgelegt:

SAPI = 0	Signalisierinformation
SAPI = 1	Paketübermittlung im ISDN nach I.451
SAPI = 16	Paketübermittlung im ISDN nach X.25
SAPI = 63	Managementfunktionen, Gruppen SAPI (*Broadcast*)

Durch den niedrigsten SAPI-Wert werden die Signalisierinformationen beim D-Kanal Zugriff priorisiert. Die beiden Paketübermittlungsmethoden (mit SAPI=1 und 16) werden wir im nächsten Abschnitt kennenlernen. Meldungen mit SAPI= 63 sind Wartungs- und Verwaltungsmeldungen für alle. Die übrigen SAPI-Werte sind für künftige Anwendungen reserviert, wobei die SAPI-Werte 32 bis 47 national verwendet werden können.

Über das **C**/**R**-Bit (***C****ommand*/***R****esponse*) wird gekennzeichnet, ob ein Endgerät oder die Vermittlung den Nachrichtenaustausch initiiert hat und ob es sich um einen Befehl (*Command*) oder Antwort auf den Befehl (*Response*) handelt. Die Werte werden wie folgt codiert:

Kommando	VST → Endgerät	C/R = 1
	Endgerät → VST	C/R = 0
Antwort	VST → Endgerät	C/R = 0
	Endgerät → VST	C/R = 1

Das **EA**-Bit (***E**xtended **A**ddress*) gibt an, ob die Adresse beendet ist. EA=0 zeigt an, daß ein weiteres Adressbyte folgt; EA=1 zeigt das letzte Byte der Adresse an.

Im zweiten Byte des Adressfeldes ist der aus 7 Bits bestehende **TEI** (***T**erminal **E**ndpoint **I**dentifier*) Wert enthalten. Über diesen Wert wird jeder logischen Schicht 2-Verbindung eine eindeutige Adresse zugeteilt. Die TEI - Werte werden von der Vermittlung verwaltet. In einzelnen Endgeräten können TEI - Werte voreingestellt werden, bedürfen jedoch der Bestätigung durch die Vermittlung. Die Werte 0-63 werden für solche voreingestellten Adressen verwendet, während die Vermittlung auf Anfrage Werte ab 64 bis 126 vergibt. Der Wert 127 ist als Gruppen-TEI für das Rundsenden (*Broadcast*) - also für alle - bestimmt.

In Abb. 4.3-3 ist ein Beispiel angegeben, welches die Schicht-2-Adressierung demonstriert. Am Basisanschluß sind zwei Endgeräte angeschlossen. Endgerät 1 hat zwei SAPIs (0 und 16, SAPI 63 ist nicht angezeigt), ist also paketübermittlungsfähig (nach X.25 im ISDN) und zwei TEIs (72 und 127). Insgesamt bedient die Schicht 2 des Endgerätes 1 die vier Verbindungsendpunkte (SAPI 0, TEI 72), (SAPI 0, TEI 127), (SAPI 16, TEI 72) und (SAPI 16, TEI 127). Das Endgerät 2 hat ein SAPI (0) und drei TEIs (66, 67 und 127). Die Schicht 2 des Endgerätes bedient die drei Endpunkte (SAPI 0, TEI 66), (SAPI 0, TEI 67) und (SAPI 0, TEI 127). Entsprechend sind in der Schicht 2 der Vermittlung unter SAPI 0 vier und SAPI 16 zwei Endpunkte zu bedienen.

Abb. 4.3-3: Beispiel zur Adressierung

Im LAPD sind drei Typen von Rahmen definiert:

- numerierte Informationsrahmen (*I-Frames*)
- Steuer- und Überwachungsrahmen (*S-Frames*)
- unnumerierte Rahmen (*U-Frames*)

Die Codierungen des Steuerfeldes der Rahmen sind in Abb. 4.3-4 wiedergegeben. Die Befehle und Meldungen und deren Codierung entsprechen denen der HDLC-Prozedur, so daß wir diese im einzelnen nicht mehr ansprechen wollen.

Anwendung	Format	Befehle	Meldungen	Codierung 8	7	6	5	4	3	2	1
Unquittierte und quittierte Informations-übermittlung	Informations-übermittlung	I (Information)		N(S)							0
				N(R)							P
	Supervisory	RR (Receive Ready)	RR (Receive Ready)	0	0	0	0	0	0	0	1
				N(R)							P/F
		RNR (Receice Not Ready)	RNR (Receice Not Ready)	0	0	0	0	0	1	0	1
				N(R)							P/F
		REJ (Reject)	REJ (Reject)	0	0	0	0	1	0	0	1
				N(R)							P/F
	Unnumbered	SAMBE (Set Asynchronous Balanced Mode Extended)		0	1	1	P	1	1	1	1
			DM (Disconnected Mode)	0	0	0	F	1	1	1	1
		UI (Unnumbered Information)		0	0	0	P	0	0	1	1
		DISC (Disconnect)		0	1	0	P	0	0	1	1
			UA (Unnumbered Acknowledgement)	0	1	1	F	0	0	1	1
			FRMR (Frame Reject)	1	0	0	F	0	1	1	1
Verbindungs-steuerung		XID (Exchange Identification)	XID (Exchange Identification)	1	0	1	P/F	1	1	1	1

Abb. 4.3-4: Die Codierung des Steuerfeldes im LAPD

Wie bei der HDLC-Prozedur ist eine quittierte oder eine nichtquittierte Nachrichtenübertragung möglich. Bei der unquittierten Nachrichtenübertragung werden weder Fehlerkorrekturmaßnahmen noch eine Flußsteuerung vorgenommen. Der Fenstermechanismus für die Flußkontrolle bei quittierter Nachrichtenübermittlung ist identisch mit dem der HDLC-Prozedur. Die Zählung wird in Modulo 128 vorgenommen. Die Fenstergröße wird je nach Art der Verbindung unterschiedlich eingestellt. Für die Signalisierung am Basisanschluß wird die Fenstergröße 1, für die Signalisierung am Primärmultiplexanschluß die Fenstergröße 7 verwendet. Für die Übermittlung von paketierten Nutzdaten im D-Kanal des Basisanschlusses wird die Fenstergröße 3, für die Übermittlung der paketierten Nutzdaten im D-Kanal des Primärmultiplexanschlusses die Fenstergröße 7 verwendet. Wie wir bereits bei der HDLC-Prozedur kennengelernt haben, wird für jede Schicht-2-Verbindung eine eigene Flußsteuerung durchgeführt. Hierfür werden an beiden Enden der Verbindung entsprechende Zählungen vorgenommen. Das Informationsfeld darf aus maximal 260 Bytes bestehen und enthält die Schicht-3-Informationen, die wir im nächsten Abschnitt kennenlernen werden.

Die letzten beiden Byte des Rahmens werden zur Durchführung der CRC-Prüfung verwendet. Hierfür wird wie bei der HDLC-Prozedur das Generatorpolynom $g(x) = x^{16} + x^{12} + x^5 + 1$ verwendet.

Die Abläufe beim Verbindungsaufbau, Datenübermittlung und Verbindungsabbau entsprechen denen der HDLC-Prozedur. Wir wollen diese deshalb hier nicht mehr ansehen. In Abb. 4.3-5 und Abb. 4.3-6 sind zwei Prozeduren, die mit der TEI-Verwaltung in der Vermittlung zusammenhängen, dargelegt. Bei Abb. 4.3-5 handelt

es sich um eine TEI-Zuweisungsprozedur; das Endgerät fordert einen bestimmten TEI-Wert an und bekommt diesen von der Vermittlung bestätigt. Bei Abb. 4.3-6 überprüft die Vermittlung einen TEI-Wert, das Endgerät mit diesem TEI-Wert meldet sich. Da nur ein Endgerät sich meldet, wird der Wert als gültig angesehen und bleibt in Betrieb. Bei mehreren Antworten wäre der Wert als ungültig, bei keiner Antwort als nicht vergeben anzusehen.

Abb. 4.3-5: TEI-Zuweisungsprozedur

Abb. 4.3-6: TEI-Prüfprozedur

4.4 Die Vermittlungschicht

Die Einbettung der Schicht-3-Nachrichten im Schicht-2-Rahmen (*Information Frame*) haben wir bereits im vorigen Abschnitt (Abb. 4.3-1) kennengelernt - dort wurden diese Meldungen als INFO bezeichnet. Die allgemeine Struktur dieser Nachrichten ist in Abb. 4.4-1 dargestellt.

Abb. 4.4-1: Aufbau der Schicht-3-Meldungen

Jede Schicht-3-Nachricht enthält die drei Komponenten:

- Protokoll Diskriminator (*Protocol Discriminator*),
- Referenznummer (*Call Reference*),
- Nachrichtentyp (*Message Type*).

Hinzu kommen weitere Nachrichtenelemente (*Information Elements*). Diese können je nach Nachrichtentyp zwingend vorgeschrieben (*Mandatory*) oder wählbar (*Optional*) sein.

Protokolldiskriminator

Der **Protokolldiskriminator** kennzeichnet das angewandte Schicht-3-Protokoll. Für die Zeichengabe nach dem CCITT-Protokoll (I.451) ist die Codierung `0000 1000` festgelegt; für nationale Protokolle (z. B. 1TR6 der DBP-Telekom) werden die Codierungen `0100 0001` und `0100 0000` verwendet.

Referenznummer

Dem Protokolldiskriminator folgt die Angabe der Länge der**Referenznummer**. Die maximale Länge der Referenznummer beträgt zur Zeit 1 Byte für den Basisanschluß und 2 Byte für den Primärmultiplexanschluß. Die Referenznummer wird verwendet, um den Ruf an der Benutzer-Netzschnittstelle zu identifizieren. Sie hat lediglich lokale Bedeutung (d. h. keine unmittelbare Ende-zu-Ende Relevanz) und ermöglicht die eindeutige Zuordnung einer Schicht-3-Nachricht zu einem Ruf. Sie zeigt ferner an, ob die Meldung von dem Rufenden oder von dem Gerufenen gesendet wird.

Nachrichtentyp

Die im ISDN in der Schicht 3 verwendeten Nachrichten sind sehr umfangreich; viele der Nachrichten ähneln oder sind gar identisch mit denen der X.25 und SS Nr. 7 Protokolle. Die wesentlichen **Nachrichtentypen** sind in Abb. 4.4-2 zusammengestellt. Man unterteilt diese in:

- Nachrichten für den Verbindungsaufbau und -abbau,
- Nachrichten während der Datentransferphase und

- verschiedene andere Nachrichten.

Wir wollen die Nachrichtentypen nicht im einzelnen besprechen, sondern werden einige im Zusammenhang mit Protokollabläufen näher kennenlernen.

Nachrichten für den Verbindungsaufbau und -abbau									
		8	7	6	5	4	3	2	1
SETUP	Setup	0	0	0	0	0	1	0	1
SETUP ACK	Setup Acknowledge	0	0	0	0	1	1	0	1
CALL PROC	Call Proceeding	0	0	0	0	0	0	1	0
ALERT	Alerting	0	0	0	0	0	0	0	1
CONN	Connect	0	0	0	0	0	1	1	1
CONN ACK	Connect Acknowledge	0	0	0	0	1	1	1	1
DISC	Disconnect	0	1	0	0	0	1	0	1
REL	Release	0	1	0	0	1	1	0	1
REL COM	Release Complete	0	1	0	1	1	0	1	0

Abb. 4.4-2: Nachrichten für den Verbindungsaufbau und -abbau

Nachrichten während der Datentransferphase									
		8	7	6	5	4	3	2	1
USER INFO	User Information	0	0	1	0	0	0	0	0
SUSP	Suspend	0	0	1	0	0	1	0	1
SUSP ACK	Suspend Acknowledge	0	0	1	0	1	1	0	1
SUSP RES	Suspend Reject	0	0	1	0	0	0	0	1
RES	Resume	0	0	1	0	0	1	1	0
RES ACK	Resume Acknowledge	0	0	1	0	1	1	1	0
RES REJ	Resume Reject	0	0	1	0	0	0	1	0

Abb. 4.4-3: Nachrichten während der Datentransferphase

Verschiedene weitere Nachrichten									
		8	7	6	5	4	3	2	1
FAC	Facility	0	1	1	0	0	0	1	0
STATUS	Status	0	1	1	1	1	1	0	1
INFO	Information	0	1	1	1	1	0	1	1

Abb. 4.4-4: Nachrichten während der Datentransferphase

Nachrichtenelement

Dem Nachrichtentyp folgen nun gegebenenfalls weitere **Nachrichtenelemente**. Diese sind entweder ein Byte oder mehrere Byte lang. Die verwendeten Formate sind in Abb. 4.4-5 und Abb. 4.4-6 dargelegt, die wesentlichen verwendeten Nachrichtenelemente in Tabelle 4.4-1 zusammengestellt. Dem Nachrichtenelement *Shift* kommt eine besondere Bedeutung zu. Es dient dazu, den Codesatz für die nachfolgenden Elemente umzuschalten. Hierdurch wird die Anzahl der möglichen Nachrichtenelemente erheblich erweitert. Zur Zeit sind vier verschiedene Codesätze definiert: international standardisierte, national standardisierte, lokale und herstellerspezifische Nachrichtenelemente.

Abb. 4.4-5: Einbyte Nachrichtenelement

Abb. 4.4-6: Mehrbyte Nachrichtenelement

Tab. 4.4-1: *Einige Nachrichtenelemente*

Einbyte-Nachrichtenelemente	Mehrbyte-Nachrichtenelemente
Shift	Bearer Capability
More Data	Cause
Congestion Level	Connected Address
	Call Identity
	Call State
	Channel Identification
	Terminal Capabilities
	Display
	Keypad
	Keypad Echo
	Signal
	Switchhook
	Origination Address
	Destination Address
	Redirection Address
	Transit Network Selection
	Low Layer Compability
	High Layer Compability
	User-User Information
	CCITT-Standardized Facilities
	Network-Specific Facilities

Wir sehen uns nun exemplarisch die Nachricht *SETUP* an, mit der der Aufbau einer Schicht-3-Verbindung eingeleitet wird (Tabelle 4.4-2, Abb. 4.4-7 und Abb. 4.4-8). Die Nachricht beginnt mit dem Protokolldiskriminator und der Referenznummer des Rufers. Es folgen die Codierungen für den Nachrichtentyp *SETUP* (`00000101`) und für das zwingend vorgeschriebene Element *Bearer Capability* (`00000100`). In Richtung Netz zum Benutzer ist ferner die Angabe des zu verwendenden B-Kanals zwingend erforderlich. Die weiteren Nachrichtenelemente wie *Origination Address*, *Destination Address*, *User-User-Infos* usw. sind optional. Das Element *Bearer Capability* ist beispielhaft in Abb. 4.4-7 dargestellt, die Codierung des Inhalts dieser Meldung in Abb. 4.4-8 angegeben.

Tab. 4.4-2: *Inhalt der Nachricht SETUP*

Nachrichtenelement	Benutzer → Netz	Netz → Benutzer	Länge
Protocol discriminator	M	M	1
Call reference	M	M	1-?
Message type	M	M	1
Bearer capability	M	M	4-?
Channel identification	O	M	3-?
CCITT-standardized facilities	O	O	3-?
Network specific facilities	O	O	3-?
Display	-	O	3-?
Keypad	O	-	3-?
Signal	O	O	3-?
Switchhook	O	O	3
Origination Address	O	O	4-?
Destination Address	O	O	4-?
Redirection Address	-	O	4-?
Transit network selection	O	-	3-?
Low layer compability	O	O	4-?
High layer compability	O	O	3-?
User-user information	O	O	3-30(130)

M: Mandatory, O: Optional

Abb. 4.4-7: Beispiel für das Nachrichtenelement *Bearer Capability*

Coding Standard			
0	0		CCITT Standard
0	1		Other Intern. Standard
1	0		National Standard
1	1		Network Standard

Information Transfer Capability						
0	0	0	0	0		Speech
0	1	0	0	0		Unrestricted Digital Information
0	1	0	0	1		Restricted Digital Information
1	0	0	0	0		3,1 kHz Audio
1	0	0	0	1		7 kHz Audio
1	1	0	0	0		Video

Transfer Mode			
0	0		Circuit Mode
1	0		Packet Mode

Information Transfer Rate						Circuit Switched
0	0	0	0	0		Reserved For Packet Mode
1	0	0	0	0		64 kbit/s B
1	0	0	0	1		2x64 kbit/s 2xB
1	0	0	1	1		384 kbit/s H_0
1	0	1	0	1		1536 kbit/s H_{11}
1	0	1	1	1		1920 kbit/s H_{12}

Abb. 4.4-8: Codierung der Inhalte des Nachrichtenelements *Bearer Capability*

Wir wollen nun den kompletten Verbindungsaufbau und -abbau für eine Durchschalteverbindung im ISDN ansehen. In Abb. 4.4-9 sind die Abläufe zwischen den beiden Endgeräten TE-A und TE-B über die Vermittlung ET beispielhaft dargestellt. Die Kommunikation zwischen den beiden Vermittlungen ET-A und ET-B über das Netz wird mit Hilfe des SS Nr. 7 durchgeführt - dieses ist im Bild zur Vereinfachung nicht dargestellt. Um das Zusammenspiel zwischen den Schichten aufzuzeigen, sind die Primärmeldungen zwischen Schicht 2 (S2) und Schicht 3 (S3) aufgezeigt.

Abb. 4.4-9: Verbindungsaufbau und -abbau für eine Durchschalteverbindung

Die Schicht 3 des A-Teilnehmers leitet den Verbindungsaufbau durch Senden der Nachricht *SETUP* ein. Daraufhin wird zunächst von der Schicht 2 des A-Teilnehmers eine Schicht-2-Verbindung durch Senden der Nachricht *SABME* aufgebaut und durch *UA* von der Vermittlung quittiert. Über diese Schicht-2-Verbindung wird nun eine Schicht-2-Meldung *I* (Info) an die Vermittlung gesendet und durch *RR* quittiert. Die Meldung *I* enthält die Schicht-3-Meldung *SETUP* mit den für den Verbindungsaufbau erforderlichen Parametern. Bei Blockwahl sind alle Adressinformationen im *SETUP* als Nachrichtenelemente enthalten. Bei Einzelziffernwahl können die Adressinformationen durch weitere Info-Nachrichten nachgereicht wer-

den. Handelt es sich nun um zwei verschiedene Vermittlungen, an denen die A- und B-Teilnehmer angeschlossen sind, so werden die relevanten Informationen zwischen den Vermittlungen über das Signalisiernetz ausgetauscht und die Transitverbindungen entsprechend aufgebaut. Die Schicht-3-Meldung *CALL PROC* teilt dem Endgerät A mit, daß die Wählinformationen vollständig sind und die Wahlaufforderung weiterbearbeitet wird; ferner wird dem Endgerät mitgeteilt, welcher B-Kanal für die Verbindung belegt wird. Die Vermittlung des B-Teilnehmers leitet den Verbindungsaufbau ihrerseits mit der Schicht-2-Nachricht *UI*, in der die Schicht-3-Meldung *SETUP* eingebettet ist, ein. Diese enthält nun die relevanten Adress-, Kompatibilitäts- und Berechtigungsangaben, die für die Auswahl von Endgeräten, die gerufen werden, erforderlich ist. Die *SABME*-Meldung initiiert die Zähler für die Schicht-2-Verbindung, während die *ALERT*-Meldung (Schicht 3) anzeigt, daß beim B-Teilnehmer Endgeräte in der Lage sind, den Ruf anzunehmen und gerufen werden. Die Annahme des Rufes wird durch die *CONNECT*-Meldung (Schicht 3) angezeigt und mit *CONNACK* quittiert. Die Verbindung ist nun für die Datenübermittlung transparent geschaltet. Der Verbindungsabbau kann von beiden Seiten angestoßen werden. In unserem Beispiel wird er vom A-Teilnehmer durch Senden der *DISC*-Meldung (Schicht 3) eingeleitet, durch *RELEASE* (Schicht 3) abgebaut; mit *RELEASE COM* (Schicht 3) wird der Abbau quittiert. Die jeweiligen Schicht-2-Meldungen *I*, die diese Nachrichten enthalten, werden ihrerseits durch *RR* quittiert. Die Schicht-2-Verbindungen werden anschließend durch die *DISC*-Meldung (Schicht 2) abgebaut und durch *UA* quittiert.

ISDN-Adresse
ISDN-Rufnummer

In Abb. 4.4-10 ist die Struktur der **ISDN-Adresse** dargelegt. Die **ISDN-Rufnummer** mit maximal 15 Dezimalziffern besteht wie im Fernsprechnetz aus der Länderkennung, der Ortsnetzkennung und der Teilnehmernummer. Die Subadresse mit maximal 40 Dezimalziffern wird von der Vermittlung transparent vom rufenden zum gerufenen Teilnehmer übertragen. Eine ISDN-Rufnummer adressiert eine oder mehrere Schnittstellen an einem Referenzpunkt T oder S. Die ISDN-Subadresse dient der genauen Adressierung von Subkomponenten, so z. B. einem bestimmten Endgerät, einem bestimmten Dienst eines Multifunktionalendgerätes usw. Bei einer Punkt-zu-Punkt-Verbindung ist ein Endgerät durch die ISDN-Rufnummer direkt wählbar. Bei Euro-ISDN (E-DSS1) können bis zu zwölf unterschiedliche Rufnummern Verwendung finden. Beim nationalen ISDN (1TR6) wurde die sogenannte Mehrgerätekonfiguration für die Endgeräteauswahl entweder mit Hilfe der Subadresse oder durch die Kompatibilitätsinformation in der Nachricht *SETUP* vorgenommen. Im Netz der DBP-Telekom wird einem S_0-Bus eine ISDN-Rufnummer zugeteilt. Der S_0-Bus wird durch die Ziffern bis zu der Dekadenstelle identifiziert. Die Ziffer 0 der Einerstelle kennzeichnet den *Global Call* (d. h. alle Endgeräte eines Dienstes werden gerufen). Die anderen Ziffern der Einerstelle werden zur Endgeräteauswahl am S_0-Bus verwendet (Direktruf).

Abb. 4.4-10: Die Struktur der ISDN-Adresse

Es sind grundsätzlich alphanumerische Zeichen zugelassen. Zur Zeit findet dies jedoch keine Anwendung.

Selbsttestaufgabe 4.4-1:

Beschreiben Sie den grundsätzlichen Aufbau von Schicht-3-Nachrichten.

4.5 Das Signalisierungssystem Nr. 7

Signalisierung
Zeichengabe

Bei unseren bisherigen Betrachtungen haben wir gesehen, daß beim Kommunikationsvorgang Steuerinformationen erzeugt und ausgetauscht werden. Man bezeichnet dies als **Signalisierung** oder **Zeichengabe**. Sie dient dem Auf- und Abbau von Nachrichtenverbindungen, der Steuerung von Nachrichtenpaketen und der Aktivierung und Deaktivierung von unterschiedlichsten Dienstmerkmalen und Betriebsfunktionen. Man unterscheidet zwischen der Signalisierung

Teilnehmer-Signalisierung
Netz-Signalisierung
Ende-zu-Ende-Signalisierung

- zwischen dem Endgerät und der Vermittlungsstelle, an der es angeschaltet ist (**Teilnehmer-Signalisierung**),
- zwischen Vermittlungsstellen (**Netz-Signalisierung**) und
- zwischen den Endgeräten (**Ende-zu-Ende-Signalisierung**).

Im-Band-/ Außerband-
Signalisierkanal
zentraler Signalisierkanal

Wird der Nutzkanal für die Signalisierung mitverwendet, so wird die Signalisierung als **Im-Band** (*In Band*), sonst als **Außerband** (*Out Band*) bezeichnet. Wird für die Signalisierung ein gesonderter Kanal verwendet, so wird dieser als **Signalisierkanal** bezeichnet. Dieser kann für einzelne Nutzkanäle individuell oder für mehrere Nutzkanäle gemeinsam ausgelegt sein. Im letzteren Fall spricht man von einem **zentralen Signalisierkanal**.

Beim Signalisierungssystem Nr. 7 (SS Nr. 7) handelt es sich um ein Signalisiersystem für die zentrale Zeichengabe zwischen Vermittlungsstellen, deren Entwicklung bei CCITT bereits 1973-76 vorangetrieben wurde; die ersten Empfehlungen wurden 1980 verabschiedet. Die Entwicklung des SS Nr. 7 verlief parallel zu ISO-Aktivitäten zur Normung des OSI-Modells (siehe Abschnitt 1.2); das Protokoll weist deshalb lediglich Grundmerkmale des OSI-Modells auf (Abb. 4.5-1).

Die beiden ersten Ebenen der SS Nr. 7 stimmen mit den beiden ersten Schichten des OSI-Modells funktionsmäßig überein. Die dritte Ebene enthält die Grundfunktion der Schicht 3; im wesentlichen sind es die Vermittlungsfunktion für die Zeichengabe und das Netzmanagement. Die ersten drei Ebenen bilden die Transportfunktionen für die Zeichengabe (**MTP**- ***M**essage **T**ransfer **P**art*). Hierauf setzen die dienste- abhängigen Anwendungsteile (*User Parts*) auf - so z. B. für Fernsprechen, Daten und ISDN. Diese enthalten alle Signale und Prozeduren, die für die Zeichengabe zwischen den Vermittlungsknoten für die einzelnen Dienste (Fernsprechen, Daten usw.) erforderlich sind. Der **SCCP** (***S**ignalling **C**onnection **C**ontrol **P**art*) ergänzt den MTP, um die OSI Schicht-3-Funktionen zu komplettieren. Der so entstandene **NSP** (***N**etwork **S**ervices **P**art*) ermöglicht den darauf aufgebauten Anwendungen (*Application Parts*), beliebige Informationen verbindungsorientiert oder verbindungslos auszutauschen. Erwähnt seien die Anwendungen für Betrieb und Wartung (*Operations and Maintenance*) und die Ende-zu-Ende Signalisierung/**TCAP** (***T**ransaction **C**apability **A**pplication **P**art*). TCAP ermöglicht es, nicht nutzkanalgebundene Anwendungen, wie Chipkartenauthentifikation und -buchung, zu realisieren. TCAP wird auch für die GSM-Mobilfunk-Signalisierung (*Mobile User Part*) verwendet. Im folgenden wollen wir den Transportteil (MTP) des Protokolls und beispielhaft die ISDN-Anwendung (ISDN-UP) näher ansehen.

MTP

SCCP

NSP

TCAP

MTP	Message Transfer Part
NSP	Network Service Part
OMAP	Operational and Maintenance Application Part
SCCP	Signalling Connection Control Part
TCAP	Transaction Capability Application Part

Abb. 4.5-1: SS Nr. 7 und OSI-Modell

Für die Bitübertragung (Ebene 1) im SS Nr. 7 werden gewöhnlich 64 kbit/s Strecken, die über digitale Koppelfelder zugänglich sind, verwendet. Bei PCM-30- Systemen wird der 16. Zeitschlitz als transparenter Kanal für SS Nr. 7 genutzt.

Die hierauf aufsetzende Schicht 2 ist dem HDLC-Protokoll sehr ähnlich. Die Rahmenstruktur der Meldungen ist in Abb. 4.5-2 dargelegt. Man unterscheidet zwischen Meldungen, die Schicht-3-Informationen enthalten, (**MSU** - ***M**essage **S**ignal **U**nit*), Meldungen, die den Status der Schicht 2 anzeigen (**LSSU** - ***L**ink **S**tatus **S**ignal **U**nit*) und Füllmeldungen (**FISU** - ***F**ill **I**n **S**ignal **U**nit*). Die FISU wird zum Überbrücken von Wartezeiten und ggf. zum Quittieren verwendet. Der Rahmen beginnt und endet jeweils mit dem üblichen ***Flag*** (Codierung `01111110`). Um die Bittransparenz zu gewährleisten, wird das *Zero Insertion* Verfahren (Siehe Abschnitt 2.2) angewandt. Da es sich bei SS Nr. 7 stets um Punkt-zu-Punkt Verbindungen handelt, ist eine Adressierung in der Schicht 2 nicht erforderlich.

MSU
LSSU
FISU
Flag

MSU Frame

LSSU Frame

FISU Frame

BIB	Backward Indication Bit	LI	Length Indicator
BSN	Backward Sequence Number	LSSU	Link Status Signal Unit
CRC	Cyclic Redundancy Check	MSU	Message Signal Unit
FIB	Forward Indication Bit	SINFO	Status Info (Layer 7)
FISU	Fill In Signal Unit	UINFO	User Info (Layer 3)
FSN	Forward Sequence Number		

Abb. 4.5-2: Rahmenstruktur der SS Nr. 7 Meldungen

Flag Unmittelbar nach dem **Flag** folgen zwei Oktetts, die für die Sequenzierung und Flußkontrolle verwendet werden. Es wird Modulo 128 gezählt, so daß 7 Bit für die Sequenzierung erforderlich sind. Die **FSN** (***F**orward **S**equence **N**umber*) wird für die Sequenzierung der gesendeten Meldungen verwendet und entspricht der N(S) beim HDLC-Verfahren. Es werden nur Meldungen, die eine Schicht-3-Nachricht enthalten (d. h. nur MSUs), gezählt. Bei den anderen Meldungen wird die alte Sequenznummer lediglich wiederholt. Das **FIB** (***F**orward **I**ndication **B**it*) zeigt an, ob es sich um eine neue Meldung oder eine Wiederholung handelt. Im Falle der Wiederholung wird das Bit invertiert, unterscheidet sich also hierin gegenüber der

FSN
FIB

vorherigen Meldung. Die **BSN** (***B**ackward **S**equence **N**umber*) wird zur Quittierung der Meldungen verwendet und gibt die Nummer der letzten richtig erhaltenen Meldung an. Das **BIB** (***B**ackward **I**ndication **B**it*) kann für eine explizite positive oder negative Quittierung (ACK/NAK) verwendet werden. Die positive Quittierung wird durch Beibehalten des BIB der letzten Meldung, die negative Quittierung durch Invertieren des BIB der letzten Meldung angezeigt. Für die Fehlerkorrektur sind zwei Verfahren vorgesehen. In der Grundversion wird das BIB für die negative Quittierung verwendet. Bei Empfang einer negativen Quittierung wiederholt der Sender alle Meldungen, die nach der letzten, durch BSN bestätigten Meldung, liegen. Wenn keine neuen Meldungen mehr vorliegen, wiederholt der Sender alle nicht quittierten Meldungen. Die zweite als **PCR** (***P**reventive **C**yclic **R**etransmission*) bezeichnete Version wird hauptsächlich auf Strecken mit langen Signallaufzeiten (z. B. Satelliten-Strecken) eingesetzt. In dieser Version wird nur die positive Quittierung verwendet. Nicht quittierte Meldungen werden zyklisch wiederholt, bis sie quittiert werden.

BSN

BIB

PCR

In Abb. 4.5-3 und Abb. 4.5-4 sind Beispiele zur Quittierung in der Basis-Version skizziert. Dabei wird die Übermittlung und Zählung der MSU nur in einer Richtung gezeigt; die gleichzeitige Abwicklung in der anderen Richtung verläuft entsprechend.

Das 4. Byte des Rahmens, *Length Indicator* (LI), gibt die Länge des nachfolgenden Informationsfeldes an. Die ersten beiden Bit werden als Reserve gehalten, so daß LI aus 6 Bit besteht und die Zahlen 0 bis 63 anzeigen kann. Ist LI = 0, so handelt es sich um Füllzeichen (FISU) ohne Informationsfeld. Ist LI = 1 oder 2, so handelt es sich um Zustandsmeldungen (LSSU). Bei LI = 3...63 handelt es sich um Schicht-3-Meldungen (MSU). Falls eine Zeichengabenachricht länger als 62 Byte ist, wird LI = 63 gesetzt. Die maximale Länge des Informationsfeldes ist auf 272 Byte begrenzt.

Wie beim HDLC-Protokoll schließt ein 16 Bit CRC-Feld den Rahmen ab. Es wird das CCITT-Generatorpolynom $x^{16} + x^{12} + x^5 + 1$ verwendet.

Abb. 4.5-3: Fehlerfreie Übertragung in der Basisversion bei SS Nr.7

Abb. 4.5-4: Fehlerbehaftete Übertragung und Korrektur in der Basisversion bei SS Nr.7

Der Ebene 3 des SS Nr. 7 kommen zwei Aufgaben zu: die Zeichengabenachrichtenbehandlung (*Signalling Message Handling*) und das Zeichengabenetzmanagement (*Signalling Network Management*). Zur Zeichengabebehandlung gehören die Unterscheidung der Nachrichten (*Message Discrimination*), deren Verteilung auf die jeweilige Anwendung (*Message Distribution*) und die Lenkung der Nachricht durch das Zeichengabenetz (*Message Routing*). Zum Netzmanagement gehören die Aufgaben des Strecken-, Wege- und Verkehrsmanagements, so z. B. die Inbetriebnahme von Zeichengabestrecken oder deren Ersatzschaltung.

Beispiel 4.5-1: Routing im SS Nr. 7
In einem paketvermittelnden Netz tritt die Aufgabe auf, an jedem Netzknoten den günstigsten Weg für die Übermittlung der Pakete eines jeden Zielknotens zu bestimmen. Gewöhnlich werden an den Netzknoten Routing-Tabellen angelegt aus denen hervorgeht, an welchen Nachbarknoten die Pakete für den jeweiligen Zielknoten abzugeben sind. Solche Tabellen können statisch angelegt sein oder sich aber auch dynamisch an die Verhältnisse im Netz anpassen. Im SS Nr. 7 werden in den Routing-Tabellen jeweils die kürzesten Wege eingetragen (Shortest Path Routing). In der folgenden Skizze ist ein Kommunikationsnetz mit 6 Knoten dargestellt. Die Kantengewichtungen geben die jeweiligen Längen der

Strecken an. In der Praxis kann statt der Länge irgendeine zu optimierende Zielfunktion (bestehend aus Länge, Kosten, Kapazität usw.) angegeben werden.

Versuchen Sie für das Netz die kürzesten Wege zu jedem Knoten vom Knoten A aus zu bestimmen. Sie werden feststellen, daß manchmal mehrere Wege gleichwertig sind. In diesem Fall bevorzugen Sie einen Knoten, der im Alphabet vor einem entsprechenden anderen Knoten liegt (d. h. A vor B, B vor C). Das Verfahren wird als SP-Routing with Selection bezeichnet. Folgende Skizze zeigt die nunmehr eindeutig bestimmten kürzesten Wege (Baum kürzester Wege) für den Knoten A.

Ein Paket zum Knoten E wird nun vom Knoten A an den Knoten B weitergereicht. Um bei Ausfall von Strecken im Netz die Kommunikation aufrecht zu erhalten, enthalten Routing-Tabellen mehrere Einträge, die Ersatzwege angeben.

Man berücksichtigt dabei den Ausfall der vom Knoten unmittelbar ausgehenden Strecken und bestimmt dann wiederum die kürzesten Strecken, um den ersten Ersatz zu finden. Für den Knoten A werden die drei Strecken AB, AC und AD berücksichtigt und man erhält folgende, kürzeste Ersatzbäume:

Ein Paket zum Knoten E wird nun vom Knoten A beim Ausfall von AB an den Knoten C weitergeleitet.

Beispiel 4.5-2: Anforderungen an Routing-Verfahren
Wir betrachten im folgenden einige Nachteile von SP-Routing und einige Anforderungen, die an Routing-Tabellen gestellt werden.

Beim SP-Routing wird die Last im Netz ungleichmäßig verteilt. Eine gleichmäßige Last kann durch Berücksichtigung prozentualer Angaben für die verschiedenen Wege in einer Routing-Tabelle erzielt werden. Zwar gibt SP-Routing mit alternativen Wegen einen gewissen Schutz vor Streckenausfällen, man kann jedoch auch Routing-Tabellen mit minimaler Ausfallwahrscheinlichkeit angeben.

Bei Verwendung von Ersatzwegen in Routing-Tabellen entsteht das Problem, daß Nachrichten im Netz kreisen können (d. h. an einem Knoten wieder ankommen). Es müssen deshalb besondere Maßnahmen ergriffen werden, um ein Kreisen zu vermeiden.

Es wird an Routing-Tabellen gewöhnlich die Anforderung gestellt, daß sie bidirektional sind, d. h. auch beim Ausfall von Strecken muß ein Weg von B nach A existieren, wenn der umgekehrte Weg von A nach B vorhanden ist.

Abb. 4.5-5: Allgemeine Struktur der Ebene-3-Nachricht einer Message Signal Unit

In Abb. 4.5-5 ist die allgemeine Struktur der Ebene-3-Nachricht einer ***M**essage **S**ignal **U**nit* (**MSU**) angegeben. Das **SIO**(***S**ervice **I**nformation **O**ctet*) ist die Anwendungskennung. Sie kennzeichnet den Urheber bzw. Empfänger der Zeichengabenachricht und ermöglicht die Unterscheidungen **MSU, SIO**

- es handelt sich um eine nationale oder internationale Zeichengabenachricht,
- es handelt sich um eine Netzmanagementnachricht (der Ebene 3) oder
- es handelt sich um eine Zeichengabenachricht einer bestimmten Anwendung, z. B. ISDN UP oder TUP usw.

Die Adresse ist in einem Standardkopf (*Label*) angegeben. Diese besteht aus der Zeichenadresse der Zeichengabenachricht (**DPC** - ***D**estination **P**oint **C**ode*), der Ursprungsadresse der Zeichengabenachricht (**OPC** - ***O**rigination **P**oint **C**ode*) und aus 4 Bit für die Zeichengabestreckenauswahl (**SLS** - ***S**ignalling **L**ink **S**election*). Die 14 Bit langen Adressen ermöglichen es, alle nationalen und internationalen Vermittlungsstellen der öffentlichen Netze zu kennzeichnen. **DPC** **OPC** **SLS**

Bei den Zeichengabeverbindungen (Abb. 4.5-6) unterscheidet man zwischen den Quellen und Senken der Nachrichten (*Signalling End Points*) und den Transferstellen (*Signalling Transfer Points*). Die 4 Bit der SLS ermöglichen es, verschiedene Zeichengabewege festzulegen und somit eine Lastverteilung im Zeichengabenetz vorzunehmen (Abb. 4.5-7). Im ISDN wird ein weiteres Byte (**CIC** - ***C**ircuit **I**dentification **C**ode*) für die Angabe des Nutzkanals verwendet. Die CIC zusammen mit OPC und DPC kennzeichnet eindeutig, für welchen Nutzkanal eine Zeichengabenachricht bestimmt ist. **CIC**

Abb. 4.5-6: Ende-zu-Ende und Streckensignalisierung

Abb. 4.5-7: Wegeauswahl im Zeichengabenetz

Wir wollen uns nun den ISDN-Anwenderteil beispielhaft ansehen. In der ursprünglichen Version, die 1984 von CCITT verabschiedet wurde, setzte er direkt auf dem MTP auf. Die entsprechende nationale Spezifikation der DBP (1TR7) war etwas umfangreicher, legte viel mehr Details fest und sah für die Ende-zu-Ende-Signalisierung einen gesonderten Transportteil (**TF** - ***T**ransport **F**unction*) vor. In der CCITT-92-Version, die auch für Euro-ISDN verwendet wird, wird diese Funktion über **SCCP** (***S**ignalling **C**onnection **C**ontrol **P**art*, siehe Abb. 4.5-1 und die Erläuterung dazu) wahrgenommen (siehe Abb. 4.5-8).

TF

SCCP

Abb. 4.5-8: Der Aufbau des ISDN UP in verschiedenen Varianten

Die einzelnen, im ISDN UP festgelegten Zeichengabenachrichten, sind sehr umfangreich. Wir wollen uns diese lediglich an Hand von zwei Beispielen ansehen. Zunächst sehen wir uns den Nachrichtenaustausch für den einfachen Verbindungsaufbau und -abbau im ISDN zwischen zwei Vermittlungsstellen an und anschließend den Aufbau einer Ende-zu-Ende-Zeichengabeverbindung unter Verwendung des SCCP.

Die allgemeine Struktur der Zeichengabenachricht des ISDN UP ist typisch für alle Anwenderteile und ist in Abb. 4.5-9 dargelegt. Sie besteht aus der Bezeichnung der Nachricht (*Message Type*), dem für jede Nachricht festgelegten, zwingend vorgeschriebenen Teil (*Mandatory Part*) und einem optionalen Teil (*Optional Part*). Der zwingend vorgeschriebene Teil ist wiederum unterteilt in einen Teil fester und einen Teil variabler Länge. Derzeit sind über dreißig Nachrichten (*Message Types*) im ISDN UP festgelegt.

Abb. 4.5-9: Die allgemeine Struktur der ISDN UP Zeichengabenachricht

In Abb. 4.5-10 ist der Nachrichtenaustausch zwischen zwei Vermittlungsstellen für den einfachen Verbindungsaufbau und -abbau skizziert.

Abb. 4.5-10: Austausch von Zeichengabemeldungen für den einfachen Verbindungsaufbau und -abbau

IAM Die Nachricht **IAM** (***I**nitial **A**ddress **M**essage*) ist die erste Nachricht, die beim Verbindungsaufbau gesendet wird. Sie dient zur Belegung eines Nutzkanals. Im zwingend vorgeschriebenen Teil fester Länge werden Angaben über das Zeichengabeformat, Verbindungsart, Übertragungsgeschwindigkeit usw. gemacht. Als zwingend vorgeschriebener Teil variabler Länge wird die Rufnummer des B-Teilnehmers angegeben. Es folgen dann optionale Angaben. Zur Zeit sind für die Meldung IAM über zwanzig solcher optionaler Nachrichteninhalte vorgesehen, so z. B. Rufnummer des A-Teilnehmers, Angabe über Rufumleitung, Nummer des Teilnehmers, zu dem der Ruf umgeleitet wird, ursprüngliche Rufnummer, Angabe daß es sich um eine geschlossene Benutzergruppe handelt, Diensteparameter, Kompatibilitätsangaben usw. Es wird deutlich, daß der optionale Teil im wesentlichen zur Realisierung verschiedener Leistungsmerkmale verwendet wird. Falls die Rufnummer des B-Teilnehmers in der IAM-Meldung nicht vollständig vorlag (dies ist bei Einzelzifferwahl im Gegensatz zu Blockwahl möglich), folgt die Meldung **SAM** (***S**ubsequent **A**ddress **M**essage*) mit weiteren Wahlziffern. Die Meldung **ACM** (***A**ddress **C**omplete **M**essage*) zeigt an, daß der gerufene Teilnehmeranschluß frei ist und bei ihm gerufen wird. Die Meldung **ANM** (***An**swer **M**essage*) zeigt an, daß der gerufene Teilnehmer die Verbindung angenommen hat und löst die Gebührenerfassung aus. Die **REL**-Meldung (*Release*) kann von beiden Seiten gesendet werden. Sie leitet den Verbindungsabbau des Nutzkanals ein. Die **RLC** (***Rel**ease **C**omplete*) bestätigt den Verbindungsabbau. Beim erfolglosen Versuch des Verbindungsaufbaus, werden REL und RLC zum Anzeigen des Besetztfalles und Bestätigen des Verbindungsabbaus verwendet. Weitere wichtige Meldungen sind:

SAM
ACM
ANM
REL
RLC

- FAR - *Facility Request*, um Dienstmerkmale anzufordern,
- FAA - *Facility Accepted*, um Dienstmerkmale zu bestätigen und
- FRJ - *Facility Reject*, um Dienstmerkmale abzuweisen.

Wir werden die Realisierung einiger Dienstmerkmale im ISDN im nächsten Abschnitt kennenlernen.

In Abb. 4.5-11 ist der Aufbau einer Ende-zu-Ende-Zeichengabeverbindung unter Verwendung des SCCP dargestellt. Grundidee dabei ist, daß nachdem abschnittsweisen Aufbau der Verbindung durch den ISDN UP die Verbindung direkt, d. h. ohne Mitwirkung des ISDN-UP in den Transit-Vermittlungsstellen, verwendet wird.

Abb. 4.5-11: Aufbau einer Ende-zu-Ende-Zeichengabeverbindung unter Verwendung des SCCP

Der ISDN UP der Ursprungsvermittlungsstelle erfragt zunächst die Daten für die **CR-Meldung** (***C**onnection **R**equest*) vom SCCP ab (Schritt 1 und 2 in Abb. 4.5-11). Die Ende-zu-Ende-Zeichengabeverbindung wird nun durch Senden der Nachricht CR als Parameter der Meldung IAM oder FAR (die wir bereits kennengelernt haben) angefordert (Schritt 3 in Abb. 4.5-11). Diese Nachricht wird abschnittsweise von Vermittlungsstelle zu Vermittlungsstelle bis zur Zielvermittlungsstelle weitergeleitet (Schritt 4). Hier übergibt der ISDN UP die Meldung CR an den SCCP (Schritt 5), der die Meldung verarbeitet und den ISDN UP über die Verbindungsaufforderung informiert (Schritt 6). Die Bestätigung des ISDN UP (Schritt 7) leitet nun der SCCP als **CC** (***C**onnect **C**onfirm*) direkt (d. h. über den MTP) an den SCCP der Ursprungsvermittlungsstelle (Schritt 8). Dieser informiert wiederum seinen ISDN UP. Somit steht nun die Ende-zu-Ende Verbindung. Die über sie laufenden Meldungen werden nun in den Transitvermittlungsstellen ohne jegliche Bearbeitung im ISDN UP weitergeleitet. Der Verbindungsabbau erfolgt ähnlich.

CR-Meldung

CC

Beispiel 4.5-3: Blindbelegungszeiten
Man kann davon ausgehen, daß in herkömmlichen Fernsprechnetzen etwa 18 % aller Belegungszeiten Blindzeiten in dem Sinne sind, daß Verbindungsabschnitte belegt werden, ohne daß hierfür Gebühren eingenommen werden. Grob werden diese Zeiten zu je einem Drittel benötigt für die Wahl (Inband), für die Freiprüfung und Durchschaltung der Verbindung und für den Ruf beim Teilnehmer. Durch konsequente Verwendung moderner Signalisierverfahren (insbesondere SS Nr. 7 und D-Kanal Signalisierung im ISDN) können die Blindzeiten wesentlich vermindert werden. Maßnahmen, die getroffen werden können, sind:

- Automatische Wahl (Kurzwahl vom Endgerät, Blockwahl statt Einzelziffernwahl im Netz),
- Freiprüfung (Indirektwahl mit konjugierter Durchwahl für das Gesamtnetz + Freiprüfung beim Teilnehmer),
- Durchschaltung nach dem Abheben beim Teilnehmer.

Alle drei Maßnahmen finden heute Anwendung und dürften die Blindbelegungszeit auf unter 8% reduzieren.

Beispiel 4.5-4: Belastung der Steuerungen durch neue Dienste
Durch den Einsatz von Kurzwahl und SS Nr. 7 wird der Verbindungsaufbau für nationale Verbindungen unter 1 Sekunde und internationale Verbindungen unter 2 Sekunden (mit 99% Wahrscheinlichkeit ohne Berücksichtigung der Satellitenstrecken) gedrückt. Die mittlere Verbindungsdauer von 100 Sekunden für Fernsprechverbindungen dürfte bei neuen Diensten (ISDN-Fax, ISDN-Filetransfer, Telemetrie im ISDN) im ISDN unter 10 Sekunden liegen. Dieser Wert scheint hoch genug, wenn man berücksichtigt, daß in 10 Sekunden bei 64 kbit/s netto etwa 50 KByte (beispielsweise bei etwa 2 KByte pro Seite etwa 25 Seiten) übertragen werden können. Zur Zeit ist die Anzahl solcher Anwendungen im ISDN noch klein, längerfristig kann jedoch davon ausgegangen werden, daß sie erheblich steigt. Nimmt man an, daß die Anzahl soweit ansteigt, daß die Hälfte der Verbindungen von den neuen Diensten verursacht werden, so sinkt die mittlere Verbindungsdauer auf 55 Sekunden. Da aber doppelt so viele Verbindungen aufgebaut werden, steigt die Belastung der Steuerung im Verhältnis überproportional auf rund das Vierfache!

Selbsttestaufgabe 4.5-1:

Nennen Sie Vorteile der Zentralkanalsignalisierung gegenüber kanalindividuellen Verfahren!

4.6 Dienste und Leistungsmerkmale

ISDN zeichnet sich durch folgende Eigenschaften aus:

- Voll digitale Nutzkanäle hoher Bitrate (64 kbit/s) bis zum Teilnehmer.
- Ein starker, vom Nutzkanal unabhängiger Signalisierkanal (16 kbit/s) bis zum Teilnehmer.
- Eine geringe Bitfehlerrate (10^{-5} im Nutzkanal, 10^{-11} für die Signalisierung).
- Schneller Verbindungsaufbau (< 2 Sekunden).
- Eine einheitliche Steckdose für alle Endgeräte.
- Eine einheitliche Rufnummer.
- Umstecken eines Endgerätes am Bus während der Diensteabwicklung ist möglich.
- Ein Dienstwechsel auf einer aufgebauten Verbindung ist möglich.
- Eine homogene Diensteabwicklung (einheitliche Prozeduren, einheitliche Benutzeroberflächen) wird möglich.
- Ein Besetztfall kann (durch Anklopfen s. unten) umgangen werden.

Beispiel 4.6-1: Mächtigkeit des D-Kanals
Die Wahlempfänger werden so ausgelegt, daß die Rufnummer mit einer Geschwindigkeit von maximal 10 Zeichen pro Sekunde eingegeben werden dürfen. Dies entspricht etwa 600 Zeichen pro Minute - eine Grenze, die auch Spitzen-Schreibkräfte selten erreichen dürften. Verwendet man eine 4 Bit Codierung, so entspricht dies 40 bit/s. Im Verhältnis zum Signalisierkanal mit 16 kbit/s im ISDN sind dies nur 2,5 Promille.

Eine Signalisiernachricht im ISDN hat häufig eine Länge von 268 Byte. Die Übertragung dieser Nachricht mit 16 kbit/s dauert 0,134 Sekunden. Im Vergleich zur Modemübertragung mit 2,4 kbit/s ist dies fast sieben Mal schneller.

Dienste

Herkömmliche **Dienste** wie Fernsprechen, Btx, Fax, Datendienste usw. werden im ISDN entweder direkt realisiert oder über Terminaladaptoren (TA, siehe Abschnitt 4.1) zugänglich gemacht. Häufig können wegen Wahrung der Kompatibilität die Vorteile des ISDN nicht genutzt werden; es wird also die gleiche Dienstgüte wie bei den bestehenden Diensten angeboten, gelegentlich sogar eine Verschlechterung in Kauf genommen. Bestehen solche Kompatibilitätseinschränkungen nicht, so können diese Dienste mit einer erhöhten Dienstgüte angeboten werden. Die Verbesserung bezieht sich dann auf eine höhere Übermittlungsgeschwindigkeit, eine bessere Signalauflösung oder verbesserte Leistungsmerkmale. Zu den weiteren Diensten, die im ISDN realisiert werden, gehören Bildfernsprechen, Still- und Bewegtbildübertragung, Datenfernübertragung (DFÜ), Internet, E-Mail, Diensteumsetzungen und andere Mehrwertdienste.

Übermittlungsdienste

Dienste werden dem ISO-Modell (siehe Kapitel 1) entsprechend in **Übermittlungsdienste** (*Bearer Services*), auch Transportdienste genannt, und

Teledienste

Teledienste (*Teleservices*), auch Standarddienste genannt, unterteilt. Übermittlungsdienste enthalten Funktionen der Schicht 1 bis 3 des OSI-Referenzmodells; bei Telediensten sind Funktionen aller Schichten 1 bis 7 des OSI-Modells spezifiert. Typische Teledienste sind Fernsprechen, Telefax, Internet, E-Mail usw. In Abb. 4.6-1 ist die Realisierung dieser Dienste in einem Transportnetz anschaulich dargestellt.

Abb. 4.6-1: Anschauliche Darstellung der Realisierung von Diensten in einem Kommunikationsnetz

Obwohl im ISDN auch Teledienste angeboten werden, ist ISDN im Grunde ein ausgeprägtes Transportnetz. Die Übermittlungsdienste im ISDN werden künftig erheblich an Bedeutung zunehmen. Diese bestehen aus leitungsvermittelten Diensten (*Circuit Swichted Services*), auch Durchschaltedienste genannt, und paketvermittelten Diensten (*Packet Switched Services*). Bei den **leitungsvermittelten Diensten im ISDN** wird die für den Verbindungsaufbau und -abbau sowie die Aktivierung der Leistungsmerkmale erforderliche Signalisierung im D-Kanal ausgeführt, während die Nutzdaten über den B-Kanal übertragen werden. Der B-Kanal wird transparent durchgeschaltet; es werden lediglich Funktionen der Schicht 1 des OSI-Modells ausgeführt. Die Abläufe für den Verbindungsaufbau und -abbau bei der leitungsvermittelnden Datenübermittlung haben wir bereits im vorangegangenen Abschnitt kennengelernt.

leitungsvermittelte Dienste im ISDN

Beispiel 4.6-2: Übertragungszeiten im ISDN und LAN
Beim folgenden Vergleich gehen wir davon aus, daß im ISDN für Nutzdatenübertragung 64 kbit/s und im LAN 10 Mbit/s zur Verfügung stehen. Tatsächlich ist es so, daß die Kapazität eines LANs dem Teilnehmer voll zur Verfügung steht, allerdings erst, wenn er an der Reihe ist. Insofern ist der Vergleich zugunsten von LANs verfälscht. Nun haben Untersuchungen gezeigt, daß heute installierte LANs selten eine Spitzenauslastung von 4% oder darüber erreichen, was wiederum den Vergleich rechtfertigt.

Eine 160 KByte Datei wird im ISDN in 20 Sekunden, im LAN in 0,1 Sekunden übertragen. Bereits hier erreicht man bei ISDN die Grenze, die ein geduldiger Anwender setzen dürfte.

Die Übertragung eines 70 MByte großen Videos erfordert im ISDN 2,5 Stunden Übermittlungsdauer, im LAN ist es gerade 1 Minute!

Bei der Echtzeitübermittlung, so z. B. Sprache, zeigt sich der Vorteil der Durchschaltevermittlung. In LANs entsteht in der Regel eine Verzögerung von einigen 100 ms, während diese im ISDN lediglich im μs-Bereich anfällt. Natürlich steht im LAN auch die gesamte Kapazität nicht längere Zeit zur Verfügung.

paketvermittelte Dienste im ISDN

Für die Realisierung der **paketvermittelten Dienste im ISDN** gibt es zahlreiche Möglichkeiten. Zunächst stellt sich die Frage, ob über ISDN lediglich ein Zugang zu paketvermittelten Netzen angeboten wird, oder ob die Paketvermittlung im ISDN integriert zur Verfügung steht (siehe Abb. 4.6-2). Im ersten Fall wird eine virtuelle Verbindung zwischen zwei am ISDN angeschlossenen Endgeräten über einen im Paketvermittlungsnetz befindlichen Knoten geführt. Im zweiten Fall sind die Assemblierung - Deassemblierung (*Packethandling*) und alle Paketvermittlungsfunktionen im ISDN integriert. Hier bieten sich wiederum eine dezentrale Lösung mit Paketvermittlungsfunktionen in jeder ISDN-Vermittlung und eine zentrale Lösung mit wenigen solchen Paketvermittlungsknoten an. In den meisten Ländern (so auch im Netz der Telekom) wird zunächst die Übergangslösung realisiert und längerfristig die dezentrale, integrierte Lösung angestrebt.

Abb. 4.6-2: Möglichkeiten für die Realisierung der Paketvermittlung im ISDN

Unabhängig davon, ob die Übergangslösung oder die integrierte Lösung für die Paketvermittlung im ISDN gewählt wird, besteht die Möglichkeit, für die Übermittlung der Nutzdatenpakete zwischen den drei Kanälen, die zum Teilnehmer führen, zu wählen. Zunächst muß entschieden werden, ob der D- oder ein B-Kanal für die Nutzdatenübermittlung verwendet wird. Fällt die Entscheidung für einen B-Kanal, so muß entschieden werden, ob ein B-Kanal neu belegt wird oder ein für Datenpaketübermittlung bereits genutzter B-Kanal mitverwendet werden kann. Die Auswahl zwischen der Paketübermittlung im B- oder D-Kanal hängt vom angeschlossenen Endgerät (bzw. dem entsprechenden Terminaladapter) ab. Da eine ISDN-Vermittlung nicht über die Information verfügt, welche Terminaladaptoren an einem S-Bus angeschlossen sind, muß sie beim ankommenden Ruf alle Möglichkeiten anbieten und die gewünschte dann umsetzen. Hierfür wird eine von CCITT empfohlene sogenannte *Call Offering Procedure* angewendet. Beim abgehenden Ruf entscheidet das rufende Endgerät darüber, ob ein B- oder D-Kanal verwendet wird.

Für die Signalisierung bei der Paketvermittlung im ISDN gibt es wiederum zwei Lösungen. Die als Lösung in zwei Schritten (*Two Step Solution*) bekannte Variante verwendet zunächst die Signalisierung im D-Kanal, um die Strecke im ISDN aufzubauen und anschließend die X.25-Signalisierung, um die virtuelle Verbindung aufzubauen. Diese Variante ist für die vorher besprochene Übertragungslösung zwingend erforderlich. Bei der Lösung in einem Schritt (*One Step Solution*) wird auch

der Aufbau der virtuellen Verbindung durch die D-Kanal Signalisierung erwirkt. Diese Variante ist nur im Falle der vorher besprochenen integrierten Lösung realisierbar.

In Abb. 4.6-3 ist der Ablauf beim Aufbau einer ersten virtuellen Verbindung im B-Kanal entsprechend der Zweischrittlösung beispielhaft dargestellt. Beim Aufbau der Schicht-2-Verbindung wird SAPI s=0 für die Paketvermittlung im B-Kanal angegeben. Die Schicht-3-Meldung *SETUP* belegt den B-Kanal. Nachdem der B-Kanal transparent geschaltet ist, wird mit der üblichen X.25-Signalisierung im B-Kanal fortgefahren. Auf der gerufenen Seite wird wiederum über die D-Kanal-Signalisierung ein B-Kanal aufgebaut. An dieser Stelle wird die *Call Offering Procedure* eingeleitet um die Auswahl eines Kanals zu ermöglichen. Nach dem Durchschalten des neuen B-Kanals wird die X.25-Signalisierung in diesem Kanal fortgesetzt. Die Datentransferphase entspricht der X.25-Prozedur. Der anschließende Verbindungsaufbau verläuft wiederum in zwei Schritten - zunächst wird die X.25-Verbindung abgebaut und anschließend erst nach dem Abbau der letzten X.25-Verbindung wird auch der B-Kanal abgebaut.

Abb. 4.6-3: Aufbau einer ersten virtuellen Verbindung über den B-Kanal. (Zweischrittlösung)

In Abb. 4.6-4 ist der Ablauf beim Aufbau einer virtuellen Verbindung über den D-Kanal entsprechend der Zweischrittlösung beispielhaft dargestellt. Zunächst wird eine Schicht-2-Verbindung mit SAPI p=16 aufgebaut, soweit diese nicht bereits existiert. Unter Verwendung der Schicht-2-Meldung I wird nun der X.25-Verbindungsaufbau eingeleitet. Auf der gerufenen Seite wird zunächst über UI mit SAPI s=0 und *Global TEI* die *Call Offer Procedure* eingeleitet. Die Wahl des D-Kanals leitet den Abbau der Signalisierverbindung (mit SAPI s=0) und den Aufbau der Verbindung für X.25 Signalisierung (mit SAPI p=16) ein. Die restlichen Abläufe sind nach der bisherigen Abhandlung leicht nachvollziehbar.

Abb. 4.6-4: Aufbau einer virtuellen Verbindung im D-Kanal. (Zweischrittlösung)

Die CCITT Empfehlungen sehen eine schnelle Paketvermittlungstechnik im ISDN in der integrierten Version mit der Einschrittlösung vor. Charakteristika dieser als **New Packet Mode** (neuer Paketvermittlungsmodus) bezeichneten Technik sind: **New Packet Mode**

- Die Signalisierung (auch für den Auf- und Abbau einer virtuellen Verbindung) wird ausschließlich im D-Kanal (bzw. SS Nr. 7) vorgenommen.
- Die Signalisierung für den Auf- und Abbau der Verbindungen in D- bzw. B-Kanälen und der virtuellen Verbindungen wird in einem Schritt vorgenommen (Ein-Phasenwahl bzw. Einschrittlösung).
- Alle für die Übermittlung der Pakete erforderlichen Funktionen werden in die Schicht 2 verlagert. Es findet also eine Übermittlung der Schicht-2-Rahmen im Netz statt (*Frame Relay* bzw. *Frame Switching*).
- Das Multiplexen und das Vermitteln von virtuellen Verbindungen wird entsprechend in der Schicht 2 vorgenommen. Hierfür wird anstelle von TEI und SAPI ein **DLCI** (***D**ata **L**ink **C**onnection **I**dentifer*), welcher der logischen Kanalnummer in X.25 entspricht, verwendet (siehe Abb. 4.6-5). **DLCI**

- Es werden zwei Rahmenübermittlungsdienste (*Frame Mode Bearer Services*) angeboten: *Frame Relaying Bearer Service* und *Frame Switching Bearer Service*. Der wesentliche Unterschied zwischen den beiden Übermittlungsverfahren liegt im Quittierungsmechanismus für die Datenübermittlung. Beim *Frame Relaying* werden die Sicherung und eine eventuelle Wiederholung der Rahmen den Endeinrichtungen überlassen. Beim *Frame Switching* wird die Datenübermittlung abschnittsweise durch die ISDN Vermittlungsstellen gesichert und quittiert; eine Flußkontrolle wird vorgenommen.
- Es werden Maßnahmen zur Überlastabwehr getroffen. Rahmen, die bei Überlast verworfen werden dürfen, können markiert werden.

8	7	6	5	4	3	2	1
DLCI						C /R	EA (0)
DLCI				FECN	BECN	DE	EA (1)

BECN Backward Explicit Congestion Notification
C/R Command / Response
DE Discard Eligibility
DLCI Data Link Connection Identifier
EA Extended Address
FECN Forward Explicit Congestion Notification

Abb. 4.6-5: Adreßfeld des Rahmens bei *Frame Relaying Bearer Service*

Beispiel 4.6-3: Bildübertragung in verschiedenen Netzen
Ein Super-VGA Bild besteht aus 800 x 600 Pixeln und 256 Farben und ergibt (bei 8-Bit-Codierung) ein Datenvolumen von 3,84 Mbit pro Bild. Im folgenden sind die Übertragungszeiten in verschiedenen Netzen zusammengestellt.

ISDN	1 B-Kanal	64 kbit/s	1 Bild/min
	2 B-Kanäle	128 kbit/s	2 Bilder/min
	30 B-Kanäle	1,92 Mbit/s	30 Bilder/min
CSMA/CD		10 Mbit/s	2,6 Bilder/s
Token Ring		16 Mbit/s	4,2 Bilder/s
FDDI		100 Mbit/s	26 Bilder/s
DQDB		140 Mbit/s	36 Bilder/s
CRMA-II		1,25 Gbit/s	5 Kanäle à 65 Bilder/s

Die Übertragungszeiten können durch Datenkompression (Redundanzreduktion) herabgesetzt werden. Bei Studioübertragungen stellt man hohe Anforderungen an die Qualität und verlangt eine verlustfreie Kompression. Man erreicht eine verlustfreie Kompression bis zu etwa 1 : 10. Erlaubt man einen geringen Datenverlust, so sind Werte zwischen 1 : 10 bis 1 : 50 erreichbar - wobei die höhere Kompression lediglich bei kleinen Bildschirmen (Videofilme) akzeptabel ist. Auf diese Weise wird Bildtelefonie bzw. Bewegtbildübertragung im ISDN möglich.

Wir wenden uns nun den allgemeinen **Leistungsmerkmalen** zu, die im ISDN typischerweise angeboten werden. Im folgenden sind einige dieser Leistungsmerkmale aufgezählt und kurz kommentiert:

Leistungsmerkmale

Anklopfen Bei einem besetzten Anschluß wird ein ankommender Ruf durch Anklopfen (Ton) und ggf. Einblenden der Nummer des Anrufers angezeigt.

Anrufliste Alle Anrufe, die ankommen und nicht bedient werden, werden beim Gerufenen in eine Liste mit der Nummer des Anrufers und der Uhrzeit eingetragen.

Anrufumleitung Der ankommende Ruf wird an eine vom Gerufenen vorher eingegebene bzw. aktivierte Rufnummer weitergeleitet. Es gibt verschiedene Varianten dieses Leistungsmerkmales, so z. B. die Weiterleitung auch der Rufnummer des Anrufers, die Mitteilung an den Rufenden, daß sein Anruf umgeleitet wird oder eine zeitgesteuerte Anrufumleitung.

Anrufweiterleitung Der ankommende Ruf wird im Besetztfall ggf. bei keiner Anwort (zeitgesteuert) an eine andere (festgeschaltete oder eingebbare) Rufnummer weitergeleitet.

Anzeige der Rufnummer des Anrufers beim Angerufenen. Das Leistungsmerkmal wird erst durch die D-Kanal und SS Nr. 7 Signalisierung möglich. Es gibt für den Rufenden die Möglichkeit, die Anzeige wahlweise zu unterbinden.

Gebührenanzeige Dieses Leistungsmerkmal wird herkömmlich über Zählerimpulse, im ISDN über die Signalisierung realisiert.

Gebührenübernahme durch den Angerufenen. Dieses Leistungsmerkmal ist in vielen Varianten möglich, so z. B. komplette Übernahme der Gebühr, Berechnung von Ortsgebühr, wahlweise Übernahme der Gebühr. Die Realisierung wird durch die Signalisierung möglich.

Konferenzschaltung Dieses Leistungsmerkmal ermöglicht es, ein Telefongespräch zwischen drei oder mehr Teilnehmern (als Konferenz) zu führen. Es erfordert eine gewisse Gesprächsdisziplin, vor allem, wenn mehrere Partner beteiligt sind. Die Realisierung des Leistungsmerkmales erfordert eine gesonderte Schaltung in der Vermittlung, da digitale, nicht linear codierte Signale nicht einfach superponiert werden können.

Kurzwahl Häufig benutzte Rufnummern können eingegeben werden und werden durch Knopfdruck angewählt. Die Realisierung wird durch Speicher im Endgerät, gelegentlich auch in der Vermittlung vorgenommen.

Makeln Ein bestehendes Gespräch wird per Knopfdruck kurzfristig unterbrochen, um ein weiteres Gespräch (Rückfrage) zu führen. Anschließend wird das Gespräch wieder aufgenommen.

Notizblock Dieses Leistungsmerkmal ermöglicht das Vermerken von Rufnummern. Die Realisierung wird durch Speicher im Endgerät, gelegentlich auch in der Vermittlung vorgenommen.

Rückruf Ist ein Anschluß besetzt, so kann der Rufende den automatischen Rückruf einleiten. Beim Freiwerden des Anschlußes wird zunächst er und dann der zuvor Gerufene angerufen. Die Realisierung des Leistungsmerkmales erfordert eine schnelle Signalisierung und eine flexible Vermittlungssteuerung.

Sperren Ein Endgerät oder Anschluß wird für Auslandswahl, Fernwahl oder Externverkehr gesperrt. Das Sperren wird durch eine Berechtigungsprüfung in der Vermittlung wirksam.

Wahlwiederholung Die letztgewählte Rufnummer wird vermerkt und durch Knopfdruck wieder angewählt. Die Realisierung wird durch einen Speicher im Endgerät, gelegentlich auch in der Vermittlung vorgenommen.

Intelligentes Netz

Ein Netz in dem solche komplexen Leistungsmerkmale angeboten werden, kann als ein **Intelligentes Netz** (**IN** - ***I**ntelligent **N**etwork*) angesehen werden. Seit den Arbeiten 1986-88 bei Bell, IBM und Siemens auf diesem Gebiet verbindet man eine bestimmte Systemarchitektur mit dem Begriff des Intelligenten Netzes.

In Abb. 4.6-6 oben sind zunächst die Funktionen für die Abwicklung der Basisdienste in einem Kommunikationsnetz wie ISDN in der IN-Terminologie dargestellt. Die Orts bzw. Endvermittlungsstelle wird als CCAP - (***C**all **C**ontrol **A**ccess **P**oint* d. h. Punkt an dem die Basisdienste zugänglich sind) und die Transitvermittlungen als CCP - (***C**all **C**ontrol **P**oint* d. h. Verbindungssteuerungspunkt) bezeichnet. In Abb. 4.6-6 unten sind nun die Komponenten des IN und deren Einbettung im Kommunikationsnetz dargestellt. Die SSPs (***S**ervice **S**witching **P**oints*) können an beliebigen Vermittlungsstellen angebracht werden. Es handelt sich dabei um ein Software-Modul (*Service Switching Function*), das die Vermittlungsfunktion ergänzt. Ihre wesentliche Aufgabe besteht darin, beim Verbindungswunsch zu erkennen, daß es sich um einen IN-Dienstewunsch handelt, daraufhin über SS Nr. 7 mit dem SCP (***S**ervice **C**ontrol **P**oint* - Dienstesteuerungspunkt) zu kommunizieren und die entsprechenden vermittlungstechnischen Funktionen einzuleiten. Die Kommunikation wird in Echtzeit durchgeführt, die einzelnen kurzen Meldungen werden im Millisekundenbereich abgewickelt. Der SCP liefert dem SSP die für die Weiterverarbeitung des IN-Wunsches erforderlichen Daten und Steuerinformationen; der SSP kann somit die unterschiedlichsten IN-Dienste vermitteln. Die eigentlichen Steuerprozeduren für die Abwicklung der unterschiedlichen Dienste liegen im SCP. In einem IN können mehrere SCP vorhanden sein. Da sie jeweils mehrere Vermittlungsstellen bedienen, erleichtern sie die Einführung von neuen Diensten im IN. Der Betrieb und die Wartung des IN werden vom SMP (***S**ervice **M**anagement **P**oint*) wahrgenommen. Hier befinden sich die Dienstedatenbanken und es wird das Sammeln und Auswerten der Verkehrs-, Gebühren- und Statistikdaten vorgenommen. Im IP (***I**ntelligent **P**eripheral*) sind Zusatzfunktionen verfügbar, die in den Vermittlungsstellen nicht vorhanden sind, für die Abwicklung der IN-Dienste aber

benötigt werden; dies wären z. B. Spracherkennungs- und -ausgabesysteme oder Frequenzwahlauswertung. Beim VFN (***V**endor **F**eature **N**ode*), auch SPN (***S**ervice **P**rovider **N**ode*) genannt, handelt es sich um SCP-ähnliche Einrichtungen, die es auch anderen Anbietern (außer dem Netzbetreiber) ermöglichen, Mehrwertdienste zu realisieren und im IN anzubieten.

CCAP	Call Control Access Point	SSP	Service Switching Point
CCP	Call Control Point	SMP	Service Management Point
IP	Intelligent Peripheral	STP	Signalling Transfer Point (des SS NR. 7)
SCP	Service Control Point	VFN	Vendor Feature Node

Abb. 4.6-6: Zur Architektur des Intelligenten Netzes

Typische IN-Dienste, die realisiert wurden, sind:

Free Phone, Service 0800 Bei diesen Diensten handelt es sich darum, daß der Dienstteilnehmer im gesamten (Landes-) Netz unter einer Rufnummer, die mit 0-800 beginnt, erreicht werden kann. Die Gebühren für den Anruf übernimmt der Angerufene. Die SSP erkennt, daß es sich um einen IN-Dienst handelt und übergibt die 800er Rufnummer an den SCP, der dann dem SSP die relevanten Rufdaten zur Verfügung stellt. Die Gebührendaten werden anschließend dem SMP übermittelt. Verschiedene Leistungsmerkmale für diesen Dienst sind möglich, so z. B. ursprungsabhängige Zielansteuerung, zeitabhängige Zielansteuerung, kennungsabhängige Zielansteuerung, Feinselektierung durch Bedienerführung (mit Hilfe der IP), Standard- oder vom

Dienstteilnehmer eingebbare Ansagen für den Anrufer, Wartebetrieb. Typische Anwendungen des Dienstes sind Informations-, Bestell-, Beratungs-, Buchungs-, Reservierungs- und Kundenbetreuungsservice.

Landesweite, einheitliche Rufnummer, Service 180 Der Dienst ist dem Service 0800 sehr ähnlich - der wesentliche Unterschied besteht darin, daß die Gebühren vom Anrufenden getragen werden. Typische Anwendungen sind Informations- und Bereitschaftsdienste (Wettervorhersage, Ärztebereitschaft).

Telefondienst, Privater Informationsdienst, Audiotex, Service 0900 Der Dienst ist dem Service 180 sehr ähnlich - der wesentliche Unterschied besteht darin, daß die Informationen gebührenpflichtig sind. Die Informationen werden häufig unter Benutzerführung in Dialogform angeboten. Der Dienst kann als Sprach-, aber auch als Textdienst realisiert sein.

Televotum (TED) Bei diesem Dienst können (z. B. in Fernsehsendungen) Abstimmungen per Telefon vorgenommen werden. Eine Televotumverbindung braucht beim SSP nicht durchgeschaltet, sondern lediglich gezählt und das Ergebnis dem Dienstteilnehmer mitgeteilt zu werden. Eine Alternative besteht darin, daß jeder 100. Anruf zum Dienstteilnehmer geleitet wird und dieser die Zählung in dieser Stufung selbst vornimmt. Der Dienst führt zur Vermeidung von Überlast bzw. besserer Nutzung der Ressourcen. Eine Gebührenübernahme durch den Dienstteilnehmer ist möglich.

PCN

PIN

Eine Weiterentwicklung des Intelligenten Netzes in ein personenbezogenes Netz (**PCN** - ***P****ersonal* ***C****ommunications* ***N****etwork*) zeichnet sich (insbesondere bei digitalen, zellularen Mobilfunknetzen wie z. B. GSM) bereits ab. In einem PCN-Netz wird die Rufnummer nicht endgerätebezogen, sondern personenbezogen verwaltet. Der Teilnehmer erhält eine Chipkarte und eine **PIN** (***P****ersonal* ***I****dentification* ***N****umber*) als Berechtigung. Damit kann er an beliebigen, dafür geeigneten Endgeräten die angebotenen Dienste in Anspruch nehmen. Die Abrechnung erfolgt personenbezogen. Umgekehrt werden die an ihn gerichteten Anrufe an das Gerät (oder Geräte), an dem er sich befindet, weitergeleitet. Die Realisierung dieses Dienstes erfordert außer einer schnellen Signalisierung eine verteilte Datenbank. Im SCP, der für den Bereich, in dem der Teilnehmer angemeldet ist, zuständig ist, wird ein Datensatz (*Local Register*) für ihn angelegt. In einem SCP, in dessen Bereich sich ein Teilnehmer befindet bzw. sich das erste Mal meldet, wird ein weiterer Datensatz (*Visitor's Location Register*) angelegt. Die beiden SCP tauschen die erforderlichen Daten über SS Nr. 7 aus, so daß die Berechtigungsprüfungen bei der besuchten SCP und die Anrufweiterleitung und Abrechnungen bei der Heimat-SCP vorgenommen werden können.

5 B-ISDN: Die ATM-Technik

5.1 ATM als Basis für B-ISDN

In der Vergangenheit haben sich zwei Übermittlungsverfahren für die technische Kommunikation überwiegend durchgesetzt[3]:

- die Durchschaltevermittlung (*circuit switching*) unter Verwendung der Zeitmultiplextechnik, die auch bei der Digitalisierung des Fernsprechnetzes und deren Aufrüstung zum ISDN eingesetzt wird, und
- die Paketvermittlungstechnik (*packet switching*), die sich insbesondere für die Datenübermittlung eignet und im Prinzip auch in Lokalen Netzen angewandt wird.

Ein wesentlicher Nachteil der Durchschaltevermittlung besteht in den starren Bitraten die übermittelt werden und die damit verbundene Unfähigkeit, Ruhezeiten der Datenquellen zu nutzen oder sporadische Quellen adäquat zu bedienen. Ein wesentlicher Nachteil der Paketvermittlung auf der anderen Seite besteht in den langen schwankenden Wartezeiten und damit verbundenen Problemen bei der Echtzeit- oder plesiochronen Übermittlung.

Bereits recht früh wurden Anstrengungen unternommen, diese Nachteile durch die Integration sowohl der Zeitmultiplex- als auch der Paketvermittlung in einem Koppelnetz zu integrieren. Anfang der achtziger Jahre strebte man an, eine **schnelle Paketvermittlung** durch unterschiedlichste Maßnahmen zu entwickeln. Typische Maßnahmen waren:

schnelle Paketvermittlung

- schnelle Technologien für die Durchschaltung und die Übertragung
- parallele Verarbeitung
- eine Reduktion der Komplexität der Protokolle, so z. B. durch Weglassen der abschnittsweisen zyklischen Fehlerkorrektur und der Flußkontrolle
- Bildung von kleinen, für die Übermittlung optimalen Paketen
- Verschiebung von allen für die Übermittlung erforderlichen Funktionen in die niedrigen Schichten (des OSI Modells).

Zwei solche konkreten Ansätze waren das FPS (***F**ast **P**acket **S**witching*), das von Bell Labs 1983 vorgestellt wurde und eine variable Paketlänge von im Mittel 100 Bytes hatte, und das ATD (***A**synchronous **T**ime **D**ivision*), das etwa zur gleichen Zeit von CNET vorgestellt wurde und eine kurze feste Paketlänge von 8 bzw. 32 Bytes hatte. Die FPS und ATD Arbeiten gelten als der Ursprung für die ATM-Technik (***A**synchronous **T**ransfer **M**ode*).

3 F. Kaderali: Neue Kommunikationstechniken für Anwender: Ein Überblick
In: Corporate Networks und neue Techniken, Proceedings des Telekom-Anwender-Kongesses 1994, Vieweg Verlag, 1995

Unter dem Begriff *Broadband ISDN* begannen in den 80er Jahren beim CCITT (heute: ITU-T) Standardisierungsarbeiten mit dem Ziel der Fortschreibung und Ergänzung der bestehenden ISDN-Standards für breitbandige Dienste. Gemäß dem ISDN-Prinzip, unterschiedliche Dienste über nur einen Netzzugang mit einheitlicher Schnittstelle zwischen Benutzer und Netzbetreiber abzuwickeln, konzentrierten sich die Standardisierungsaktivitäten für das Breitband-ISDN (B-ISDN) auf die *User Network Interface* (UNI). Die Prinzipien der Integration von Audio-, Video- und Datendiensten, die bereits dem Schmalband-ISDN zugrunde liegen, wurden in entsprechender Weise auf das Breitband-ISDN übertragen und in Grundzügen im *Blue Book* des CCITT (1988) festgelegt. Gleichzeitig wurde der Asynchrone Transfer Modus als Basis des B-ISDN empfohlen.

Technologische Innovationen bei Speicherbausteinen und Prozessoren führten dazu, daß die Leistungsfähigkeit der Endsysteme stark zunahm. Der Preisverfall bei Komponenten und Systemen begünstigte gleichzeitig die weite Verbreitung der neuen Technologien und damit den wachsenden Bedarf nach schnellen Weitverkehrsverbindungen. Kommunikationsanwendungen mit hochauflösenden Grafiken, HiFi-Audio und Video stellten hohe Anforderungen an die Übertragungskapazität, die die Grenzen der Leistungsfähigkeit des 64 kbit/s-ISDN aufzeigten. Diese Entwicklung deutete sich bereits während der CCITT-Studienperiode 1984-1988 an und es war offensichtlich, daß auch der Normungsprozeß in der Telekommunikation dem Bedarf entsprechend beschleunigt werden mußte. Während mit der Empfehlung I.121 im Jahre 1988 lediglich ein grober Rahmen abgesteckt wurde, entstanden, nach Maßgabe eines schnelleren Abstimmungsverfahrens, in den folgenden 2 Jahren 13 Normen zum ATM, die 1991 in Genf und 1993 in Helsinki erweitert und ergänzt wurden. In Tabelle 5.1-1 ist eine Übersicht über wesentliche CCITT-Empfehlungen zu B-ISDN und insbesondere ATM zusammengestellt.

Tab. 5.1-1: *ITU-T Empfehlungen der I-Serie zum B-ISDN*

I.113	Vocabulary of terms for Broadband aspects of ISDN
I.121	Broadband Aspects of ISDN
I.150	B-ISDN Asynchronous Transfer Mode functional characteristics
I.211	B-ISDN Service Aspects
I.311	B-ISDN General network aspects
I.321	B-ISDN Protocol Reference Model and its application
I.327	B-ISDN Functional Architecture
I.350	Quality of Service (QoS) for Bearer Service and Tele-Service
I.356	B-ISDN ATM-Layer cell transfer performance
I.361	B-ISDN ATM-Layer Specification
I.362	B-ISDN ATM Adaptation Layer (AAL) functional description
I.363	B-ISDN ATM Adaptation Layer (AAL) specification
I.364	Support of Broadband Connectionless Data Service on B-ISDN
I.370	Congestion Management for the ISDN Frame Relaying Bearer Service
I.371	Traffic Control and Congestion Control in B-ISDN
I.413	B-ISDN User-Network-Interface
I.414	Overview of Recommendations on layer 1 for ISDN and B-ISDN customer access
I.432	B-ISDN User-Network-Interface - Physical Layer Specification
I.580	General Arrangements for Interworking between B-ISDN and 64 kbit/s ISDN
I.610	B-ISDN Operation and Maintenance Principles and Functions

Einen besonderen Impuls erhielt die ATM-Entwicklung mit der Gründung des ATM-Forums im Oktober 1991. Dieser Zusammenschluß von Firmen und Institutionen entstand mit der Zielsetzung, die Entwicklung im Bereich ATM-basierender Produkte und Dienste zu forcieren sowie die herstellerübergreifende Interoperabilität sicherzustellen und zu demonstrieren. Bereits nach einem Jahr zählte das ATM-Forum 150 Mitglieder, darunter Hersteller, Anwender und Forschungseinrichtungen [SAM]. Das ATM-Forum versteht sich nicht als Normungsgremium, sondern arbeitet im Vorfeld der Standardisierung an Spezifikationen. Aufgrund der großen Mitgliederzahl des ATM-Forums und ihrer Gewichtigkeit sind die veröffentlichten Spezifikationen allerdings bereits De-Facto-Standards. Die "*ATM User-Network Interface Specification*", die über die bisherigen Empfehlungen der ITU-T weit hinausgeht, hat praktisch die Bedeutung eines offenen Standards. In Tabelle 5.1-2 ist eine Übersicht über die wichtigsten Spezifiaktionen des ATM-Forums zusammengestellt.

Tab. 5.1-2: *Spezifikationen des ATM-Forums*

ATM UNI v4.0 Specification, Prentice Hall
ATM Data Exchange Interface DXI v1.0 Specification
ATM Broadband Inter-Carrier Interface B-ICI v1.1 Specification

Fortsetzung auf der naechsten Seite

Fortsetzung von der vorhergehenden Seite
LAN Emulation over ATM v1.0 Specification
Interim Inter-Switch Signaling Protocol (IISP) v1.0
TM DS1 PHY v1.0 Specification
ATM 52 Mb/s Category 3 UTP
ATM 155 Mb/s Category 5 UTP v.1.0
25.6 Mb/s over Twisted Pair Cable Physical Interface
A Cell-Based Transmission Convergence Sublayer for Clear Channel Interfaces
UTOPIA Specification Level 1, Version 2.01
UTOPIA Specification Level 2, Version 1.0
PICS Proforma for the DS3 Physical Layer Interface v1.0
PICS Proforma for the 100 Mbps Multimode Fibre Physical Layer Interface v1.0
Introduction to ATM Forum Test Specifications v1.0
PICS Proforma for the SONET STS-3c Physical Layer Interface v1.0
PICS Proforma for the UNI 3.0 ATM Layer
PICS Proforma for the DS1 Physical Layer Interface
Customer Network Managment (CNM) for ATM Public Network Service
M4 Interface Requirements and Logical MIB
CMIP Specification for the M4 Interface
Circuit Emulation Service Interoperability Specification
LAN Emulation Client Management Specification v1.0
Frame Based User-to-Network Interface (FUNI) Specification
Interoperability Abstract Test Suite for the ATM Layer
Interoperability Abstract Test Suite for the Physical Layer
6,312 Kbps UNI v1.0 Specification

ATM-Verfahren

Typische Merkmale des **ATM-Verfahrens** sind die kurzen Nachrichtenpakete (Zellen genannt), die aneinander gereiht werden, die Bildung von virtuellen Verbindungen (mit einer besonders schnellen virtuellen Adressierung) und die Verlagerung aller Funktionen für den Transport der Zellen durch das Netz in die Schicht 2. Die Zellen sind 53 Byte lang und bestehen aus einem Kopf aus 5 Byte und einem Informationsteil aus 48 Byte.

Außer der Integration unterschiedlichster Dienste in einem Netz bietet die ATM-Technik zwei wesentliche Vorteile für den Anwender, die aus der variablen Bitrate herrühren:

- die Anpassung der Bitübermittlungsrate der Verbindung an die Schwankungen der Datenquelle d. h. ökonomische Nutzung der verfügbaren Bandbreite und
- die Unabhängigkeit der Endgeräteschnittstelle von der Bitrate des Dienstes, die wiederum eine von der Schnittstelle unabhängige und daher rasche Einführung neuer Dienste und spätere Anpassung an neue Komprimierungsverfahren ermöglicht.

Bereits jetzt ist ersichtlich, daß die ATM-Technik eine Brücke zwischen den öffentlichen und den Lokalen Netzen (in der zweiten Hierarchiestufe der Lokalen Netze) schlägt. Durch die Anpassung der Paketformate der HSLANs (***H**igh **S**peed **L**ocal **A**rea **N**etworks*) an ATM-Zellen, wird eine optimale Integration möglich, die es nunmehr durch gemeinsame Netzmanagementfunktionen zu unterstützen gilt. So verwenden DQDB (***D**istributed **Q**ueue **D**ual **B**us*) und ATM-Ring die gleichen Zellformate und bieten auch gemeinsame Netzmanagementfunktionen an (z. B. Abgrenzungsmöglichkeiten zwischen dem privaten und dem öffentlichen Bereich durch *edge gateways*). Das Fehlen dieser Kompatibilität ist ein Grund für das Stocken der FDDI-Ausbreitung im *backbone* Bereich von LANs.

Kurzfristig wird die ATM-Technik für die schnelle Datenübermittlung zwischen Rechnern bzw. zwischen LANs über das B-ISDN bzw. über HSLANs wie DQDB eingesetzt. Hierzu bieten sich die beiden Dienste FRS (***F**rame **R**elay **S**ervices*) und SMDS (***S**witched **M**ultmegabit **D**ata **S**ervices*) in ATM-Netzen an.

Frame Relay

Frame Relay ist eine Weiterentwicklung des X.25 Protokolls, das sich durch einen minimalen Fehlererkennungs- und Flußkontrolloverhead (von 6 Bytes pro Rahmen) auszeichnet. Charakteristisch für Frame Relay sind ferner die verbindungsorientierte Arbeitsweise mit variablen Rahmenlängen von bis zu 8192 Bytes und die Verlagerung der Vermittlungsfunktionen in die Schicht 2. Der Dienst wird mit unterschiedlichen Geschwindigkeiten von beispielsweise 56 kbit/s, 384 kbit/s und 1.544 Mbit/s angeboten. Die Standards für Frame Relay werden seit 1988 bei CCITT und ANSI erarbeitet (s. z. B. ITU-T Empfehlung I.122 *Frame Mode Bearer Service*).

SMDS

SMDS ist ein auf verbindungsloser Datenübermittlung basierender Dienst mit festen Rahmen- und Zellgrößen, der in Hochgeschwindigkeitsnetzen insbesondere für den Weitverkehr mit typischen Geschwindigkeiten von 1,17, 4, 10, 16, 25 und 34 Mbit/s zunächst in USA angeboten wurde. Um den Dienst auch zügig in Europa einzuführen wurde eine europäische Interessengruppe gebildet (ESIG). Der ETSI Standard CBDS (*Connectionless Broadband Data Service*) und der Australische FASTPAC sind äquivalent zu SMDS.

Außer für die schnelle Datenübermittlung eignet sich die ATM-Technik auch besonders für die Bildübermittlung wegen seiner hohen Bitübermittlungsgeschwindigkeit und der variablen Bitrate. Es werden inzwischen sehr hohe Erwartungen bezüglich Multimedia-Anwendungen, Videoabruf, interaktives Fernsehen und ähnliche Dienste über Datenautobahnen auf ATM-Basis gestellt. Um jedoch die vollen Vorteile der variablen Bitraten, die die ATM-Technik ermöglicht, ausnutzen zu können, müssen noch besondere Kompressions-, Codierungs- und Verschlüsselungsverfahren entwickelt werden, die die einzelnen Zellen mit Prioritäten zu versehen erlauben und gegen den Verlust von niedrig priorisierten Zellen robuste Eigenschaften aufweisen. Politisch sind allerdings sowohl in USA als auch in der Europäischen Union die Weichen bereits so gestellt, daß Breitbanddienste im großen Stil möglich werden.

Beispiel 5.1-1: Frame Relay

Frame Relay ist ein Schicht-2-Übertragungsdienst, der auf der X.25-Paketvermittlung basiert und mit einer Übertragungsgeschwindigkeit von bis zu 2,048 Mbit/s im DATEX-P angeboten wird. Der Dienst wird meist für virtuelle Festverbindungen angeboten, bei denen keine Signalisierung zum Aufbau von Verbindungen erforderlich ist.

Der Aufbau eines Frame Relay Übertragungsrahmen ist in Abb. 5.1-1 dargestellt (ITU-T Q.922):

Abb. 5.1-1: Frame Relay Übertragungsrahmen (ITU-T Q.922)

Das Beginn- und Ende-Flag (`0111 1110`) dient der Rahmenbegrenzung und darf im restlichen Frame Relay Rahmen nicht mehr vorkommen. Das wird dadurch erreicht, daß nach jeder fünften `1` eine `0` eingefügt wird. Die übrigen Felder:

- Der **Header** besteht im wesentlichen aus dem DLCI (**D**ata **L**ink **C**onnection **I**dentifier) zur Indentifizierung einer eindeutigen virtuellen Verbindung und eventuell aus einer Sendefolgennummer. Aufgrund des DLCI und den Verbindungstabellen der Vermittlungsknoten werden die Rahmen durch das Netz geleitet.

- **Benutzerdaten** haben eine variable Länge bis zu 8192 Byte.

- Die CRC-16 **Prüfsumme** ($G(x) = x^{16} + x^{12} + x^5 + 1$) dient zur Erkennung von Bitfehlern.

Der Aufbau des Header Feldes (erweiterte Adresse, keine Sendefolgenummern):

Bit	8	7	6	5	4	3	2	1
	DLCI						C/R	EA=0
	DLCI				FECN	BECN	DE	EA=0
	DLCI							EA=0
	DLCI Control						D/C	EA=1

Abb. 5.1-2: Aufbau des Header Feldes

Alternativer Aufbau des Header Feldes (kurze Adresse, Sendefolgenummern):

Bit	8	7	6	5	4	3	2	1
	DLCI						C/R	EA=0
	DLCI				FECN	BECN	DE	EA=1
	N(S)							EA=0
	N(R)							P

Abb. 5.1-3: Alternativer Aufbau des Header Feldes

wobei

- **DE (Discard Eligible)**: Kennzeichnung von Blöcken, die bei Überlast bevorzugt transportiert werden
- **FECN (Forward Error Congestion Notification)**: End-to-End Flußkontrolle
- **BECN (Backward Error Congestion Notification)**: End-to-End Flußkontrolle
- **N(S) und N(R)** dienen zur Sendefolgesteuerung und Quittierung von Paketen
- **P (Poll-Bit)**: Aufforderung zur unmittelbaren Quittierung
- **D/C**: DLCI oder DL-Core control indicator

Das Steuerfeld (control field) kennzeichnet die Art der Blöcke, die Art der Befehle und Meldungen:

- **I-Block**: Übermittlung von Daten der Schicht 3
- **S-Block**: Dient zur Quittierung und Sendefolgesteuerung der Datenübermittlung

- **U-Block**: Dient dem ungesicherten Transport (keine Flußkontrolle) von Informationen oder Steuerzeichen. Außerdem ist der U-Block für die Initialisierung, Beendigung einer Verbindung und das Zurücksetzen der Sendefolgenummernzähler in den Stationen zuständig.

Die Vorteile von Frame Relay:

- Zur Installation von Frame Relay auf bestehende Routersysteme für X.25 ist meist nur ein Software/Hardware-Upgrade nötig.
- Effizient bei der Übertragung von traditionellem Datenverkehr.
- Keine Implementation von umfangreichen Fehlerkorrekturmechanismen und Algorithmen zur wiederholten Übertragung verloren gegangener Datenpakete. Dies ist bei den niedrigen Fehlerraten heutiger Weitverkehrsnetze nicht mehr notwendig.

Die Nachteile von Frame Relay:

- Frame Relay ist nicht in der Lage stark wechselnde Verkehrsprofile zu bewältigen. Müßten mit Frame Relay gleichzeitig Daten und Sprachinformation übertragen werden, so würden Sprachsignale mit starkem Jitter das Resultat sein.
- In Frame Relay ist keine Möglichkeit vorgesehen verlorene oder fehlerhafte Datenpakete wiederholt zu übertragen. Die Korrektur von Übertragungsfehlern wird den Protokollen der höheren Ebenen überlassen.

Beispiel 5.1-2: SMDS
Der SMDS Dienst ist in den Schichten 1 bis 3 des OSI-Referenzmodells definiert:

- **Schicht 1**: Im DATEX-M definiert durch die MAN-Festlegung.
- **Schicht 2**: Abbildung der Nutzdaten auf die MAN Slots.
- **Schicht 3**: Definition der SMDS-Datagramme, in denen die Nutzdaten transportiert werden und die Quellen- und Zieladressen enthalten sind.

Das SMDS-Datagramm Nachrichtenformat der Schicht 3 ist, wie in Abb. 5.1-4 dargestellt, aufgebaut:

Abb. 5.1-4: SMDS-Datagramm des Nachrichtenformates der Schicht 3

Die Felder Zieladresse und Quelladresse enthalten die Informationen Adresstyp (4 Bit) und die Adresse (60 Bit). Empfohlen wird die Adressierung nach ITU-T-Empfehlung E.164 (ISDN-Adressierung). Im Breitband-ISDN wird dieser Dienst über eine spezielle Einrichtung angeboten, das die Datenpakete zu einem speziellen Server zur Unterstützung von verbindungsloser Kommunikation, dem Connection Less Server (CLS), weiterleitet. Der CLS wertet die verbindungslosen SMDS- Datagramme aus und leitet sie entsprechend der Zieladresse weiter.

Selbsttestaufgabe 5.1-1:

Nennen Sie typische Merkmale des ATM-Verfahrens !

Selbsttestaufgabe 5.1-2:

Welche Vorteile ergeben sich für Anwender durch die variablen Bitraten des ATM-Verfahrens?

5.2 Anforderungen von Breitbandanwendungen

In einem Breitbandnetz werden verschiedene Dienste übermittelt und Anwendungen unterstützt. Man unterscheidet grob zwischen Text- und Daten-, Sprach- bzw. Audio- und Still- bzw. Bewegtbild-Anwendungen. **Multimedia-Anwendungen** sind Anwendungen in denen die verschiedenen Elemente Text, Sprache, Ton, Graphik, Bild und Bewegtbild integriert vorkommen; in der Regel wird auch eine gewisse Interaktivität (zwischen den Teilnehmern bzw. Endgeräten) vorausgesetzt.

Multimedia-Anwendungen

mittlere Bitrate

Die erforderliche **mittlere Bitrate** ist ein wesentliches Kriterium für die verkehrstheoretische Charakterisierung einer Anwendung bzw. eines Dienstes. Ist diese mit vertretbarer Genauigkeit während der Verbindungsdauer konstant, so spricht man von **CBR-Diensten** (***Constant Bit Rate Services*** - CBR). Variiert die Bitrate während der Verbindung , so handelt es sich um **VBR-Dienste** (***Variable Bit Rate Services*** - VBR). Das Verhältnis des während der Verbindung auftretenden Spitzenbitrate zur mittleren Bitrate wird als ***Burstiness*** (B) bezeichnet.

CBR-Dienste

VBR-Dienste

Burstiness

$$\text{Burstiness B} = \frac{\text{Spitzenbitrate}}{\text{mittlere Bitrate}}$$

Für CBR-Dienste ist Burstiness $B = 1$. Häufig spielt auch die Dauer der Spitzenbitrate (*peak bit rate duration*) eine Rolle.

Bei der verkehrstheoretischen Modellierung von VBR-Diensten in zellbasierten Netzen wie ATM betrachtet man häufig drei Ebenen (Abb. 5.2-1). Auf der **Verbindungsebene** werden die Ankünfte der Verbindungsanforderungen und die Verbindungsdauer betrachtet. Häufig sind die Verbindungsauf- und abbauzeiten spezifiziert.

Verbindungsebene

In vielen VBR-Anwendungen treten die Nutzdaten sporadisch auf. Sind Daten zur Übermittlung vorhanden, werden sie mit der Spitzenbitrate gesendet. Es entstehen somit Sende- und Ruhephasen, die auf der **Burstebene** betrachtet werden. Schließlich werden die auftretenden Zellen auf der **Zellebene** betrachtet.

Burstebene

Zellebene

$t_{b,i}$ Dauer der Aktivitätsphase
$t_{s,i}$ Dauer der Ruhephase
t_i Zellenzwischenankunftszeit

Abb. 5.2-1: Hierarchische Klassifizierung von Quellverkehr

Die Güte einer Übermittlungsstrecke wird häufig durch die **Bitfehlerrate** (***Bit Error Rate*- BER**) angegeben.

Bitfehlerrate

$$\text{BER} = \frac{\text{Anzahl der fehlerhaft übertragenen Bits}}{\text{Gesamtzahl der übertragenen Bits}}$$

Typische BER-Werte sind :

- ca. 10^{-5} für analoge Fernsprechleitungen
- 10^{-6} bis 10^{-7} für digitale Leitungen
- 10^{-9} für Koax-Kabel und
- 10^{-12} für Glasfaser.

In zellbasierten Netzen führen Bitfehler auf den Strecken, aber auch Verarbeitungsfehler in den Vermittlungen zu Fehlern in den Zellen und man betrachtet die **Zellfehlerrate** (***Cell Error Rate* - CER**):

Zellfehlerrate

$$\text{CER} = \frac{\text{Anzahl der fehlerhaft übertragenen Zellen}}{\text{Gesamtzahl der übertragenen Zellen}}$$

In ATM-Netzen kann es wegen Speicherüberläufen in den Koppelfeldern, nicht korrigierbarer Bitfehler in der Adressierung oder durch Verwerfen von zu spät eintreffenden Zellen usw. zu Zellverlusten kommen, man betrachtet deshalb auch die **Zellverlustrate** (***Cell Loss Rate*-CLR**):

Zellverlustrate

$$\text{CLR} = \frac{\text{Anzahl der verlorengegangenen Zellen}}{\text{Gesamtzahl der übertragenen Zellen}}$$

Gelegentlich werden fehlgeleitete Zellen den Nutzdaten hinzugeführt, man betrachtet deshalb auch die **Einfügungsrate von fehlgeleiteten Zellen** (***Cell Insertion Rate* - CIR**):

Einfügungsrate von fehlgeleiteten Zellen

$$\text{CIR} = \frac{\text{Anzahl der hinzugefügten Zellen}}{\text{Gesamtzahl der empfangenen Zellen}}$$

Die für die Übermittlung einer Zelle durch das Netz erforderliche Zeit wird als **Zellübermittlungsverzögerung** (***Cell Transfer Delay* - CTD**) bezeichnet, deren Mittelwert wird als **MCTD** (***Mean Cell Transfer Delay***) bezeichnet. Häufig wird auch die **Varianz der Zellübermittlungsverzögerung** betrachtet (***Cell Delay Variation* - CDV**).

Zellübermittlungsverzögerung

Varianz der Zellübermittlungsverzögerung

In Abb. 5.2-2 sind einige **Anwendungen und ihre wichtigsten Anforderungen** zusammengestellt [GRE].

Anforderungen von Anwendungen

Anwendung	$\bar{\lambda}$ [Mbit/s]	CBR/ VBR	B	B_s [Byte]	L_{max}	D_{max} [s]
Text/Daten:						
Lan-Lan-Kopplung	1-100	VBR	5	100-1K	10^{-12}	0,01-0,1
Client-/Server-Anwendung	10-100	VBR	1000	1K-500K	10^{-9}	0,001-0,5
Datenbankzugriff	1-10	VBR	1000	100-100K	10^{-9}	1-10
Prozeduraufruf	6-60	VBR	15-20	60-10K	10^{-9}	0,001-0,1
E-Mail	0,0096-1,5	VBR	1	-	10^{-9}	1-10
CAD/CAM/CAE Workstation	0,064-1,5	VBR	5	40K-100K	10^{-9}	1-10
CAD/CAM/CAE Mainframe	1,5-36	VBR	10-100	100K-1M	10^{-9}	10-60
Dialoganwendung	0,064-5	VBR	40	100-300	10^{-9}	1-3
Audio:						
Telefon 3,1 kHz PCM	0,064	CBR	1	-	10^{-4}-10^{-7}	0,01-0,5
Telefon 3,1 kHz ADPCM	0,032	CBR	1	-	10^{-4}-10^{-7}	0,01-0,5
Telefon 3,1 kHz DSI	0,0048-0,0128	VBR	5-15	2K-3K	10^{-6}-10^{-8}	0,1-0,15
Telefon 7-15 kHz	0,192-0,384	CBR	1	-	10^{-5}-10^{-6}	0,1-0,15
Audiopost	0,016-0,064	CBR, VBR	1-3	-	10^{-6}	0,5-5
Audio-on-Demand (CD-Qualität)	1,536	CBR	1	-	10^{-6}	0,5-5
Videokonferenz (Sprachanteil)	0,064-0,192	CBR	1	-	10^{-7}-10^{-9}	0,01-0,05
Still- und Bewegtbilder:						
Videotelefon	0,064-2	CBR, VBR	2-5	2K-10K	10^{-9}	0,15-0,35
Videokonferenz (Videoteil)	0,128-14	CBR, VBR	2-5	1,6K-40K	10^{-9}	0,15-0,35
Videostreaming	1-4	CBR	1	-	10^{-10}	1-5
Breitband-Bildschirmtext	0,064-10	VBR	10	≥ 1M	10^{-7}-10^{-10}	0,1-2
TV (NTSC)	1,5-44	VBR	2-5	0,5M-1,3M	10^{-6}	0,004-1
TV (HDTV)	15-150	VBR	1-2	5M-8M	10^{-16}	0,004-1
Medizinische Bilder (Röntgen)	1,5-10	CBR, VBR	-	250K-3M	10^{-12}	2
Medizinische Bilder (MRI/CT)	10-200	CBR, VBR	-	1M-100M	10^{-12}	2
Hochauflösende Computergraphik	100-10000	VBR	-	-	10^{-12}	0,01-0,5

λ:	mittlere Bitrate
B:	Burstiness
B_s:	Größe des Bursts *(Burstsize)*
L_{max}:	maximal tolerierte Zellverlustrate *(Maximum Tolerable Loss Rate)*
D_{max}:	Maximal tolerierbare Ende-zu-Ende-Verzögerung *(Maximum Tolerable End-to-End Delay)*

Abb. 5.2-2: Verkehrscharakteristiken ausgewählter Telekommunikationsanwendungen

Beispiel 5.2-1: Burst-Silence Modell

Im Fall eines Datenstroms mit Burst-Silence Charakteristik wechselt die Zellrate unregelmäßig zwischen der Spitzenzellrate (Burst-Phase) und Null (Silence-Phase). Dies ist der Extremfall der Zellratenvariabilität auf der Burst-Ebene. Die Länge der Burstphase wird durch die Zufallsvariable T_{on} und die Dauer der Silence-Phase mit der Zufallsvariable T_{off} beschrieben (Abb. 5.2-3).

Abb. 5.2-3: Burst-Silence Modell

Bezeichnet man die Mittelwerte von T_{on} und T_{off} mit $\overline{T}_{on}$ und $\overline{T}_{off}$, so ergibt sich für die Burstiness B der Burst-Silence Datenquelle

$$B = 1 + \frac{\overline{T}_{off}}{\overline{T}_{on}}$$

Wir betrachten nun eine Bilddatenbank als Burst-Silence Datenquelle für eine interaktive Abruf-Anwendung mit folgenden Parametern: Bei einer Senderate der Bildübertragung von $10Mbit/s$ und einer mittleren Burst-Länge von 250 kByte (komprimierte Bilddaten) beträgt die mittlere Sendezeit $\overline{T}_{on} = 0,2s$. Der Mittelwert der Silence-Phase, die mittlere Bildbetrachtungszeit, nehmen wir mit $\overline{T}_{off} = 3,8s$ an. In der Silence-Phase seien Signallaufzeiten und Verzögerungen in den Netzknoten und beim Benutzerendgerät enthalten. Für die Burstiness B ergibt sich $B = 20$.

Betrachtet man das Blättern im World Wide Web (WWW) als Burst-Silence Modell mit den folgenden Parametern: Senderate von $64kbit/s$ und einer mittleren Burstlänge von 10000 Byte (HTML Seite ohne Grafik) beträgt die mittlere Sendezeit $\overline{T}_{on} = 0,156s$. Der Mittelwert der Silence-Phase, die mittlere Lesezeit für eine HTML Seite, nehmen wir mit $\overline{T}_{off} = 15s$ an. Für die Burstiness B ergibt sich dann $B = 97,15$.

5.3 Die Referenzmodelle

Für unsere Betrachtungen sind zwei Referenzmodelle relevant - das B-ISDN-Referenzmodell (CCITT I.321) und das ATM-Referenzmodell (CCITT I.320).

B-ISDN-Referenzmodell

Das **B-ISDN-Referenzmodell** beschreibt die Referenzpunkte und Schnittstellen im Netz und ist eng an dem ISDN-Referenzmodell (s. Kurs Kommunikationsnetze und Protokolle I Kapitel 4.1 und Abb. 5.3-1) angelehnt. So werden durch NT1 der netzseitige Abschluß (Leitungsabschluß), NT2 der endgeräteseitige Abschluß, durch TE1 das ISDN-Endgerät, durch TE2 ein nicht-ISDN-Endgerät mit einem Terminaladapter (TA) bezeichnet. Ein B vor den jeweiligen Abkürzungen kennzeichnet

das Breitbandnetz. Es ist sowohl die Möglichkeit Schnittstellen an den Referenzpunkten S_B und T_B (Abb. 5.3-2) als auch nur am Referenzpunkt S_B (Abb. 5.3-3) zu bilden vorgesehen.

Abb. 5.3-1: ISDN Referenzmodell

Abb. 5.3-2: B-ISDN Referenzmodell mit Schnittstelle an den Referenzpunkten S_B und T_B

Abb. 5.3-3: B-ISDN Referenzmodell mit Schnittstelle nur am Referenzpunkt S_B

ATM-Protokollmodell

Das **ATM-Protokollmodell** beschreibt die einzelnen Schichten (*Layers*) und Ebenen (*planes*) des Protokolls (Abb. 5.3-4). Die Vereinfachungen und Beschleunigungen der Paketvermittlung führen zu einem einfacheren Modell als das ISO-Referenzmodell. Natürlich ist das Modell auch nicht mehr zum OSI-Modell kompatibel.

Die unterste Schicht ist, wie beim OSI-Modell, die **physikalische Schicht (*physical layer*)**. Sie setzt auf dem Übertragungsmedium auf und ermöglicht die physikalische Bit- und Zellenübertragung.

physikalische Schicht

Hierauf setzt die **ATM-Schicht (*ATM-Layer*)** auf. In ihr werden die Zellübermittlungsfunktionen wahrgenommen. Hierzu gehören die Aufgaben der Muliplex - und Demultiplexbildung, Zellkopfgenerierung, Adressaus- und -umwertung und Flußkontrolle. Sie wird deshalb auch symbolisch auf den Zellkopf bezogen (Bild 5.6).

ATM-Schicht

Abb. 5.3-4: ATM-Protokollreferenzmodell

Die hierauf aufbauende **ATM-Adaptionsschicht (*ATM Adaptation Layer - AAL*)** dient der Anpassung der verschiedenen Anwendungen an die ATM-Schicht. Zu den wesentlichen Aufgaben gehören die Segmentierung der Meldungen in ATM-Zellen und die Reassemblierung, einschließlich der erforderlichen Glättung. Die ATM-Adaptionschicht wird deshalb auch symbolisch auf das Informationsfeld der ATM-Zelle bezogen.

ATM-Adaptionsschicht

Die **höheren Schichten (*higher layers*)** stehen symbolisch für die Anwendungen. In den Anwendungen und der AAL-Schicht unterschiedet man zwischen der Benutzerebene (*user plane*) für den Nachrichtentransport und der Steuerungsebene (*control plane*) für die Signalisierung. In der Management-Ebene (*management plane*) für Verwaltungsaufgaben wird zwischen Funktionen mit Schichtenbezug (*layer management*) und Funktionen ohne Schichtenbezug (*plane management*) unterschieden. Im Abb. 5.3-5 sind Schichten und Funktionen im ATM zusammengefaßt dargestellt. In den folgenden Abschnitten wollen wir die drei Schichten des ATM-Protokollmodells näher ansehen.

höhere Schichten

Abb. 5.3-5: Schichten und Funktionen im ATM

5.4 Die physikalische Schicht

In ATM-Netzen erfolgt das Absenden, Weiterleiten und Empfangen von Zellen vom Prinzip her nach keinem festen Raster sondern asynchron. Der Zellentransport ist an kein bestimmtes physikalisches Übertragungsmedium oder Multiplexsystem gebunden. Im sogenannten **reinem ATM** werden die ATM-Zellen auf einer bitsynchronen Übertragungsstrecke lückenlos aneinander gereiht (Abb. 5.4-1). Auch Zeitmultiplexe-Strecken (d. h. Strecken mit periodisch wiederkehrender Rahmenstruktur) können für die ATM-Übertragung verwendet werden (Abb. 5.4-2). Die Nutzung von Übertragungsstrecken der Synchronen Digitalen Hierarchie (***S**ynchronus **D**igital **H**ierarchie* - **SDH**, s. Abschnitt 10.3, KE 10, KT II) wird als ***ATM over SDH*** bezeichnet, dabei kann der ganze externe Rahmen (Abb. 5.4-3) oder jeweils ein Nutzlast-Modul (*Pay Load Module*) individuell für die ATM-Übertragung verwendet werden (Abb. 5.4-4 und Abb. 5.4-5).

reines ATM

SDH

ATM over SDH

Abb. 5.4-1: ATM ohne Rahmenstruktur

Abb. 5.4-2: ATM im Zeitmultiplexrahmen

Abb. 5.4-3: ATM over SDH (Nutzung des ganzen externen Rahmens)

Abb. 5.4-4: ATM over SDH (Nutzung aller Nutzlast-Module)

Abb. 5.4-5: ATM over SDH (Nutzung eines Nutzlast-Moduls)

Die Aufgaben der Bitübertragungsschicht im ATM-Protokollmodell werden unterteilt in mediumabhängige Funktionen (***P**hysical **M**edium Sublayer* - **PM**) und mediumunabhängige Funktionen (***T**ransmission **C**onvergence Sublayer* - **TC**).

PM
TC

Zu den medienabhängigen Funktionen der Teilschicht PM gehören die Bitübertragung, einschließlich der Bitsynchronisation und Leitungscodierung, und erforderliche Anpassungen an das Medium (z. B. Elektro-Optische-Umwandlung für Glasfaserübertragung). Oberhalb der Bitebene, in der Teilschicht TC, wird

- die Erzeugung und Regenerierung der Übertragungsrahmen
- die Abbildung des Zellstromes auf die Übertragungsrahmen (*cell mapping*)
- die Erzeugung oder Eliminierung von Leerzellen (*cell rate decoupling*)
- die Sicherung des Zellkopfes und
- die Zellgrenzenerkennung (*cell delineation*)

durchgeführt.

Als Bitraten für den Teilnehmeranschluß (Teilschicht PM, Referenzpunkt T_B) werden symmetrische Anschlüsse mit 155,520 Mbit/s und 622,020 Mbit/s und ein asymmetrischer Anschluß (155,520 Mbit/s vom Teilnehmer und 622,020 Mbit/s zum Teilnehmer) von ITU-T empfohlen. Ferner die Bruttobitraten der plesiochromen Systeme:

2,048 Mbit/s	(PCM 30) (E1)
34,368 Mbit/s	(PCM 480) (E3)
139,264 Mbit/s	(PCM 1920) (E4)
1,544 Mbit/s	(DS 1)
44,736 Mbit/s	(DS 3)

Beispiel 5.4-1: ATM over SDH-Netzwerke mittels STM-1 Rahmen
Die Übertragung von ATM-Zellen über SDH-Netzwerke ist in der ITU-T Empfehlung I.342 festgelegt. Der Zellenstrom wird dazu bytesynchron in VC-4 Con-

tainer verpackt. Zunächst der Aufbau eines STM-1 Rahmens mit einer Bitrate von 155,520 Mbit/s: Der STM-1 Rahmen besteht aus 9 Zeilen zu je 270 Byte (insgesamt 2430 Byte) und hat eine Übertragungsdauer von $125\,\mu s$. Die ersten 9 Bytes einer Zeile werden als SOH (***S**ection **O**ver**h**ead*) bezeichnet und werden für Synchronisation und Verwaltungsaufgaben verwendet. Hier werden auch Pointer (Zeiger) angelegt (Abb. 5.4-6).

Abb. 5.4-6: Synchrone Multiplexhierarchie

Die synchrone Multiplexhierachie unterhalb eines STM-1 Elementes besteht aus verschiedenen Multiplexelementen, die durch Buchstaben- und Zifferngruppen gekennzeichnet werden. Sie sind byteorientiert und haben eine Rahmendauer von 125 μs oder 500 μs (bei niedrigeren Bitraten). Verschiedene Elemenete können entsprechend zusammengesetzt werden, um ein STM-1 Element zu ergeben. Die unterste Stufe der synchronen Multiplexhierachie bilden Container, in denen herkömmliche plesiochrone Signale unter Verwendung von festgelegten Stopfverfahren eingepackt werden. Dem Container wird eine Spalte von Bytes, die POH (**P**ath **O**ver**h**ead) genannt werden, hinzugefügt, um einen virtuellen Container (VC) zu bilden. Im POH werden verschiedenen Steuerungs- und Verwaltungsinformationen übertragen. Insbesondere erhält das POH Synchronisations- und Stopfinformationen, Paritätsbits usw.. Virtuelle Container dienen als transparente Kanäle über Netzknoten hinweg. Hierzu wird jedem virtuellen Container ein Pointer zugeordnet. Der Pointer hat die Aufgabe, den Rahmenanfang

des betreffenden virtuellen Containers zu kennzeichnen. Im obigen Bild ist die Einbettung eines VC-4 in einem STM-1-Element dargestellt. Der Pointer AU-4 PTR im SOH zeigt die Stelle an, an der das erste Byte (J1) von VC-4 beginnt. Auf diese Weise wird es möglich, den Rahmen des VC vom Rahmen des STM-1-Elementes unabhängig zu machen.

Für die Nutzlast eines VC-4 ergeben sich 261 x 9 Bytes = 2349 Bytes und somit ist die Nutzlast eines C-4 2340 Byte. Der ATM-Zellenstrom wird nun bytesynchron, d. h. die Ausrichtung der ATM-Zellen erfolgt auf Bytegrenze, im C-4 Container verpackt. Da die C-4 Nutzlast kein ganzzahliges Vielfaches der Zellengröße von 53 Byte ist, kann sich eine Zelle über zwei C-4 Container erstrecken. Vor der Übertragung der ATM-Zellen mittels VC-4 Container muß der Nutzdatenbereich der ATM-Zelle verschlüsselt werden, um die Synchronisierung der Empfangsstation auf die einzelne Zelle sicherzustellen. Das verwendete Verschlüsselungsverfahren ist das Self-Synchronizing-Scramblers unter Verwendung des Generatorpolynoms $x^{43} + 1$. Unter Verwendung des HEC-Synchronisationsalgorithmus ist damit das Erkennen des jeweiligen Zellenanfangs möglich. Bei Abweichungen der Zellenübertragungsrate von der Nutzdatenbandbreite der VC-4 Container werden Idle-Zellen eingefügt. Bei Überschreitung der verfügbaren Bandbreite werden Zellen verworfen. Die ATM-Zellenstrom-Bitrate ist somit synchron zu der des SDH-Containers.

Beispiel 5.4-2: ATM over PDH mittels PCM 480 (E3 Übertragungsrahmen)
Ein E3 Übertragungsrahmen nach ITU-T G.751 ist 1536 Bit lang und ist folgendermaßen aufgebaut (Abb. 5.4-7):

1	1	1	1	0	1	0	0	0	0	RAI	Res	Bits 13…384
C1	C1	C1	C1	Bits 5…384								
C2	C2	C2	C2	Bits 5…384								
C3	C3	C3	C3	St	St	St	St	Bits 9…384				

1111010000	Rahmentrennungssignal
RAI	Alarm an Gegenstelle (Remote Alarm Indication)
Res	Reserviert für nationale Anwendungen
Cn	Anpassungs-Kontroll-Bits (Justificationbits)
St	Stopfbits

Abb. 5.4-7: Übertragungsrahmen nach ITU-T G.751

Die Kodierung des Bitstromes erfolgt im HDB3-Code. Die Übertragung geschieht mit 34,368 Mbit/s. Ein E3-Rahmen wird somit in $125\mu s$ übertragen.

Da sich die ATM-Zellen in das von G.751 beschriebene Rahmenformat nur schlecht einpassen lassen, wird die Anpassung der ATM-Zellen in das Rahmen-

format von E3 (nach ITU-T Draft G.7xx) gemäß ITU-T Draft G.7zz vorgenommen. Der E3-Rahmen nach der Empfehlung ITU-T G.7xx sieht folgendermaßen (Abb. 5.4-8) aus (1.Zeile ist Spaltennumerierung):

1.	2.	3.	4.	...	58.	59.	60.	
FAI	FA2			...				1.
EM				...				2.
TR				...				3.
MA				...				4.
NR				...				5.
GC				...				6.
FA2				...				7.
				...				8.
				...				9.

Abb. 5.4-8: E3-Rahmen nach der Empfehlung ITU-T

Jedes Kästchen entspricht einem Byte, wobei die belegten Kästchen für Kontroll- und Steueraufgaben reserviert sind. Der gesamte E3-Rahmen hat eine Größe von

$$(59 \cdot 9 + 6)\ \text{Byte} = 537\ \text{Byte}$$

Für den Payload des E3-Rahmens stehen demnach genau 530 Byte zur Verfügung. Diese entsprechen genau 10 ATM Zellen. Dadurch ergibt sich eine Ausnutzung des Rahmens von:

$$\frac{10 \cdot 48\ \text{Byte}}{537\ \text{Byte}} = 89,3\%$$

Dies entspricht einer Bitrate von 30,720 Mbit/s. Die Anpassung der Zellenrate auf die Nutzkanalrate des E3-Übertragungsrahmen von 33,920 Mbit/s erfolgt durch Idle-Zellen, falls keine gültigen Zellen zur Übertragung vorhanden sind. Die 48 Byte des Informationsfeldes der ATM-Zelle werden vor der Übertragung nach dem Verfahren des **S**elf-**S**ynchrionizing-**S**crambling (SSS) mittels des Generatorpolynoms $x^{43} + 1$ verschlüsselt.

PLOAM-Funktionen

Die Realisierung der Schicht-1-Aufgaben erfordern vielfältige Betriebs-, Verwaltungs- und Wartungsfunktionen. Diese werden als **PLOAM-Funktionen** (***P**hysical **L**ayer **O**perating, **A**dministration and **M**aintenance Functions*) bezeichnet. Für die Erbringung dieser Funktionen müssen verschiedene PLOAM-Meldungen über die Übertragungsstrecke (zwischen Schicht-1-Instanzen) ausgetauscht werden. Sie unterstützen beispielsweise Überwachungsfunktionen der physikalischen Schicht oder übertragen Störungsmeldungen. Je nach verwendetem Übertragungssystem können die Meldungen außerhalb der ATM-Transportstrecke (z. B. im PCM-Rahmen oder SDH Overhead) oder in besonderen Zellen (**PLOAM-Zellen**) im ATM-Zellstrom eingebettet werden.

PLOAM-Zellen

Die PLOAM-Zellen werden durch festgelegte, reservierte Bitmuster im Zellkopf identifiziert. So wird bei einer reinen ATM-Strecke jeweils nach 26 Zellen eine PLOAM-Zelle eingefügt (Abb. 5.4-9). Die Zellen werden innerhalb der Schicht 1 verwendet, werden also nicht an die übergeordnete ATM-Schicht weitergeleitet. Bei der reinen ATM-Übertragung ist eine blockweise Sicherung der Nutzdaten zwischen aufeinanderfolgenden PLOAM-Zellen vorgesehen.

Abb. 5.4-9: Einfügung und Herausnahme der PLOAM-Zellen in der physikalischen Schicht

Die Abbildung des Zellstromes auf die Übertragungsrahmen (*cell mapping*) kann es erforderlich machen, daß Idle-Zellen eingefügt oder herausgenommen werden (*cell rate decoupling*). Die Idle-Zellen werden wiederum durch ein reserviertes Muster im Zellkopf gekennzeichnet.

Wie bereits erwähnt besteht der Zellkopf aus 5 Bytes. Die ersten vier Bytes werden bei PLOAM-Meldungen für die Kennzeichnung der PLOAM-Zellen; bei Leerzellen für deren Kennzeichnung verwendet. Bei Nutzdaten enthalten sie Informationen (wie Adressierung) für die ATM-Schicht. Das fünfte Byte wird als **HEC (*Header Error Control*)** bezeichnet und dient einerseits zur Fehlerentdeckung oder korrektur, zum anderen zur Unterstützung der Zellgrenzenerkennung. Für die Fehlererkennung oder –korrektur wird ein zyklischer Code (**CRC** - ***C**yclic **R**edundancy **C**heck*) mit dem Generatorpolynom $x^8 + x^2 + x + 1$ verwendet.

HEC

CRC

In Abb. 5.4-10 ist das Zustandsdiagramm für die Fehlerüberwachung wiedergegeben. Dem errechneten CRC-Wert wird die Folge `01010101` addiert, um die Synchronisation zu unterstützen. Das Ergebnis wird als HEC-Wert übertragen.

Abb. 5.4-10: Zustandsdiagramm für die Zellkopf-Fehlerüberwachung

Die Funktion der Zellgrenzenerkennung in ATM-Netzen ist so ausgelegt, daß sie vom verwendeten Übertragungssystem unabhängig wird. Das vorgesehene Verfahren beruht auf der Korrelation zwischen dem HEC-Wert und der Zellkopfinformation und wird gewöhnlich in Hardware implementiert, um ausreichende Schnelligkeit zu gewährleisten. Die Synchronisation erfolgt in drei Schritten:

1. Zu Beginn der Synchronisation wird die Suche nach einem gültigen HEC-Oktett Bit für Bit durchgeführt (**Hunt-Zustand**). **Hunt-Zustand**
2. Die Verifikation des gefundenen Oktetts als HEC-Wert erfolgt Zelle für Zelle (**Presync-Zustand**) und wird z. B. sechs mal wiederholt. **Presync-Zustand**
3. Nach Abschluß der Verifikation wird in den **Sync-Zustand** gewechselt, die ATM-Schnittstelle ist nun synchronisiert. Erst bei mehrfachem Empfang (z. B. sieben Mal) inkorrekter HEC-Werte wird eine erneute Synchronisationsphase eingeleitet. **Sync-Zustand**

In Abb. 5.4-11 ist das Zustandsdiagramm für die Zellgrenzenerkennung wiedergegeben. Die Zellgrenzenerkennung kann vom Übertragungssystem unterstützt werden, so sind z. B. bei *ATM over SDH* die Bytegrenzen durch Informationen im *Pathoverhead* (POH) bekannt.

Abb. 5.4-11: Zustandsdiagramm für die Zellgrenzenerkennung

Beispiel 5.4-3: Leerzellen der physikalischen Schicht (= Idle-Zellen)
Für die Kennzeichnung einiger bestimmter Zelltypen sind bestimmte Werte der ersten 4 Bytes des Zellkopfes einer ATM-Zelle reserviert. Zellen, die nur in der physikalischen Schicht erzeugt werden, tragen die folgende Kennung in den ersten 4 Byte:

Byte 1	Byte 2	Byte 3	Byte 4
PPPP 0000	0000 0000	0000 0000	0000 PPP1

Dabei dienen die P-Bits zur Unterscheidung der verschiedenen Zelltypen.

So zum Beispiel Idle-Zellen, die von der physikalischen Schicht eingefügt werden, um nicht genutzte Zeitschlitze zu markieren. Sie tragen die folgende Kennung in den ersten 4 Bytes:

MSB LSB

Byte 1	Byte 2	Byte 3	Byte 4
0000 0000	0000 0000	0000 0000	0000 0001

Die Zellköpfe der ATM-Zellen (Byte 1-4) werden mittels CRC-8 Prüfsumme und anschließendem Scrambling übertragen. Die CRC-8 Prüfsumme wird mittels des Generatorpolynoms $G = x^8 + x^2 + x + 1$ gebildet. Dabei wird aus den ersten 32 Bit des Zellkopfes ein Polynom vom Grad 31 gebildet, mit x^8 multipliziert und anschließend durch das Generatorpolynom G nach modulo 2 Arithmetik dividiert. Der Rest dieser Division ist die CRC Prüfsumme. Anschließend wird auf die CRC Prüfsumme, aufgrund der besseren Synchronisierung auf den Zellanfang, die Bitfolge `0101 0101` (MSB ganz links) addiert. Dieser Wert wird dann ins HEC Feld geschrieben.

Für die Idle-Zellen wird nun die Bitfolge für das HEC-Feld berechnet: Zunächst werden die ersten 32 Bit des Zellkopfes in ein Polynom vom maximalen Grad 31 konvertiert:

$$M = 1.$$

Anschließend wird mit x^8 multipliziert: $M' = M \cdot x^8 = x^8$. M' wird nun durch das Generatorpolynom G modulo 2 Arithmetik dividiert:

$$\begin{array}{ll} x^8 & : (x^8 + x^2 + x + 1) = 1 \\ \underline{x^8 + x^2 + x + 1} & \\ \quad\quad x^2 + x + 1 & \end{array}$$

Als Rest der Polynomdivision ergibt sich $R = x^2 + x + 1$ und somit die Bitfolge der CRC-Summe zu `0000 0111`. Auf die CRC-Summe wird die Bitfolge `0101 0101` modulo 2 addiert:

```
0000 0111                                   5.4-1
0101 0101
0101 0010
```

In das HEC-Feld der ATM-Zelle wird somit der Wert `0101 0010` eingetragen.

Beispiel 5.4-4: OAM-Netzwerk-Schichten
Die Wartung und der Betrieb eines ATM-Netzwerkes ist in Hierachiestufen organisiert und die Kommunikation findet mittels OAM-Zellen statt. Dazu wird die physikalische Schicht in drei und die ATM-Schicht in zwei OAM-Hierachiestufen unterteilt (Abb. 5.4-12):

Abb. 5.4-12: Wartung und Betrieb eines ATM-Netzes in Hierarchiestufen

Die Aufgaben der OAM-Hierachiestufen im einzelnen:

F5 virtueller Kanal (virtual channel level): Mit F5-OAM-Zellen werden Management und Überwachungsfunktionen von Kanalverbindungen übertragen.

F4 virtueller Pfad (virtual path level): Mit F4-OAM-Zellen werden Management und Überwachungsfunktionen von Pfadverbindungen (VPC) übertragen.

F3 Übertragungspfad (transmission path level): F3-OAM Zellen melden den Verlust der Zellsynchronisation (Zellsynchronisationsalgorithmus befindet sich nicht mehr im SYNC-Zustand), nicht ausreichende Qualität der Übertragungsstrecke oder das Ereignis, daß ein fehlerhafter Datenkopf nicht wiederherstellbar ist.

F2 Digitalabschnitt (digital section level): F2-OAM-Zellen melden z. B. die nicht ausreichende Übertragungsqualität auf einem Digitalabschnitt, da zu viele Bitfehler auftraten.

F1 Regenerationsabschnitt (regenerator level): Wartungs- und Fehlerfunktionen zwischen Verstärkern einer Verbindungsstrecke, z. B. Meldung von Signal- oder Rahmenverlust (z. B. SDH Rahmensynchronisation wurde verloren).

Beispiel 5.4-5: Codierung der PLOAM-Zellen
Es gibt drei PLOAM-Zelltypen mit der folgenden Headercodierung (Abb. 5.4-13):

Zelltyp	Byte 1	Byte 2	Byte 3	Byte 4	HEC
F1	0000 0000	0000 0000	0000 0000	0000 0011	0101 1100
F2	nicht spezifiziert				
F3	0000 0000	0000 0000	0000 0000	0000 1001	0110 1010

Abb. 5.4-13: PLOAM-Zelltypen

Selbsttestaufgabe 5.4-1:

Bestimmen Sie die Bandbreiteneffizienz und die Nettobitrate bei der Übertragung von ATM-Zellen über einen STM-1 Rahmen mittels eines VC-4.

Selbsttestaufgabe 5.4-2:

a. *Beschreiben Sie die Übertragung von ATM-Zellen über einen STM-4 Rahmen mittels des virtuellen Containers VC-4-4C.*

b. *Bestimmen Sie die Bandbreiteneffizienz und die Nettobitrate bei dieser Übertragungsweise.*

Selbsttestaufgabe 5.4-3:

Berechnen Sie das HEC-Feld für eine PLOAM Zelle vom Typ F3 !

5.5 Die ATM-Schicht

Bevor wir die einzelnen Aufgaben der ATM-Schicht (Abb. 5.3-5) behandeln, betrachten wir die Struktur der ATM-Zellen näher, denn die Informationen im Zellkopf sind für die Durchführung der Aufgaben der ATM-Schicht essentiell. In Abb. 5.5-1 ist die ATM-Zelle bestehend aus 5 Byte langem Kopf und 48 Byte langem Informationsfeld dargelegt. In dem Bild ist auch die Reihenfolge der Übertragung der einzelnen Bits und Bytes gekennzeichnet. Die Struktur des Zellkopfes an der **Benutzer-Netz-Schnittstelle** (**UNI** - ***U**ser-**N**etwork-**I**nterface*, Abb. 5.5-2) und **Netz-Netz-Schnittstelle** (**NNI**- **N**etwork-***N**etwork-**I**nterface*, Abb. 5.5-3) unterscheiden sich geringfügig.

Benutzer-Netz-Schnittstelle
Netz-Netz-Schnittstelle

Generic Field Access

Die ersten Bits der UNI werden als **GFC** (***Generic Field Control***) bezeichnet und dienen der Flußkontrolle zwischen den Teilnehmern und dem Netz. Ein Flußkontrollmechanismus ist bisher nicht standardisiert. Das GFC-Feld kann für die Überlastabwehr (*Traffic Control*) oder für die Überwachung (*Policing*, ***U**sage **P**arameter **C**ontrol* - UPC) verwendet werden. Auch die Unterstützung eines dezentralen Medienzugriffs (wie bei LANs) durch GFC ist möglich. Innerhalb des Netzes (an der NNI) sind diese Funktionen nicht erforderlich, so daß die 4 Bits zusätzlich zur Adressierung verwendet werden.

Usage Paramter Control

Adressfeld
VPI
VCI
Payload Type

Das **Adressfeld** besteht aus 24 Bits an der UNI und aus 28 Bits an der NNI. Es ist unterteilt in **VPI** (***V**irtual **P**ath **I**dentifier*) und **VCI** (***V**irtual **C**hannel **I**dentifier*). Wie wir sehen werden, ermöglicht diese Unterteilung die Bündelung (d. h. gemeinsame Vermittlung) aller Verbindungen mit dem gleichen VPI. Die drei als **PT-Feld** (***P**ay Load **T**ype*) bezeichneten Bits werden verwendet um den Nutzdatentyp und die Lastsituation zu kennzeichnen (so z. B. Nutzzellen, Steuerzellen, Überlast oder keine Überlast).

Cell Loss Priority

Das **CLP-Bit** (***C**ell **L**oss **P**riority*) kann verwendet werden, um priorisierte Zellen zu kennzeichnen. Bei Speicherüberlauf am Netzknoten können dann z. B. die nicht priorisierten Zellen verworfen werden. Wie wir bereits gesehen haben, wird das HEC-Feld für die Sicherung des Zellkopfes verwendet. Die Überprüfung der in der ATM-Schicht behandelten Zellen wird in dieser Schicht vorgenommen.

Abb. 5.5-1: ATM-Zellen: Inhalt und Übertragungsreihenfolge

Abb. 5.5-2: ATM-Zellkopf an der Benutzer-Netz-Schnittstelle UNI

Abb. 5.5-3: ATM-Zellkopf an der Netz-Netz-Schnittstelle NNI

Beispiel 5.5-1: Bedeutung des PT-Feldes (Payload Type)
Im Informationsfeld einer ATM-Zelle einer virtuellen Verbindung ist entweder netzinterne Information oder Benutzerinformation enthalten. Innerhalb einer virtuellen Verbindung werden verschiedene Zelltypen unterschieden, gekennzeichnet durch vordefinierte Header-Werte (Byte 1-4) und bestimmte Belegungen des 3 Bit PT-Feldes (4. Byte im ATM-Zell Header). Dabei haben die 3 Bits folgende Bedeutung:

Bit 3	Unterscheidung von Daten- und Signalisierungszellen
Bit 2	Anzeige von Überlast
Bit 1	ATM-layer-user to ATM-layer-user Indication

Bit 1 kann z. B. von der ATM-Anpassungsschicht genutzt werden, um die letzte Zelle eines AAL 5 Dienstes zu kennzeichnen.

Beispiel 5.5-2: Reservierte Headerbelegungen
An der UNI-Schnittstelle sind einige Belegungen des ATM-Zellkopfes (Bytes 1-4) für bestimmte Zellen der physikalischen und der ATM-Schicht reserviert (nach ITU-T, I.361 und I.432) (P: Bits stehen der physikalischen Schicht zur Verfügung, A: Bits stehen der ATM-Schicht zur Verfügung):

	VPI/VCI	Byte 1	Byte 2	Byte 3	Byte 4
reserviert für Pysikalische Schicht	0/0	PPPP 0000	0000 0000	0000 0000	0000 PPP1
PLOAM-Zelle	0/0	0000 0000	0000 0000	0000 0000	0000 1001
Idle-Zellen	0/0	0000 0000	0000 0000	0000 0000	0000 0001
Leerzellen	0/0	AAAA 0000	0000 0000	0000 0000	0000 AAA0

Für die Metasignalisierung und Broadcast-Signalisierung sind ebenso bestimmte VPI/VCI-Werte bereits fest vergeben (nach ITU-T, I.361):

	VPI / VCI
Identifikation des Meta-Signalisierungskanals	0/1
Identifikation des Broadcast-Signalisierungskanals	0/2

Mit Ausnahme dieser fest definierten Belegungen stehen alle anderen Werte für die ATM-Schicht zur Verfügung (Spezifikation des ATM Forums):

Zelltyp	VPI/VCI	Byte 1	Byte 2	Byte 3	Byte 4
Leerzellen	0/0	0000 0000	0000 0000	0000 0000	0000 xxx0
Meta-Signalisierung (default)	0/1	0000 0000	0000 0000	0000 0000	0001 0a0c
Meta-Signalisierung	Y ≠ 0/1	0000 yyyy	yyyy 0000	0000 0000	0001 0a0c
Broadcast Signalisierung (default)	0/2	0000 0000	0000 0000	0000 0000	0010 0aac
Broadcast Signalisierung	Y ≠ 0/0	0000 yyyy	yyyy 0000	0000 0000	0010 0aac
Punkt zu Punkt Signalisierung (default)	0/5	0000 0000	0000 0000	0000 0000	0101 0aac
Punkt zu Punkt Signalisierung	Y ≠ 0/5	0000 yyyy	yyyy 0000	0000 0000	0101 0aac
ungültige Sequenz	0/0	xxxx 0000	0000 0000	0000 0000	0000 xxx1
ILMI-Message	0/16	0000 0000	0000 0000	0000 0001	0000 aaa0
OAM-F4 Segment-Zelle	A/3	0000 aaaa	aaaa 0000	0000 0000	0011 0a0a
OAM-F4 End-to-End Zelle	A/4	0000 aaaa	aaaa 0000	0000 0000	0100 0a0a
OAM-F5 Segment-Zelle	A/B ≠ 0	0000 aaaa	aaaa bbbb	bbbb bbbb	bbbb 100a
OAM-F5 End-to-End Zelle	A/B ≠ 0	0000 aaaa	aaaa bbbb	bbbb bbbb	bbbb 101a

x: beliebig
y: jeder VPI außer 0000 0000
a: Bit für die Benutzung der ATM-Schicht
b: jeder VCI außer 0000 0000 0000 0000
c: die Sendestation soll CLP Bit auf 0 setzen, es kann vom Netz geändert werden

Die Bedeutung der wichtigsten Zellarten:

- Idle-Zellen dienen zur Entkopplung der Zellrate von der Bandbreite des Übertragungsmediums. Sind zum Ausfüllen der Bandbreite des Übertragungsmediums nicht genügend Zellen vorhanden, so werden Idle- Zellen übertragen. Im Unterschied zu Leerzellen werden Idle-Zellen nicht an die ATM- Schicht weitergegeben.

- Leerzellen (unassigned cells) sind Zellen der ATM Schicht, die einen gültigen VPI/VCI Wert enthalten, aber keine Nutzinformation beinhalten.
- Meta-Signalisierungszellen (Meta-Signalling Cells) werden bei der Kommunikation zwischen Teilnehmer und Netzbetreiber verwendet, wenn noch keine verbindungsbezogene Signalisierung aufgebaut ist. Ähnlich wie beim Schmalband-ISDN (D-Kanal) erfolgt auch beim B-ISDN die Signalisierung über einen gesonderten Kanal, allerdings wird hier im Gegensatz zum Schmalband-ISDN ein flexibleres Signalisierungskonzept eingesetzt. Hierzu wird kein permanenter Kanal eingerichtet, sondern bei Bedarf ein Signalisierungskanal eingerichtet. Alle Befehle der Meta-Signalisierung haben im Informationsfeld der ATM-Zelle Platz, so daß keine eigene Anpassungsschicht für die Metasignalisierung notwendig ist. Die drei Aufgaben der Metasignalisierung sind der Aufbau, der Abbau und die Überprüfung der Signalisierungskanäle.
- Broadcast Signalisierungszellen (general broadcast signalling cells) sind Zellen mit standardisierter, fester Adresse (VPI/VCI) für Verteilkommunikation mit der alle Anschlüße an der UNI erreicht werden können.
- Punkt-zu-Punkt Signalisierungsverbindungen können in einem Netz für den Austausch von Signalisierungsinformation festgelegt werden. In diesem Fall ist eine Meta-Signalisierung nicht notwendig.
- OAM Zellen der pysikalischen Schicht (PLOAM-Zellen: F1-, F2-, F3-OAM-Zellen) werden bei der direkten Zellübertragung über das physikalische Medium nach jeder 26. ATM-Zelle eingefügt. Diese Zellen werden beim Empfang nicht an die ATM-Schicht weitergegeben. End-to-End OAM Zellen müssen zwischen den Endstellen liegende Netzknoten unverändert passieren.
- OAM-F4 und OAM-F5 Zellen (Zellen der ATM-Schicht) dienen zum Testen und Überwachen von ATM-Pfaden bzw. Kanälen. Diese Zellen sind zusätzlich im PT-Feld gekennzeichnet.

Multiplexbildung

Die **Multiplexbildung** im ATM-Netz bedeutet die Zusammenführung verschiedener Zellströme zu einem Zellstrom (Abb. 5.5-4). In der Regel sind ATM-Ströme auf den Eingangsleitungen zueinander nicht synchron. Die Multiplexbildung wird durch die bitweise Übertragung der belegten Zellen der Eingangsleitung auf die Ausgangsleitung vorgenommen. Leerzellen können dabei herausgenommen oder hinzugefügt werden. Um Kollisionen zu vermeiden, müssen bei Bedarf belegte Zellen gespeichert werden. Die Bitraten auf den einzelnen Leitungen können durchaus unterschiedlich sein; die Ausgangsleitung muß jedoch so dimensioniert werden, daß sowohl die Verweildauer der Zellen im Speicher als auch ein eventueller Zellverlust durch Speicherüberlauf vertretbar bleiben.

Demultiplexbildung

Bei der **Demultiplexbildung** wird die Verbindungs-Kennzeichnung (VPI/VCI) insgesamt oder in Teilen ausgewertet, um Zellen auf bestimmte Ausgangsleitungen zu bringen (Abb. 5.5-5). Leerzellen der Eingangsleitung werden verworfen. An den Ausgangsleitungen werden Leerzellen nach Bedarf hinzugefügt.

Die Hintereinanderschaltung der Multiplexbildung und Demultiplexbildung ergibt im Prinzip eine ATM-Vermittlung. Zellen einer Verbindung können so von einer Leitung zu einer anderen, beliebigen Leitung entsprechend der Adressierung geleitet werden. Im Abb. 5.5-6 ist eine einfache, intern-geschwindigkeitsmäßig günstige Struktur zur Darstellung eines ATM-Koppelelementes aus zwei Eingängen und zwei Ausgängen bestehend aus Multiplexern und Demultiplexern aufgezeigt.

Abb. 5.5-4: ATM-Multiplexen

Abb. 5.5-5: ATM-Demultiplexen

Abb. 5.5-6: ATM-Vermittlung

Der Kommunikationsablauf in ATM-Netzen ist verbindungsorientiert. Im ersten Schritt wird eine virtuelle Verbindung (Route) durch das Netz aufgebaut; an den betroffenen Netzknoten werden die für die Vermittlung der Zellen der Verbindung erforderlichen Daten (Tabellen) abgelegt. In der Datenübermittlungsphase werden die Zellen mit den Netzdaten entsprechend den Routingtabellen vermittelt. Alle

Zellen werden auf der gleichen Route übermittelt, sie können zwar an den einzelnen Knoten verschiedene Verzögerungen erfahren, eine gegenseitige Überholung ist jedoch nicht möglich. Deshalb muss man auch keinen Sequenzierungs- und Resequenzierungsaufwand betreiben. Nach Beendigung der Datenübermittlung wird die virtuelle Verbindung in der Regel abgebaut.

Um die Verarbeitung der Zellen zu beschleunigen, tragen die Zellen mit den Nutzdaten nicht die komplette Adresse der Verbindung. Am Eingang eines Knotes enthalten die Zellen die Eingangsadresse, die zur Steuerung der Zelle durch das Koppelfeld dient. Als Ausgangsadresse wird die für den nächsten Knoten relevante Adresse eingetragen. Dies ist symbolisch in Abb. 5.5-7 dargestellt. Somit haben die Zelleadressen lediglich einen lokalen Bezug. Im Abb. 5.5-8 ist der komplette Vermittlungsvorgang mit einer entsprechenden Umwerttabelle dargestellt.

Abb. 5.5-7: Die virtuelle Adressierung im ATM

Umwertetabelle			
kommende Verbindung	Zellkopf	gehende Verbindung	Zellkopf
E_1	x y z	A_1 A_2 A_w	k l m
⋮	⋮	⋮	⋮
E_n	x y s	A_1 A_2 A_w	n j g

Abb. 5.5-8: Das ATM-Vermittlungsprinzip

Wie bereits dargestellt besteht die Adresse im Zellkopf aus zwei Komponenten, dem *Virtual Path Identifier* (VPI) und dem *Virtual Channel Identifier* (VCI). Die Adreßumwertung in den Netzknoten kann sich auf beide Komponenten erstrecken oder sich allein auf den VPI beschränken, der VCI bleibt in diesem Fall bei der Adressumwertung unverändert. Diese zunächst nur formale Unterscheidung hat große praktische Bedeutung für den Netzbetrieb.

Abb. 5.5-9 zeigt einen Abschnitt eines physikalischen Übertragungsweges. Jeweils mehrere virtuelle Verbindungen (VC) gehören zu einem virtuellen Pfad (VP), so daß dieser einem Leitungsbündel ähnelt. Zwischen den Endpunkten eines virtuellen Pfades werden alle zugehörigen virtuellen Verbindungen über den gleichen physikalischen Weg geführt. In den dazwischen liegenden Netzknoten erfolgt eine Umwertung des VPI, die VCI-Werte bleiben unverändert (Abb. 5.5-10 und Abb. 5.5-11).

Abb. 5.5-9: Übertragungsweg mit virtuellen Pfaden (VP) und virtuellen Verbindungen (VC)

Abb. 5.5-10: VP-Switching

Abb. 5.5-11: Adressumwertung beim VP-Switching

In Abb. 5.5-12 ist ein anderes Beispiel vorgestellt, in dem sowohl VP- als auch VC-Vermittlung vorgenommen wird; diese werden aber nach VP- und VC-Vermittlung zusammengefaßt dargestellt, um ihre Bedeutung zu verdeutlichen.

Abb. 5.5-12: VPI/VCI-Switching

Der Vorteil einer VPI-bezogenen Vermittlung liegt zunächst darin, daß nur ein Teil der Adresse des Zellkopfs betrachtet wird und der Aufwand in der Verwaltung der Verbindungen und bei der Umwertung gegenüber der VPI/VCI-bezogenen Vermittlung geringer ist. Ein entscheidender Vorteil der virtuellen Pfade besteht in ihrer Bündelungseigenschaft. Auf häufig frequentierten Streckenabschnitten bietet sich die Einrichtung von virtuellen Pfaden an. Dies hat den Vorteil, daß beim Auf- und Abbau von virtuellen Verbindungen (VCC), die über diese Pfade gehen, die zugehörigen Netzknoten nicht involviert werden und sich somit der Aufwand beim Verbindungsaufbau reduziert. Dies wird besonders im Fall eines virtuellen Pfades zwischen Quelle und Ziel deutlich: Der Netzbetreiber richtet zwischen Quelle und Ziel einen (semipermanenten) virtuellen Pfad ein. Der anschließende Aufbau von virtuellen Verbindungen und die Verwaltung der VCI-Werte bleiben dem Benutzer überlassen.

Wir wollen die für den Verbindungsauf- und -abbau und den Betrieb erforderliche Signalisierung nicht mehr näher ansehen, denn sie ähnelt der für X.25-Netzen bzw. für ISDN. Es sei auf die ITU-T Q.931 Empfehlung verwiesen.

5.6 Die ATM-Adaptionsschicht

ATM-Adaptionsschicht

Die **ATM-Adaptionsschicht** (**AAL** - ***A**TM-**A**daption **L**ayer*, Abb. 5.3-5) bildet die funktionale Schnittstelle zwischen der ATM-Schicht und den Diensten der höheren Schichten. ATM-Zellen werden ausschließlich in der physikalichen und der ATM-Schicht behandelt. Der AAL obliegt lediglich die Behandlung des Informationsfeldes der relevanten ATM-Zellen. Die Hauptaufgaben der AAL sind:

Protocol Data Units

- Segmentierung der Nutzdaten (***P**rotocol **D**ata **U**nits* - **PDU**s) der höheren Schichten und deren Einbettung in das Informationsfeld der ATM-Zellen.
- Reassemblierung der PDUs aus den Informationsfeldern (d. h. der umgekehrte Vorgang zur Segmentierung).
- Durchführung der erforderlichen Konvergenzfunktionen für die höheren Schichten so z. B. Anpassung der Bitfehleranforderungen, Glättung bzw. zeitliche Anpassung usw..
- Kommunikation mit den zugeordneten AAL-Instanzen über Steuerdaten im Informationsfeld.

An Hand der Anforderungen von Breitbandanwendungen (s. Tabelle 5.1-1) werden bei ITU und im ATM-Forum ATM-Dienste detalliert spezifiziert. Wir betrachten die ITU-T Diensteklassifikation nach der Empfehlung I.362 (Abb. 5.6-1).

Merkmal	Klasse A	Klasse B	Klasse C	Klasse D
Zeitbezug zwischen Sender/ Empfänger	erforderlich		nicht erforderlich	
Bitrate	konstant (CBR)	variabel (VBR)		
Modus	verbindungsorientiert			verbindungs - los
entsprechende AAL Typen	AAL 1	AAL 2	AAL 3/4	AAL 3/4, AAL 5

Abb. 5.6-1: Diensteklassifikation nach ITU-T I.362 und entsprechende Protokolle der AAL-Schicht

Darin werden vier Klassen unterschieden:

- Klasse A: für Echtzeit-Anwendungen mit konstanter Bitrate (z. B. Sprache, Audio, Video).
- Klasse B: für Echtzeit-Anwendungen mit variabler Bitrate (z. B. komprimierte Sprache, Audio oder Video).

- Klasse C: für verzögerungsunempfindliche, verbindungsorientiete Datenübermittlung.
- Klasse D: für verzögerungsunempfindliche, verbindungslose Datenübermittlung.

Ursprünglich wurden entsprechend vier AAL-Protokolle entworfen (AAL 1 bis 4). Später wurden AAL 3 und 4 zusammengefaßt und, da diese einen zu hohen Aufwand (Overhead) erforderten, wurde ein einfacheres Protokoll AAL 5 entworfen. Im folgenden wollen wir diese vier AAL-Protokolle kurz betrachten.

AAL 1

AAL 1 unterstützt die Diensteklasse A, d. h. Dienste mit einer nominell konstanten Bitrate (CBR - Constant Bitrate Services). Taktinformationen werden zwischen dem Sender und dem Empfänger übertragen und Fehlermeldungen an höhere Schichten werden erstattet.

Es werden folgende Funktionen durchgeführt:

- Segmentierung bzw. Reassemblierung der Nutzinformation
- Maßnahmen zur Begrenzung von Zell-Laufzeitschwankungen (*Cell Delay Variation*)
- Taktrückgewinnung auf der Empfangsseite
- Behandlung fehlender oder fehlgeleiteter Zellen
- Bitfehlerüberwachung der AAL-Steuerdaten einer PDU
- Bitfehlerüberwachung des Informationsfeldes

Protocol Data Units

Segmentation and Reassembly

Für AAL 1, AAL 3/4 und AAL 5 wurden Protokol-Daten-Einheiten (**PDU**s - ***P**rotocol **D**ata **U**nits*) für die Teilschicht **SAR** (***S**egmentation **A**nd **R**eassembly Layer*) festgelegt (Abb. 5.6-2). Die Sequenznummer SN wird zur Überwachung fehlender oder fehlgeleiteter Zellen verwendet, während mit den vier SNP-Bits (***S**equenz **N**umber **P**rotection*) die Sicherung der Sequenznummer SN vorgenommen wird.

SN Sequence Number (Zellfolge-Nummer)
SNP Sequence Number Protection

Abb. 5.6-2: Struktur des SAR-PDU des AAL 1

AAL 2
Service Data Units

AAL 2 unterstützt die Diensteklasse B, d. h. Echtzeitdienste mit variabler Bitrate. Dies bedeutet, daß variable Dateneinheiten der Dienste der höheren Schichten (**SDU** - ***S**ervice **D**ata **U**nits*) innerhalb der Anpassungsschicht bearbeitet werden. Zwischen dem Sender und Empfänger werden Taktinformationen übertragen und Fehlermeldungen an höhere Schichten erstattet. Die einzelnen Funktionen, die durchgeführt werden sind noch nicht festgelegt, erforderlich sind zumindest:

- Taktrückgewinnung für Audio- und Videosignale mit variabler Bitrate
- Zellfolgeüberwachung und
- Forward Error Correction.

AAL 3/4

AAL 3/4unterstützt die Diensteklasse C und D, d. h. sowohl verbindungsorientierte als auch verbindungslose Datendienste mit variabler Bitrate und ohne Zeitbezug zwischen Sender und Empfänger. Es werden Nutzdateneinheiten (SDUs) variabler Bitrate unterstützt und bei Dienst der Klasse D auch Multiplex- und Demultiplexfunktionen durchgeführt um die ATM-Zellen optimal zu verwenden. Ferner wird eine Bitfehlererkennung der SAR-PDUs vorgenommen und Fehlermeldung erstattet. Um beide Dienstklassen zu unterstützen, ist die Funktionalität modular angelegt (Abb. 5.6-3).

AAL	ATM-Adaption Layer	SAP	Service Access Point
CPCS	Common Part Convergence Layer	SAR	Segmention and Reassembly Sublayer
CS	Convergence Sublayer	SSCS	Service Specific Convergence Sublayer

Abb. 5.6-3: Modulare AAL 3/4 Architektur

Die CPCS-Funktionalität genügt, um die Anforderungen der Diensteklasse C zu erfüllen. Zur Unterstützung der Diensteklasse D werden SSCS-Funktionen erforderlich. In Abb. 5.6-4 ist das Rahmenformat der CPCS-PDU dargestellt. Diese wird in der SAR-Schicht segmentiert bzw. reassembliert. Die Struktur der SAR-PDU ist in Abb. 5.6-5 dargestellt.

Abb. 5.6-4: Rahmenformat der CPCS-PDU des AAL3/4

AAL 5 **AAL 5** unterstützt die verbindungslose Datenübertragung mit variabler Bitrate und ohne Zeitbezug. Sie weist einen wesentlich geringeren *Overhead* als AAL 3/4 auf. Dies wird durch den Verzicht auf Fehlererkennung, Sequenzierung, Längenangaben und Multiplexfunktionen erreicht. In Abb. 5.6-6 ist die Struktur der SAR-PDU des AAL 5 dargestellt. Die SAR-Teilschicht benutzt ein Bit (*das AUU-Bit-ATM-Layer-User to ATM-Layer-User Bit*) des *Payload Type PT* im Zellkopf zur Anzeige des Anfanges oder der Fortsetzung (AUU = 0) und des Endes (AUU = 1) einer SAR-PDU an. Die weiteren Funktionen der CS-Teilschicht entsprechen denen des AAL 3/4.

Codierung des Segment Type Feldes (ST)		
BOM	10	Beginning of Message
COM	00	Continuation of Message
EOM	01	End of Message
SSM	11	Single Segment of Message

ST	Segment Type	LI	Length Indicator
SN	Sequence Number	CRC	Cyclic Redundancy Check
MID	Multiplexing Identification		

Abb. 5.6-5: Rahmenformat der SAR-PDU des AAL 3/4

Abb. 5.6-6: Die Struktur der SAR-PDU des AAL 5

Beispiel 5.6-1: Einbettung von Nachrichten in AAL 3/4

Die Datenpakete variabler Länge, die der AAL 3/4 von der darüberliegenden Anwendung übergeben werden, werden zunächst auf ganzzahlige Vielfache von 32 Bits aufgefüllt, wodurch eine effektive, hardwareorientierte Implementierung der Abläufe ermöglicht wird, und in das Payload Feld des CPCS-Rahmen eingetragen. Das CPCS-Datenformat kann länger sein als das Payload-Feld (44 Byte) der SAR-Dateneinheit. In diesem Fall wird der CPCS-Rahmen (Abb. 5.6-7) auf mehrere SAR-Dateneinheiten aufgeteilt. Im SAR-Header wird die erste Segmentierungseinheit eines CPCS-Rahmen als BOM (Begin of Message) gekennzeichnet, jede weitere Segmentierungseinheit wird als COM (Continuation of Message) und die letzte als EOM (End of Message) gekennzeichnet.

Abb. 5.6-7: CPCS-Rahmen

Wird ein CPCS-Rahmen an die SAR-Teilschicht übergeben, der kürzer ist als das Payload-Feld der SAR-Dateneinheit, so wird das Payload-Feld von links nach rechts mit dem CPCS-Rahmen gefüllt und die restlichen Bits mit dem Wert 0 belegt

Beispiel 5.6-2: Übertragung von Ethernet-Rahmen mit AAL 3/4
In Abb. 5.6-8 wird ein Ethernet-Rahmen aus der Anwendungsschicht der AAL Schicht übergeben. Unter Verwendung des Dienstes AAL 3/4 wird der Rahmen segmentiert:

Abb. 5.6-8: Segmentierung eines Ethernet-Rahmens

Bei der Nutzdatenübertragung zwischen verschieden LAN-Netzwerken (hier z. B. Ethernet) können die Zielinformationen nicht vom ATM-Netzwerk ausgewertet werden. Deshalb werden diese Datenpakete an einen bestimmten Server im ATM-Netzwerk (Connectionless Server - CLS) weitergeleitet, der die verbindungslose Kommunikation zwischen verschiedenen LAN-Netzwerken unterstützt. Schickt eine Endeinrichtung A aus einem LAN eine verbindungslose

Nachricht an eine Endeinrichtung B, das an einem anderen LAN angeschlossen ist, über ein ATM-Netzwerk, so wird diese Nachricht zunächst an den CLS gesendet. Der CLS wertet die Zielinformation aus, baut eine Verbindung mit der Endeinrichtung B auf und verschickt die Nachricht an B über diese Verbindung.

Beispiel 5.6-3: Einbettung von Nachrichten in AAL 5

Die Daten aus der Anwendungsschicht werden in das Payload-Feld des CPCS-Rahmens des AAL 5 (1-65535 Byte) übernommen und das Padding-Feld (PAD) wird mit 0 bis maximal 47 Füllbytes ergänzt, so daß der CPCS-Rahmen eine Gesamtlänge hat, die ein ganzzahliges Vielfaches von 48 ist.

Der CPCS-Rahmen wird nun durch die SAR-Schicht in eventuell mehrere Payload-Felder der SAR-Dateneinheiten (48 Byte) abgelegt (Abb. 5.6-9). Aus dem CPCS-Rahmen entsteht eine Sequenz von SAR-Dateneinheiten. Da eine SAR-Dateneinheit keinen Header besitzt, wird die Markierung des letzten Datenpaketes einer Sequenz von SAR-Dateneinheiten, die zu einem CBCS-Rahmen gehören, mittels des PT-Feldes (3 Bits des 5. Byte im Zellkopf) der ATM-Zelle vorgenommen. Neben anderen Anwendungen können in diesem Feld ATM-Layer-User to ATM-Layer-User (AUU-Parameter zwischen ATM Schichten) ausgetauscht werden. Der AUU-Parameter wird normalerweise auf 0 gesetzt, in der letzten ATM-Zelle, die zu einer Sequenz von SAR-Dateneinheiten gehört, wird dieses Feld mit 1 gekennzeichnet.

Abb. 5.6-9: Zerlegung des CPCS-Rahmens

Beispiel 5.6-4: IP Pakete verpackt in LLC/SNAP über AAL 5

Ein IP Paket (Abb. 5.6-10) besteht aus einem Header (20 Byte) und einem Datenteil (0-9160 Byte). Das IP Paket wird nun nicht direkt der AAL 5 Schicht übergeben, sondern zunächst wird eine LLC/SNAP-Kapselung (Abb. 5.6-11) vorgenommen:

Abb. 5.6-10: IP-Paket

Das Paket, das durch die LLC/SNAP-Kapselung entstanden ist, wird nun der AAL 5 Schicht übergeben, in das Payload Feld des CPCS-Rahmen eingebettet und anschließend von der SAR Schicht des AAL 5 segmentiert.

Abb. 5.6-11: LLC/SNAP-Kapselung eines IP-Paketes

Je nach Länge des Payloadfeldes des IP Paketes, werden somit 1-192 ATM-Zellen erzeugt.

5.7 ATM-Verkehrskontrolle

Überlast

Überlast (*congestion*) ist im B-ISDN definiert als ein Zustand von Netzkomponenten (wie Vermittlungsstellen, Konzentratoren, Leitungen oder auch Steuerung), in dem das Netz die ausgehandelten Leistungsparameter für bereits bestehende oder neue Verbindungen nicht einhalten kann.

Verkehrskontrolle
Überlastkontrolle

Unter **Verkehrskontrolle** (*traffic control*) im strengeren Sinne versteht man die Maßnahmen zur Vermeidung von Überlastsituationen. Unter **Überlastkontrolle** (*congestion control*) versteht man die Maßnahmen zur Minderung der Intensität, Reichweite oder Dauer von Überlastsituationen.

Im B-ISDN (ITU-T, I.371) sind folgende Funktionen für Verkehrs- und Überlastkontrolle vorgesehen (s. Abb. 5.7-1):

CAC
- Zugangskontrolle (**CAC** - ***C****onnection* ***A****dmission* ***C****ontrol*)

UPC
- Parameterüberwachung an der UNI (**UPC** - ***U****sage* ***P****arameter* ***C****ontrol*)

NPC
- Parameterüberwachung an der NNI (**NPC** -***N****etwork* ***P****arameter* ***C****ontrol*)
- Prioritätenkontrolle (Priority Control, CLP - *Cell Loss Priority*)
- Reaktive Flußkontrolle (*Feedback Control*)
- Management von Betriebsmitteln (*Network Resource Management*)

In ATM-Netzen wird in der Verbindungsphase zwischen dem Teilnehmer und Netz ein sogenannter Verkehrsvertrag ausgehandelt, der die während der Verbindung einzuhaltenden Verkehrsparameter und Dienstgüteanforderungen (QoS - Quality of Service) wie z. B. die Varianz der Zellübermittlungsverzögerung (**CDVT** - ***C****ell* ***D****elay* ***V****ariation* ***T****olerance*) festlegt. Mögliche Verkehrsparameter sind die Spitzenzellrate, die mittlere Zellrate und die Burstdauer. Bei ITU-T ist bisher nur die

CDVT

Spitzenzellrate als Verkehrsparameter spezifiziert. Das ATM-Forum hat zusätzlich die aufrechthaltbare Zellrate (**SCR** - ***S**ustainable **C**ell **R**ate*) als die obere Grenze der mittleren Zellrate und die Bursttoleranz spezifiziert.

SCR

UNI	User Network Interface (Benutzer-Netz-Schnittstelle)
NNI	Network Network Interface (Netz-Netz-Schnittstelle)
LAN	Local Area Network (Lokales Netz)
UPC	Usage Parameter Control (Parameterüberwachung an der UN)
NPC	Network Parameter Control (Parameterüberwachung an der NN)
CLP	Cell Loss Priority (Zellverlust-Priorität)
CAC	Connection Admission Control (Zugangskontrolle)

Abb. 5.7-1: Funktionen der Verkehrskontrolle im B-ISDN

Wir betrachten nun die **Zugangskontrolle**. Beim Eintreffen einer neuen Verbindungsanforderung wird im Netz überprüft, inwieweit die geforderte Dienstgüte (**QoS** - ***Q**uality **of S**ervice*) in der momentanen Lastsituation garantiert werden kann. Basis der Bewertungskriterien sind einerseits die Verkehrsparameter zur Beschreibung des zu erwartenden Quellenverhaltens (z. B. Spitzenbitrate, mittlere Bitrate, Burstiness) und anderseits die Verkehrsdaten über die aktuelle Netzauslastung. Vor der Verbindungsaufnahme oder deren Ablehnung werden für alle beteiligten Netzknoten Kapazitätsberechnungen durchgeführt. Die durch die Rechenzeit bedingte Verzögerung des Verbindungsaufbaus muß möglichst klein gehalten werden.

Zugangskontrolle

Quality of Service

Technologische Randbedingungen (wie die verfügbare Prozessorleistung) setzen der Komplexität der Berechnungen enge Grenzen. Dies gilt insbesondere für die lastabhängigen dynamischen Kapazitätsberechnungen (so z. B. beim Nachverhandeln und Aktualisieren der Verkehrsparameter).

Während der Verbindung erfolgt eine kontinuierliche Überwachung der vereinbarten Verkehrsparameter durch die UPC-Funktion. Die Aufgabe der effizienten Nutzung der Ressourcen wird von den Verwaltungsfunktionen für die Betriebsmittel (*Resource Allocation*) unterstützt.

Zuteilungsstrategie

Die einfachste **Zuteilungsstrategie** besteht darin, jeweils jeder Verbindung die Spitzenbitrate zuzuteilen. Dies entspricht der Anforderung einer CBR-Verbindung und bedeutet, daß die effektive Bandbreite gleich der Spitzenbitrate gesetzt wird. Bei VBR-Verbindungen ist jedoch ein linearer Ansatz effektiver. Man setzt die effektive Bandbreite zwischen der mittleren und Spitzen-Bitrate an. Um die effektive Bitrate genauer bestimmen zu können, werden weitere statistische Parameter berücksichtigt. Als Beispiel sei die Zwei-Momentenmethode erwähnt. Das statistische Verhalten einer großen Anzahl überlagerter unabhängiger Quellen läßt sich durch eine Normalverteilung annähern. Dies ist durch zwei Parameter, den Mittelwert und die Varianz, vollständig charakterisiert. Zur Bewertung der effektiven Bandbreite werden der Mittelwert und die Varianz der einzelnen Quellen herangezogen.

Beispiel 5.7-1: Summenbildung der Bandbreitenanforderungen von zwei Verbindungen

Abb. 5.7-2: Die genutzte Bandbreite der 1. Verbindung

Abb. 5.7-3: Die genutzte Bandbreite der 2. Verbindung

Abb. 5.7-4: Die Summenbildung der genutzten Bandbreite der 1. und 2. Verbindung

Auf dem Übertragungskanal, den sich die 1. und 2. Verbindung teilen, stehen in diesem Beispiel insgesamt nur 5 Einheiten Bandbreite zur Verfügung. Überschreitet die Bandbreitensumme der 1. und 2. Verbindung den Wert von 5 Einheiten Bandbreite, so werden diese ATM-Zellen in einem Puffer zwischengespeichert - es kommt zu Zellverzögerungen - oder die ATM-Zellen werden verworfen - es treten Zellverluste in der 1. oder 2. Verbindung auf (Abb. 5.7-5).

Abb. 5.7-5: Beispiel

Die absoluten Häufigkeiten des Bandbreitenbedarfs:

Bandbreitenbedarf	Verb. 1	Verb. 2	Verb. 1+2	kummulierte Häufigkeite von 1+2
0	3	2	0	0
1	4	7	4	0,28
2	5	2	3	0,49
3	2	1	1	0,56
4	0	2	3	0,77
5	0	0	1	0,84
6	0	0	2	1

Für die empirischen Mittelwerte und Varianzen des Bandbreitenbedarfs:

	Verbindung 1	Verbindung 2	Summe	Verbindung 1+2
empirischer Mittelwert	1,428	1,571	3	3
empirische Varianz	1,032	1,648	2,68	3,384

In Abb. 5.7-6 ist der Kurvenverlauf der Verteilungsfunktion der Normalverteilung (Kurve) mit den Parametern 3 (Mittelwert) und 2,68 (Varianz) und den tatsächlichen aufsummierten relativen Häufigkeiten der Summenverbindung (Linienzug) dargestellt.

Abb. 5.7-6: Kurvenverlauf der Verteilungsfunktion der Normalverteilung

Parameterüberwachung

Die **Parameterüberwachung** d. h. die Überwachung der Einhaltung der ausge-

handelten Verkehrsparameter sowie die Gültigkeit der VPI/VCI-Werte kann an der Benutzer-Netz-Schnittstelle UNI bzw. an der Netz-Netz-Schnittstelle NNI durchgeführt werden. Am Teilnehmer-Netzzugang UNI können ATM-Zellen prinzipiell mit einer höheren Rate als angemeldet eintreffen. Ursachen hierfür sind:

- eine absichtliche Überschreitung der Grenzen, die mit dem Netzbetreiber vereinbart wurden
- Fehler im Endgerät oder Teilnehmeranschluß
- Veränderung der Quellcharakteristiken.

Eine virtuelle Verbindung, die überzählige ATM-Zellen ins Netz absetzt, beeinflußt alle Verbindungen, die die gemeinsame physikalische Teilstrecke im Netz benutzen, durch längere Wartezeiten in den Netzknoten oder gar höhere Zellverlustwahrscheinlichkeiten.

Bei der Verletzung des Verkehrsvertrages können netzseitig Abwehrmaßnahmen eingeleitet werden. So können überzählige Zellen verworfen werden oder zunächst in die niedrige Priorität (CLP = 1) eingestuft und bei Überlast verworfen werden.

Die UPC/NPC-Algorithmen sollen möglichst jede Verletzung des Verkehrsvertrages schnell erkennen. Die fehlerhafte Bewertung von Zellen als Zellen, die den Vertrag verletzen, muß so selten sein, daß die Dienstgüte der Verbindung nicht beeinflußt wird. Ferner soll die Implementierung des Algorithmus möglichst einfach sein.

Als Parameter für die Einhaltung des Verkehrsvertrages kommen in erster Linie die Spitzenzellrate und die mittlere Zellrate in Frage. Bei der Überwachung der Spitzenzellrate wird die minimale Zwischenankunftszeit zwischen den Zellen einer Verbindung überwacht. Hier gilt es eine zu definierende Toleranz der Spitzenbitrate zu gewähren. Die Überwachung der mittleren Zellrate ist bisher nicht standardisiert. Hier besteht das Problem darin, daß die Maßgenauigkeit mit der Länge des Maßintervalls steigt, eine Reaktion auf überzählige Zellen sollte jedoch so früh wie möglich erfolgen.

Beispiel 5.7-2: Generic Cell Rate Algorithm
Der Generic Cell Rate Algorithm (GCRA) dient zur algorithmischen Definition der Verkehrsparameter PCR (Peak Cell Rate) und CDVT (Cell Delay Variation Tolerance) bzw. SCR (Sustainable Cellrate) und BT (Burst Tolerance) einer Verbindung sowie zur Überwachung der Konformität eines Zellstromes zu diesen Parametern.

Es existieren zwei gleichwertige Versionen des GCRA, nämlich das Virtual Scheduling (VS) und der Continous-State-Leaky-Bucket-Algorithmus (LB). Diese sind in der ITU-T Empfehlung I.371 definiert. Für jede Folge von Zellankunftszeiten ($\{t_i, i \geq 1\}$, i: i-te Zellankunft) befinden beide Algorithmen die gleichen Zellen für konform oder nicht konform. Der GCRA hat zwei Eingabeparameter:

- I (Inkrement): von der Zellrate beinflußt

- L (Limit): von der Burstgröße bzw. CDVT beeinflußt

Der GCRA wird deshalb auch mit GCRA (I,L) bezeichnet.

Der VS-Algorithmus Für eine ankommende Zelle (Ankunftszeitpunkt t_i) wird die theoretisch vorausgesehene Ankunftszeit (TAT_i - theoretically predicted arrival time) berechnet unter der Annahme, daß Zellen im gleichmäßigen Abstand ankommen, wenn eine Quelle aktiv ist (I = Abstand zwischen zwei aufeinanderfolgenden Zellen). Mit der Initialisierung $TAT_o = 0$ und der folgenden Iteration für die i-te Zellankunft:

$$TAT_i = TAT_{i-1} + I$$

Ist die tatsächliche Ankunftszeit t_i der Zelle i nach $TAT_i - L(TAT_i - L \leq t_i)$, so wird die Zelle als konform befunden, ansonsten ist sie zu früh angekommen und wird für nicht konform gehalten. Ist $TAT_i < t_i$, so wird außerdem $TAT_i = t_i$ für die nächste Iteration gesetzt.

Der Continous-State-LB-Algorithmus (LB - Leaky Bucket)-Algorithmus Der Behälter (bucket) des LB-Algorithmus hat eine endliche Kapazität von $L + I$ Zellen. Der Inhalt des Behälters leert sich mit einer konstanten Ausströmgeschwindigkeit von einer Zelle pro Zeiteinheit. Mit jeder ankommendenden Zelle wird der Inhalt des Behälters um L Zellen vergrößert. Ist der Inhalt des Behälters kleiner als L, so wird jede ankommende Zelle als konform betrachtet, ansonsten wird die Zelle verworfen.

6 GSM

6.1 Das Konzept

Mobilfunk

Unter den Begriffen **Mobilfunk** bzw. Mobilkommunikation wird heute der gesamte Bereich der Kommunikationstechnik gefasst, der die Begriffe „mobil“ und „Funk“ als drahtlose Kommunikation in sich vereint. Obwohl das Funktelefon hier die heute dominierende Anwendung darstellt, umfasst der Begriff auch den mobilen Datenfunk oder mobiles Faxen. Ebenso kann das Autoradio (**DAB** - ***D****igital* ***A****udio* ***B****roadcast*) hinzu gerechnet werden.

GSM-Netze
D-Netz
E-Netz

Die **GSM-Netze** sind als **D-** und **E-Netze** im Laufe weniger Jahre in Deutschland (und auch im übrigen Europa) zu den dominierenden Mobilfunknetzen geworden. Zur Zeit gibt es mehr als 70 Millionen Teilnehmeranschlüsse, so dass in Deutschland im Durchschnitt auf jeden Einwohner ein Mobiltelefon kommt.

IMT 2000

Unter der Bezeichnung **IMT-2000** (***I****nternational* ***M****obile* ***T****elecommunications at* ***2000*** *MHz*) wurde von der **ITU** (***I****nternational* ***T****elecommunication* ***U****nion*) ein Mobilfunksystem spezifiziert, welches den Anwendungen weitaus höhere Bandbreiten als GSM bieten kann. In Europa ist dieses System von **ETSI** (***E****uropean* ***T****elecommunications* ***S****tandards* ***I****nstitute*) unter der Bezeichnung **UMTS** (***U****niversal* ***M****obile* ***T****elecommunications* ***S****ystem*) standardisiert worden und bietet neben der Sprachkommunikation auch Anwendungen wie z. B. mobiler Internet-Zugang und Multimedia Kommunikation an. Das erste **UMTS-Netz** in Deutschland ist seit anfang 2004 öffentlich zugänglich.

UMTS-Netz

C-Netz

Das in Deutschland bis Ende 2000 betriebene C-Netz, basierte auf analoger Übertragungtechnik, wobei die Signalisierung zwischen den Funkvermittlungsstellen von Anfang an bereits nach dem digitalen Zeichengabesystem Nr. 7 erfolgte. Schon beim **C-Netz** war – wie bei den GSM-Netzen – die Netzstruktur zellular. Auch Bündelfunk-, *Paging*- sowie *Telepoint*-Systeme basieren auf zellularen Netzen. Das Prinzip ist in Abb. 6.1-1 dargestellt. Durch eine räumliche Aufteilung in Zellen wird es möglich, die verfügbaren Frequenzen nach gewissen Schutzabständen zeitgleich mehrfach zu verwenden.

Abb. 6.1-1: Idealisierte Überdeckung der Ebene durch Zellen

Eine GSM-Zelle kann einen Radius von bis zu etwa 35 km haben. Jede Zelle wird von einer Funkstation (auch: Basisstation) versorgt, die mit dem sich in dieser Zelle befindenden Endteilnehmern per Funk kommuniziert. Die Basisstationen sind über normale terrestrische Verbindungen oder auch per Richtfunk mit Vermittlungsstellen verbunden, die den Übergang zu anderen öffentlichen Netzen (z. B. ISDN) herstellen.

Die Zellenstruktur wurde (wie auch schon beim C-Netz) so konzipiert , dass eine Funktelefonverbindung beim Wechsel des Teilnehmers aus dem Bereich einer Funkzelle nicht abgebrochen, sondern an die nächste Zelle „weitergereicht“ wird – man spricht von **Handover**. Ferner existiert die Möglichkeit, einen Teilnehmer bundesweit über seine Nummer zu erreichen. Dies wird dadurch möglich, dass das eingeschaltete Endgerät „dem Netz“ den Wechsel des Teilnehmers in eine andere Funkzelle mitteilt; im Netz wird eine Datenbank gepflegt, die die aktuellen Aufenthaltsorte der Teilnehmer speichert. Diese Auffindbarkeit des Teilnehmers wird als **Roaming** bezeichnet, wobei in den GSM-Systemen auch Roaming zwischen verschiedenen Netzen – die u. U. in verschiedenen Ländern betrieben werden – möglich ist. Internationales Roaming setzt natürlich voraus, dass die entsprechenden Netzbetreiber ein wechselseitiges Roaming-Abkommen geschlossen haben, in dem insbesondere die Fragen der gegenseitigen Gebührenabrechnung geregelt sind.

Handover

Roaming

Das C-Netz arbeitete im 450 MHz-Bereich. Ähnliche Netze existierten in den anderen europäischen Ländern. Mangels einheitlicher technischer Standards war jedoch hierbei (von Ausnahmen abgesehen, z. B. innerhalb der Nordischen Länder) keine grenzüberschreitende Nutzung möglich. Demgegenüber ist das GSM-System von vornherein als europaweites, auf moderner digitaler Technologie basierendes System konzipiert worden.

GSM

Der Name **GSM** leitet sich aus der Spezifikationsgruppe ***G**roupe **S**peciale **M**obile*, die heute innerhalb von ETSI arbeitet, ab. Seitdem außerhalb Europas eben-

falls GSM-Netze eingerichtet wurden, wird GSM auch häufig als *Global System for Mobile Communication* verstanden. In Deutschland wurden die ersten GSM-Systeme unter dem Namen D-Netze (im Bereich um 900 MHz) implementiert. Später kamen die E-Netze hinzu, welche in einem höheren Frequenzbereich (um 1800 MHz) arbeiten.

Die folgenden Aspekte standen bei der Konzeption der GSM-Netze im Vordergrund:

- Die Teilnehmer können ihre Endgeräte europaweit benutzen.
- Die begrenzten Ressourcen an Frequenzen werden ökonomisch genutzt.
- Die Sprache wird mit erheblich besserer Qualität (gegenüber analogen Systemen) übertragen.
- Infrastruktur und Endgeräte werden durch den großen Teilnehmermarkt preislich vergleichsweise günstig.
- Durch die Miniaturisierung in der Digitaltechnik sind Handtelefone geringer Abmessung möglich.

Charakteristisch für die Sprachübertragung bei GSM sind neben der Nutzung der Digitaltechnik, leistungsfähige Verfahren zum Erkennen von Sprachpausen und Hintergrundgeräuschen. Dadurch kann die Menge der übertragenen Signale, mit dem Resultat eingeschränkter Interferenzen und geringeren Stromverbrauchs, reduziert werden.

Daneben bietet die GSM-Technik aufgrund der digitalen Funkübertragung prinzipiell die Möglichkeit der Unterstützung von Daten- und Telematikdiensten. Allerdings sind die GSM-Netze im Vergleich zu ISDN zunächst stärker auf die Übertragung von Sprache optimiert, weshalb für die Datenübertragung zusätzliche Maßnahmen erforderlich sind – auf diesen Punkt wird an späterer Stelle noch einmal eingegangen.

SIM-Karte

Ein GSM-Netz bietet den Teilnehmern eine besondere Art der Mobilität: Dem einzelnen Benutzer ist eine **SIM-Karte** zugeordnet (***S**ubscriber **I**dentity **M**odule*), die in verschiedenen Endgeräten genutzt werden kann.

In Abb. 6.1-2 ist die Einbettung der GSM-Netze in die Struktur der öffentlichen Netze dargestellt:

Abb. 6.1-2: Einbettung der GSM-Netze in die Struktur öffentlicher Netze

Selbsttestaufgabe 6.1-1:

Welche Netzstruktur haben GSM-Netze und und was für Vorteile bringt dies mit sich?

6.2 Die Dienste

In den GSM-Netzen werden neben Sprachdiensten auch Datendienste angeboten. Dabei orientieren sich die angebotenen Dienste an denen des ISDN. Insbesondere wird bei GSM zwischen **Basisdiensten** und **Zusatzdiensten** unterschieden, wobei Basisdienste nochmals in Trägerdienste (Übermittlungsdienste) und Teledienste unterteilt werden. Fernsprechen und Telefax zählen zu den Telediensten. Sie umfassen die Kommunikationsprotokolle aller OSI-Schichten und stellen auch die Kompatibilität der Endgeräte sicher. Interessante Zusatzdienste sind beispielsweise die Mailbox, der Kurznachrichtendienst *Short Message Service* oder der Verkehrsfunk.

Basisdienste
Zusatzdienste

Das Fernsprechen ist der am meisten genutzte GSM-Dienst. Erwähnenswert ist hierbei, dass der **Notruf** von jedem GSM-Telefon durchgeführt werden kann – und zwar auch ohne SIM-Karte und unabhängig vom „Heimatnetz“ des Teilnehmers; der Notruf wird stets an die örtlich nächstgelegene Notrufzentrale weitergeleitet.

Notrufdienst

Für die Telefonie sind eine Reihe von Zusatzdiensten (wie beispielsweise Rufumleitung) verfügbar, die teilweise auch mit Datenübertragungsdiensten kombinierbar sind.

Ein weiterer GSM-Teledienst ist der Telefaxdienst. Fax der Gruppe 3 wird in GSM voll unterstützt. Damit eine solche Faxübertragung vom Handy aus funktioniert, muss ein Faxmodem die analogen Modemsignale des üblichen Gruppe-3-Faxgerätes zunächst in die digitalen Ausgangssignale der Faxcodierung zurückverwandeln; diese Signale werden dann unter Umgehung des Sprachcoders an die

Basisstation gesendet. Diese »Umgehung« ist notwendig, da andernfalls das für die Sprachübertragung optimierte Kompressionsverfahren im Sprachcoder die Faxsignale zerstören würde.

Für den Faxempfang seitens des Mobilteilnehmers muss das Mobilgerät von Sprach- auf Faxbetrieb umgeschaltet werden. Eine andere Möglichkeit bietet das bei GSM vorgesehene **Multinumbering**, bei dem für die verschiedenen Verbindungsarten eigene Rufnummern zur Verfügung stehen.

Multinumbering

Von größerer Bedeutung ist ferner der Kurznachrichtendienst **SMS** (***S**hort **M**essage **S**ervice*) für Daten begrenzten Umfangs. Hierbei ist es möglich, selbst mit einem Handy kürzere Nachrichten (bis zu 160 Zeichen) zu senden und zu empfangen. Der Empfang solcher Nachrichten ist auch aus anderen Netzen möglich; für diesen Zweck richtet der GSM-Netzbetreiber eine spezielle Vermittlungsstelle ein, das *Short Message Service Center*, welches die gewünschte Nachricht an den GSM-Teilnehmer absetzt.

Short Message Service

GSM-Trägerdienste sind die synchronen und die asynchronen Datenübertragungsdienste mit verschiedenen Geschwindigkeiten. Die beim Telefax geschilderten Besonderheiten wie Umgehung des Sprachcoders und *Multinumbering* treffen auch auf die Datendienste zu.

GSM-Trägerdienste

Die **Datendienste** wurden bis zur Einführung der Zusatzdienste, der Phase 2+ (siehe unten), in den GSM-Netzen nur leitungsvermittelt zur Verfügung gestellt. Mit anderen Worten fand die Kommunikation über eine Wählverbindung statt, für die die üblichen zeitabhängigen Gebühren zu entrichten waren. Dabei wurden folgende Übertragungsgeschwindigkeiten unterstützt: 300, 1200, 2400, 4800 und 9600 bit/s.

GSM-Datendienste

Die Datendienste unterscheiden sich ferner hinsichtlich der folgenden Grundmerkmale:

- Signaltyp (analog/digital)
- Betriebsverfahren (asynchron/synchron)
- Übertragungsfunktion (transparent/protokollgebunden).

In der detaillierteren Betrachtung muss auch nach der Verbindungskonfiguration unterschieden werden. Je nachdem, in welchem Netz sich der Kommunikationspartner befindet (GSM, ISDN, analoges Fernsprechnetz), können bestimmte Datendienste unter Einbeziehung entsprechender Adapter bzw. Modems in Anspruch genommen werden.

Aus Sicht der Anwender sind die bei GSM angebotenen Möglichkeiten des **Zugangs zu paketvermittelten Datennetzen** von besonderem Interesse. Für den Zugang zu X.25-Netzen (wie Datex-P) sind die hauptsächlichen Varianten der asynchrone Zugang über den nächstgelegenen **PAD** (***P**acket **A**ssembly/**D**isassembly Facility*) des X.25-Netzes bzw. der synchrone Zugang über eine leitungsvermittelte ISDN-Verbindung, bei der der nächstgelegene **PH** (***P**acket **H**andler*) im ISDN angewählt werden muss.

Zugang zu paketvermittelten Datennetzen

Zusatzdienste werden immer an einen Teledienst oder einen Trägerdienst gebunden. In GSM wurden Zusatzdienste in verschiedenen Phasen standardisiert (Phase 1, 2, 2+). Die Zusatzdienste der **Phase 1** behandeln verschiedene Arten der Rufweiterleitung und das Sperren bestimmter Anrufe. Die Rufweiterleitung kann auf eine beliebige Telefonnummer oder eine im GSM-Netz implementierte Mailbox erfolgen. Diese übernimmt die Funktion eines Anrufbeantworters und kann vom Teilnehmer später von seinem Endgerät aus abgefragt werden. Das Sperren von Rufen ist besonders dann interessant, wenn die anfallenden Gebühren beschränkt werden sollen, z. B. bei der Nutzung von Firmentelefonen. Die Zusatzdienste der Phase 1 sind in Tabelle 6.2-1 zusammengefasst.

Phase 1

Tab. 6.2-1: *GSM Zusatzdienste der Phase 1*

Zusatzdienst	Abkürzung
Call Forwarding Unconditional	CFU
Call Forwarding on Mobile Subscriber Busy	CFB
Call Forwarding on No Reply	CFNRy
Call Forwarding on Mobile Subscriber Not Reachable	CFNRy
Barring of All Outgoing Calls	BAOC
Barring of Outgoing International Calls	BOIC
Barring of Outgoing International Calls except those directed towards the Home PLMN Country	BOIC-exHC
Barring of All Incoming Calls	BAIC
Barring of Incoming Calls when Roaming outside of the home PLMN country	BIC-Roam

Phase 2

In der **Phase 2** wurden viele weitere Zusatzdienste eingeführt. Wie im ISDN-Netz kann man sich die Rufnummer eines Anrufers anzeigen lassen. Außerdem kann man sich anzeigen lassen, mit welcher Nummer man tatsächlich verbunden wurde, was bei einem weitergereichten Anruf interessant sein kann. Diese Funktionen lassen sich vom jeweils anderen Teilnehmer deaktivieren. Außerdem kann man – wie beim ISDN – bei einem Teilnehmer, der bereits einen Dienst nutzt, anklopfen. Dieser kann seine bestehende Verbindung halten und rückfragen. Es gibt außerdem Konferenz-Dienste und geschlossene Benutzergruppen. Diese und weitere Zusatzdienste werden in Tabelle 6.2-2 aufgelistet.

Tab. 6.2-2: *GSM Zusatzdienste der Phase 2*

Zusatzdienst	Abkürzung
Calling Line Identification Presentation	CLIP
Connected Line Identification Presentation	CoLP
Calling Line Identification Restriction	CLIR
Connected Line Identification Restriction	CoLR
Malicious Call Identification	MCI
Advice of Charge	AoC
Freephone Service	FPH
Reverse Charging	REVC
Closed User Group	CUG
Three Party Service	3PTY
Conference Calling	CONF
Call Transfer	CT
Mobile Access Hunting	MAH
Call Waiting	CW
Call Hold	HOLD
Completion of Call to Busy Subscriber	CCBS
User to User Signalling	UUS

Phase 2+

In weiteren Ausbaustufen der GSM-Netze sind im Rahmen der Phase 2+ nochmals neue Dienste und Zusatzdienste angeboten worden. Besonders wichtig sind die neuen Datendienste, da die bisher unterstützten Datendienste niedrige Bitraten hatten (max. 9,6 kbit/s pro Kanal) und verbindungsorientiert waren. Im Rahmen der Phase 2+ wurden verbindungsorientierte Datendienste mit hoher Bitrate – **HSCSD** (***H****igh* ***S****peed* ***C****ircuit* ***S****witched* ***D****ata*) – und paketorientierte Datendienste variabler Bitrate – **GPRS** (***G****eneral* ***P****acket* ***R****adio* ***S****ervice*) – eingeführt.

HSCSD

GPRS

Beispiel 6.2-1:
Verbindungsorientierte Datendienste hoher Bitrate lassen sich entwickeln, indem acht Kanäle einer Trägerfrequenz zu einem Datenkanal mit einer Datenrate von $76,8\,kbit/s\,(=\,8\cdot 9,6\,kbit/s)$ zusammengefasst werden (HSCSD).

Paketorientierte Datendienste variabler Bitrate wurden unter dem Stichwort *General Packet Radio Service* entwickelt. Die Einführung von GPRS erforderte gegenüber HSCSD einen höheren Aufwand sowohl hinsichtlich der GSM-Infrastruktur wie auch der Anpassung der Endgeräte. Vorteil von GPRS ist die - mit der Paketvermittlung einhergehende - bessere Nutzung der Resourcen sowie die Möglichkeit volumenorientierter Abrechnungsmodelle.

Selbsttestaufgabe 6.2-1:

In welche zwei Gruppen werden GSM-Dienste unterteilt? Nennen Sie ein Beispiel für den GSM-Teledienst.

6.3 Netz- und Protokollarchitektur

In diesem Abschnitt wird das GSM-Netz in seiner Netz- und Protokollarchitektur besprochen und die Schnittstellen zwischen den Systemkomponenten, die Schichtung der Protokollfunktionen und die Abgrenzung vom Transportnetz und der Signalisierung behandelt.

6.3.1 Netzkomponenten und Systemhierarchie

Ein GSM-Netz besteht grundsätzlich aus den im Folgenden beschriebenen Elementen (siehe Abb. 6.3-1).

Abb. 6.3-1: GSM Systemübersicht

MS Zunächst hat man die Endgeräte, die hier als ***M**obile **S**tation* (**MS**) bezeichnet werden. Es kann sich hier um die sich immer weiter verbreitenden Handys, aber auch um fest installierte Autotelefone handeln. Die **SIM-Karte** enthält auf einem Chip alle relevanten Daten des Teilnehmers. SIM-Karten sind in zwei Formaten verfügbar: in einer scheckkartengroßen Variante sowie in einer miniaturisierten Plug-In-Variante. Mittlerweile hat sich bei den Handys die kleinere Plug-In-Variante durchgesetzt.

SIM-Karte

Ein Endgerät tritt über eine ***B**ase **S**tation* (**BS**), die man als Funkfeststation übersetzen kann, mit dem Netz in Kontakt. Die Base Stations sind Teil des Festnetzes, d. h. Funkverkehr findet nur zwischen MS und BS statt. In einer feineren Unterteilung sind innerhalb einer BS zu unterscheiden: ***B**ase **T**ransceiver **S**tation* (**BTS**) und ***B**ase **S**tation **C**ontroller* (**BSC**). Die BTS ist der reine Funkteil für Senden und Empfangen, während die BSC den Steuerungsteil der BS darstellt; eine BSC kann auch mehrere BTS steuern.

BS

BTS

BSC

Beispiel 6.3-1:
Bei den GSM-900-Systemen (also den D-Netzen in Deutschland) werden für die funktechnische Übertragung zwei 25 MHz breite Frequenzbänder im Bereich um 900 MHz genutzt. Unter Wahrung eines vorgegebenen Kanalabstandes stehen so pro Richtung 124 Kanäle zur Verfügung. Als Modulationsverfahren zur Erzeugung des Funksignals wird ***G**aussian **M**inimum **S**hift **K**eying* (**GMSK**) verwendet. Die Übertragungskapazität eines Funkkanals wird im synchronen Zeitmultiplexverfahren gesplittet, indem acht aufeinander folgende Zeitschlitze einen Pulsrahmen der Periodendauer 4,6125 ms bilden. Jeder dieser acht Zeitschlitze kann nun als Verkehrskanal oder zur Übertragung von Synchronisations- bzw. Signalisierdaten benutzt werden. Wegen dieses gestuften Zugriffs über Frequenz und Zeitschlitz auf einen Verkehrskanal seitens der Mobilstation spricht man auch von einem kombinierten **FDMA/TDMA**-Verfahren (***F**requency* bzw. ***T**ime **D**ivision **M**ultiple **A**ccess*).

In der gegenüber den Basisstationen höheren Hierarchiestufe, der Ebene des ***S**witching **S**ubsystem* (**SSS**), sind die ***M**obile **S**witching **C**enter* (**MSC**) angesiedelt, die eigentlichen Funkvermittlungsstellen. Neben der Vermittlungsfunktion zwischen den verschiedenen MSCs stellen sie auch die Übergänge zu anderen Netzen wie ISDN dar. Zu dieser Ebene gehören ebenfalls:

SSS, MSC

- ***H**ome **L**ocation **R**egister* (**HLR**)
- ***V**isitor **L**ocation **R**egister* (**VLR**)
- ***A**uthentication **C**enter* (**AC**)
- ***E**quipment **I**dentity **R**egister* (**EIR**)

HLR, VLR, AC und EIR

In einem HLR als „Heimatdatenbank“ werden die Daten der zugeordneten Teilnehmer verwaltet. Ein solches HLR kann mehrere 100000 Teilnehmer verwalten, weshalb in großen Netzen mehrere solcher Datenbanken nötig sind. Im VLR werden die Kennungen derjenigen Teilnehmer gespeichert, die sich aktuell in dem betreffenden Bereich aufhalten, aber eigentlich woanders „beheimatet“ sind; diese Bereiche werden **Location Areas** genannt und bestehen aus mehreren Funkzellen. Die Komponenten AC und EIR betreffen Sicherheitsaspekte.

Location Areas

Das AC, welches einem HLR zugeordnet und in der Regel physikalisch auf demselben Rechner realisiert ist, verwahrt den persönlichen Schlüssel eines Teilnehmers, der für Zwecke der Authentifikation und Verschlüsselung benötigt wird. Das AC gibt einen solchen Schlüssel niemals preis, sondern nur aus diesem Schlüssel

abgeleitete Größen, die im vom Teilnehmer besuchten VLR (und zwar einige „auf Vorrat“) abgelegt werden.

Im EIR können gestohlene Geräte als gestohlen registriert werden. Da beim Einbuchen und beim Verbindungsaufbau auch die Identifikationsnummer des verwendeten Geräts übertragen wird, kann das Netz im EIR registrierte Geräte sperren.

Die Informationen - Sprache oder Daten - werden zwischen MS und BS mobil über Funk auf 16 kbit/s-Kanälen übertragen. Innerhalb des Festnetzes können dann 64 kbit/s-Kanäle genutzt werden, wie sie auch im ISDN üblich sind. Die BS werden sternförmig an die Vermittlungsstellen (MSC) herangeführt, die ihrerseits untereinander vermascht sind.

SS7

Für die Signalisierung wird das im ISDN-Netz verwendete **S**ignalisier**s**ystem Nr. **7** (**SS7**) eingesetzt, mit dessen Hilfe beispielsweise auch – bei eingeschaltetem Endgerät – die Information über den Wechsel eines mobilen Teilnehmers in eine andere *Location Area* an dessen HLR übermittelt wird.

Subsysteme: BSS, NSS, OSS

Das ***B**ase **S**tation **S**ubsystem* (**BSS**) besteht aus den beiden funkspezifischen Netzelemente BTS und BSC. Das *Mobile Switching Center*, *Visitor Location Register* und *Home Location Register* bilden zusammen das ***N**etwork and **S**witching **S**ubsystem* (**NSS**). Dazu gehören auch die ***G**ateway **MSC**s* (**GMSC**), die den Übergang zu anderen öffentlichen Netzen schaffen. Das ***O**peration and **M**aintenance **C**enter* (**OMC**) bildet mit dem *Equipment Identification Register* und dem *Authentication Center* das ***O**peration **S**ub**s**ystem* (**OSS**). Das OSS hat folgende Aufgaben: Teilnehmerverwaltung (*Subscription Management*), Verwaltung der Endgeräte (*Mobile Equipment Management*) und Netzbetrieb und Wartung (*Operation and Maintenance*). Das OMC dient der Steuerung und Überwachung der anderen Netzelemente (vgl. Abb. 6.3-1). Über das OMC können Eingriffe in diese Netzelemente vorgenommen werden. Weitere Informationen zum Netz-Management finden sich in Abschnitt 6.6.

Abb. 6.3-2: Systemhierarchie in GSM

Systemhierarchie

In GSM wurde eine **Systemhierarchie** definiert (vgl. Abb. 6.3-2), die in den unteren Ebenen auf die Systemkomponenten und ihre »Zuständigkeitsbereiche« bezug nimmt. Die oberen Ebenen beziehen sich auf die Gesamtnetze und deren Versorgungsgebiete. An der obersten Stelle steht das GSM-Systemgebiet. Es bezeichnet den Bereich der Erde, in dem ein GSM-Endgerät betrieben werden kann. Dieses Gebiet setzt sich aus den Versorgungsgebieten der GSM-Netzbetriebe (**PLMN** - ***P**ublic **L**and **M**obile **N**etwork*) zusammen, die sich überlappen können; in vielen Ländern, in denen GSM-Netze betrieben werden, gibt es konkurrierende Anbieter. Innerhalb der Netzbetriebe ergibt sich die weitere Strukturierung entsprechend der Systemkomponenten MSC, BSC und BTS. Die größte Einheit ist das Gebiet, dass einem *Mobile Switching Center* zugeordnet wird. Ein MSC kann einige zehn- bis hunderttausend Teilnehmer verwalten, so dass sich große Gebiete ergeben können. Einem MSC sind einige *Base Station Controller* zugeordnet, an die wiederum einige *Base Transceiver Stations* angeschlossen sind. Jede BTS hat eine oder mehrere (typisch drei oder sechs) Antennen, deren Funkfeld jeweils eine Zelle abdeckt. Alle Zellen, die zu einem BSC gehören, bilden das BSC-Gebiet. Alle BSC-Gebiete, die zu einem MSC gehören, bilden das MSC-Gebiet.

PLMN

Location Area

Eine Besonderheit in diesem Schema sind die **Location Areas**, da sie kein physikalisches Gegenstück in Form einer Systemkomponente haben; sie sind rein logi-

sche Konstruktionen. Eine *Location Area* ist formal eine Menge von Zellen. Für jeden Teilnehmer wird in einer Datenbank gespeichert, in welcher *Location Area* er sich zur Zeit aufhält. Diese Information ist Grundlage für den Lokalisierungsprozess (vgl. Abschnitt 6.5.2.1 und Abschnitt 6.5.3.1). Meist sind *Location Areas* so gewählt, dass sie ein oder mehrere BSC-Gebiete komplett umfassen – dies ist aber nicht bindend (vgl. Abb. 6.3-2). Die einzige Einschränkung besteht darin, dass alle Zellen einer *Location Area* zum gleichen MSC gehören müssen.

6.3.2 Schnittstellen und Schichtung der Protokolle

Funk- und Mobilvermittlungsnetz

Ein GSM-Netz setzt sich aus einem **Funknetz und** einem **Mobilvermittlungsnetz** zusammen. Das Funknetz umfasst die Komponenten, die die Funkübertragung durchführen bzw. kontrollieren. Die Komponenten im Mobilvermittlungsnetz haben sowohl mobilfunkspezifische Aufgaben – wie z. B. die Verwaltung der Aufenthaltsorte der Endgeräte – als auch Routing-Aufgaben.

Abb. 6.3-3: Übersicht über die internen Schnittstellen in GSM

Übersicht Schnittstellen

Die einzelnen Komponenten haben ihre spezifischen Aufgaben zu erfüllen, die bereits in Abschnitt 6.3.1 erläutert wurden. Außerdem müssen sie über die Schnittstellen zusammenarbeiten. Aus der Aufgabenverteilung ergibt sich, welche Informationen über die Schnittstellen ausgetauscht werden. Abb. 6.3-3 zeigt die internen Schnittstellen in GSM. Die Schnittstellen im Funknetz – U_m und Abis – sind funkspezifisch und speziell an die Bedürfnisse von GSM angepasst. Die anderen Schnittstellen – A, B, C, D, E, F und G – basieren auf dem Signalisierungssystem Nr. 7 (SS7). Der Grund dafür liegt darin, dass die Komponenten, die durch diese Schnittstellen verbunden sind, weitgehend Aufgaben erfüllen, die in ISDN oder anderen Festnetzen auch auftreten, wie z. B. die Vermittlung von Gesprächen (MSC) oder die Speicherung von Daten (VLR, HLR, EIR). Zwei weitere Schnittstellen – R und S – wurden für das Mobile Endgerät definiert.

Abb. 6.3-4: Alternative Anschlusskonfigurationen des Mobilen Endgeräts

Referenzkonfiguration Endgeräte

Abb. 6.3-4 zeigt die GSM-**Referenzkonfiguration** für Mobile Endgeräte. Bei der Spezifikationen von GSM wurden die bestehenden Telefonnetze und die Entwicklungen beim ISDN berücksichtigt. Endgeräte mit verschiedenen Schnittstellen sollten mit einem GSM-Netz verbunden werden können. Die Funkschnittstelle (U_m) ist zwar GSM-spezifisch, nachfolgende Schnittstellen (R und S) entsprechen aber den Empfehlungen der ITU-T X- bzw. V-Serie.

R- und S-Schnittstelle

Als Netzabschluss (U_m-Schnittstelle) dient immer der Mobile Termination (MT) genannte Funktionsblock in einer der drei Ausprägungen MT0, MT1, oder MT2. MT0 unterstützt Sprach- und Datendienste; das Endgerät ist integriert. Es handelt sich um ein einfaches Handy. MT1 hat eine ISDN S-Schnittstelle, an die ISDN-Endgeräte (TE1) angeschlossen werden können. Alternativ setzt ein Terminal Adapter (TA) an der **S-Schnittstelle** die Protokolle um und stellt die **R-Schnittstelle** für herkömmliche Endgeräte (TE2) zur Verfügung. Der Netzabschluss MT2 integriert einen Terminal Adapter und hat eine externe R-Schnittstelle für herkömmliche Endgeräte.

Überblick Schichtung

Die Protokolle, die über die verschiedenen Schnittstellen ablaufen, sind geschichtet aufgebaut. GSM orientiert sich dabei an der OSI-Schichtung der Funktionen in Kommunikationssystemen. Durch den großen Anteil funkspezifischer Funktionen und Anforderungen, die im OSI-Modell nicht vorgesehen waren, ergeben sich einige Abweichungen. Abb. 6.3-5 zeigt die Verteilung der Protokollfunktionen auf die Teilsysteme Mobiles Endgerät, *Base Station Subsystem* (BSS) und *Mobile Switching Center* und die Schnittstellen zwischen diesen Teilsystemen. Die Unterteilung des BSS in *Base Station Controller* und *Base Transceiver Station* wurde nicht berücksichtigt. Die Schnittstellen werden mit U_m, A, B, C und E bezeichnet. Die U_m-Schnittstelle ist GSM-spezifisch. Die anderen Schnittstellen basieren auf dem Signalisiersystem Nr. 7 (SS7). Signalisiernachrichten können über das ISDN-Netz international ausgetauscht werden. Die durchgehende Linie im Bild deutet die physikalischen Verbindungen an, während die gestrichelten Doppelpfeile den logischen Austausch von Signalisierinformationen bezeichnen.

Abb. 6.3-5: Protokollarchitektur des GSM-Systems auf der Signalisierebene

Schichtung im Endgerät

Abb. 6.3-5 zeigt ganz links die **Schichtung im Endgerät**. Auf der untersten Schicht – bezeichnet mit TDMA/FDMA – werden die Signalisierkanäle physikalisch realisiert (vgl. Abschnitt 6.4.2.2 und Abschnitt 6.4.2.3). In der Schicht zwei wird das $LAPD_m$-Protokoll (***L**ink **A**ccess **P**rocedure **D m**obile*) realisiert. $LAPD_m$ erweitert LAPD (ISDN) um funkspezifische Funktionen. Mit diesem Protokoll werden die Signalisiernachrichten der Schicht drei zwischen BSS und Endgerät gesichert übertragen.

Darauf setzten das ***R**adio **R**esource Management* (**RR**), das ***M**obility **M**anagement* (**MM**) und das ***C**onnection **M**anagement* (**CM**) auf, die zusammen die Schicht drei im Mobilen Endgerät bilden.

Radio Ressource Management

Das **Radio Ressource Management** im Mobilen Endgerät und im BSS verwalten die Funkressource. Das umfasst den Aufbau, den Abbau und die Unterhaltung von Funkverbindungen zwischen Endgerät und BSS, die Auswahl einer Funkzelle beim Verbindungsaufbau, die Überwachung des Broadcast-Kanals (***B**roadcast **C**ontrol **Ch**annel* - **BCCH**) und des Paging-Kanals, die Überwachung der Qualität der Funkverbindung, Leistungsregelung, Synchronisierung, Chiffrierung und andere Funktionen.

Mobility Management

Das **Mobility Management** im Mobilen Endgerät und im MSC bearbeiten Aufgaben, die sich aus der Mobilität der Teilnehmer ergeben. Im einzelnen sind das Aufgaben der Lokalisierung, Identifizierung und Authentisierung der Teilnehmer, Vergabe der TMSI, *Location Update*, Registrierung bzw. De-Registrierung von Teilnehmern auf einem Endgerät und andere Funktionen.

Connection Management

Das **Connection Management** im Mobilen Endgerät arbeitet (ähnlich wie das Mobility Management mit dem MM) mit dem CM im MSC zusammen. Es behandelt neben der Steuerung von Sprach- und Datendiensten (***C**all **C**ontrol* - **CC**) den *Short Message Service* und die Zusatzdienste (***S**upplementary **S**ervices* - **SS**). Die *Call Control* Funktionen sind u. a. verantwortlich für den Auf- und Abbau von Verkehrsverbindungen, die Unterhaltung und Modifikation bestehender Verbindungen und für Notrufe.

Schichten im BSS

Im mittleren Bereich von Abb. 6.3-5 ist die **Schichtung im BSS** zu sehen. Weitere Funktionsblöcke, die nur im Zusammenspiel der BSS-Komponenten BTS und BSC eine Rolle spielen, werden nicht aufgeführt. Die bisher nicht besprochenen Blöcke ***M**essage **T**ransfer **P**art* (**MTP**), ***S**ignalling **C**onnection **C**ontrol **P**art* (**SCCP**) und ***B**ase **S**tation **S**ystem **M**anagement **A**pplication **P**art* (**BSSMAP**) bzw. ***D**irect **T**ransfer **A**pplication **P**art* (**DTAP**) dienen der Zusammenarbeit von BSS und MSC. Die entsprechenden Signalisiernachrichten sind Elemente der A-Schnittstelle zwischen BSS und MSC. Die Signalisierung erfolgt mit SS7 Nachrichten; im BSS ist eine Teilmenge des *Message Transfer Part* des SS7 realisiert. Auf der MSC-Seite ist der gesamte MTP implementiert. Der MTP besteht aus den drei Unterschichten MTP 0, MTP 1, und MTP 2. Diese unterstützen u. a. den Transport und die Sicherung von Signalisiernachrichten, sind für die Überlastabwehr zuständig und behandeln das Link-Management. Sie decken damit die Schichten 1, 2 und einen Teil der Schicht 3 des OSI-Modells ab.

SCCP

Der **SCCP** deckt die restlichen Funktionen der dritten OSI-Schicht ab. Es handelt sich um erweiterte Adressiermöglichkeiten und verbindungsorientierte und nicht-verbindungsorientierte Signalisierdienste. Wiederum ist im BSS nur ein Teil der SCCP-Funktionalität implementiert.

BSSAP
DTAP

Die mobilfunkspezifische Signalisierung zwischen BSS und MSC wird durch die Kombination aus BSSMAP und DTAP – dem ***B**ase **S**tation **S**ystem **A**pplication **P**art* (**BSSAP**) – abgedeckt. Der **DTAP** transportiert die Nachrichten des *Call Management* und des *Mobility Management*, die zwischen Endgerät und MSC ausgetauscht werden, transparent durch das BSS. Der BSSMAP bietet auf der MSC-Seite eine Schnittstelle für die Kontrolle über bestimmte Teile des Managements der Funkressource im BSS. Das MSC nutzt diese Möglichkeit zum Eingriff in das *Radio Ressource Management* der BSS, z. B. wenn es zur gleichmäßigeren Lastverteilung *Handover* initiiert. Der BSSMAP setzt die Nachrichten des MSC in die entsprechenden RR-Nachrichten um und umgekehrt.

Schichten im MSC

Auf der Rechten Seite von Abb. 6.3-5 sind auf dem gemeinsamen *Message Transfer Part* zwei weitere »Säulen« zu sehen, die einerseits aus ***M**obile **A**pplication **P**art* (**MAP**), ***T**ransactions **C**apabilities **A**pplication **P**art* (**TCAP**) und SCCP und andererseits aus dem ***IS**DN **U**ser **P**art* (**ISUP**) bestehen. Die entsprechenden Funktionen dienen der Signalisierung des MSC mit anderen MSCs bzw. VLR/HLR bzw. des MSC mit ISDN-Netzen.

MAP
TCAP
ISUP

Der **MAP** ist eine Erweiterung des SS7-Protokolls für den Mobilfunk. Er wird auch für die Signalisierung von VLR/HLR untereinander genutzt. Mit den Nachrichten des MAP können Einträge der Benutzer im VLR und HLR abgefragt, geändert verschoben oder gelöscht werden. Es handelt sich u. a. um den Aufenthaltsort der Benutzer und benutzerspezifische Einstellungen. Der **TCAP** und der SCCP werden für den Transport der Nachrichten genutzt. Der **ISUP** ist die Schnittstelle zum ISDN-Netz. Die GSM-Signalisierung des *Call Management* wird in den ISUP umgesetzt.

6.3.3 Transport und Signalisierung

Signalisier- und Transportebene

In GSM-Netzen laufen der Transport von Nutzdaten und die Signalisierung auf logisch voneinander unabhängigen Ebenen ab. Abb. 6.3-6 zeigt die Unterteilung eines GSM-Netzes in eine **Transportebene und** eine **Signalisierebene**. Auf der Transportebene findet der Transport von Nutzdaten statt – z. B. Telefongespräche und Datenkommunikation – während die Signalisierebene Signalisierinformationen überträgt; dies sind Informationen, die z. B. zum Verbindungsaufbau bzw. -abbau nötig sind.

Abb. 6.3-6: Transport- und Signalisierebene

Das Zusammenspiel wird am Beispiel eines vom Festnetz aus initiierten Anrufs bei einem Handy deutlich, der hier in groben Zügen skizziert wird. Eine Übersicht über die hier verwendeten und weitere Kennungen in GSM gibt Abschnitt 6.5.1.

Beispiel 6.3-2:
Am Endgerät im Festnetz wird die ISDN-Nummer (***M**obile **S**ubscriber **ISDN*** - **MSISDN**) des Angerufenen eingegeben. Der *Local Exchange* kann an der gewählten Nummer erkennen, in welchem Mobilfunknetz der Angerufene registriert ist. Mit dieser Information kann der Anruf auf der Transportebene bis zu einem GMSC des entsprechenden GSM-Netzes geleitet werden. Von dort aus wird dann auf der Signalisierebene das zu dem Teilnehmer gehörende HLR angesprochen, worauf dieses die ***M**obile **S**tation **R**oaming **N**umber* (**MSRN**) übermittelt. MSC und HLR sind *Signalling Point Codes* zugeordnet, mit denen sie auf der Signalisierebene adressiert werden. Auf der Signalisierebene benutzt man das Zeichengabesystem Nr. 7 (***S**ignalling **S**ystem No. **7*** - **SS7**).

Mit der MSRN kann der Anruf dann auf der Transportebene bis zu dem MSC geleitet werden, der dem angerufenen Teilnehmer am nächsten liegt. Das VLR besitzt die nun nötigen weiteren Informationen. Es ist üblich, dass MSC und VLR gemeinsam implementiert werden, so dass hier meist nur eine interne Signalisierung nötig ist. Anschließend wird das Handy in der *Location Area* gepaget, in der es zur Zeit registriert ist. Wenn es sich in einer Zelle meldet, ist sein Aufenthaltsort bekannt und der Verkehrskanal kann auf der Transportebene bis zum Handy geschaltet werden.

Selbsttestaufgabe 6.3-1:

Stellen Sie den Unterschied zwischen einem HLR und einem VLR dar und beschreiben Sie kurz einen MSC.

6.4 Übertragungstechnik

6.4.1 Funktechnik

Modulationsverfahren

Für die Funkübertragung benutzt man in GSM die Phasenmodulation. Es handelt sich um ein Minimum Shift Keying Verfahren, bei dem eine zusätzliche Filterung durch einen Gauß-Tiefpass erfolgt. Man bezeichnet dieses Verfahren als ***G**auß **M**inimum **S**hift **K**eying* (**GMSK**). Es wurde als Kompromiss zwischen spektraler Effizienz und Implementierungsaufwand gewählt.

Eigenschaften des GSM-Modulationsverfahrens

Die Phase wird bei GMSK kontinuierlich verändert und das Sendespektrum ist entsprechend schmal, was zu einer besseren Trennung von Nachbarkanälen führt. Außerdem können einfache Verstärker mit hohem Wirkungsgrad eingesetzt werden. Dies verringert den Preis der Geräte und erhöht deren Standzeit beim Akku-Betrieb. Nachteilig ist die breite Impulsantwort beim GMSK; dadurch überlappen sich die einzelnen Impulse (Intersymbolinterferenz), die benachbarte Datenbits darstellen. Der Empfänger benötigt einen entsprechenden Entzerrer um die gesendete Information wiederzugewinnen.

Beschreibung des Verfahrens

Der mit einer Bitrate von ca. 271 kbit/s ankommenden Datenstrom wird zuerst differentiell codiert und in eine bipolare Datenfolge umgesetzt, die das Sendefilter durchläuft. Das so entstandene Signal wird als Phaseninformation des Sendesignals genutzt. Das Produkt aus Bandbreite und Bit-Dauer beträgt BT = 0,3; die maximale Phasenänderung beträgt $\pi/2$ pro Bitintervall.

Demodulation

Der Demodulator hat die Aufgabe, dem ankommenden, während der Funkübertragung gestörten Signal, das wahrscheinlichste Sendesignal zuzuordnen. Die dem Empfänger bekannte Trainingssequenz in den einzelnen übertragenen *Bursts* (siehe Abb. 6.4-2) erlaubt ihm eine Kanalschätzung und so einen teilweisen Ausgleich der Störungen. Der Standard gibt kein Demodulationsverfahren vor, fordert aber bestimmte Leistungsmerkmale. Unter anderem muss ein GSM-Empfänger in der Lage sein, unter der Bedingung einer Zweiwegeausbreitung zu arbeiten; Für den Test sollen beide empfangenen Signale gleiche Leistung und bis zu $16\,\mu s$ Zeitversatz (ca. 4 Bit-Dauern) haben. Dieses Signal muss korrekt demoduliert werden.

Störung von Nachbarzellen

Wegen der hohen Bitrate des ankommenden Datenstroms von ca. 271 kbit/s überlappen sich die benachbarten Spektren der im Abstand von 200 kHz angeordneten Kanäle deutlich (ca. $10\,dB$ Abschwächung im Abstand von 100 kHz von der Trägerfrequenz). Dabei muss man bedenken, dass es sich bei GSM um ein zellulares Netz handelt. Durch eine geeignete Frequenzplanung kann man direkt benachbarte Frequenzen in größerem räumlichen Abstand verwenden.

6.4.2 Die GSM-Luftschnittstelle

6.4.2.1 FDMA-Komponente und TDMA-Rahmenaufbau

Das GSM-System hat eine differenzierte Kanalstruktur, die sich im ersten Schritt aus der Unterteilung der genutzten Frequenzbänder in Teilbänder à 200 kHz Bandbreite ergibt (***F**requency **D**ivision **M**ultiple **A**ccess*-Anteil (**FDMA**)). Für *Uplink* und *Downlink* gibt es je ein eigenes Frequenzband. Die entsprechenden Zahlenwerte für GSM und GSM 1800 sind in Tabelle 6.4-1 angegeben. Diese Teilbänder sind mit **TDMA** (***T**ime **D**ivision **M**ultiple **A**ccess*) in je acht physikalische Kanäle unterteilt. Alle 4,62 ms wiederholt sich das gleiche Muster der acht Kanäle; die Zeitschlitzdauer beträgt 0,577 ms (siehe Abb. 6.4-1). Die Daten, die während einer Periode in einem Zeitschlitz übertragen werden, nennt man *Burst*. Aufgrund der physikalischen Kanalstruktur umfasst die kleinste Einheit acht Kanäle, um die die Kapazität einer Zelle sinnvoll erweitert werden kann.

FDMA- und TDMA-Anteil der Luftschnittstelle

Tab. 6.4-1: *Frequenzbänder von GSM und GSM 1800*

	Uplink	Downlink
GSM	890 - 915 MHz 25 MHz entspr. 124 Teilbändern	935 - 960 MHz 25 MHz entspr. 124 Teilbändern
GSM 1800	1710 - 1785 MHz 75 MHz entspr. 374 Teilbändern	1805 - 1880 MHz 75 MHz entspr. 374 Teilbändern

Abb. 6.4-1: FDMA- und TDMA-Komponenten in GSM

Auf diese physikalischen Kanäle werden logische Kanäle abgebildet, die Signalisierungsaufgaben und den Transport von Telefongesprächen und Nutzdaten übernehmen. Je nach Ausbaustufe und Auslegung sind unterschiedliche Unterteilungen in Signalisier- und Nutzkanäle vorgesehen, die weiter unten beschrieben werden.

logische und physikalische Kanäle

Zuerst werden der Aufbau der TDMA-Rahmen und die Rahmen-Hierarchie von GSM beschrieben.

Burst-Typen

Die acht Zeitschlitze eines TDMA-Rahmens lassen sich für unterschiedliche Zwecke nutzen. Dementsprechend gibt es in GSM fünf verschiedenen **Burst-Typen**, die in einem Zeitschlitz übertragen werden können. Diese werden in Abb. 6.4-2 gezeigt. Alle *Burst*-Typen haben die gleiche Länge von 156,25 bit bzw. 0,577 ms. Außerdem sind ihnen eine Schutzzeit am Ende und *Tail-Bits* am Anfang und Ende gemeinsam. Während der Schutzzeit werden keine Informationen übertragen. Ihre Länge von 8,25 Bit-Dauern bei allen *Bursts* außer dem Zugriffs-*Burst* ergibt sich daraus, dass alle für die Rahmenstruktur benötigten Zeiten aus einem Frequenznormal von 13 MHz abgeleitet werden. Wenn sich die einzelnen *Bursts*, z. B. von verschiedenen Mobilen Endgeräten, zeitlich leicht gegeneinander verschieben, führt dies wegen der Schutzzeit nicht zu Überschneidungen; der störungsfreie Betrieb ist gewährleistet. Bei den *Tail-Bits* handelt es sich um Null-Bits; in dieser Zeit wird die Sendeleistung hoch- bzw. heruntergefahren.

Abb. 6.4-2: Aufbau der Bursts in GSM

Normal Burst

Der **Normal Burst** dient der Übertragung von Nutzdaten und Signalisierinformationen. Er besitzt in der Mitte eine 26 bit lange Trainingssequenz, die der (Bit-) Synchronisation und der Schätzung der Kanalimpulsantwort dient. Mit den Ergebnissen wird der Entzerrer eingestellt; Intersymbolinterferenzen durch Mehrwegeausbreitung können so korrigiert werden. Dadurch, dass die Trainingssequenz in der Mitte des *Bursts* angeordnet ist, ist der mittlere Fehler der Kanalschätzung bei einem zeitlich schnell veränderlichen Kanal am geringsten. Es gibt acht verschiedene Trainingssequenzen mit geringer Kreuzkorrelation. Nahe beieinander liegende Zellen, die die gleichen Kanäle nutzen und sich eventuell stören, verwenden unterschiedliche Trainingssequenzen; die Basisstation hat so ein zusätzliches Unterscheidungskriterium für ankommende Signale. Die Basisstation signalisiert dem Endgerät bei der Kanalzuteilung, welche Trainingssequenz verwendet werden soll.

Stealing Flags

Die beiden an die Trainingssequenz angrenzenden Bits werden **Stealing Flags** genannt. Ein gesetztes *Stealing Flag* zeigt an, dass der angrenzende Block von Datenbits (eines Verkehrskanals) zur Signalisierung genutzt wird (***F**ast **A**ssociated **C**ontrol **Ch**annel* - **FACCH**, siehe Abschnitt 6.4.2.2); dies geschieht z. B. zur schnellen Signalisierung beim Handover. Die anderen Signalisierkanäle weisen eine zehn- bis zwanzigfach größere Zeitverzögerung auf. Wenn das *Stealing Flag* gesetzt ist, wird die eigentlich zu übertragende Information des Verkehrskanals verworfen, so dass es zu Datenverlusten kommt.

Effizienzabschätzung

Die beiden Blöcke mit je 57 Datenbits werden entweder einem Verkehrs- oder Signalisierkanal zugeordnet. Wenn man berücksichtigt, dass aufgrund der Fehlerschutz-Codierung nur ca. 50% der übertragenen Daten Nutzinformationen sind, ergibt sich ein Anteil von weniger als 57 zu 156,25 Bits oder ca. 36% Nutzinformationen. Dieses Verhältnis wird noch ungünstiger, da ein Teil der Bandbreite für Signalisierkanäle genutzt wird.

Frequency Correction Burst

Der **Frequency Correction Burst** besteht neben Schutzzeit und *Tail-Bits* aus einer 142 bit langen Null-Folge zur Synchronisation. Aufgrund des GMSK-Modulationsverfahrens entspricht diese Folge der Aussendung des um +1625/24 kHz frequenzverschobenen, unmodulierten Sinus-Trägers. Damit können sowohl die Trägerfrequenzen der Endgeräte auf die Basisstation abgestimmt werden als auch die Rahmensynchronisation durchgeführt werden.

Synchronization Burst

Der **Synchronization Burst** besteht aus einer 64 bit langen Trainingssequenz zur Zeitsynchronisation und 2x39 Bits codierter Information. Diese enthalten die Kennung der Basisstation und die speziell codierte, aktuelle TDMA-Rahmennummer, die das Mobile Endgerät z. B. zur Verschlüsselung benötigt.

Dummy Bursts

Dummy Bursts haben bis auf die *Stealing Flags* den gleichen Aufbau wie *Normal Bursts*. Sie werden im gleichen Frequenzkanal ausgesandt, in dem sich der Broadcast-Kanal (siehe Abschnitt 6.4.2.2) befindet – falls keine anderen *Bursts* vorhanden sind. So werden auf diesem Träger ständig Signale gesendet. Mobile Endgeräte können diesen Träger bei verschiedenen Basisstationen gut und schnell finden; durch Leistungsmessung können sie feststellen, welche Basisstationen z. Z. am besten zu empfangen sind. Diese Information ist z. B. für das *Handover* wichtig.

Access Burst

Wenn ein Mobiles Endgerät mit einer Basisstation in Kontakt treten will, dann sendet es einen **Access Burst**. Zu diesem Zeitpunkt weiß das Endgerät noch nicht, welche Entfernung es zur Basisstation hat. Daher besitzt der *Access Burst* eine besonders lange Schutzzeit von 68,25 Bits, so dass eine Laufzeit des *Bursts* von bis zu 0,252 ms zulässig ist, ohne nachfolgende *Bursts* zu stören. Wenn man die Zeit abzieht, die benötigt wird um die Leistung des Senders herunterzufahren, berechnet sich aus der Lichtgeschwindigkeit und der Schutzzeit eine maximale Entfernung zur Basisstation – entsprechend dem Zellradius – von ca. 35 km.

6.4.2.1.1 Adaptive Rahmenausrichtung

Laufzeitproblematik

Bei einem Mobilfunksystem können sich die einzelnen Teilnehmer an beliebigen Orten aufhalten und demnach unterschiedliche Entfernungen zur Basisstation haben. Dadurch differiert die Laufzeit der Signale zwischen den Endgeräten und der Basisstation. Die Laufzeit beträgt ca. $3,3\,\mu s/km$ und liegt damit in der Größenordnung der Bit-Dauer von ca. $3,69\,\mu s$. Wenn keine Gegenmaßnahmen getroffen werden, führt dies dazu, dass sich die *Bursts* stark überlappen, die von Mobilstationen mit unterschiedlicher Entfernung zur Basisstation gesendet werden. Wenn die Schutzzeit von ca. $30\,\mu s$ zwischen den *Bursts* nicht ausreicht, kommt es zu Störungen.

Beispiel 6.4-1:
Der Zeitschlitz 1 ist einer entfernten Mobilstation zugeordnet und der Zeitschlitz 2 einer nahen. Die beiden Mobilstationen empfangen die *Bursts* der Basisstation mit einer entfernungsabhängigen Verzögerung und antworten jeweils im dritten darauf folgenden Zeitschlitz. Auf dem Rückweg ergibt sich nochmals die gleiche Verzögerung, wodurch sich die *Bursts* der Zeitschlitze 1 und 2 an der Basisstation überlappen und nicht mehr sicher ausgewertet werden können. Siehe Abb. 6.4-3.

Abb. 6.4-3: Kollision bei einem System ohne Timing Advance

Adaptive Frame Alignment

Diesem Problem wirkt GSM entgegen, indem die Mobilstationen ihre *Bursts* früher senden, als es ihrem eigenen Zeitschlitzschema entspricht. Das Verfahren wird als adaptive Rahmenausrichtung (**Adaptive Frame Alignment**) bezeichnet und in Abb. 6.4-4 gezeigt.

Abb. 6.4-4: Laufzeitausgleich mit Timing Advance

Die Basisstation übermittelt jeder Mobilstation über den ***S**low **A**ssociated **C**ontrol **Ch**annel* (**SACCH**, siehe Abschnitt 6.4.2.2) einen Code, der angibt, um wieviele Bitdauern der Antwort-Burst der Mobilstation vorverlegt werden soll (*Timing Advance*); der maximale Wert liegt bei 63. Der maximale Laufzeitausgleich beträgt daher 63/2 Bitperioden. Das entspricht ca. $116\,\mu s$ – was ausreicht, um eine Entfernungsdifferenz von ca. 35 km auszugleichen. Eine GSM-Zelle darf daher einen Radius von maximal 35 km aufweisen. Dieser Wert entspricht dem, den man aus der Länge der Schutzperiode des *Access Bursts* berechnet.

Timing Advance

Das Mobile Endgerät schickt der Basisstation den aktuell eingestellten Wert des **Timing Advance** ebenfalls auf dem SACCH. Die Basisstation kann aus der gemessenen Zeitverzögerung und dem übermittelten Wert den nächsten Wert für das Endgerät berechnen. Nebenbei gewinnt die Basisstation die Entfernung der Mobilstation, die als zusätzliches *Handover*-Kriterium genutzt werden kann.

Verfahren beim Verbindungsaufbau

Dieses Verfahren ist nur dann stabil, wenn bereits ein Kanal vorhanden ist, auf dem die Basisstation die aktuelle Verzögerung messen und daraus die Korrekturwerte berechnen kann. Beim Verbindungsaufbau wird daher – wie oben erwähnt – der kurze *Access-Burst* geschickt. Aus der Verzögerung berechnet die Basisstation den nötigen *Timing Advance*-Wert, und das Endgerät kann dann beim nächsten Senden einen *Normal Burst* mit kurzer Schutzzeit verwenden.

Zeitversatz zwischen Senden und Empfangen

Abb. 6.4-4 zeigt auch, dass das Endgerät im Uplink ca. 3 Zeitschlitze später sendet als es im *Downlink* empfängt. Durch den zeitlichen Versatz stören sich Sender und Empfänger nicht, und eine (aufwendige) Duplexeinheit entfällt; das Endgerät kann einfacher und damit preiswerter hergestellt werden. Außerdem nutzt das Endgerät

die Zwischenzeiten, um die Feldstärke der Signale benachbarter Basisstationen zu messen (s.a. *Handover*, Abschnitt 6.5.2.2).

6.4.2.2 Logische Kanäle

Logische Kanäle

In GSM müssen über die Luftschnittstelle neben dem Nutzverkehr eine Vielzahl von Kontroll- und System-Informationen übertragen werden, z. B. zur Synchronisation und Kanalzuteilung oder zur Verbreitung von Systemeinstellungen. Für den Nutzverkehr – seien es Telefongespräche, FAX oder Daten – stehen verschiedene Verkehrskanäle zur Verfügung, die jeweils einen oder einen halben physikalischen Kanal exklusiv nutzen. Die Kontroll- und System-Informationen werden über eine größere Anzahl von Signalisierkanälen übertragen; verschiedene, vorgegebene Kombinationen der Signalisierkanäle teilen sich einen physikalischen Kanal.

Abb. 6.4-5 zeigt eine funktionale Gliederung der logischen Kanäle in GSM. Betrachtet man die Abbildung der logischen Kanäle auf die physikalischen Kanäle, dann ergeben sich andere Gruppen.

Verkehrskanäle (traffic channel, TCH)		**Signalisierungskanäle**			
Sprache Bidirek-tional	Daten Bidirek-tional	Broadcast Channel Downlink	Common Control Channel (CCCH) Down- oder Uplink	Dedicated Controll Channel (DCCH) Bidirektional	
				Stand-alone Dedicated Control Ch. (SDCCH)	Associated Control Channel (ACCH)
Vollrate Halbrate	Vollrate 9,6 kbit/s 4,8 kbit/s 2,4 kbit/s Halbrate 4,8 kbit/s 2,4 kbit/s	Broadcast Controll Channel (BCCH) Frequency Correction Channel (FCCH) Synchronisat. Channel (SCH)	Randam Access Channel (RACH) Access Grant Cannel (AGCH) Paging Channel (PCH)	Stand-alone Dedicated Control Ch. (SDCCH)	Slow Associated Control Channel (SACCH) Fast Associated Control Channel (FACCH)

Abb. 6.4-5: Gliederung der logischen Kanäle in GSM

Systematik

Die Verkehrskanäle werden sowohl zur Sprach- als auch zur Datenübertragung genutzt. Es gibt Vollraten-Sprachkanäle mit 22,8 kbit/s und Halbratenkanäle mit 11,4 kbit/s. Die Datenkanäle zeigen die gleiche Unterteilung, sind aber noch weiter untergliedert; einige Fälle sind in der Tabelle angegeben. Die Signalisierkanäle bilden drei Gruppen, den *Broadcast Channel*, der nur im *Downlink* arbeitet, den *Common Control Channel*, der dem Zugriffsmanagement dient und den *Dedicated Control Channel* zur bidirektionalen Signalisierung zwischen der Basisstation und einem spezifischen Endgerät.

Im Folgenden werden die Signalisierkanäle einzeln beschrieben. Die Zahl der *Bursts*, die der jeweilige Kanal in einem Mehrfachrahmen (s. u.) belegt, ist ein Maß für die Übertragungskapazität. Ein Mehrfachrahmen besteht aus 51 TDMA-Rahmen.

Traffic Channel (TCH)

Traffic Channel

Verkehrskanäle übertragen die vom Teilnehmer generierten Datenströme (Sprache, Daten, Fax) und keine Signalisierung. Bei Leitungsvermittlung stellt der Verkehrskanal eine transparente Verbindung zur Verfügung; bei der Paketermittlung werden z. B. Pakete entsprechend X.25 transportiert. Vollratenkanäle nutzen einen kompletten physikalischen Kanal; zwei Halbratenkanäle teilen sich einen physikalischen Kanal. Für Halbratenkanäle wird die zu übertragende Sprache stärker komprimiert.

Broadcast Channel

- **Broadcast Control Channel (BCCH)**

  Broadcast Control Channel

  Auf dem BCCH verbreitet die Basisstation Netzinformationen, wie z. B. die *Location Area* Kennung (LAI), Parameter zur Zellauswahl, Parameter der Kanalkonfiguration, Liste der benachbarten Zellen, RACH Kontrollparameter und andere in der aktuellen Zelle gewählte Parameter. Der BCCH belegt die Bursts Nr. 3, 4, 5 und 6. Genauso wie beim Synchronisierkanal und dem Frequenzkorrekturkanal liegen diese Bursts immer an der gleichen, vordefinierten Stelle im Zeitschlitz Null der ersten Trägerfrequenz der Zelle. Diese Trägerfrequenz ist weder leistungsgeregelt noch nimmt sie am optionalen Frequenzspringen teil, damit sie vom Mobilen Endgerät gefunden und ausgewertet werden kann. Danach hat das Mobile Endgerät alle Informationen, um sich auf die aktuelle Zelle mit ihren Kanalkonfigurationen und Parametern einzustellen.

- **Frequency Correction Channel (FCCH)**

  Frequency Correction Channel

  Der FCCH dient der Frequenzkorrektur und Synchronisation. Er tritt immer zusammen mit dem SCH auf und belegt 5 *Bursts* in einem Mehrfachrahmen (Nr. 1, 11, 21, 31, 41).

- **Synchronisation Channel (SCH)**

  Synchronisation Channel

  Der SCH dient der Synchronisation. Er belegt die 5 auf den FCCH folgenden *Bursts* in einem Mehrfachrahmen (Nr. 2, 12, 22, 32, 42).

Common Control Channel (CCCH)

- **Random Access Channel (RACH)**

  Random Access Channel

  Über den RACH nimmt das Endgerät Kontakt mit dem Netz auf. Dazu wird ein *slotted ALOHA* Protokoll genutzt; ein erfolgloser Zugriff wird in einem zufällig gewählten Abstand von 3 bis 50 Bursts 1, 2, 4, oder 7 Mal wiederholt. Dieser Parameter wird über den BCCH verbreitet.

- **Access Grant Channel (AGCH)**

  Access Grant Channel

  Über den *Access Grant Channel* werden die Nachrichten gesendet, mit

denen einer Mobilstation ein SDCCH zugeordnet wird. Die Zahl der *Bursts* für den AGCH hängt davon ab, wieviele *Bursts* für den *Paging*-Kanal genutzt werden.

- **Paging Channel (PCH)**

Paging Channel

Der *Paging*-Kanal kann in mehrere Unterkanäle unterteilt werden; so braucht das einzelne Endgerät nur den zugehörigen *Paging*-Unterkanal abhören und kann seinen Energieverbrauch minimieren. Es gibt zwei Varianten des PCH mit 4 bis 81 bzw. 2 bis 27 *Paging*-Unterkanälen. Die *Paging*-Nachrichten werden abhängig von der Zahl der Unterkanäle im Abstand von ca. 0,95 bis 4,25 Sekunden ausgesandt. Die *Paging*-Nachrichten enthalten jeweils (siehe Abschnitt 6.5.3.1) zwei beliebige Kennungen oder zwei ***T**emporary **M**obile **S**ubscriber **I**dentities* (**TMSI**) und eine ***I**nternational **M**obile **S**ubscriber **I**dentity* (**IMSI**) oder vier TMSIs gesuchter Mobilstationen.

Dedicated Control Channel DCCH

- **Stand alone Dedicated Control Channel (SDCCH)**

Stand alone Dedicated Control Channel

Der SDCCH ist ein eigenständiger Signalisierkanal zur Punkt-zu-Punkt-Signalisierung zwischen Basisstation und einer Mobilstation. Er belegt 4 *Bursts* pro Mehrfachrahmen und wird beim Verbindungsaufbau und beim *Location Update* genutzt (siehe Abschnitt 6.5.3.1).

- **Slow Associated Control Channel (SACCH)**

Slow Associated Control Channel

Der SACCH gehört immer zu einem TCH (→ SACCH/T) oder SDCCH (→ SACCH/C). Er überträgt u. a. Daten zur Sendeleistungsregelung und Messreports des Mobilen Endgeräts (siehe Abschnitt 6.5.3.1).

- **Fast Associated Control Channel (FACCH)**

Fast Associated Control Channel

Der FACCH ist Teil eines Verkehrskanals (TCH). Wenn das *Stealing Flag* eines *Bursts* gesetzt ist (siehe Abb. 6.4-2), dann werden die entsprechenden Nutzdaten durch Signalisierdaten ersetzt.

Cell Broadcast Channel (CBCH)

Cell Broadcast Channel

Der CBCH kann ca. alle zwei Sekunden 80 Bytes übertragen und wird für *Short Messages* genutzt. Er nutzt 4x4 Bursts in einer Folge von acht Mehrfachrahmen.

In Abb. 6.4-6 und Abb. 6.4-7 werden die wichtigsten Parameter der Signalisier- und Verkehrskanäle zusammengefasst.

Kanal	Verzögerung (ms)		Inter-leaving-Tiefe	Netto-Datenrate in kbit/s	Brutto-Datenrate in kbit/s
	Wieder-holung	Inter-leaving			
BCCH	235	14	4	0,782	1,932
PCH	235	14	4	n * 0,782	n * 1,932
AGCH	235	14	4	n * 0,782	n * 1,932
RACH	235	-	1	n * 0,034	n * 0,084
SDCH	235	14	4	0,782	1,932
SACCH/ TCH	480	360	4	0,382	0,95
SACCH/ SDCCH	471	14	4	0,391	0,968
FACCH/FR	20	38	8	9,2	22,8
FACCH/HR	40	74	8	4,6	11,4

Abb. 6.4-6: Übersicht über die Signalisierkanäle in GSM

Kanal	Verzögerung (ms)		Inter-leaving-Tiefe	Netto-Datenrate in kbit/s	Brutto-Datenrate in kbit/s
	Wieder-holung	Inter-leaving			
TCH	20	38	8	13	22,8
Full Rate Data 9,6 kbit/s	5	93	22(19)	12	22,8
Half Rate Data 4,8 kbit/s	10	185	22(19)	6	11,4
Full Rate Data 4,8 kbit/s	5	93	22(19)	6	22,8
Half Rate Data 2,4 kbit/s	20	185	22(19)	3,6	11,4
Full Rate Data 2,4 kbit/s	20	38	8	3,6	22,8

Abb. 6.4-7: Übersicht über die Verkehrskanäle in GSM

6.4.2.3 Abbildung Logischer auf Physikalische Kanäle

Mehrfachrahmen

Die im vorangegangen Abschnitt beschriebenen logischen Kanäle werden auf physikalische Kanäle abgebildet. Die Zuordnung findet auf der Ebene der sogenannten Mehrfachrahmen statt. Ein **Mehrfachrahmen** wird aus 26 bzw. 51 TDMA-Rahmen gebildet. Dies ist nur ein Teil einer hierarchischen Rahmenstruktur, die zusätzlich Super- und Hyper-Rahmen umfasst und weiter unten beschrieben wird.

Abb. 6.4-8: Mehrfachrahmen für Vollraten- (gerade Zeitschlitznummer) und Halbratenkanäle

Der 26er Mehrfachrahmen (Abb. 6.4-10, links) hat eine Länge von 120 ms und ist für Verkehrskanäle vorgesehen. Abb. 6.4-8 zeigt die Organisation des 26er Mehrfachrahmens für Vollratenkanäle (oben) und Halbratenkanäle (unten).

Voll- und Halbratenkanal (TCH/F und TCH/H)

Bei Abb. 6.4-8 ist zu beachten, dass hier nur ein Ausschnitt aus einem Mehrfachrahmen gezeigt wird, der alle Zeitschlitze mit der gleichen Nummer zusammenfasst, die einen physikalischen Kanal bilden. Die anderen 7x26 Zeitschlitze sind nicht dargestellt. Für einen Vollratenkanal sind 24 der 26 *Bursts* mit Verkehrsdaten besetzt, während ein *Burst* (Nr. 13) als SACCH dient. Der letzte *Burst* ist unbelegt; bei ungeraden Rahmennummern vertauschen SACCH und der unbelegte *Burst* ihre Plätze. Da ein Mehrfachrahmen 120 ms lang ist und 4 *Bursts* des SACCH zusammen codiert werden, ergibt sich so die in Abb. 6.4-6 genannte Verzögerung von 480 ms. Im Vergleich dazu enthalten Halbratenkanäle 2x12 Zeitschlitze Daten der Vehrkehrskanäle und die *Bursts* 13 und 26 die dazugehörigen SACCHs. In diesem Fall sind alle *Bursts* belegt.

Beispiel 6.4-2:
Ausgehend von dem Kanalschema für GSM (Abb. 6.4-1) kann man die Datenrate der Verkehrskanäle bestimmen. Aus einer Eingangsdatenrate von $271\,kbit/s$ berechnet sich bei acht Zeitschlitzen eine Datenrate von $33,9\,kbit/s$ je Zeitschlitz. In einem *Normal Burst* (Abb. 6.4-2) stehen $2 \cdot 57$ Bit der $156,25$ Bit langen Folge zur Nutzdatenübertragung zur Verfügung, was eine Datenrate von

$$33,9\,kbit/s \cdot \frac{114}{156,25} = 24,7\,kbit/s$$

ergibt. In den Mehrfachrahmen werden dann noch weitere zwei aus 26 Rahmen für den SACCH (bzw. einen Leerkanal) verwendet, so dass schließlich für den Vollratenkanal

$$24,7\,kbit/s \cdot \frac{24}{26} \;=\; 22,8\,kbit/s$$

und für den Halbratenkanal $11,4\,kbit/s$ zur Verfügung stehen. Der SACCH hat eine Datenrate von je $0,95\,kbit/s$. Es handelt sich bei dieser Rechnung um Bruttodatenraten; etwa die Hälfte dieser Daten sind noch Redundanz zur Fehlerkorrektur.

Abb. 6.4-9: Beispiel für die Kanalorganisation in einem 51er Mehrfachrahmen

51er Mehrfachrahmen

Der **51er Mehrfachrahmen** (Abb. 6.4-10, rechts) hat eine Länge von ca. 235,4 ms und multiplext verschiedene Signalisierkanäle auf einen physikalischen Kanal. Die entsprechenden Schemata sind komplizierter als bei Verkehrskanälen und berücksichtigen, dass im *Uplink* und *Downlink* unterschiedliche Signalisierkanäle genutzt werden; für manche Kombinationen werden sogar zwei aufeinanderfolgende Mehrfachrahmen kombiniert. Aus der jeweiligen Anordnung der *Bursts*, die zu den einzelnen Signalisierkanälen gehören, und dem verwendeten Codierungsschema berechnen sich die Datenraten und Verzögerungen, die in Abb. 6.4-6 aufgeführt sind. Abb. 6.4-9 zeigt ein Beispiel-Schema. Im *Downlink* werden im Zeitschlitz 0 FCCH, SCH, BCCH und CCCH zusammen übertragen; im *Uplink* ist der Zeitschlitz 0 für den RACH reserviert. Dieses Schema ist für Zellen geringer Kapazität vorgesehen.

Abb. 6.4-10: Rahmenstruktur in GSM

Rahmenhierarchie

Abb. 6.4-10 zeigt die gesamte **Rahmenhierarchie** in einem Diagramm. Die nächst höhere Ebene über den Mehrfachrahmen bilden die Super-Rahmen; ein Super-Rahmen besteht entweder aus 51 Mehrfach-Rahmen der kurzen Variante oder aus 26 Mehrfach-Rahmen der langen Variante; die beiden Varianten der Mehrfach-Rahmen führen so wieder zu gleich langen Super-Rahmen (1326 Rahmen = 6,12 s). 2048 Super-Rahmen bilden dann im letzten Schritt je einen Hyper-Rahmen, der dann 2.715.648 Rahmen umfasst und eine Dauer von ca. 3,5 h hat. Die einzelnen TDMA-Rahmen haben durchlaufende Nummern, die sich bei jedem Hyper-Rahmen wiederholen. Diese Rahmennummern sind ein Eingangsparameter bei der Verschlüsselung der Übertragung über die Luftschnittstelle.

6.4.2.3.1 Kanalkonfigurationen

Kanalkonfiguration

Die Basisstation bestimmt für jede Zelle, wie die logischen Kanäle auf die einzelnen physikalischen Kanäle abgebildet werden. Es gibt in GSM genau sieben erlaubte Kanalkombinationen, d. h. Schemata, wie logische Kanäle auf physikalischen Kanälen übertragen werden. Drei dieser Schemata wurden bereits vorgestellt: Nr. 1 und 2 in Abb. 6.4-8 und Nr. 4 in Abb. 6.4-9. Alle sieben Kombinationen sind nochmals in Abb. 6.4-11 zusammengefasst.

Nummer	Logische Kanäle	
	Downlink	Uplink
1	TCH/F,FACCH,SACCH	TCH/F,FACCH,SACCH
2	2xTCH/H, 2x FACCH, 2x SACCH	2xTCH/H, 2x FACCH, 2x SACCH
3	2xTCH/H, FACCH, SACCH	2xTCH/H, FACCH, SACCH
4	FCCH, SCH, BCCH, CCCH	RACH
5	FCCH, SCH, BCCH, CCCH, 4xSDCCH, 4x SACCH/C	RACH, 4xSDCCH, 4xSACCH/C
6	BCCH, CCCH	RACH
7	8xSDCCH, 8xSACCH	8xSDCCH, 8x SACCH

Abb. 6.4-11: Erlaubte Kombinationen logischer Kanäle

Im Folgenden sind drei übliche Kanalkonfigurationen angegeben, die eine effiziente Aufteilung der jeweiligen Bandbreite zwischen Verkehrs- und Signalisierkanälen in Gebieten unterschiedlicher Systemlast sicherstellen sollen.

Kanalkonfiguration für Zellen geringer Kapazität

Bei Zellen geringer Kapazität mit einem Transceiver werden im *Downlink* im Zeitschlitz 0 die Kanäle FCCH, SCH, BCCH, AGCH, PCH, 4xSDCCH und 4xSACCH übertragen. Im *Uplink* werden RACH, 4xSDCCH und 4xSACCH übertragen. Damit ist die von den Verkehrskanälen unabhängige Signalisierung abgedeckt; die restlichen sieben Zeitschlitze werden für Verkehrskanäle genutzt: 7x(TCH/F + FACH + SACCH).

Kanalkonfiguration für Zellen mittlerer Kapazität

Bei Zellen mittlerer Kapazität mit vier Transceivern werden im *Downlink* im Zeitschlitz 0 die Kanäle FCCH, SCH, BCCH AGCH, PCH und im *Uplink* der RACCH übertragen. Auf zwei weiteren Zeitschlitzen liegen in *Up-* und *Downlink* je 8x(SDCCH, SACCH). Damit ist die von den Verkehrskanälen unabhängige Signalisierung abgedeckt; die restlichen 29 Zeitschlitze werden für Verkehrskanäle genutzt: 29x(TCH/F + FACH + SACCH).

Kanalkonfiguration für Zellen hoher Kapazität

Bei Zellen mit hoher Kapazität mit 12 Transceivern werden im *Downlink* im Zeitschlitz 0 die Kanäle FCCH, SCH, BCCH, PCH und AGCH im *Uplink* der RACH übertragen. Weitere Signalisierkanäle (BCCH, PCH, AGCH im *Downlink* und RACH im *Uplink*) werden in den Zeitschlitzen 2, 4 und 6 übertragen. Fünf Zeitschlitze sind in *Up-* und *Downlink* mit je 8x(SDCCH, SACCH) belegt. Die verbleibenden 87 Zeitschlitze werden für Verkehrskanäle genutzt: 87x(TCH/F + FACH + SACCH).

6.4.2.4 Fehlerschutzmechanismen (Codierung, Verwürfelung und Frequenzspringen)

Codierung der Signalisierung

GSM benutzt zwei unterschiedliche Codierungsschemata zum Fehlerschutz bei der Signalisierung und bei der Sprachübertragung (Vollratenkanal). Abb. 6.4-12 zeigt das Codierungsschema für die Signalisierung. Alle übertragenen Informationen werden als gleichwertig betrachtet und dementsprechend gleich codiert. Jeweils 184 Bit werden mit einem *Fire-Code* codiert, der 40 Bit Redundanz zufügt. Es handelt sich um eine systematische Codierung, d. h. die Nutzinformation bleibt direkt lesbar, die Redundanz wird angehängt.

Außerdem werden noch 4 Null-Bits angehängt, so dass sich eine Folge von 228 bit Länge ergibt. Die Null-Bits sind nötig, damit am Blockende durch den Faltungscoder – der hier für die blockweise Codierung genutzt wird – keine Information verlorengeht. Der Faltungscoder hat eine Rate von 0,5 und eine Einflusslänge von 5, wodurch eine 456 bit lange Folge übertragen wird.

Abb. 6.4-12: Codierungsschema für die Signalisierung

Codierung der Sprache

Die Sprache wird in GSM komprimiert übertragen. Der Komprimierungsalgorithmus erzeugt Bits dreier Wertigkeitsklassen. Es handelt sich um 50 Bits hoher, 132 Bits mittlerer und 78 Bits geringer Empfindlichkeit. Bits hoher Empfindlichkeit sind besonders wichtig, da ein Fehler in einem solchen Bit zu vielen Folgefehlern führt; diese Bits werden daher in zwei Stufen geschützt (s. Abb. 6.4-13). Im ersten Schritt werden drei Parity-Bits angehängt. Im zweiten Schritt werden sie dann zusammen mit den Bits mittlerer Empfindlichkeit mit dem gleichen Faltungscode codiert, der auch bei der Signalisierung genutzt wird. Die 78 Bits geringer Empfindlichkeit werden ungeschützt übertragen.

Abb. 6.4-13: Codierungsschema für die Sprachübertragung (Vollratenkanal)

Bündelfehler

Bei der Übertragung über den Mobilfunkkanal können **Bündelfehler** auftreten, d. h.

ganze Folgen von Bits werden gestört, während dazwischen längere Zeit kaum Fehler auftreten. Dies ist ungünstig für die Fehlerkorrektureigenschaften der verwendeten Faltungscodierung: Tritt ein Bündelfehler auf, reicht die Codierung oft nicht aus, um diese zu korrigieren; im anderen Fall ist die Codierung überdimensioniert.

Interleaving

Man verwürfelt daher die übertragenen Informationen (**Interleaving**). Die Bits werden so umsortiert, dass ursprünglich benachbarte Bits bei der Übertragung über die Luftschnittstelle nicht mehr benachbart sind. Bündelfehler führen so zu Einzelfehlern an verschiedenen Stellen des Codewortes bzw. mehrerer Codewörter und lassen sich so besser korrigieren. Der Nachteil des Interleaving ist, dass erst alle der zeitlich verteilten Teile einer Nachricht gesammelt werden müssen, bis diese gelesen werden kann. Dadurch ergeben sich zum Teil erhebliche Verzögerungen.

Interleavingverfahren

Beim Blockinterleaving betrachtet man N Codewörter (à 456 Bit) als Zeilen einer Matrix, die man dann spaltenweise ausliest. Bis zu N benachbarte Fehler werden damit zu Einzelfehlern in je einem Codewort. Dieses Verfahren führt zu einer Verzögerung entsprechend der Übertragungszeit für die N Codewörter. GSM kombiniert Blockinterleaving und Bitinterleaving mit einer Interleavingtiefe von insgesamt bis zu 22(19); die maximale Verzögerungen durch Interleaving beträgt 360 ms.

Frequenzdiversität

Neben Bündelfehlern weist der Mobilfunkkanal frequenzselektive Störungen auf, z. B. durch *Fading* (Schwund). Dadurch haben die relativ schmalbandigen Kanäle unterschiedlicher Frequenzen auch unterschiedliche Qualität. Beim Frequenzspringen (***F**requency **H**opping* - **FH**) wechseln Basisstation und Endgerät in schneller Folge die verwendete Frequenz, so dass sich eine Mittelung der Übertragungseigenschaften der verschiedenen Kanäle ergibt (Frequenz-Diversität). Dies führt wie beim Interleaving zu einer gleichmäßigeren Verteilung von Fehlern – jetzt zwischen Kanälen – so dass die Fehlerkorrektur besser arbeiten kann und die mittlere Übertragungsqualität steigt.

Interferenz-Diversität

Neben der Frequenz-Diversität hat Frequenzspringen den Vorteil der Interferenz-Diversität. In zellularen Netzen werden Frequenzen in einem gewissen räumlichen Abstand wiederverwendet. Da man diesen Abstand aus Gründen der Frequenzökonomie möglichst klein wählen will, stören sich die Zellen mit gleicher Frequenz etwas (Interferenz). Durch Frequenzspringen mitteln sich auch diese Störungen; das kann zu einer Kapazitätssteigerung genutzt werden, indem man den Frequenzwiederholabstand verringert. Dies kann man gut mit DTX (*Discontinuous Transmission*) kombinieren; bei DTX wird der Träger während der Sprachpausen unterdrückt.

FH-Typen

In GSM ist langsames Frequenzspringen (***S**low **FH*** - **SFH**) vorgesehen; die Frequenz wechselt von Burst zu Burst, d. h. alle 4,62 ms. Der erste Träger, auf dem der Broadcast-Kanal liegt, ist vom Frequenzspringen ausgeschlossen. Beim zyklischen (*cyclic*) Frequenzspringen wechselt die Sprungfrequenz in einem vorgegebenen Muster, beim zufälligen (*random*) Frequenzspringen entsprechend einer Pseudo-Zufallsfolge. Wenn eine Zelle genug Transceiver hat, d. h. für viele Teilnehmer ausgelegt ist, kann man zwischen den Trägerfrequenzen springen. Bei Zellen mit weniger Transceivern übernimmt ein Synthesizer den Frequenzwechsel.

FH-Rahmenbedingungen

Frequenzspringen ist nur dann sinnvoll, wenn der Frequenzabstand der einzelnen Kanäle groß genug ist und ihre *Fadingprofile* unkorreliert sind. Außerdem kann Frequenzspringen nur eingeführt werden, wenn alle Mobilstationen in der jeweiligen Zelle dazu in der Lage sind; nicht frequenzsprungfähige Endgeräte können nicht betrieben werden.

Selbsttestaufgabe 6.4-1:

Wieviele Kanäle (Downlink und Uplink) stehen für GSM 900 und GSM 1800 zur Verfügung?

Selbsttestaufgabe 6.4-2:

Was versteht man unter einem Burst und wieviele Bursts passen in einem TDMA-Rahmen?

6.5 Protokolle

Bevor Protokolle für das *Call Management* und die Mobilität in GSM vorgestellt werden, gibt der folgende Abschnitt einen Überblick über die verschiedenen Kennungen in GSM, mit denen Teilnehmer, Endgeräte, *Location Areas* etc. bezeichnet werden.

6.5.1 Kennungen in GSM

Teilnehmerbezogene Kennungen

In GSM wurden eine Vielzahl von Adressen und Kennungen vereinbart, die in diesem Abschnitt vorgestellt werden. Benutzer bzw. deren ***S**mart **I**dentification **M**odule* (**SIM**) und Geräte werden unterschieden; sie haben jeweils eigene, international eindeutig zugeordnete Kennungen. Die SIM kann für beliebige Mobile Endgeräte benutzt und zwischen diesen übertragen werden. Ferner hat ein GSM Benutzer sowohl eine GSM-Kennung (IMSI) als auch eine ISDN-Rufnummer (MSISDN). Tabelle 6.5-1 listet die Teilnehmer-Kennungen in GSM auf. Sie enthält die Bezeichnung, die Abkürzung, den Aufbau und eine kurze Beschreibung der jeweiligen Kennung.

Tab. 6.5-1: *Übersicht über die in GSM vorgesehenen Teilnehmerkennungen*

Bezeichnung	Aufbau (Dezimalstellen)	Funktion
***I**nternational **M**obile **S**ubscriber **I**dentity* (**IMSI**)	3 Stellen ***M**obile **C**ountry **C**ode* (**MCC**) 2 Stellen ***M**obile **N**etwork **C**ode* (**MNC**) max. 10 Stellen ***M**obile **S**ubscriber **I**dentification **N**umber* (**MSIN**)	Eindeutige Teilnehmerkennung, die in der SIM gespeichert wird. Eine SIM mit IMSI ist zum Betrieb eines Endgeräts nötig. Die IMSI ist GSM-spezifisch und weicht von der ISDN-Numerierung ab. Beispiele: 262 für Deutschland (MCC), 01 für D1-Telecom, 02 für D2-Privat und 03 für E-Plus.
***M**obile **S**ubscriber **ISDN** Number* (**MSISDN**)	max. 3 Stellen ***C**ountry **C**ode* (**CC**) normalerweise 2-3 Stellen ***N**ational **D**estination **C**ode* (**NDC**) max. 10 Stellen ***S**ubscriber **N**umber* (**SN**)	Die MSISDN ist dem Teilnehmer bzw. der SIM zugeordnet; Teilnehmeridentität und Rufnummer sind getrennt. Aus der MSISDN ist kein Rückschluss auf die Teilnehmeridentität (IMSI) möglich, da die Zuordnung MSISDN - IMSI nur im Netz bekannt ist. Ein Teilnehmer kann mehrere MSISDNs haben.
***M**obile **S**tation **R**oaming **N**umber* (**MSRN**)	max. 3 Stellen *Country Code* normalerweise 2-3 Stellen *National Destination Code* max. 10 Stellen *Subscriber Number*	ISDN-Nummer, die dem Teilnehmer temporär zugeordnet wird. Zeigt auf das zugeordnete MSC; ankommende Rufe werden mit der MSRN bis zum MSC geführt. Wird vom VLR entweder beim *Location Update* oder beim Verbindungsaufbau dem Endgerät zugeordnet.
***T**emporary **M**obile **S**ubscriber **I**dentity* (**TMSI**)	Maximal 4 Byte	Vom VLR vergebenes, temporäres Pseudonym für Teilnehmer. Erlaubt beim Abhören auf der Luftschnittstelle keinen Rückschluss auf die Teilnehmeridentität. Für das Netz identifiziert die TMSI zusammen mit einem LAI einen Teilnehmer eindeutig.

Fortsetzung auf der naechsten Seite

Fortsetzung von der vorhergehenden Seite		
Bezeichnung	Aufbau (Dezimalstellen)	Funktion
***L**ocal **M**obile **S**ubscriber **I**dentity* (**LMSI**)	4 Byte	Kurzer, lokaler und optionaler Suchschlüssel im VLR, um die Suche nach Teilnehmerdaten zu beschleunigen.

Nichtteilnehmerbezogene Kennungen

Tabelle 6.5-2 enthält die nicht-teilnehmerbezogenen Kennungen in GSM. Der Aufbau ist analog zu Tabelle 6.5-1. Weitere, nicht aufgeführte Systemkomponenten wie z. B. MSC, HLR, VLR haben eine ISDN-Nummer, mit der sie adressiert werden können. Außerdem können sie *Signalling Point Codes* besitzen; Damit ist eine eindeutige Adressierung im Signalisiersystem (SS7) möglich.

Tab. 6.5-2: *Übersicht über die nicht-teilnehmerbezogenen Kennungen in GSM*

Bezeichnung	Aufbau (Dezimalstellen)	Funktion
***I**nternational **M**obile **S**tation **E**quipment **I**dentity* (**IMEI**)	6 Stellen ***T**ype **A**pproval **C**ode* (**TAC**) 2 Stellen ***F**inal **A**ssembly **C**ode* (**FAC**) 6 Stellen Seriennummer 1 Stelle frei	International eindeutige Gerätekennung, vom Hersteller vergeben und im EIR des jeweiligen Betreibers gespeichert (*white list*). Dient der Identifizierung gestohlener (*black list*) und veralteter bzw. fehlerhafter Geräte (*grey list*).
***L**ocation **A**rea **I**dentifier* (**LAI**)	3 Stellen *Country Code* 2 Stellen *Mobile Network Code* max. 5 Stellen oder 16 bit hex ***L**ocation **A**rea **C**ode* (**LAC**)	Bezeichnet international eindeutig eine *Location Area*. Jeder Zelle wird ein LAI zugeordnet und auf dem Broadcast-Kanal ausgestrahlt. Wird bei ankommenden Rufen vom VLR an das MSC weitergegeben. Der Teilnehmer wird in den Zellen mit dem passenden LAI gepaget.
***C**ell **I**dentifier* (**CI**)	Max. 2 Byte	Identifiziert eine Zelle innerhalb einer *Location Area*.
***B**ase Transceiver **S**tation **I**dentity **C**ode* (**BSIC**)	3 Bit *Network Colour Code* 3 Bit *Base Station Colour Code*	Unterscheidet benachbarte Basisstationen und wird auf dem Broadcast-Kanal ausgestrahlt.

6.5.2 Call Management

6.5.2.1 Mobile Originated Call und Mobile Terminated Call

Abb. 6.5-1 zeigt einen Ausschnitt aus dem Informationsfluss eines vom mobilen Endgerät initiierten Anrufs (**Mobile Originated Call - MOC**). Dieser Protokollausschnitt soll als Beispiel für GSM-Protokolle auf der Luftschnittstelle ausführlicher dargestellt werden. Die senkrechten Linien stellen die beteiligten Netzkomponenten dar, die Nachrichten senden bzw. empfangen.

Mobile Originated Call

Es gibt keine ständige Verbindung zwischen der Mobilstation und dem Mobilfunknetz. Anrufversuche können aber jederzeit erfolgen, so dass man eine Prozedur zur Kontaktaufnahme zwischen dem mobilen Endgerät und der Basisstation definieren muss. Es gibt einen speziellen Signalisierkanal, den ***R**andom **A**ccess **Ch**annel* (**RACH**, siehe Abschnitt 6.4.2.2), auf dem sich das Endgerät zu einem zufälligen Zeitpunkt mit einer *Channel Request* Nachricht meldet. Wenn mehrere Mobilstationen gleichzeitig eine Nachricht auf dem RACH senden, kommt es zu Konflikten; ein neuer Versuch ist nötig. Im erfolgreichen Fall sendet nun die Basisstation auf dem *Access Grant Channel* eine *Immediate Assignment* Nachricht, in der die Nummer eines *Stand Alone Dedicated Control Channels* übermittelt wird, über den nun die weitere Signalisierung abläuft. Diese beiden Schritte sind bei allen Kontaktaufnahmen der Mobilstation mit dem Netz als erste nötig.

RACH-Zugriff

Abb. 6.5-1: Ausschnitt aus dem Informationsfluss für vom Endgerät initiierte Anrufe

Weiterer Ablauf des Mobile Originated Call

Anschließend wird der *Service Request* an die Basisstation geschickt und weitergeleitet. Vom *Visitor Location Register* wird nun eine Authentisierung ausgelöst, die mit einem *Challenge-Response*-Protokoll überprüft, ob der Nutzer über eine gültige SIM verfügt. Auch dieser Abschnitt wird bei allen wesentlichen Transaktionen des Endgerätes mit dem Netz durchlaufen. Im weiteren Verlauf wird die Chiffrierung eingeschaltet und der Mobilstation eine neue TMSI zugewiesen. Anschließend wird der eigentliche *Call Set-Up* eingeleitet. Die Basisstation wird vom MSC veranlasst, dem Endgerät einen Verkehrskanal zuzuweisen (*Assignment Command* und *Assignment Request*). Der bisher genutzte Signalisierkanal wird freigegeben und nun ein Teil des Verkehrskanals – ein sogenannter *Fast Associated Control Channel* – zur weiteren Signalisierung zwischen Endgerät und Basisstation genutzt.

Der weitere Ablauf entspricht dem üblichen Verbindungsaufbau im Festnetz. Es wird ersichtlich, dass die Signalisierung im GSM-System durch die Mobilität der Teilnehmer deutlich aufwendiger ist als im Festnetz.

Set-Up-Typen

Die hier beschriebene Variante des Gesprächsaufbaus wird ***Non Off Air Call Set-***

Up (**Non-OACSU**) genannt. Dabei wird ein Verkehrskanal geschaltet, nachdem alle für den Verbindungsaufbau nötigen Informationen über den Signalisierkanal ausgetauscht wurden, aber bevor geklärt ist, ob der Angerufene antwortet. Dadurch werden zu einem gewissen Anteil Ressourcen verschwendet. Beim ***Off Air Call Set-Up*** (**OACSU**) vermeidet man dies, indem man den Verkehrskanal erst dann durchschaltet, wenn der Angerufene den Anruf auf Signalisierungsebene entgegengenommen hat. Eine dritte Variante ist das ***Very Early Assignment*** (**VEA**) Verfahren, bei dem direkt nachdem sich die Mobilstation über den Random Access Channel gemeldet hat, ein Verkehrskanal zugewiesen wird. Die Signalisierung läuft wegen der höheren verfügbaren Bandbreite in diesem Fall schneller ab. Nachteilig ist – vor allem bei starker Last – die großzügige Belegung von Ressourcen.

Mobile Terminated Call

Bei ankommenden Anrufen (***Mobile Terminated Calls*** - **MTC**) muss der Teilnehmer als erstes lokalisiert werden. Dazu wird der Anruf von der Vermittlungsstelle des Anrufers bis zu einem *Gateway* MSC des GSM-Netzes weitergeleitet. Dort wird die *Location Area* des Teilnehmers bestimmt und die *Paging*-Prozedur eingeleitet (siehe Abschnitt 6.5.3.1). Wenn das Endgerät eine passende *Paging*-Nachricht empfängt, wird es sich mit einer channel-request Nachricht über den *Random Access Channel* an die Basisstation wenden. Die Basisstation weist über den *Access Grant Channel* einen Signalisierkanal (SDCCH) zu. Diese Schritte entsprechen bereits dem Ablauf bei einem vom Mobilen Endgerät initiierten Anruf. Der weitere Ablauf mit Authentisierung, Einschalten der Chiffrierung und der Zuweisung der neuen TMSI und des Verkehrskanals ist identisch. Danach wird dem Anrufer das Freizeichen gesendet, und der Angerufene wird über den Anruf benachrichtigt (»Klingeln«). Wenn der Angerufene antwortet, wird der Kanal auf der Luftschnittstelle durchgeschaltet und das Gespräch kann beginnen.

6.5.2.2 Handover

Handover

Als **Handover** bezeichnet man den Wechsel der Funkverbindung während eines bestehenden Gesprächs. Im Idealfall wird der *Handover* vom Teilnehmer nicht bemerkt (*seamless Handover*), d. h. es entstehen keine Störgeräusche bzw. es gehen bei einer Datenverbindung keine Daten verloren.

Intra-Zell-Handover

Beim einfachsten Handovertyp (HO 1 in Abb. 6.5-2) wird eine neue Funkverbindung zur gleichen Basisstation (BTS) aufgebaut; dies bezeichnet man als **Intra-Zell-Handover**. Ein Intra-Zell-Handover wird aus administrativen Gründen durchgeführt oder wenn der benutzte Kanal selektiv gestört wird bzw. selbst Störungen verursacht - z. B. bei einem Teilnehmer, der den gleichen Kanal in einer anderen Zelle nutzt. Wenn in der aktuellen Zelle noch freie Kanäle vorhanden sind, dann kann ein neuer Kanal gewählt und so die Störung beseitigt werden. Die Entscheidung und Durchführung von Intra-Zell-Handovern liegt bei dem betroffenen *Base Station Subsystem* (BTS 1a und BSC 1 in Abb. 6.5-2).

Abb. 6.5-2: Intra-Zell- und Inter-Zell-Handover

Inter-Zell-Handover

Ein weiterer Handovertyp ist das **Inter-Zell-Handover**. Dabei wird eine neue Funkverbindung zu einer neuen Basisstation aufgebaut (HO 2 und HO 3 in Abb. 6.5-2). Inter-Zell-Handover werden durchgeführt, wenn sich Mobilstationen aus dem Sende- bzw. Empfangsbereich einer Basisstation entfernen und in den Bereich einer anderen Basisstation bewegen. Die Funkverbindung zu der alten Basisstation wird schlechter während die anvisierte Verbindung zu einer neuen Basisstation besser ist. Daneben kann man Inter-Zell-Handover durchführen, um die Verkehrslast zwischen benachbarten Zellen auszugleichen. Wenn eine Zelle sehr stark belastet ist, dann kann man für Mobilstationen in den Randbereichen der Zelle ein Handover zu einer benachbarten Zelle einleiten.

Weitere Handover

Findet ein Inter-Zell-Handover zwischen Zellen statt, die zum gleichen BSC gehören, dann wird dieses Handover vom BSS selbständig durchgeführt. Dies ist gut möglich, da das *Radio Resource Management* im BSS angesiedelt ist. Das zugehörige MSC wird über den Vorgang informiert. Wenn die beiden Zellen zu verschiedenen MSCs gehören, dann werden diese MSCs an dem *Handover* beteiligt.

Ein Handover zwischen Zellen, die zu verschiedenen Netzen (PLMNs) gehören, ist nicht vorgesehen.

Auslösung des Handover

Die Entscheidung, ein *Handover* durchzuführen wird immer netzseitig getroffen. Die Mobilstation sammelt aber fortlaufend Daten über die Qualität der bestehenden Funkverbindung und über den Empfang der Broadcast-Kanäle alternativer Basisstationen. Die entsprechenden Messergebnisse werden der Basisstation periodisch mitgeteilt und dort zusammen mit eigenen Daten ausgewertet.

Abb. 6.5-3 beschreibt in vereinfachter Form den Nachrichtenfluss, der bei einem *Handover* zwischen Zellen auftritt, die zu verschiedenen MSCs gehören. Die in den einzelnen Nachrichten enthalten Parameter werden hier nicht berücksichtigt;

außerdem wird nur der erfolgreiche Fall behandelt. Die senkrechten Linien stellen die beteiligten Netzkomponenten dar, die Nachrichten senden bzw. empfangen.

Abb. 6.5-3: Vereinfachter Nachrichtenfluss beim Handover zwischen zwei MSCs

Ablauf des Handover

Solange eine Verkehrsverbindung besteht, sammelt das Endgerät ständig Daten über die Qualität dieser und der maximal sechs besten alternativen Verbindungen. Diese Daten werden als *Measurement Information* regelmäßig (mindestens einmal pro Sekunde) an die Basisstation geschickt. In der Basisstation werden diese und weitere Daten ausgewertet. Wenn ein *Handover* als notwendig erachtet wird, wird eine *Handover Required* Nachricht an das zugehörige MSC geschickt. Das alte MSC sendet dann an das MSC, das zu der ausgesuchten neuen Zelle gehört, *Perform Handover*; das neue MSC fordert von der neuen Basisstation mit *Handover Request* einen Verkehrskanal an. Die Bestätigung der neuen Basisstation wird zurückgereicht, und das *Handover* eingeleitet (*Handover Command*). Nach erfolgreichem *Handover* meldet das Endgerät über die neue Funkverbindung *Handover Complete*. Anschließend sammelt es neue Messdaten, die wieder regelmäßig an die Basisstation geschickt werden. Netzseitig wird die alte Basisstation über das erfolgreiche *Handover* informiert und gibt dann den nun nicht mehr belegten Kanal frei.

Selbsttestaufgabe 6.5-1:

Was versteht man unter einem Handover? Welche beiden Handovertypen gibt es?

6.5.3 Mobilität

6.5.3.1 Location Update und Paging

Location Update und Paging

Die Unterstützung der Teilnehmermobilität ist eine zentrale Aufgabe eines Mobilfunknetzes. Bei mobilen Teilnehmern, die gerade einen Dienst nutzen, muss ein störungsfreies *Handover* in eine neue Zelle sichergestellt sein. Aber auch bei nicht aktiven Teilnehmern laufen im Hintergrund Prozeduren ab, die die Mobilität unterstützen. Die beiden wichtigsten Funktionen sind das **Location Update und** das **Paging**. Beim *Location Update* meldet das Endgerät bei einem Wechsel der *Location Area* die Kennung der neuen *Location Area* an das Netz, so dass die Informationen über den Aufenthaltsort der Teilnehmer immer aktuell sind. Beim *Paging* wird ein Teilnehmer innerhalb einer *Location Area* – die viele Zellen umfasst (vgl. Abb. 6.3-2) – gesucht.

Beispiel:
Die Abbildung unten zeigt eine Übersicht über den Datentransfer beim *Location Update*.

Vor dem *Location Update* befindet sich das Endgerät in seiner alten *Location Area*, dessen LAI (*Location Area Identifier*) es gespeichert hat. Neben der festen Kennung IMSI besitzt es eine temporäre Kennung TMSI, die nur die halbe Länge der IMSI hat und nur innerhalb einer *Location Area* eindeutig ist. Wenn das Mobile Endgerät nun in einer Zelle, die zu einer anderen *Location Area* gehört, über den *Broadcast Channel* die neue LAI empfängt, leitet es ein *Location Update* ein.

Das Mobile Endgerät kontaktiert über einen *Random Access Channel* die Basisstation und es wird ein Signalisierkanal (SDCCH) etabliert, über den die weitere Signalisierung abläuft. Das Endgerät sendet die alte TMSI und LAI an die Basisstation die im allgemeinsten Fall zu einem neuen VLR gehört. TMSI und LAI gelangen über das MSC zu dem neuen VLR, welches dann vom alten VLR die IMSI und die *Authentication Parameter* anfordert (obere Alternative in Abb. 6.5-4). Kann dieser Schritt nicht ausgeführt werden, wird die IMSI direkt vom Endgerät angefordert und mit dieser das zugehörige HLR bestimmt, das dann die IMSI und die *Authentication Parameter* zum neuen VLR schickt (untere Alternative in Abb. 6.5-4). Das Endgerät kann jetzt authentifiziert werden. Die neuen Daten – LAI und MSRN – können ins HLR eingetragen werden. Die MSRN ist die Routing-Adresse des Teilnehmers mit der ankommende Rufe bis zum nächstgelegenen MSC geschaltet werden können. Anschließend wird die Chiffrierung eingeschaltet und über den so gesicherten Kanal dem Mobilen Endgerät eine neue TMSI zugewiesen. Wenn die neue *Location Area* zum gleichen VLR gehört, vereinfacht sich der Vorgang.

Nach dem *Location Update* hat das Mobile Endgerät eine neue TMSI und LAI; das alte VLR spielt keine Rolle mehr. Im neuen VLR sind TMSI, IMSI, Authentikations-Parameter und die MSRN abgespeichert; im HLR wurden LAI und MSRN aktualisiert.

Es gibt noch weitere Ereignisse, die zu einem *Location Update* führen:

Typen des Location Update

- Wenn das Endgerät eingeschaltet wird und feststellt, dass die aktuelle Zelle einen anderen LAI hat als vor dem Ausschalten. Dies stellt sicher, dass die Lokalisierungsinformation eingeschalteter Endgeräte immer aktuell ist.
- Wenn ein Timer abgelaufen ist, dessen Dauer (ca. 6 min bis 24 h) von der Basisstation verbreitet wird (periodisches *Location Update*). Dies stellt sicher, dass sich Endgeräte regelmäßig melden; bei Datenverlust oder -inkonsistenz im Netz wird so automatisch wieder ein regulärer Zustand erreicht.

Location Registration

Wenn das Mobile Endgerät in dem Netz, in dem ein Anruf erfolgen soll, z. Z. keine TMSI hat, erfolgt anstelle der *Location Update* die verwandte **Location-Registration**-Prozedur ab.

Paging

Die **Paging**-Prozedur wird beim Verbindungsaufbau benötigt. Im Rahmen des Verbindungsaufbaus werden im *Visitor Location Register* Teilnehmerdaten abgefragt. Dies veranlasst das VLR den *Paging*-Vorgang einzuleiten. Dafür sind die TMSI des Teilnehmers und die Kennung der aktuellen *Location Area* nötig. Diese Daten wer-

den an das MSC geschickt, das sie an alle BSS, die zur *Location Area* gehören, weiterleitet. In allen Zellen, die zu diesen BSS gehören, wird auf einem *Paging*-Kanal eine *Page*-Nachricht mit der TMSI des gesuchten Teilnehmers ausgesandt. Die TMSI ist kurz, so dass mit einer Nachricht maximal vier Teilnehmer gleichzeitig gesucht werden können. Wenn das Endgerät eine *Page*-Nachricht empfängt, die die eigene TMSI enthält, dann meldet es sich auf dem RACH und eine Signalisierverbindung kann aufgebaut werden.

Die *Page*-Nachricht kann in bestimmten zeitlichen Abständen mehrfach wiederholt werden; so werden auch Teilnehmer gefunden, die kurzzeitig eine schlechte Funkverbindung hatten. Wenn das Pagen mit der TMSI nicht erfolgreich war, kann auch noch mit der IMSI gepaget werden.

Trade-off zwischen Paging und Location Update

Paging und *Location Update* sind durch ihren beachtlichen Bedarf an Signalisierbandbreite gekoppelt. Es besteht ein *trade-off*, der von der Größe der *Location Areas* abhängt. Einerseits wird ein Teilnehmer um so seltener in eine andere *Location Area* wechseln, je größer die *Location Areas* sind, und die Signalisierung für *Location Update* ist gering. Andererseits müssen die Teilnehmer bei ankommenden Gesprächen in um so mehr Zellen gesucht werden. Dabei ist noch zu berücksichtigen, dass das Location Update wesentlich aufwendiger als das *Pagen* ist, dafür aber nur in einer Zelle auftritt, während das *Pagen* in vielen Zellen gleichzeitig geschieht.

6.6 Netz-Management

Bei einem GSM-Netz handelt es sich um ein sehr komplexes technisches System. Zum zuverlässigen Betrieb und zur Gewährleistung der Zufriedenheit der Kunden, ist ein umfangreiches Netz-Management nötig. Man fasst die Aufgaben des Netz-Management unter dem Kürzel **FCAPS** zusammen, das sich aus den Begriffen ***F**ault* (Störung), ***C**onfiguration* (Konfiguration), ***A**ccounting* (Abrechnung), ***P**erformance* (Performanz) und ***S**ecurity* (Sicherheit) ableitet.

FCAPS

Fehler im Netz

Aus Betreibersicht ist es wichtig technische Störungen, z. B. wenn ein *Transceiver* ausfällt, automatisch zu erkennen, damit diese schnell behoben werden können; den Kunden soll das Netz ständig zur Verfügung stehen. Fehler müssen erkannt, die Ursachen identifiziert und wenn möglich Gegenmaßnahmen eingeleitet werden.

Parametereinstellung

Für den Betrieb eines GSM-Netzes müssen eine Vielzahl von Parametern eingestellt werden. Dies reicht von der Verteilung der Frequenzen unter den Basisstationen (Frequenzplanung) bis hinunter zur Festlegung der Zahl der Wiederholversuche beim *Pagen*. Mit den Parametern wird die Netz-Konfiguration an die aktuellen Belange des Netzbetriebs angepasst. Es ergeben sich ständig Änderungen, die das Netz-Management den einzelnen Komponenten im Netz übermittelt.

Gesprächsdaten

Die Nutzung der Dienste, durch eigene Kunden wie durch Kunden anderer Betreiber, muss abgerechnet werden. Dies erfordert die Registrierung der Daten der Gespräche als auch Transaktionen mit anderen Betreibern. Es werden Datensätze und Konten für neue Kunden eingerichtet und für ehemalige Kunden entfernt.

Performanz

Die **Performanz** des Systems wird ständig überwacht. Dazu wird die Auslastung der einzelnen Systemkomponenten und Kanäle gemessen. Ferner werden Antwortzeiten, Fehlerraten, die Zahl der Abweisungen, etc. überwacht. So gewinnt man ein Bild der Systemauslastung. Bei hoher Last kann man Abwehrmaßnahmen ergreifen. Außerdem werden so Daten zur Planung des weiteren Systemausbaus gewonnen.

In den Bereich des Sicherheits-Managements fallen schließlich die Zugangskontrolle und die sichere Datenübertragung.

Netz-Management und TMN

Das GSM-Netz-Management wurde ungefähr zeitgleich mit dem ***T****elecommunication* ***M****anagement* ***N****etwork* (**TMN**) Konzept von ITU-T, ETSI und CEPT standardisiert. Das TMN besteht aus den – mit fallendem Abstraktionsgrad aufgelisteten – fünf Schichten ***B****usiness* ***M****anagement* ***L****ayer* (**BML**), ***S****ervice* ***M****anagement* ***L****ayer* (**SML**), *N*etwork ***M****anagement* ***L****ayer* (**NML**), ***E****lement* ***M****anagement* ***L****ayer* (**EML**) und *Network Element Layer*. Die oberen vier Schichten sind in sogenannten ***O****perating* ***S****ystems* (**OS**) realisiert. Dort findet das Netz-Management statt. Der *Network Element Layer* umfasst die ***N****etwork* ***E****lements* (**NE**), welche den einzelnen Komponenten des Kommunikationsnetzes zugeordnet werden. *Network Elements* und *Operating System* tauschen Informationen über ein Datennetz (***D****ata* ***C****ommunication* ***N****etwork* - **DCN**) aus. Netzelemente können das OS über Ereignisse benachrichtigen (*notification*) oder Anforderungen (*request*) des OS bearbeiten, d. h. die vom OS gewünschten Aktionen in der angesprochenen Netzkomponente auslösen.

Dieser Ansatz hat das Ziel, die Komponenten verschiedener Hersteller einheitlich in das Netzwerk-Management einzubinden. Jede Komponente bedarf daher einer Schnittstelle (Qx oder Q3) zum TMN. Die Realisierung der Protokollfunktionen in der einzelnen Komponente ist herstellerspezifisch und wird von diesem ausgeführt.

TMN im GSM

In GSM wurde nur ein Teil des TMN-Konzepts verwirklicht. Man beschränkte den Einsatz von Schichten im Wesentlichen auf das mobilfunkspezifische BSS. HLR und VLR sind Datenbanken und das MSC eine abgewandelte ISDN-Vermittlungsstelle; dort wurden Fehler- und Konfigurations-Management nicht spezifiziert.

6.7 GSM-Standard

Das Globale System für Mobile Kommunikation wurde in einem umfangreichen Werk (mehr als 5000 Seiten) standardisiert. Die entsprechenden Empfehlungen sind in 12 Hauptkapitel unterteilt, deren Inhalt in Tabelle 6.7-1 stichpunktartig benannt wird.

Tab. 6.7-1: *Übersicht über die Themenbereiche der GSM-Empfehlungen*

Standard	Inhalt
00 Series	Präambel
01 Series	Generelles; verwendete Terminologie; GSM-Diensteinführungsphasen
02 Series	Behandelt die Dienste, die den Kunden angeboten werden können; Vergebührung und internationale Verrechnung (technische Aspekte)
03 Series	Behandelt Netzaspekte wie z. B. Netzplanung, Sicherheitsaspekte, Routing
04 Series	Behandelt die Schnittstelle zwischen Mobilstation und Basisstation und die entsprechenden Protokolle
05 Series	Behandelt die physikalische Schicht der Funkübertragung; Codierung, Interleaving, Synchronisation, Multiplexen
06 Series	Behandelt die Sprachcodierung
07 Series	Behandelt Terminaladapter für die Mobilstation
08 Series	Behandelt die Schnittstelle zwischen Base Station System und Mobile Switching Center
09 Series	Verbindung von GSM-Netzen mit anderen Netzen (Interworking Functions, IWF)
10 Series	Service Interworking
11 Series	Spezifikation von Geräten, Zulassung von Geräten
12 Series	Operation and Maintenance Aspekte bei GSM-Netzen

7 Wireless Local Area Networks

7.1 Einleitung

***W**ireless **L**ocal **A**rea **N**etworks* (**WLAN**) sind eine Alternative zur klassischen LAN Verbindungen bei denen Daten durch Kabel oder Fasern übertragen werden. Im WLAN werden die Daten durch das Medium Luft übertragen, wobei die Datenrate typischerweise niedriger ist als bei der kabelgebundenen Übertragung. Typische Kabelverbindungen erreichen Werte im Gigabit-Bereich (Gigabit-Ethernet). Die aktuellen Standards im Wireless-Bereich sind:

WLAN

HiperLAN aus dem Projekt ***B**roadband **R**adio **A**ccess **N**etworks* (BRAN) des European Telecommunications Standards Institute (ETSI). Die erste Version HiperLAN/1 wurde 1996 fertiggestellt, der Nachfolger HiperLAN/2 (H/2) wurde im 2000 verabschiedet.

HiperLAN/2

IEEE 802.11 vom Institute of Electrical and Electronics Engineers. Der IEEE 802.11 Standard zur drahtlosen Datenübertragung wurde 1997 vom Institute of Electrical and Electronics Engineers (IEEE) verabschiedet. 802.11a ist eine Weiterentwicklung dieses Standards, wurde 1999 fertiggestellt und ermöglicht höhere Übertragungsraten sowie ***Q**uality **o**f **S**ervice* (**QoS**). 802.11 selbst baut auf anderen IEEE-Standards auf, wie zum Beispiel 802.2 **LLC** (***L**ogical **L**ink **C**ontrol*) und 802.3 CSMA/CA (Ethernet).

IEEE 802.11

Bluetooth wurde ursprünglich ab 1994 von Ericsson entwickelt und nutzt das 2,4 GHz ISM-Frequenzband[4]. 1998 wurde mit weiteren Industriepartnern, die an der Weiterentwicklung interessiert waren, die *Bluetooth Special Interest Group* gegründet. Besonderes Augenmerk wurde auf die Ad-Hoc Vernetzung von mobilen Geräten gelegt. Die tatsächliche Datenübertragung geschieht über definierte und auf spezielle Anwendungsbereiche beschränkte Profile, die auf die Bedürfnisse der (mobilen) Geräte zugeschnitten sind.

Bluetooth

Durch diese unterschiedlichen Standards reagierte die Industrie auf die verschiedenen Bedürfnisse einzelner Nutzer und Endgeräte[5]. Die Bedürfnisse richten sich im Wesentlichen nach dem Einsatzzweck:

- Nutzung des Internets mit PDAs, Notebooks etc.
- Telefonieren mit Headsets innerhalb eines **PAN**s (***P**ersonal **A**rea **N**etwork*)
- Drucken von PDAs und Notebooks
- Anbindung von PDAs und Notebooks an Handys.
- Variable Nutzung des Internets an verschiedenen Orten (Firma, Wohnung, Hotspots)

Personal Area Network

4 Es handelt sich um maximal 14 Kanäle je 22 MHz die lizenzfrei (ursprünglich für ISM ***I**ndustry, **S**cienceand**M**edicine*) genutzt werden können.

5 Je nach Namenskonvention werden diese Endgeräte bei der ETSI als Mobile Terminal (MT) oder Mobile Station (STA) bei der IEEE bezeichnet.

7.1.1 Komponenten

Ein WLAN besteht aus Access Points (AP) (Zugangspunkte) und Mobilen Terminals (MT) bzw. Mobile Stations. Im Folgenden werden die Begriffe jeweils im Kontext des betreffenden Standards verwendet. Beispiele für MTs sind Desktop Computer, Laptops, Palmtops und auch netzwerkfähige Drucker. Mehrere MTs/STAs können mit einem AP zur selben Zeit verbunden sein. Diese Anordnung wird in H/2 als Funkzelle (Radio Cell) und in 802.11 als ***B**asic **S**ervice **S**et* (**BSS**) bezeichnet. Die Begriffe werden synomym gebraucht. Kommuniziert ein AP mit einem anderen AP, so wird diese Anordnung bei H/2 mit ***R**adio **A**ccess **N**etwork* (**RAN**) bezeichnet, bei 802.11 mit ***E**xtended **S**ervice **S**et* (**ESS**). Das ***D**istribution **S**ystem* (**DS**) wird verwendet um verschiedene BSS und stationäre LANs zu einem ESS zu verbinden. Ein unabhängiges "Ad-Hoc"-Netz wird als Independent BSS (IBSS) bezeichnet.

Distribution System

Abb. 7.1-1: Komponenten eines WLANs

Die logische Verbindung des DS mit dem stationären LAN erfolgt über den Convergence Layer (H/2) bzw. das Portal (802.11).

In Folgenden werden häufig verwendete und grundlegende Übertragungsverfahren besprochen. Darauf folgt in Abschnitt 7.3 eine detaillierte Beschreibung des Standards IEEE 802.11.

7.2 Übertragungsprinzipien

7.2.1 Direct Sequence Spread Spectrum (DSSS)

DSSS

Chipping

Bei dem ***D**irect **S**equence **S**pread **S**pectrum*-Verfahren handelt es sich um ein Frequenzspreizverfahren. Dabei wird ein Ausgangssignal mit einer sogenannten Chipping-Sequenz gespreizt. Das Ergebnis dieser Spreizung wird durch eine XOR Berechnung bestimmt und ist in Abb. 7.2-1 dargestellt.

Ein Hauptvorteil der DSSS liegt in dem sehr guten Schutz gegen Abhören außerhalb der Zelle, da das Signal direkt an der Rauschgrenze liegt (Abbildung Abb. 7.2-1). Je länger die Chipping-Sequenz ist, desto mehr Bandbreite wird jedoch benötigt.

Abb. 7.2-1: Spreizen eines Signals

7.2.2 Frequency Hopping Spread Spectrum (FHSS)

FHSS

Bei diesem Verfahren handelt es sich auch um ein Frequenzspreizverfahren, bei dem ein schmalbandiges Signal in ein breitbandiges Signal umgewandelt wird. Die zuvor auf einen begrenzten Frequenzbereich konzentrierte Energie wird dabei auf ein größeres Intervall verteilt. Damit wird eine höhere Resistenz gegenüber schmalbandigen Störungen erreicht, die jedoch einen höheren Aufwand bei Sende- und Empfangsanlagen bedeutet.

Ein weiteres Merkmal ist das Kanalsprungverfahren. Dabei wird das verfügbare Frequenzband in unterschiedliche Kanäle aufgeteilt, die bei der Übertragung über eine dem Sender und Empfänger bekannte Folge von Pseudozufallszahlen gewechselt werden.

7.2.3 Frequency Division Muliplexing (FDM)

FDM Mehrere Signale mit geringer Bandbreite werden im ***F**requency **D**ivision **M**uliplexing* bei der Leitungsübertragung zu einem breitbandigen Signal gebündelt. Bei der Funkübertragung wird jedes Signal direkt auf seine Trägerfrequenz moduliert (s. Abb. 7.2-2).

Abb. 7.2-2: Frequency Division Muliplexing im Frequenzbereich

Um Überlagerungen zu vermeiden und auf Empfängerseite eine bessere Trennung der Signale zu ermöglichen, wird zwischen den jeweiligen Frequenzbändern eine Schutzbarriere eingefügt.

7.2.4 Orthogonal Frequency Division Multiplexing (OFDM)

OFDM Das ***O**rthogonal **F**requency **D**ivision **M**ultiplexing*-Verfahren ähnelt dem Frequency Division Muliplexing-Verfahren, verbessert dieses jedoch. Die durch den notwendigen Abstand der Kanäle entstehende ineffektive Ausnutzung der Bandbreite von FDM wird durch eine erlaubte Überschneidung der Kanäle behoben. Die Grundidee besteht darin, ein breitbandiges Signal in mehrere schmalbandige orthogonale Signale zu überführen. Die Orthogonalität zweier benachbarter Trägerfrequenzen ω_1 und ω_2 kann durch

$$\int_0^T e^{j\cdot 2\pi\cdot\omega_1 t}\cdot e^{-j\cdot 2\pi\cdot\omega_2 t}dt = \begin{cases} konst \text{ falls} & \omega_1 = \omega_2 \\ 0 & \text{für benachbarte } \omega \end{cases} \qquad 7.2\text{-}1$$

ausgedrückt werden, dabei muss der Abstand benachbarter Kanäle $d = \frac{1}{T} = 0,3125MHz$ betragen. Trotz der spektralen Überlappung benachbarter Signale können sie durch die jeweilige Orthogonalität unterschieden werden. Abb. 7.2-3 zeigt die Bandbreitenersparnis von OFDM im Vergleich zu FDM.

Abb. 7.2-3: OFDM (FDM)-Bandbreiten

Bei Überschneidung der Unterkanäle wird die Signalrate so gewählt, dass sie im Frequenzabstand zwischen zwei benachbarten Kanälen liegt und dadurch genau die Sendeleistung dieses Kanals ungleich Null ist. Die Sendeleistungen der anderen Kanäle heben sich gegenseitig genau auf. Dies ergibt die lineare Unabhängigkeit und damit die Orthogonalität des Systems.

7.2.5 Gaussian Frequency Shift Keying

Bei der digitalen Form der Frequenzmodulation (***F**requency **S**hift **K**eying* – **FSK**) wird die Frequenz einer sinusförmigen Schwingung entsprechend der Wertigkeit des zu übertragenen Symbols verändert. Bei binären Symbolen werden zwei, bei ternären Symbolen entsprechend drei verschiedene Frequenzen verwendet. **FSK**

Der Abstand zwischen den Frequenzwerten wird als **Frequenzhub** bezeichnet, das Verhältnis zwischen Frequenzhub und Bitrate wird als **Modulationsindex** bezeichnet.

Bei der Methode ***G**aussian **F**requency **S**hift **K**eying* (**GFSK**)wird zusätzlich zur normalen Frequenzmodulation das Signal mit einem Gauß-Filter versehen. Dadurch entstehen glattere Frequenzübergänge und die Bandbreite des Signals wird dadurch begrenzt. Damit werden ebenfalls Oberwellen reduziert, die in der Funkübertragung Störungen durch Übersprechen bewirken können. **GFSK**

7.2.6 Verwendung der Übertragungsverfahren

Folgende Tabelle zeigt eine Zusammenfassung der einzelnen Übertragungsverfahren und deren Verwendung in verschiedenen Standards.

Verfahren	Einsatz
Direct Sequence Spread Spectrum	802.11b
Frequency Hopping Spread Spectrum	Bluetooth
Frequency Division Muliplexing	GSM, DECT, Bluetooth
Orthogonal Frequency Division Multiplexing	802.11g, DVB-T, DAB
Gaussian Frequency Shift Keying	Bluetooth, DECT

7.3 IEEE 802.11

Der Standard 802.11 für drahtlose lokale Netze (WLAN) wurde 1997 von der IEEE veröffentlicht. Mit Hilfe von WLAN ist es möglich, Computer und andere mobile Geräte drahtlos über Funk untereinander und mit bestehenden lokalen Netzen zu verbinden. Wie das Ethernet basiert es ebenso auf den unteren beiden Schichten des OSI-Referenzmodells. In den weiteren Kapiteln wird die physikalische Schicht und die Zugriffskontrollschicht beschrieben.

7.3.1 Funkübertragung

Der Standard 802.11 ist durch mehrere Varianten konkretisiert und verbessert worden. Abb. 7.3-1 gibt einen Überblick über die in diesem Kapitel näher erläuterten Ausprägungen.

Abb. 7.3-1: IEEE 802.11 Ausprägungen

Die in Abschnitt 7.2 besprochenen unterschiedlichen Technologien ergeben Unterschiede in der Beschreibung der physikalischen Ebene. Der Zugriff zur nächsthöheren Ebene, der Zugriffskontrollebene ist jedoch wieder identisch. Wie alle Übergänge im ISO/OSI-Schichtenmodell ist der Übergang zwischen physikalischer Schciht und Datensicherungsschichtdurch einen Dienstzugangspunkt (*Service Access Point*) definiert (s. Abb. 7.3-2).

Dienstzugangspunkt

Abb. 7.3-2: Untere Schichten im ISO/OSI Modell

Die pyhsikalische Schicht teilt sich in zwei Subebenen mit unterschiedlichen Aufgaben:

Physical Layer Convergence Procedure (PLCP) Die Hauptaufgabe besteht in der Verwaltung der darunter liegenden PMD Subebene. Dazu werden beim Senden die von der höheren Schicht empfangenen Rahmen (*MAC Protocol Data Units*) mit den jeweiligen für den Sendevorgang benötigten Informationen erweitert.

PLCP Subebene

MPDU

Physical Medium Dependent (PMD) Diese Ebene ist für das Senden und Emfpangen der Datenpakete auf dem jeweiligen Medium der jeweiligen Technologie zuständig. Beispielsweise wird hier die Modulation und Demodulation durchgeführt.

PMD Subebene

7.3.1.1 Frequency Hopping Spread Spectrum (FHSS)

Die Funktionsweise des bereits in Abschnitt 7.2.2 erwähnten ***F**requency **H**opping **S**pread **S**pectrum* (FHSS) soll hier am Beispiel von IEEE 802.11 konkretisiert werden.

FHSS

Die Übertragung der Informationen geschieht durch Frequenzmodulation einer harmoschen Welle auf dem lizenzfreien $2,4$ GHz-Band. Das Frequenzband ist in 79 Kanäle zu je 1 MHz unterteilt auf dem die Sender eine festgelegte Zeit auf einem Kanal senden und danach auf einen anderen Kanal wechseln.

Die Reihenfolge der Kanäle zu denen gewechselt wird, ist durch die dem Sender und Empfänger bekannte Hopping-Sequenz festgelegt. Nutzen mehrere Sender zur selben Zeit den selben Kanal kann es zu Kollisionen kommen, die mit dem CDMA-Verfahren aufgelöst werden. Eine Erhöhung der Datenrate von 1 Mbit/s auf 2 Mbit/s ist möglich, jedoch muss dazu die Bandbreite der Kanäle erhöht werden. Beim Eintritt einer neuen Station in das Funknetz wird versucht ein Beacon-Frame zu empfangen, indem alle Kanäle nacheinander abgehört werden.

Hopping-Sequenz

Animation 7.3-3 zeigt das Frequenzsprungverfahren exemplarisch anhand von 3 Sendesystemen und einer zufälligen Hopping-Sequenz für jeden Sender. Die Kollisionen sind durch einen Blitz gekennzeichnet.

Animation 7.3-3: Frequency Hopping Spread Spectrum mit möglichen Kollisionen

Das Format eines PLCP-Frames ist in Abb. 7.3-4 dargestellt.

Abb. 7.3-4: FHSS-PLCP Frame Format

Feld	Bedeutung
SYNC	Das Feld dient zur Rahmenerkennung und besteht aus alternierenden Nullen und Einsen.
SFD	Markiert das Ende der Präambel mit dem Bitmuster `0000 1100 1011 1101`.
PLW	Das *MPDU Length Word* (12 bit) zeigt die Länge der MPDU in Byte an. Das Maximum beträgt 4095 Byte.
PSF	Das *PLCP Signaling Field* (4 bit) beschreibt die Datenrate der Übertragung. Präambel und Header werden immer mit 1 MBps übertragen.
CRC	Eine CRC-Fehlerprüfung des Headers.
MPDU	Enthält die der Zugriffsebene weitergereichten Nutzdaten.

7.3.1.2 Direct Sequence Spread Spectrum

Ebenso wie FHSS wird DSSS im lizenzfreien 2,4 GHz Frequenzband verwendet. Nach der Spreizung mit Chipping-Sequenz (oder auch ***P**seudo-**N**oise-Code* (**PN**) genannt) wird das breitbandige Signal zusätzlich so in der Intensität reduziert, dass es knapp unterhalb der Rauschgrenze liegt. Damit sind Störungen zu anderen Systemen minimiert und das Signal kann nur noch von einem Empfänger erkannt werden, der in Besitz des PN-Codes ist.

Pseudo-Noise-Code

Abb. 7.3-5: Spektrum bei DSSS

Das zur Verfügung stehende Frequenzband wird in 13 je 25 MHz breite Kanäle[6] unterteilt. Die Datenübertragungsrate beträgt bei diesem Verfahren 1, 2, 5,5 oder 11 Mbit/s, durch die niedrige Rauschgrenze können sich in einem festgelegten Radius jedoch mehr DSSS- als FHSS-Systeme befinden. Der Aufwand beim Eintritt in eine Funkzelle ist auch wesentlich geringer, da es weniger zu prüfende Kanäle gibt und ein System konstant auf einem Kanal sendet.

6 Die Regulierungsbehörden haben in Europa 13 und in den USA 11 Kanäle zugelassen.

Tab. 7.3-1: *Funkfrequenzkanäle IEEE 802.11*

Kanal	Trägerfrequenz [MHz]	Frequenzstreubereich [MHz]
1	2412	2399,5 - 2424,5
2	2417	2404,5 - 2429,5
3	2422	2409,5 -2434,5
4	2427	2414,5 - 2439,5
5	2432	2419,5 - 2444,5
6	2437	2424,5 - 2449,5
7	2442	2429,5 - 2454,5
8	2447	2434,5 - 2459,5
9	2452	2439,5 - 2464,5
10	2457	2444,5 - 2469,5
11	2462	2449,5 - 2474,5
12	2467	2454,5 - 2479,5
13	2472	2459,5 - 2484,5

Der PLCP-Rahmen für das DSSS-Verfahren ist in Abb. 7.3-6 dargestellt.

Abb. 7.3-6: DSSS-PLCP

Feld	Bedeutung
SYNC	Besteht aus abwechselnden Nullen und Einsen.
SFD	Kennzeichnet das Ende des SYNC-Feldes und markiert mit der Bitfolge `1111 0011 1010 0000` den eigentlichen Frame.
Signal	Gibt die Datenrate der Übertragung an. Präambel und Header werden immer mit 1 Mbps gesendet.
Service	Beinhaltet bei hohen Datenraten optionale Erweiterungen.
Lenght	Gibt die Länge der Übertragung in Mikrosekunden an.
CRC	Fehlerprüfung für den PLCP-Header
MPDU	Beinhaltet den übernommenen Frame aus der Zugriffskontrollebene und hat eine variable Länge von 14 bis 4095 Byte.

7.3.2 Nachrichtenformate

Das allgemeine Nachrichtenformat (Abb. 7.3-7) besteht aus drei Bereichen:

- Der **Header** kann eine Länge von 10 Byte bis maximal 30 Byte besitzen. Je nach Frame- bzw. Subtyp können die Adressen 2-4 sowie das Sequenzfeld entfallen.
- Der **Datenteil** kann eine Länge von 0 Byte bis maximal 2312 Byte haben.
- Jeder Rahmen schließt mit einer 4 Byte großen **Prüfsumme** ab.

Abb. 7.3-7: Physikalische Schicht 802.11

Die einzelnen Felder des Headers sind Tabelle 7.3-2 in näher beschrieben.

Tab. 7.3-2: *Nachrichtenformat 802.11*

Feld	Bedeutung
Kontrollfeld	Informationen über den Rahmen und die Netzkonfiguration (s. Abb. 7.3-8).
Dauer	Reserviert das Medium für die im Feld angegebene Zeit.
Adresse 1-4	Beinhaltet je 6 Byte große Adressen.
Sequenzkontrollfeld	Wird eine Nachricht fragmentiert, erfolgt in diesem Feld die Sequenznummerierung.
Datenteil	Enthält die Nutzdaten.
Prüfsumme	32 Bit CRC-Prüfsumme über die gesamte Nachricht.

Das 2 Byte große Kontrollfeld enthält Informationen zur Funkzelle und zum Frame selbst und ist in Abb. 7.3-8 aufgeschlüsselt.

<table>
<tr><td colspan="2">Version</td><td colspan="2">Typ</td></tr>
<tr><td colspan="4">Subtyp</td></tr>
<tr><td>To DS</td><td>From DS</td><td>More Fragments</td><td>Order</td></tr>
<tr><td>Power Management</td><td>More Data</td><td>WEP</td><td>Retry</td></tr>
</table>

Abb. 7.3-8: Informationen des Kontrollfeldes

Version Es wird die Version des IEEE 802.11 spezifiziert. Zur Zeit ist nur `00` erlaubt.

Typ Der Typ des Frames wird angegeben.

Bitfolge	Frame-Typ	Bedeutung
`00`	Management	Verwaltung von Zellen.
`01`	Kontroll	Mediumzugriffe und zuverlässige Übertragung.
`10`	Daten	Nutzlast aus höheren Schichten wird übertragen.
`11`	Reserviert	Reserviert für künftige Anwendungen.

To DS / From DS Mit diesen beiden Feldern kann die genaue Zuordnung zum Sender (Src) und Empfänger (Dst) der in den Adressfeldern eingetragenen Adressen durchgeführt werden.

To/From DS	Adresse 1	Adresse 2	Adresse 3	Adresse 4	Bedeutung
00	Dst	Src	Zelle	–	Management- oder Kontroll-Frames, Broadcast, sowie Datenframes in Ad-Hoc Netzen
01	Dst	Zelle (Dst)	Src	–	Datenframes, die über einen AP an eine Zelle gesendet werden
10	Zelle (Src)	Src	Dst	–	Datenframes, die von einer Station an einen AP gesendet werden.
11	Zelle (Dst)	Zelle (Src)	Dst	Src	Datenframes von AP zu AP über das Verteilungssystem

More Frag Dieses Bit wird auf 1 gesetzt, wenn diesem Frame noch weitere zur aktuellen Nachricht gehörigen Frames folgen.

Retry Wird ein Frame erneut gesendet, wird dieses Bit auf 1 gesetzt.

More Data / Power Management Diese Frames sind für die Verwaltung der Stromsparmechanismen zuständig.

WEP Durch dieses Bit wird angegeben, ob der Rahmen durch **WEP** (***W**ired **E**quivalent **P**rivacy*) verschlüsselt ist.

Wired Equivalent Privacy

Order Dieses Bit zeigt an, ob der Rahmen zu einer segmentierten Nachricht gehört.

Subtyp In diesem Feld werden zu jedem Frame-Typ die jeweiligen Subtypen angegeben.

- *Management Frames* werden nur innerhalb einer Zelle verschickt und stellen verschiedene Funktionen zum An- und Abmelden von Stationen an Access Points bereit.

Wert	Bezeichnung
0000	Association Request
0001	Association Response
0010	Reassociation Request
0011	Reassociation Response
1010	Deassociation
0100	Probe Request
0101	Probe Response
1000	Beacon
1001	Announcement Traffic Indication Message (ATIM)
1011	Authentifizierung
1100	De-Authentifizierung

Informationen über die Zelle (Beacon) werden gesendet, sowie Funktionen zum Auffinden neuer Stationen, Stromsparmechanismen und Authentifizierung bereitgestellt.

- Die *Kontroll-Frames* werden ebenfalls nur innerhalb einer Zelle versendet und sind unverschlüsselt. Sie werden als Bestätigungs-Rahmen im CSMA/CA (Abschnitt 7.3.3.1) oder signalisieren das Ende von Datenübertragungen im CFP-Modus (Abschnitt 7.3.3.2).

Wert	Bezeichnung
1101	Acknowlegement
1110	CF-End
1111	CF-End + CF-Ack
1010	Power Save (PS) Poll
1011	Request-to-Send (RTS)
1100	Clear-to-Send (CTS)

  Die Frames dienen als Bestätigungs-Frames im CSMA/CA (Abschnitt 7.3.3.1) oder signalisieren das Ende von Datenübertragungen im CFP-Zugriffsverfahren (Abschnitt 7.3.3.2).

- Der Subtyp für *Daten-Frames* signalisiert die Datenübertragung.

Wert	
0000	Daten
0001	Datan + CF-Ack
0010	Daten + CF-Poll
0011	Daten + CF-Ack + CF-Poll
0100	Null (keine Daten)
0101	CF-Ack (keine Daten)
0110	CF-Poll (keine Daten
0111	CF-Ack + CF-Poll (keine Daten)

  Neben einer einfachen Datenübertragung im CSMA/CA werden im CFP-Zugriffsverfahren spezielle Frames (*Poll-Frames*) genutzt, mit denen ein Access Point den einzelnen Stationen den Zugriff auf das Medium erlaubt.

7.3.3 Zugriffskontrollebene

CSMA/CA

Contention Free Period

Der IEEE 802.11 Standard sieht zwei Zugriffsverfahren vor, hierbei muss jede Station zwingend **CSMA/CA** (***C**arrier **S**ense **M**ultiple **A**ccess with **C**ollission **A**voidance*) implementieren, bei CFP (***C**ontention **F**ree **P**eriod*) handelt es sich um ein auf CSMA/CD aufbauendes optionales Verfahren.

7.3.3.1 Carrier Sense Multiple Access with Collission Avoidance

Sendewillige Stationen prüfen vor dem Medienzugriff, ob das Medium frei ist, dann greifen sie darauf zu. In diesem Modus der verteilten Koordinination (*Distributed Coordination Function*) wird diese Zeitphase als *Contention Period* bezeichnet. Anders als beim bekannten CSMA/CD-Verfahren im Ethernet ist es einer Station nicht möglich, während des Sendevorganges das Medium abzuhören und damit auftretende Kollisionen zu erkennen. Ein Entwurfsziel war damit die Vermeidung von Kollisionen (*Collision Avoidance*).

Distributed Coordination Function
Contention Period
Collision Avoidance

Grundlage für das CSMA/CA ist der ***N**etwork **A**llocation **V**ector* (**NAV**) den jede Station besitzt. In diesem Timer ist die Zeit hinterlegt, in der das Medium vermutlich belegt sein wird. Bei jedem übertragenen Frame wird der NAV von jeder Station mit dem im Frame übertragenen Wert aus dem Feld `Dauer` aktualisiert.

Network Allocation Vector

Da durch den NAV alleine Kollisionen nicht gänzlich ausgeschlossen sind, werden zusätzlich noch Bestätigungen vom jeweiligen Empfänger der Station gesendet. Ausnahme ist hier der Empfang von Frames mit einer Adressierung an Broadcast- oder Multicastadressen. Wartet die sendende Station vergeblich innerhalb der Backoff-Wartezeit auf ein Acknowlegement, so initiiert sie eine erneute Versendung des Frames.

Acknowlegement

Damit nach einem erfolgreichem Versand (und freiem Medium) nicht eine beliebige andere Station das Medium belegt, bevor die Empfangsstation ihre Bestätigung versendet hat, sind unterschiedliche Prioritäten für verschiedene Rahmen festgelegt. Abb. 7.3-9 zeigt verschiedene Intervalle (***I**nter**f**rame**s**pace* – (**IFS**)), die eine sendewillige Station warten muss, bevor sie senden darf.

Interframespacing

Abb. 7.3-9: Interframespace-Intervalle

Ist ein IFS abgelaufen, wartet jede Station zusätzlich noch eine Backoff-Zeit um dann erst zu überprüfen ob das Medium frei ist oder nicht. Die für jede Station unterschiedliche Backoff-Zeit wird verwendet damit nicht alle wartenden Stationen gleichzeitig nach dem Ende eines IFS einen Sendeversuch starten. Tabelle 7.3-3 zeigt eine Übersicht über die unterschiedlichen Intervallzeiten.

Backoff-Zeit

Tab. 7.3-3: *IFS-Intervalle*

Intervall	Anwendung	Dauer
Short **IFS**	Ack, CTS und Datenframes, die Fragmente derselben Nachricht transportieren.	Slottime
PCFS **IFS**	Wird beim Umschalten auf das CFP-Zugriffsverfahren benutzt.	SIFS + Slottime
DCF **IFS**	Für Daten- und Managementframes.	SIFS + $2 \cdot$ Slottime
Extended **IFS**	Sonderfall. Beim Abbruch eines Frames wird der Zugriff auf das Medium bereits nach Ablauf dieses Intervalls erlaubt.	SIFS + DIFS + ACK

Slottime

Die jeweils angegebene Slottime ist vom gewählten Übertragungsverfahren abhängig. Bei FSSS beträgt sie z. B. $50\mu s$ und bei DSSS $10\mu s$. Die Backoff-Zeit wird ähnlich wie beim CSMA/CD aus einer Zufallszahl und der Slottime berechnet. Das Maximum der Zufallszahl (als ***C**ontention **W**indow* – **CW**) erhöht sich nach jedem neuen Übertragungsversuch gemäß folgender Formel:

$$CW = 2^n - 1 \qquad 7.3\text{-}1$$

Nach dem sechsten Versuch erhöht sich der Wert nicht weiter und bleibt konstant.

Hidden-Station-Problem

Aufgrund der endlichen Signalausbreitung kann es vorkommen, dass eine sendewillige Station das Medium prüft, es als frei erkennt und anfängt zu senden, obwohl eine andere Station bereits den Sendeprozess begonnen hat. Dieses als *Hidden-Station-Problem* bekannte Phänomen ist in Abb. 7.3-10 dargestellt.

Abb. 7.3-10: Das Hidden-Station Problem

Request to send
Clear to send

Die Sendevorgänge von A und C führen unweigerlich zu einer Kollision, die von den bisher vorgestellten Mechanismen nicht verhindert werden kann. Zur Lösung werden die Kontroll-Frames **RTS** (***R**equest **t**o **s**end*) und **CTS** (***C**lear **t**o **s**end*) verwendet.

Eine sendewillige Station schickt als erstes einen *RTS*-Frame an den Empfänger. Im Feld `Dauer` ist die zur Übertragung benötigte Zeit (Übertragungszeit für den Frame + Ack + $3 \cdot$ IFS) hinterlegt. Der Empfänger quittiert die Sendeanfrage mit einem *CTS*-Frame. Durch diese beiden Frames können alle zur Zelle gehörigen Stationen ihren NAV-Wert neu einstellen.

Dieses Verfahren kann Kollisionen der *RTS*/*CTS*-Frames zwar nicht ausschließen, aber durch die geringe Länge (*RTS*: 20 Byte, *CTS*: 14 Byte) sinkt die Wahrscheinlichkeit einer Kollision stark. Da das Verfahren selbst das Medium zusätzlich zum Datenverkehr belegt, wird es in der Regel nur bei asymmetrisch auftretenden Reichweiten oder in stark belasteten Umgebungen eingesetzt. Bei Broadcast oder Multicast wird dieses Verfahren nicht verwendet.

Fragmentierung

Fragmentierte Nachrichten haben im Sequenzkontrollfeld das Feld `More Frag` gesetzt solange noch ein weiteres Fragment folgt. Jedes Fragment wird (außer bei Broad- und Multicast) mit einem `Ack` bestätigt, danach wird erst das nächste Segment gesendet. Das Feld `Dauer` wird so belegt, dass die NAV der anderen Stationen bis zum Ende des nachfolgenden Frame/Ack-Paares belegt sind. Die Kontrollmeldungen *RTS*/*CTS* werden demnach nur vor dem ersten Fragment ausgetauscht, die Reservierung des Mediums geschieht über die Fragmente selbst oder die zugehörigen Acknowlegment-Meldungen.

7.3.3.2 Contention Free Period

Contention Free Period

Das Verfahren ***C**ontention **F**ree **P**eriod* (**CFP**) stellt eine Weiterentwicklung mit zusätzlichen Vorteilen für zeitkritische Anwendungen von CSMA/CA dar, mit dem Kollisionen noch effektiver verhindert werden können. Problematisch ist der optionale Charakter von CFP. In einer Zelle können demnach Stationen vorkommen, die CFP unterstützen, aber auch Stationen, die lediglich CSMA/CA implementiert haben.

Point Coordinator

Im Gegensatz zur CSMA/CA wird der Zugriff auf das Medium von zentraler Stelle aus verwaltet. Der mit dieser Aufgabe betraute *Access Point* wird nun auch als *Point Coordinator* bezeichnet und berücksichtigt auch alle sich in der Zelle befindenden Stationen ohne Unterstützung von CFP. Wie in Abb. 7.3-11 dargestellt, wechseln sich CFP- und CP-Perioden innerhalb eines Intervalls ab.

Abb. 7.3-11: CFP/CP-Wechsel

Beacon-Frame

Zu Beginn eines jeden Intervalls sendet der AP einen Beacon-Frame als Broadcast.

Durch diese in Tabelle 7.3-4 zusammengefassten Informationen ist es den Stationen erst möglich, sich korrekt an dem Datenaustausch einer Zelle zu beteiligen.

Tab. 7.3-4: *Felder im Beancon-Frame*

Name	Größe	Bedeutung
Timestamp	8	Uhrzeit des Senders (für die Uhrensynchronisation)
Beacon-Intervall	2	
Capability Information	8	Angaben zur Zelle • Ad-Hoc oder Infrastruktur • CFP Unterstützung (ja/nein) • Verschlüsselung (ja/nein)
Zellen-Adresse	6	Adresse der Zelle
Übertragungsrate	3-8	Angabe über die von der physikalischen Ebene in der Zelle eingesetzten Übertragunsrate
FH-Parameter	7	Hopping-Sequenz (falls unterstützt)
DS-Parameter	3	Aktuelle Kanalnummer (falls ünterstützt)
CF-Parameter	8	Startzeit der nächsten CFP, maximalee Dauer der CFP, Restdauer der aktuellen CFP (falls unterstützt)
ATIM	4	Stromsparmechanismus in Ad-Hoc-Netzen
TIM	4	Stromsparmechanismus in Infrastruktur-Netzen

Ein wesentlicher Unterschied verglichen zum CSMA/CD-Verfahren ist die restriktive Sendeerlaubnis. Außer dem AP darf keine Station eine Sendung ohne Erlaubnis initiieren. Dies wird durch den NAV-Wert bestimmt, der jeder Station durch den CF-Parameter im Beacon Frame mitgeteilt wird.

Eine CFP-Station kann damit nur auf das Medium zugreifen, wenn sie einen CF-Poll vom Access Point erhalten hat. Dieser verwaltet eine Liste aller Stationen, die CFP unterstützen. Der AP sendet einen Poll immer nur an eine Station, diese kann vorhandene Daten senden oder lediglich den Empfang des Polls bestätigen. Daraufhin wird der nächsten Station der Poll vom AP mitgeteilt.

7.3.4 Netzaufbau

Die bereits in Abschnitt 7.1.1 beschriebenen Möglichekeiten ein Wireless Network aufzubauen werden nun detaillierter beschrieben. Es besteht die Möglichkeit eines Ad-Hoc-Netzes zwischen Endgeräten und einer Verbindung über einen Access Point.

Stationen

Ein Funknetz besteht aus Stationen, Zellen und Access Points. **Stationen** sind technische Systeme wie Notebooks oder Desktop-PCs, die Nachrichten empfangen und senden können. Eine auch als *Basic Service Set* bezeichntete **Zelle** ist eine räumliche Region, in der Stationen untereinander kommunizieren können. Ein **Access Point** (AP) ist eine für die Verwaltung der Zelle zuständige Station.

Access Point

7.3.4.1 Ad-Hoc-Modus

Diese Art des Netzes kann aufgebaut werden, um Endgeräte ohne Zugang zu einem Access Point miteinander zu verbinden. Damit verschiedene Endgeräte ein Ad-Hoc-Netz bilden und zu einem ***B**asic **S**ervice **S**et* (**BSS**) zusammengeschlossen werden, müssen verschiedene Bedingungen erfüllt sein:

Basic Service Set

- Alle Teilnehmer müssen die gleiche **SSID** (***S**ervice **S**et **I**dentification*) haben, oder es muss eine Universalkennung (ANY) eingetragen sein.
- Alle Teilnehmer müssen auf dem gleichen Kanal senden.
- Falls aktiviert, müssen alle Teilnehmer den gleichen WEP-Schlüssel (Sicherheitsschlüssen – ***W**ired **E**quivalent **P**rivacy*) besitzen.

Service Set Identification

7.3.4.2 Infrastruktur-Modus

In diesem Betriebsmodus kommunizieren alle Teilnehmer über einen Access Point. Die Teilnehmer können sowohl untereinander Daten austauschen als auch über den Access-Point auf andere LANs oder das Internet zugreifen. Dabei hängt es von der Konfiguration des Access-Points ab, ob das WLAN Teil eines drahtgebundenen Netzes ist, oder ob es ein eingenständiges Subnetz bildet.

Abb. 7.3-12: Infrastruktur-Netz

Jegliche Art der Kommunikation in einer Zelle wird über den jeweils zuständigen Access Point abgewickelt. Will beispielsweise Station B eine Nachricht an Station F schicken, wird die Nachricht zuerst über den AP1 and den AP2 geschickt und von da aus zum eigentlich Ziel F (s. Abb. 7.3-12).

Das benutzte Verteilungssystem ist nicht bestandteil von IEEE 802.11, beispielsweise kann für das Verteilungssystem Ethernet verwendet werden.

7.3.4.3 Dienste

Im Standard IEEE 802.11 sind Funktionen des Netzes abstrakt durch Dienste definiert, die aus der LLC-Schicht direkt aufgerufen werden können. Unterschieden wird dabei zwischen Stationsdiensten und Dienste des Verteilungssystems.

Stationsdienste

Stationsdienste sind sowohl im Ad-Hoc- als auch im Infrastruktur-Modus verfügbar. Um zu erkennen, in welchem Netz sich eine Station befindet, verarbeitet eine Station die Informationen des als Broadcast gesendeten Beacon-Frames. Danach sind Informationen zur Nachrichtenübertragung (*MSDU Delivery*) von einer Station zur anderen, Verschlüsselung und Authentifizierung vorhanden. Die Verschlüsselung und die darauf aufbauende Authentifizierung ist optional.

Dienste des Verteilungssystems

Die Dienste des Verteilungssystems umfassen Assoziation, Deassoziation und Reassoziation. Stationen können sich an Access Points an- und abmelden um Netzdienste nutzen zu können. Zusätzlich verfügt der Access Point damit über Informationen welche Stationen sich in welcher Zelle befinden. Dazu werden Verteilungs-Informationen zwischen verschiedenen Access Points verschickt. Ein weiterer Dienst ist die Integration des Verteilungssystems in ein externes Netzwerk.

7.3.5 Übersicht über die Standards

Der ursprüngliche Standard wurde weiterentwickelt und eine Übersicht ist in Tabelle 7.3-5 zu finden.

Tab. 7.3-5: *802.11x Standards*

Standard	Bedeutung
802.11a	Verabschiedet 1999. Physical Layer Standard für das 5 GHz-Band. Es spezifiziert acht (bis 12, je nach Kontinent) verfügbare Kanäle. Maximaler Datendurchsatz ist 54 Mbit/s, Nutzdaten etwa 30 Mbit/s.
802.11b	Verabschiedet 1999. Physical Layer Standard für das 2.4 GHz-Band. Es spezifiziert bis zu 14 verfügbare Kanäle, davon drei die sich nicht überlappen. Maximaler Datendurchsatz ist 12 Mbit/s, Nutzdaten etwa 50% davon.
802.11d	Ergänzung zu 802.11 um einige länderspezifische Anforderungen zu erfüllen, u. a. Sendeleistungsanpassung. Alle Vorgaben der landesspezifischen Regulierung wie z. B. verfügbare Kanäle und Sendeleistungsstärke werden der WLAN-Karte vom Access Point mitgeteilt. So reicht es für die Hersteller aus, eine WLAN-Karte zu produzieren die dann weltweit eingesetzt werden kann.
802.11e	Erweiterung der MAC-Schicht um Quality of Service (QoS) und Class of Service (CoS).
802.11f	Erweiterung um das Inter Access Point Protocol (IAPP) auch über das WLAN-Interface eines Access Points zu verwenden. Auf diese Weise können Punkt-zu-Punkt-Verbindungen zwischen zwei Standorten realisiert werden (Wireless Bridge) und gleichzeitig WLAN-User die Access Points nutzen.

Fortsetzung auf der naechsten Seite

Fortsetzung von der vorhergehenden Seite

Standard	Bedeutung
802.11g	Entwicklung eines schnelleren Physical Layers im 2,4 GHz-Band bei Bebehaltung der vollen 802.11b-Kompatibilität. Eine Übertragungsgeschwindigkeit von brutto 54 MBit/s im 2,4 GHz Band. Netto stehen rund 20 MBit/s für die Datenübertragung zur Verfügung. Um bei größer werdender Entfernung von Client zu Access Point immer noch eine optimale Datenübertragung bei niedrigerem Signalpegel gewährleisten zu können, sind mehrere Geschwindigkeitsstufen in einem Autofallback-Verfahren integriert: 54, 48, 36, 24, 18, 12, 9 und 6 MBit/s bzw. 11, 5,5, 2 und 1 MBit/s im 802.11b-Modus.
802.11h	Anpassung des Standards 802.11a an die Anforderungen des European Telecommunications Standards Institute (ETSI). IEEE 802.11h basiert auf 802.11a, ist also kein neuer Standard sondern lediglich eine Erweiterung des Bestehenden um: • DFS (Dynamical Frequency Selection) Das System muss in der Lage sein zu erkennen, dass andere Systeme auf der gleichen Frequenz senden und dann automatisch in eine freie Frequenz wechseln. • TPC (Transmission Power Control) Das System muss in der Lage sein, seine Sendeleistung um mindestens 6 dBi anpassen zu können, wenn die Leistung nicht benötigt wird.
802.11i	Dies ist eine Erweiterung für die beiden IEEE Standards IEEE802.11a, IEEE802.11b sowie IEEE802.11g, um die bekannten Sicherheitslücken des *Wired Equivalent Privacy*-Verfahrens (WEP) zu schließen. Dazu ist WEP durch das *Temporal Key Integrity Protocol* (TKIP) und den AES-Algorithmus ersetzt worden. Das TKPI-Verfahren fußt auf rotierende Schlüsseln, die jeweils nach einer relativ kurzen Lebensdauer durch neue ersetzt werden.
802.11k	Erweiterung um Funktionen und Schnittstellen für einheitliche Funk-Messdaten.
802.11n	Durch verwendung mehrerer Sende- und Empfangsantennen und der MIMO-Technik (Multiple Input Multiple Output) sollen Datenraten bis 600 Mbit/s erreicht werden.

8 Internet-Protokolle

8.1 Einleitung

Das Internet ist ein dezentrales, paketvermittelndes Datenkommunikationsnetz, das heute über 390 Millionen[7] Rechner (Hosts) und Übermittlungsknoten (Router, *Nodes*) weltweit miteinander verbindet.

Geschichte des Internet

Die **Geschichte des Internet** ist sehr stark verbunden mit dem Kommunikationsprotokoll **TCP/IP** – dem ***T**ransmission **C**ontrol **P**rotocol/**I**nternet **P**rotocol*. Als der Vorgänger des Internet, das **ARPANET** (***A**dvanced **R**esearch **P**rojects **A**gency **Net**work*), in den siebziger Jahren immer größer wurde, stellte man fest, dass es immer schwieriger wurde, ein gemeinsames Protokoll zu finden, das von allen Rechnersystemen verstanden wurde und auch in großen Netzen stabil lief. Hinzu kam, dass sich weitere Netze entwickelten, wie z. B. das **USENET** (*Unix **User** **Net**work*) und FIDONET, die untereinander und mit dem ARPANET verbunden werden sollten. In dem sich so entwickelnden Verbundnetz kam überwiegend ein Protokoll zum Einsatz – das TCP/IP Protokoll. Das Netz, das sich zwischen den verschiedenen Netzinseln entwickelte, kann praktisch als Netz zwischen den Netzen, oder auch „InterNetz“, bezeichnet werden, woraus sich im Englischen der Name „**Internet**“ entwickelte.

Internet

Die grundlegende Idee von TCP/IP wurde 1974 von Vinton Cerf und Robert E. Kahn in einem wissenschaftlichen Artikel veröffentlicht [CKR74]. Im Jahre 1980 wurde dann begonnen, alle Hostrechner des ARPANET (ca. 300) auf TCP/IP umzustellen. Im Jahr 1983 war die Umstellung auf TCP/IP abgeschlossen, und das Internet war geboren.

Die freie Verfügbarkeit von TCP/IP führte dazu, dass nach dem Militär in den achtziger Jahren zuerst wissenschaftliche Einrichtungen und Hochschulen das Internet für den Informationsaustausch nutzten. In den neunziger Jahren wurde das Internet dann zunehmend auch für kommerzielle und private Anwendungen interessant.

Übertragungsprinzip

Das Prinzip der TCP/IP-Übertragung liegt darin, dass es eine klare Aufgabenverteilung zwischen den einzelnen Protokollen von TCP/IP gibt. Das Internet-Protokoll (IP) hat die Aufgabe, Datenpakete von einem Sender durch das Kommunikationsnetz an den Empfänger zu leiten. Falls ein Paket verloren geht, wird dies den Kommunikationspartnern nicht angezeigt. Diese Aufgabe übernimmt das oberhalb von IP operierende TCP, das im Fehlerfalle eine erneute Übertragung von Datenpaketen anfordert.

TCP/IP Vorteile

Ein großer Vorteil von TCP/IP ist, dass es in heterogenen Netzen eingesetzt werden kann, die aus unterschiedlichen physikalischen Netzen und Netztopologien bestehen. Um z. B. zwei kleine lokale Netze in unterschiedlichen Städten zu verbinden, muss keine Direktverbindung zwischen den LANs aufgebaut werden, sondern jedes

7 Quelle: Internet Software Consortium (http://www.isc.org).

LAN braucht nur an das Internet-Backbone angeschlossen zu sein. Die Kommunikation zwischen den beiden LANs erfolgt dann über das Internet, unabhängig davon, welche Technologie unterhalb der TCP/IP-Protokollschichten an den beiden Standorten eingesetzt wird (siehe Abb. 8.1-1).

Ein weiterer Vorteil von TCP/IP ist, dass die beiden kommunizierenden Hosts nicht wissen müssen, auf welchem Weg die Daten zu dem jeweiligen Kommunikationspartner gelangen. Die Datenpakete werden nur mit der Zieladresse des anderen Hosts versehen und dann an das Netz übergeben. Innerhalb des Netzes sorgen dann spezialisierte Rechner, die als Router bezeichnet werden, dafür, dass die Datenpakete durch das Internet an das gewünschte Ziel gelangen. Falls eine Verbindung innerhalb des Netzes ausfällt, wird einfach über einen Ersatzweg geroutet und die Daten erreichen trotzdem ihr Ziel. **Router** haben also die Aufgabe, die verschiedenen selbständigen Netze innerhalb des „Internetzes“ miteinander zu verbinden. Das bedeutet, dass sie immer mindestens mit zwei verschiedenen Netzen verbunden sind und somit auch mindestens zwei Internetadressen besitzen.

Router

Abb. 8.1-1: Hosts in verschiedenen LANs kommunizieren über das Internet

Ein weiterer wichtiger Grund für die starke Verbreitung von TCP/IP ist, dass es von Anfang an für den Einsatz in Client-Server-Anwendungen konzipiert wurde. In diesem Konzept kann jeder Host als Client oder Server agieren, unabhängig davon ob es sich bei dem Computer um einen PC oder einen *Mainframe* handelt. Als **Server** wird hierbei ein Host bezeichnet, der anderen Rechnern über ein Netz einen Dienst zur Verfügung stellt. Ein Rechner, der den Dienst auf einem Server in Anspruch nimmt, wird als **Client** bezeichnet. Typische Client-Server-Anwendungen sind z. B. das *World Wide Web*, das *File Transfer Protocol* oder eine Datenbankabfrage.

Server

Client

Selbsttestaufgabe 8.1-1:

Welchen Vorteil bietet TCP/IP im Bezug auf heterogene Netze?

8.2 Das TCP/IP-Modell

TCP/IP-Schichtenmodell

Zu der TCP/IP-Protokollfamilie gehört neben TCP und IP ein ganzer Satz weiterer Protokolle, die sich über die verschiedenen Schichten des OSI-Modells verteilen. Die TCP/IP-Protokolle sind entsprechend dem OSI-Modell schichtweise angeordnet. Im Gegensatz zum OSI-Modell besteht das **TCP/IP-Modell** allerdings nicht aus sieben sondern nur aus vier Schichten (siehe Abb. 8.2-1), wobei diese vier Schichten aber eindeutig den OSI-Schichten zugeordnet werden können.

Die Verwendung eines solchen Schichtenmodells hat den Vorteil, dass z. B. ein Anwendungsentwickler nicht wissen muss, welche Netztechnologie bei potentiellen Kunden eingesetzt wird. Das Schichtenmodell verbirgt die Technologie unterhalb der IP-Schicht und die Anwendung kann ohne Wissen über die eingesetzte Technologie entwickelt werden.

Abb. 8.2-1: Das TCP/IP-Schichtenmodell

Netz-Interface-Schicht

Die **Netz-Interface-Schicht** stellt die Datenübertragungsdienste zur Verfügung, z. B. über einen STM-1 SDH-Kanal, eine ATM-AAL5-Schnittstelle, einen Frame Relay PVC oder eine Ethernetkarte. Zu dieser Schicht gehören u. a. Kabel, Hardware (Netzwerkkarten) und Software (Schnittstellentreiber). Der für TCP/IP wichtigste Zugriff auf die Netz-Interface-Schicht erfolgt durch das ***A**ddress **R**esolution*

***P**rotocol* (**ARP**), das für die Zuordnung von Adressen der Vermittlungsschicht (IP-Adressen) zu Adressen der Netz-Interface-Schicht (z. B. MAC-Adressen) verantwortlich ist.

Internet-Schicht

Die **Internet-Schicht** beinhaltet u. a. das ***I**nternet **P**rotocol* (**IP**) ergänzt durch das ***I**nternet **C**ontrol **M**essage **P**rotocol* (**ICMP**) und das ***I**nternet **G**roup **M**essage **P**rotocol* (**IGMP**). Wesentliche Aufgabe ist die abschnittsweise, verbindungslose Übertragung von Datagrammen nach dem *best effort*–Verfahren sowie insbesondere die Wegewahl (*Routing*), wofür zahlreiche verschiedene Protokolle wie **RIP2** (***R**outing **I**nformation **P**rotocol Version **2***), **BGP4** (***B**order **G**ateway **P**rotocol Version **4***) und **OSPF** (***O**pen **S**hortest **P**ath **F**irst*) entwickelt wurden. In aktuellen verbesserten Protokollen versucht man zusätzlich IP-Datagrammströme zu identifizieren und verschiedenen Anwendungen wie WWW oder *Voice-over-IP* zuzuordnen, um anschließend eine priorisierte Behandlung der IP-Pakete mit Dienstgüten (***Q**uality **o**f **S**ervice* - **QoS**) vornehmen zu können. Die dadurch mögliche Verkehrssteuerung lehnt sich an Algorithmen der ATM-Technologie an. Die wichtigste Entwicklung ist hier das ***M**ulti-**P**rotocol **L**abel **S**witching* (**MPLS**), welches durch die *label switching* genannte Technologie eine schnelle Wegewahl und Adressumwandlung analog zur ATM-Technologie erlaubt.

Transportschicht

Die **Transportschicht** stellt das Verbindungsglied zwischen den Anwendungen und dem Kommunikationsnetz dar und ist in der Regel im Betriebssystem eines Rechners (Host) oder Netzknoten (*Node*) integriert. Die wichtigsten Protokolle dieser Schicht sind das ***T**ransmission **C**ontrol **P**rotocol* (**TCP**) für eine gesicherte, Fehler korrigierende Verbindung, die Flusskontrolle mit einschließt, und das ***U**ser **D**atagram **P**rotocol* (**UDP**) für eine ungesicherte Verbindung mit Fehlererkennung, aber ohne Fehlerkorrektur und Flusskontrolle. Beide Protokolle stellen die Ende-zu-Ende-Konnektivität zwischen zwei Endsystemen her, die durch ihre IP-Adressen gekennzeichnet sind, und sorgen für die Segmentierung des Nachrichtenverkehrs. Im Zusammenhang mit UDP wird bei multimedialen Anwendungen wie *Voice-over-IP* heute oft zusätzlich das ***R**ealtime **T**ransport **P**rotocol* (**RTP**) eingesetzt, um die Durchlaufzeiten und *Jitter* der IP-Pakete im Netz zu minimieren.

Anwendungsschicht

In der **Anwendungsschicht** werden schließlich die Applikationsprotokolle bereitgestellt, deren Ergebnisse man als Nutzer auf dem Bildschirm sieht. Als kleine Auswahl gehören dazu ***H**yper **T**ext **T**ransfer **P**rotocol* (**HTTP**) zur Darstellung von ***W**orld **W**ide **W**eb* (**WWW**)-Inhalten, ***S**imple **M**ail **T**ransfer **P**rotocol* (**SMTP**) und ***P**ost **O**ffice **P**rotocol Version **3*** (**POP3**) für eMail, ***F**ile **T**ransfer **P**rotocol* (**FTP**) für *File Transfer*, *Telnet Protocol* für Terminal-Anwendungen und ***S**imple **N**etwork **M**anagement **P**rotocol* (**SNMP**) als Netzmanagement-Standard.

Selbsttestaufgabe 8.2-1:

Aus wie vielen Schichten besteht das TCP/IP-Modell? Nennen Sie diese. In welcher Schicht wird das Internet Protocol eingesetzt?

8.3 Die Standardisierung

Die unter dem Namen TCP/IP-Protokolle subsummierten Kommunikationsprotokolle werden von dem ***I**nternet **A**rchitecture **B**oard* (**IAB**) verwaltet, die selbst zwei spezialisierte Untergruppen, die ***I**nternet **E**ngineering **T**ask **F**orce* (**IETF**) und die ***I**nternet **R**esearch **T**ask **F**orce* (**IRTF**) unterhält. Alle Entscheidungen des IAB seit 1969 werden auch im Internet unter *www.ietf.org* in den so genannten **Requests for Comments** (**RFC**) veröffentlicht.

Requests for Comments

Die RFC durchlaufen verschiedene Stadien wie *Initial*, *Proposed Standard*, *Draft Standard*, *Standard*, *Experimental* und *Historic*. Das IAB definiert für jedes Protokoll einen geeigneten Zustand (*State*), der in Abb. 8.3-1 dargestellt ist.

State	Required	Recommended	Elective	Limit Use	Not Recommended
Standard	x	xxx	xxx		
Draft	x	x	xxx		
Proposed		x	xxx	x	
Initial		x	xxx	x	x
Experimental			x	xxx	x
Historic				x	xxx

x mögliche Behandlung des RFC
xxx bevorzugte Behandlung des RFC

Abb. 8.3-1: Zustände von TCP/IP-Protokollen (RFC)

Die einzelnen Zustände der RFC bedeuten unterschiedliche Anforderungen an Hersteller von internetfähigen Geräten. Tabelle 8.3-1 zeigt die empfohlenen Handlungsalternativen bei RFC in den unterschiedlichen Zuständen. Die wesentlichen Anstöße zur Definition neuer RFC werden von der IETF vorgenommen. Obwohl auch hier die Standardisierung mehrere Monate dauert, ist sie im Vergleich zu den anderen internationalen Gremien wie **ISO** (*International Organization for Standardization*), **ITU** (***I**nternational **T**elecommunication **U**nion*) oder **ETSI** (***E**uropean **T**elecommunications **S**tandards **I**nstitute*) wesentlich schneller, da ein RFC bereits nach drei bis sechs Monaten verworfen oder zur Veröffentlichung vorgelegt wird.

Der beschleunigten Entwicklung von RFC hilft auch die Tatsache, dass üblicherweise mindestens zwei eigenständige und unabhängige Implementierungen des Protokolls realisiert werden, die miteinander funktionell getestet werden müssen, bevor ein *Proposed Standard* zu einem *Draft Standard* wird.

Tab. 8.3-1: *Empfehlungen zum Umgang mit RFC*

State	Bedeutung
Required	Auf allen Hosts und *Nodes* (Router), die TCP/IP nutzen, müssen diese RFC implementiert sein.
Recommended	Es wird geraten, auf allen Hosts und *Nodes* (Router) diese RFC zu implementieren.
Elective	Hersteller können wählen, ob sie auf Hosts und *Nodes* (Router) diese RFC implementieren.
Limited Use	Nicht für den allgemeinen Gebrauch bestimmt; Hersteller können wählen, ob sie auf Hosts und *Nodes* (Router) diese RFC selektiv implementieren.
Not Recommended	Herstellern wird empfohlen, diese RFC aus Interoperabilitätsgründen nicht zu implementieren.

Aufgrund der Fülle der Protokolle in der TCP/IP-Protokollfamilie werden in den folgenden Kapiteln nur die grundlegenden Funktionen und Anforderungen der einzelnen Protokollschichten vorgestellt und an wichtigen Beispielen erläutert.

8.4 TCP / IP-Zugriff auf die Netz-Interface-Schicht

In der Netz-Interface-Schicht ist kein allgemeingültiges Protokoll vorhanden, da die zugrundeliegende Netztechnologie unterschiedlich sein kann. Die verschiedenen Netztypen und Protokolle für lokale, regionale und Weitverkehrsnetze wie Ethernet, FDDI oder ATM wurden in den vorherigen Kapiteln eingehend behandelt, so dass an dieser Stelle darauf verzichtet wird.

Im Folgenden wird für das Verständnis der in Schichten arbeitenden Kommunikationsprotokolle die Kapselung von Datenpaketen aufgezeigt und die notwendige Anpassung an die Rahmenstruktur der unterschiedlichen Protokolle der Netz-Interface-Schicht dargestellt.

8.4.1 Maximum Transfer Unit

Grundsätzlich wird in jeder logischen Schicht den Daten ein spezifischer Protokollkopf hinzugefügt, der die für diese und benachbarte Schichten wichtigen Informationen wie z. B. Adressierung, Applikations- oder Protokolltyp enthält, um die der Schicht zugeordneten Funktionen in den an der Kommunikation beteiligten Geräten (*Nodes*, Hosts, Endgeräte) wahrnehmen zu können . Durch diese **Kapselung** (*Encapsulation*) der einzelnen Schichten werden die Funktionen eindeutig voneinander getrennt und erleichtern bzw. beschleunigen die notwendigen Verarbeitungsschritte.

Kapselung

In Abb. 8.4-1 wird beispielhaft die Kapselung von Nutzdaten in Ethernet-Rahmen nach RFC 894 dargestellt. Diese Kapselung erlaubt im praktischen Einsatz die Trennung von Applikationen und Übertragungsnetzen sowie die parallele Verarbeitung von verschiedenen Protokollfamilien wie z. B. TCP/IP und IPX von Novell. In jeder Schicht werden die notwendigen Informationen für die benachbarten Schichten durch entsprechende Software- und Gerätetreiber implementiert.

In Bit- und Byte-getakteten Übertragungsmedien wie SDH oder ISDN muss zusätzlich eine Rahmenstruktur z. B. nach dem ***H**igh-Level **D**ata **L**ink **C**ontrol* (**HDLC**)-Protokoll oder dem ***P**oint-to-**P**oint **P**rotocol* (**PPP**) hinzugefügt werden.

Abb. 8.4-1: Bildung von Übertragungsrahmen am Beispiel von Ethernet

Maximum Transfer Unit

Leider sind die verfügbaren Paket-, Zellen- und Rahmengrößen in den verschiedenen Netzen nicht identisch, sondern es werden verschieden große maximale IP-Datagramme (**Maximum Transfer Unit - MTU**) erlaubt. Je kleiner eine MTU ist, um so größer ist der zusätzliche Signalisierungsanteil durch den Protokollkopf (*Overhead*) zur Kennzeichnung der IP-Datagramme in der Netz-Interface-Schicht; je größer eine MTU ist, um so länger dauert die Übertragung eines einzelnen IP-Datagramms. Lange IP-Datagramme eignen sich aufgrund des geringen *Overheads* für den *Filetransfer*, kurze IP-Datagramme sind aufgrund der kurzen Übertragungsverzögerung pro Datagramm günstig zur Übertragung von verzögerungssensitiven Daten (z. B. bei Sprache, Audio und Video), denn je kleiner ein IP-Datagramm ist, um so kürzer ist auch die "Einpackzeit" der Nutzdaten in das Datagramm.

Die MTU-Größen verschiedener Netztechnologien sind in Tabelle 8.4-1 zusammengestellt. Da die Übertragung über verschiedene Netze hinweg und die damit einhergehende Re-Segmentierung auf jedem Teilabschnitt zu unnötiger Rechenbelastung in den Routern führen würde, ist in den heute eingesetzten Implementierungen des für die Kapselung verantwortlichen *Internet Control Message Protocol* der Vermittlungsschicht ein *Path MTU Discovery*-Mechanismus implementiert, durch den die kleinste MTU Ende-zu-Ende ermittelt und zu Beginn einer Übertragung festgelegt wird.

Tab. 8.4-1: *Maximale Größen der MTU in verschiedenen Netzen*

Netztechnologie	MTU [Byte]
X.25	576
IEEE 10/1000 BASE-T	1492
DEC, Intel, XEROX Ethernet	1500
ATM (LANE)	1500
FDDI	4352
IEEE Token Ring	4464
ATM (Classical IP over ATM)	9180
IBM; 16 Mbit/s Token Ring	17914
Hyperchannel	65535

8.4.2 Address Resolution Protocol

Da in allen Netzen die physikalische Adressierung mit Adressen der Netz-Interface-Schicht (u. a. *Media Access Control*-Adressen) vorgenommen wird, fehlt zunächst bei der Inbetriebnahme eines TCP/IP-Protokollstacks die Abbildung der physikalischen Geräteadresse auf die logische IP-Adresse, was z. B. bei jedem Einschalten des eigenen PCs im **LAN** (***L****ocal* ***A****rea* ***N****etwork*) durchlaufen wird. Zur Feststellung von IP-Adressen wird das ***A****ddress* ***R****esolution* ***P****rotocol* (**ARP**) (RFC 826) eingesetzt. Die IP-Adressen können dabei statisch fest vergeben sein oder dynamisch zugeordnet werden. In Broadcastnetzen wie Ethernet enthalten daher die Rahmen neben den IP-Datagrammen ein 2 Byte großes Type-Feld, das anzeigt, welches darüber liegende Protokoll (IP oder ARP) benutzt wird.

Beispiel:
Am Beispiel des klassischen Ethernet ist die Rahmenstruktur in Abb. 8.4-2 aufgezeigt.

MAC Destination Address	MAC Source Address	Type	IP Datagram	CRC
(6 *Byte*)	(6 *Byte*)	(2 *Byte*)	(46 - 1500 *Byte*)	(4 *Byte*)

Abb. 8.4-2: Ethernet-Rahmen mit Type-Feld und IP-Datagramm

Werden IP-Datagramme im Ethernet-Rahmen transportiert, so enthält das Type-Feld den hexadezimalen Wert $0x0800$. Handelt es sich beim Protokoll der nächst höheren Schicht um das ARP, so enthält das Type-Feld den hexadezimalen Wert von $0x0806$.

Die MTU beträgt $1500\,Byte$, während ein Ethernet-Rahmen zwischen 64 und $1518\,Byte$ groß sein darf. Daraus berechnet sich bei einem $10\,Mbit/s$ Ethernet-Netz mit einem Protokollkopf von $30\,Byte$ Länge ein *Overhead* zwischen 3 % und 75 % sowie eine Sendeverzögerung von $6,4\ \mu s$ bis $151,8\ \mu s$ pro Rahmen, da alle Ethernet-Rahmen sequentiell übertragen werden.

Um die **MAC**-Adresse (***M**edia **A**ccess **C**ontrol*) einer zugehörigen IP-Adresse feststellen zu können, sendet ein Gerät einen ARP-Request in Form eines ARP-Pakets (Abb. 8.4-3) als Broadcast an alle an das Netzsegment angeschlossenen Systeme. Alle Systeme prüfen bei Empfang des ARP-Request, ob die eigene IP-Adresse angefragt ist, und senden bei Erfolg einen ARP-Reply mit der eigenen MAC-Adresse an den anfragenden Sender zurück.

Abb. 8.4-3: Format der ARP-Pakete und ARP-Paket-Header

Mittlerweile wird die dynamische Zuordnung von IP-Adressen durch den Einsatz von ARP-Servern (z. B. ATMARP in ATM-Netzen) und ***D**ynamic **H**ost **C**onfiguration **P**rotocol* (**DHCP**)-Servern erleichtert. Mit Hilfe des ARP melden sich neu angeschaltete Systeme bei den Servern, registrieren sich und bekommen von dort eine IP-Adresse zugewiesen, bzw. können die IP-Adressen anderer Systeme dort abfragen. Diese Entwicklung gewinnt vor allem durch den Aufbau virtueller LANs an Bedeutung, wobei einem physikalischen Port mehrere logische IP-Adressen zugewiesen werden können.

8.5 TCP/IP-Protokolle der Vermittlungsschicht

In der Vermittlungsschicht werden alle Dienste bereitgestellt, die eine Weiterleitung von Datagrammen zur Zieladresse ermöglichen. Dazu gehören insbesondere:

- Speicherung von eingehenden Paketen im Eingangspuffer,
- Adressierung und Wegewahl (*Routing*),
- unspezifische (*best effort*) oder priorisierte Weiterleitung von Paketen,
- Segmentierung und Reassemblierung von Datensegmenten,
- Berechnung von Ersatzwegen im Fehlerfall.

Die wesentlichen Protokolle der Vermittlungsschicht sind das ***I**nternet **P**rotocol* (**IP**) selbst, ergänzt durch die Steuerungsprotokolle ***I**nternet **C**ontrol **M**essage **P**rotocol* (**ICMP**) und ***I**nternet **G**roup **M**essage **P**rotocol* (**IGMP**), das Ressourcenmanagement-Protokoll ***R**esource **Re**s**er**vation **P**rotocol* (**RSVP**) sowie das ***M**ulti-**P**rotocol **L**abel **S**witching* (**MPLS**)-Protokoll als Schritt zur Zusammenführung von Netz-Interface-Schicht- und Vermittlungsschicht-Funktionalitäten. Da die TCP/IP-Protokolle so ausgelegt sind, dass grundsätzlich von einem fehleranfälligen Netz ausgegangen wird, sind in der Vermittlungsschicht, die im Zusammenhang mit TCP/IP bevorzugt Routing-Schicht genannt wird, insbesondere Routinen und Algorithmen implementiert, die eine möglichst hohe Anpassungsfähigkeit an verschiedene Netze und Lastsituationen ermöglichen.

8.5.1 Internet Protocol Version 4

Für die Bewältigung der Aufgaben der Vermittlungsschicht wurde das ***I**nternet **P**rotocol* (**IP**) in RFC 791 definiert. Die zurzeit am weitesten verbreitete Version des *Internet Protocols* ist Version 4 (**IPv4** - ***I**nternet **P**rotocol **V**ersion **4***). IP unterstützt eine Vielzahl von darunter liegenden Netz-Technologien, u. a. Ethernet, 10/100BASE-T, Token Ring, X.21, X.25, Frame Relay und ATM. Ermöglicht wird diese Vielfalt durch die einfache Bereitstellung der entsprechenden Socket-Software als Gerätetreiber der physikalischen Schnittstellenkarten und die universelle Einsetzbarkeit von IP zur Übermittlung von Daten in Netzen mit verschiedenen Eigenschaften. IP stellt einen paketorientierten, verbindungslosen und ungesicherten Dienst ohne jegliche Servicegarantie (*best effort*) zum Austausch von Daten zwischen zwei Endsystemen bereit.

IPv4

Dienste des IP

Paketorientiert bedeutet, dass IP die Daten der darüber liegenden Schichten, egal ob sie diskret (z. B. *Filetransfer*) oder kontinuierlich (z. B. Videosignale) vorliegen, ausschließlich in Datenpakten, so genannten IP-Datagrammen, und nicht in Übertragungskanälen (*Switched Circuits*) überträgt. Dazu ist zusätzlich zur Anpassung an die MTU der zu Grunde liegenden Übertragungsmedien ein Segmentierungs- und Reassemblierungsalgorithmus notwendig, der die Datagramme auf die Rahmenlängen der eingesetzten Netze anpasst.

Verbindungslos bedeutet, dass die IP-Router und Endgeräte nur die Sender- und Empfängeradressen kennen, aber keine Information über den Weg zwischen Sen-

der und Empfänger existiert. Router senden Datagramme unabhängig von ihrem zurückgelegten Weg in die Richtung weiter, die entsprechend dem eingesetztem Routing-Protokoll in Richtung des Ziels führt. Aufgrund dieser Methodik besitzen die Routing-Protokolle eine enorme Bedeutung innerhalb der TCP/IP-Protokollfamilie, deren Aufgaben mit komplexen graphentheoretischen Ansätzen gelöst werden.

IP gilt als völlig ungesichert, da keine Garantie übernommen wird, ob Daten überhaupt zugestellt werden, beim richtigen Empfänger ankommen, oder auf diesem Weg nicht manipuliert oder beschädigt wurden. Die Sicherstellung dieser Funktionen wird höheren Schichten oder verschlüsselten Kapselungen z. B. durch die Benutzung des **Internet Protocol Security** (**IPsec**)-Protokolls überlassen. Die Güte der Übertragung kann allerdings durch die Wahl und Eigenschaften des Transportnetzes wesentlich beeinflusst werden. So können in ATM-Netzen *best effort*-IP-Ströme in virtuellen Verbindungen (**PVC** - ***P**ermanent **S**witched **C**hannel* oder **SVC** - ***S**witched **V**irtual **C**hannel*) von Ende zu Ende transportiert werden, die vor fremdem Zugriff schon auf der Ebene der Netz-Interface-Schicht-Adressen sehr gut geschützt sind und gute Verkehrssteuerungsmechanismen zur Bereitstellung von Dienstgütegarantien bieten.

Internet Protocol Security

Wesentliches Element des IP-Protokolls ist das IP-Datagramm (Abb. 8.5-1) zur Paketierung und Kapselung von Daten höherer Schichten mit einem Paketkopf von 20 Byte Länge (ohne die selten genutzten Optionen) und einem Informationsfeld von 0 bis 65 515 Byte Länge, was insgesamt durch ein Längenfeld von 16 Bit beschrieben werden kann.

Abb. 8.5-1: Format eines IPv4-Datagramms

Die Funktionen der Vermittlungsschicht werden ausschließlich durch die Bit bzw. Meldungen im **Header** eines IP-Datagramms gesteuert. In Abb. 8.5-2 sind die Felder des IPv4-Datagramm-Header im einzelnen aufgeführt. Aufgrund der Diskussionen um die Erweiterung von Adressbereichen und Funktionen im IP-Protokoll wird sich das Format mit der Einführung der nächsten Protokollversion **IPv6** (***I**nternet **P**rotocol **V**ersion **6***) ändern (siehe Abschnitt 8.5.8, Kurseinheit 13). Die Einführung ist allerdings flächendeckend nur langsam zu erwarten, da aufgrund der hohen Verbreitung von IPv4 zwingend eine Abwärtskompatibilität von IPv6 auf IPv4 gefordert wird.

Header

IPv6

Abb. 8.5-2: Format eines IP-Datagramm-Header

Die Funktionen der einzelnen Felder des IP-Datgramm-Header sind:

Version (4 Bit) gibt die Version des IP-Protokolls an, zurzeit ist die Version 4 (IPv4) gültig.

Header Length (4 Bit) zeigt die Gesamtlänge des Header einschließlich der Optionen in 32-Bit-Wörter an. Bei einem Standard-Header von 20 Byte kann hieraus die Länge der Optionen (maximal 40 Byte) und der Beginn des Informationsteils berechnet werden.

Type of Service (**TOS**) (8 Bit) werden zur Kennzeichnung und Priorisierung von IP-Datagrammen genutzt und in einigen Routing-Protokollen mit *policy routing*-Algorithmen unterstützt. Mögliche Differenzierungen sind *minimize delay*, *maximize throughput* oder die Trennung von Verkehrsklassen in den sogenannten *precedence bits* (siehe Abb. 8.5-3).

Abb. 8.5-3: Type of Service (TOS)-Feld des IP-Datagramm-Header

Total Length (16 Bit) gibt die Gesamtlänge des IP-Datagramms in Byte an (maximal 65.535 Byte).

Zur Segmentierung der IP-Datagramme nutzt man die Felder *Identification*, *Flags* und *Segmentation Offset*, auch *Fragmentation* genannt. Mit Hilfe dieser Felder wird berechnet, ob und wie ein IP-Datagramm, z. B. auf Grund von kleineren Rahmenlängen der Netztechnologie der Netz-Interface-Schicht, in kleinere Einheiten fragmentiert und wieder zusammengefügt wird.

Identification (16 Bit) gibt die Zuordnung eines Fragments zu einem IP-Datagramm an. Die Vergabe der Identifikationsnummer erfolgt in der Regel durch eine höhere Schicht (z. B. TCP) und wird als Parameter an IP übergeben.

Flags (3 Bit) signalisieren die Segmentierungsmöglichkeiten und den Status im Segmentierungsprozess (Abb. 8.5-4). Bit 0 ist reserviert und muss 0 sein. Bit 1 zeigt die Fragmentierungsmöglichkeit an und Bit 2 zeigt an, ob es sich um das letzte oder ein laufendes Fragment handelt, dem noch weiteres Fragment folgt.

Bit 0	0	*Default 0* / muss 0 sein
Bit 1	0	*May Fragment* / Fragmentierung möglich
	1	*Don't Fragment* / Fragmentierung unmöglich
Bit 2	0	*Last Fragment* / letztes Fragment
	1	*More Fragment* / weitere Fragmente

Abb. 8.5-4: Funktionen des Flag-Feldes des IP-Datagramm-Header

Segmentation Offset (13 Bit) gibt die Lage der aktuellen Daten relativ zum Anfang des Datenblocks im ursprünglichen Datagramm in Schritten von 8 Byte an. Standardmäßig ist dieser Wert gleich null.

Beispiel 8.5-1:
Das Datagramm hat eine Größe von $1520\,Byte$, wobei $20\,Byte$ zum *Header* gehören. Die kleinste MTU im Pfad beträgt $576\,Byte$. Folglich muss das Datagramm in 3 Fragmente aufgeteilt werden. Der Wert des *Fragment Offsets* für das erste Datagrammfragment wird 0 sein, da es mit dem Anfang des originalen Datagramms beginnt. Das *Fragment Offset* des zweiten Fragments wird

$$\frac{552\,Byte}{8\,Byte} = 69$$

sein, weil das zweite Fragment nach den $552\,Bytes$ des originalen Datagramms beginnt. Das dritte Fragment startet schließlich nach $1104\,Bytes$ in Bezug auf das originale Datagramm. Das *Fragment Offset* des dritten Fragments beträgt folglich

$$\frac{1104\,Byte}{8\,Byte} = 138.$$

Version
IHL
Type of Service
Total Length = 1520 Byte
Identification = ID1
DF = 0
MF = 0
Fragment Offset = 0
Time to Live
Protocol
Header Checksum = SUM 1
Source Address = Host A
Destination Address = Host B
Data
20 Byte
1500 Byte
Fragmentierung
Fragment 1
Total Length = 572 Byte
Identification = ID1
DF = 0
MF = 1
Fragment Offset = 0
Source Address = Host A
Destination Address = Host B
Data (Byte 1-552)
20 Byte
552 Byte
Fragment 2
Total Length = 572 Byte
Identification = ID1
DF = 0
MF = 1
Fragment Offset = 69
Source Address = Host A
Destination Address = Host B
Data (Byte 553-1104)
20 Byte
552 Byte
Fragment 3
Total Length = 416 Byte
Identification = ID1
DF = 0
MF = 0
Fragment Offset = 138
Source Address = Host A
Destination Address = Host B
Data (Byte 1105-1500)
20 Byte
396 Byte

Time to Live (**TTL**) (8 Bit) begrenzt die Lebensdauer eines Datagramms. Da es in TCP/IP-Netzen keine einheitliche Uhr gibt, setzt üblicherweise ein Router bei jedem Sendevorgang eines Datagramms, also je *hop*, den TTL-Zähler um eins herab bis TTL=0 erreicht und das Datagramm ungültig wird. Nach RFC 1060 wird der Standardwert für TTL auf 32 gesetzt.

Protocol Identity (8 Bit) zeigt an, welches Protokoll im Datenfeld angesprochen wird. Diese werden in den *Assigned Internet Protocol Numbers* veröffentlicht. Die wichtigsten Protokolle sind 1 = ICMP, 2 = IGMP, 6 = TCP, 17 = UDP, 80 = OSI-IP, 88 = IGRP, 89 = OSPFIGP.

Header Checksum (16 Bit) ist das Ergebnis eines zyklischen Redundanzchecks über den IP-Datagramm-Header zur Sicherung des *Header.*

Source Address (32 Bit) ist die IP-Adresse des Senders.

Destination Address (32 Bit) ist die IP-Adresse des Empfängers.

Options sind bis zu 40 Byte groß und werden durch die Anwendung des Senders gewählt. Beispiele hierfür sind:

- *Reverse Route*: Der Verkehr vom Ursprung zum Ziel, fließt auch auf dem selben Pfad wieder zurück.
- *Record Route*: Das Headerfeld enthält eine Liste von Router-IP-Adressen, die ein Datagramm besucht hat.

Options erfordern spezielle Behandlung der IP-Datagramme in den Routern und setzen dadurch deren Gesamtdurchsatz erheblich herab. Sie werden deshalb in der Praxis möglichst vermieden.

Padding wird benutzt, um eine Header-Länge mit einem Vielfachen von 4 Byte zu erreichen, wenn *Options* nicht an einer 4-Byte-Grenze enden.

8.5.2 IPv4-Adressierungsschema

Um einen Host weltweit im Internet erreichen zu können, muss ihm eine eindeutige Adresse zugeordnet werden. In TCP/IP-Netzen wird dies erreicht, indem jedem Host eine eindeutige Internetadresse zugeordnet wird. Genau genommen handelt es sich hierbei nicht um die Identifikation eines individuellen Hosts, sondern um die Identifikation eines Netzanschlusses.

Die IP-Adressen sind fester Bestandteil des Internet-Protokolls und bestimmen die Wegewahl zwischen Sender und Empfänger sowie die Bildung von Subnetzen, bzw. Supernetzen. Aufgrund der begrenzten Anzahl von Adressen, die mit 32-Bit-Wörter dargestellt werden können, werden seit Jahren Segmentierungsverfahren zur Bildung von Subnetzen angewandt, um dem wachsenden Bedarf an IP-Adressen gerecht zu werden. Bei der Darstellung der IP-Adressen hat sich die dezimale, durch Punkt getrennte Schreibweise (**Dotted Decimal Notation**) durchgesetzt. Eine IP-Adresse bestehend aus 4 Byte wird folglich in dezimaler Schreibweise z. B. 192.5.6.2 dargestellt, wobei jedes Adresssegment ein Byte repräsentiert.

Dotted Decimal Notation

Unterschieden wird dabei in verschiedene Adressteile, die Netzklasse, Netzadresse und Host-Adresse darstellen.

Die unterschiedlichen IP-Netze werden in die Klassen A bis E unterteilt (siehe Abb. 8.5-5), wobei A bis C *Unicast*-Adressen benutzen, D für *Multicast* eingesetzt wird und E für zukünftige Anwendungen noch reserviert ist. Im Gegensatz zu **Unicast**, wo einzelne Hosts adressiert werden, wird bei **Multicast** eine Gruppe von Rechnern adressiert. Somit kann ein Sender seine Daten gleichzeitig an mehrere Empfänger verschicken, ohne dabei jeden Empfänger einzeln adressieren zu müssen.

Unicast
Multicast

Abb. 8.5-5: IP-Adressklassen und Adressformate

Die IP-Adressen der Klassen A-C setzen sich immer aus einem Netzanteil und einem Host-Anteil zusammen, wobei der Host-Anteil mindestens ein Oktett umfasst. Klasse-A-Netze enthalten die meisten Einzeladressen und werden nur noch für extrem große Netze mit sehr vielen Rechnern vergeben. Klasse-B-Netze sind verfügbar für große *Provider*, bei denen sich die Anzahl der Subnetze und angeschlossenen Rechner in etwa die Waage halten. Klasse-C-Netze werden an kleinere Provider oder größere Endkunden vergeben. Kleine Internetanwender erhalten oft wenige Adressen aus dem Klasse-C-Netz des *Internet Service Providers* zugeteilt.

Da die Aufteilung der IP-Adressen in die verschiedenen Klassen den registrierbaren Adressraum stark einschränkt, wird die Methode der **Subnetzbildung** zur weiteren Aufteilung des Adressraums eingeführt. In Subnetzen wird der Host-Anteil der IP-Adresse in einen höherwertigen Subnetzanteil und einen minderwertigen Host-Anteil aufgeteilt. Die Aufteilung erfolgt durch eine Subnetzmaske, wobei der Subnetzanteil durch »1« und der Hostanteil durch »0« gekennzeichnet ist und durch eine logische UND-Verknüpfung von IP-Adresse und Subnetzmaske der Wert des Subnetzes ermittelt wird.

Subnetzbildung

Beispiel:
Bei der Einstellung einer IP-Adresse für einen Rechner muss neben der eigentlichen Adresse die sogenannte Subnetzmaske mit angegeben werden. Die Subnetzmaske definiert, welcher Teil der Adresse zur Adressierung eines Subnetzes und welcher Teil für die Adressierung des Hosts benutzt wird. Die Subnetzmaske besteht wie die IP-Adressen aus 32 Bit, wobei die Stellen, die für die Netzadressierung gebraucht werden, mit einer binären 1 gekennzeichnet werden. Für die drei Netzklassen ergeben sich also die folgenden Standardsubnetzmasken:

Klasse A : 255.0.0.0,

Klasse B : 255.255.0.0 und

Klasse C : 255.255.255.0.

Für eine Klasse-B-IP-Adresse mit $141.89.64.1$ und einer Subnetzmaske von $255.255.0.0$ ergeben sich folgende Host- und Subnetzanteile:

IP-Adresse	141.89.64.1	10001101	01011001	01000000	00000001	AND
Subnetzmaske	255.255.0.0	11111111	11111111	00000000	00000000	
Subnetzanteil		10001101	01011001	00000000	00000000	

IP-Adresse	141.89.64.1	10001101	01011001	01000000	00000001	AND
Subnetzmaske	255.255.0.0	11111111	11111111	00000000	00000000	NOT
Hostanteil		00000000	00000000	01000000	00000001	

16 Bit

Es stehen insgesamt $2^{16} = 65536$ IP-Adressen für die Adressierung zur Verfügung. Die kleinste Adresse $141.89.0.0$ stellt die IP-Adresse des Netzes und die größtmögliche Adresse $141.89.255.255$ stellt die Broadcast-Adresse im Subnetz dar. Somit stehen insgesamt $65536 - 2 = 65534$ IP-Adressen für Hosts zur Verfügung.

Alle IP-Adressen werden offiziell registriert und sind weltweit gültig. In Europa werden sie in der Regel von einem *Internet Service Provider* vermittelt, der sie wiederum vom **RIPE NCC** (***R**ésaux **IP** **E**uropéens **N**etwork **C**oordination **C**entre*) in Amsterdam zugeteilt bekommt.

Private IP-Adressen

Darüber hinaus können die so genannten privaten IP-Adressen (RFC 1918, Abb. 8.5-6) von jedem Anwender beliebig eingesetzt werden, doch dürfen diese

Adressen auf keinen Fall direkt ins Internet geroutet werden, da durch die vielfache Verwendung der privaten IP-Adressen die weltweit zwingend notwendige Eindeutigkeit der Internetadressen verloren geht.

Klasse	Netzadresse	Anzahl möglicher Netze
A	10.0.0.0 bis 10.255.255.255	1
B	172.16.0.0 bis 172.16.255.255	16
C	192.168.0.0 bis 192.255.255.255	255

Abb. 8.5-6: Private IP-Adressen (RFC 1918)

Bei fehlerhafter Anwendung der privaten IP-Adressen muss der Anwender von seinem *Provider* radikal vom Internet getrennt werden! Gelöst wird dieses Problem durch den Einsatz von PROXY-Servern, die die Umsetzung von privaten IP-Adressen auf eine eindeutige IP-Adresse vornehmen, um interne IP-Netze mit privatem Adressierungschema an das Internet anzuschließen.

IP-Adressen mit Sonderfunktionen

Loopback-Adresse

Bei der IP-Adressierung hat man verschiedene Adressen mit besonderen Funktionen belegt. Alle Adressen 127.x.x.x sind die sogenannte »**Loopback-Adresse**«, die dafür sorgt, dass ein Datagramm mit dieser Adresse von jedem Router an den Sender zurückgeschickt wird. Diese Funktion wird oft intern zu Testzwecken benutzt. Die Adresse 127 steht im 1. Oktett für keine andere Applikation zur Verfügung.

Broadcast-Adresse

Der Adresswert 255 ist in allen Oktetts als **Broadcast-Adresse** reserviert, auf den alle TCP/IP-Nodes und -Hosts im betrachteten Netzwerk (Klasse A, B, C oder Subnetz) reagieren. In alten Implementierungen war der Adresswert 0 mit der Funktion *Broadcast* verbunden, was in heute eingerichteten Netzen unbedingt vermieden werden sollte.

Stattdessen wird der Adresswert 0 als *»This«* für die Kennung des eigenen Hosts oder Netzes eingesetzt. Die IP-Adresse 0.1.1.1 bedeutet folglich, dass der Rechner mit der Adresse xxx.1.1.1 auf dem aktuellen Netz angesprochen wird.

Selbsttestaufgabe 8.5-1:

Zu welcher Klasse gehört die folgende IP-Adresse 225.192.136.110*?*

8.5.3 Internet Control Message Protocol

Das ***I**nternet **C**ontrol **M**essage **P**rotocol* (**ICMP**) (RFC 792) wird zur Übertragung von Fehlermeldungen und Netzinformationen eingesetzt. Die Daten des ICMP werden oft direkt an Managementapplikationen höherer Schichten weitergeleitet.

Anwendungen

Bekannte und wichtige **Anwendungen** sind u. a. **PING** (***P**acket **I**nter**n**et **G**roper*) zum Testen der Erreichbarkeit von Hosts oder *Nodes* mit bekannten IP-Adressen und TRACEROUTE zur Ermittlung des Weges zwischen Sender und Empfänger.

Eine ICMP-Meldung wird im *Protocol Identity*-Feld eines IP-Datagramms angezeigt und im Datenfeld übermittelt. Sie besteht aus einem 1 Byte-Typfeld, einem 1 Byte-Codefeld, einer 16 Bit langen Prüfsumme über das komplette ICMP-Paket und den Typ- und Code-abhängigen Daten des ICMP-Pakets. In Abb. 8.5-7 ist das Format eines ICMP-Pakets ohne den IP-Datagramm-Header dargestellt.

Byte 1	Byte 2	Byte 3	Byte 4
ICMP Type	ICMP Code	ICMP Checksum	
ICMP Data			

Abb. 8.5-7: Format eines ICMP-Pakets

Die Meldungen des ICMP unterscheiden sich in Anfragen und Fehlermeldungen.

Jede ICMP-Meldung hat einen definierten Typ mit eigenen Codes und definierten anschließenden Informationsfeldern. Einige wichtige Typen sind in Tabelle 8.5-1 aufgelistet. Zur vollständigen Darstellung sei auf die entsprechenden RFC verwiesen.

Tab. 8.5-1: *Werte des ICMP-Typs*

ICMP Type	Bedeutung	Nachrichtenart
0	echo reply (PING)	Anfrage
8	echo request (PING)	Anfrage
9	router advertisement	Anfrage
10	router solicitation	Anfrage
13	timestamp request	Anfrage
14	timestamp reply	Anfrage
17	address format request	Anfrage
18	address format reply	Anfrage
242	address mask request	Anfrage
243	address mask reply	Anfrage
3	destination unreachable	Fehlermeldung
4	source quench	Fehlermeldung
5	redirect	Fehlermeldung
11	time exceeded	Fehlermeldung
12	parameter problem	Fehlermeldung

Beispiel:
Als Beispiel wird die Meldung »Zielknoten nicht erreichbar« (*destination unreacheable*) betrachtet. Der ICMP-Typ für diese Meldung ist 3; der ICMP-Code ist abhängig vom Grund der Nichterreichbarkeit. Dabei enthält das ICMP-Datenfeld den IP-Header und 64 Bit des original gesendeten IP-Datagramms. Folgende ICMP-Codes sind für *destination unreachable* nach Tabelle 8.5-2 definiert:

Tab. 8.5-2: *ICMP-Codes für destination unreachable*

ICMP Code	Bedeutung
0	network unreachable
1	host unreachable
2	protocol unreachable
3	port unreachable
4	fragmentation necessary but don't fragment bit is set
5	source route unreachable

8.5.4 Internet Group Management Protocol

Die ersten wichtigen *Multicast*-Applikationen wurden als MBONE-Applikationen (*Multicasting Backbone*) eingeführt, das heute noch als Pionier-System für *Multicast* weiterentwickelt wird und auf dem ***I**nternet **G**roup **M**anagement **P**rotocol* (**IGMP**) (RFC 1112) basiert. Bereits bei der Betrachtung von IP-Adressen wurde die Verwendung von IP-Multicast-Adressen in der Klasse D eingeführt. Das IGMP-Protokoll wird zur Verwaltung von Gruppenadressen eingesetzt, wobei alle Rechner einer Multicast-Gruppe auf die Multicast-Adresse dieser Gruppe reagieren. Die Informationen über die Multicast-Gruppen sind für die Multicast-Router erforderlich, die wissen müssen, welche Multicast-Datagramme auf welche Interfaces weiterzuleiten sind. Für das Funktionieren von IP-Multicast müssen auf dem Weg zwischen Sender und Empfänger alle Netzknoten IGMP unterstützen. TCP/IP-fähige Rechner und Router werden in **drei Multicast-Klassen** eingeteilt:

drei Multicast-Klassen

- *Level 0*-Geräte unterstützen das IP-Multicast nicht und werden durch IP-Multicast nicht behindert.
- *Level 1*-Geräte unterstützen die IP-Multicastdienste nur teilweise (*resource location* und *status reporting*). Sie können nicht dynamisch an so genannten *Host Groups* teilnehmen.
- *Level 2*-Geräte unterstützen IP-Multicast vollständig. Diese Geräte sind in der Lage dynamisch an *Host Groups* teilzunehmen und den *Host Groups* Multicast-Datagramme zu senden. Hierfür ist die Implementierung von IGMP notwendig.

Eine IGMP-Meldung wird im *Protocol Identity*-Feld eines IP-Datagramms angezeigt und im Datenfeld übermittelt. In Abb. 8.5-8 ist das Format eines IGMP-Pakets ohne den IP-Datagramm-Header dargestellt. Die IGMP-Meldung besteht aus foldenden Feldern:

Version: Das 4-Bit-Versionsfeld gibt die Version des IGMP-Pakets an.

Type: Das 4-Bit-Typfeld unterscheidet zwischen zwei Typen: »Typ 1« bezeichnet eine *Query* (Abfrage), die von einem Multicast-Router gesendet wird; »Typ 2« ist eine *Response* (Antwort), die von einem Host gesendet wird.

unused: Das 8-Bit-Feld wird nicht benutzt.

Checksum: Die 16-Bit-Prüfsumme wird über das komplette IGMP-Paket und der 32 Bit langen Multicast-Gruppenadresse gebildet.

Multicast Group Address: In diesem 32-Bit-Feld wird die Gruppenadresse – Klasse-D-IP-Adresse – angegeben. Handelt es sich um den IGMP-Type *Query* wird die Gruppenadresse auf 0 gesetzt, und bei einer *Response* steht die Gruppenadresse, die zu melden ist.

<table>
<tr><th colspan="2">Byte 1</th><th>Byte 2</th><th>Byte 3</th><th>Byte 4</th></tr>
<tr><td>Version
(4 Bit)</td><td>Type
(4 Bit)</td><td>unused</td><td colspan="2">IGMP Checksum</td></tr>
<tr><td colspan="5">Multicast Group Address
(32 Bit)</td></tr>
</table>

Abb. 8.5-8: Format eines IGMP-Pakets

IGMP-Reports und Queries

Multicast-Router benutzen IGMP-Meldungen, um in jedem der physisch angeschlossenen Netze die Gruppenmitgliedschaften zu überwachen. Dabei wird die Überwachung über IGMP-*Reports* und IGMP-*Queries* geregelt:

1. Ein Host sendet ein IGMP-*Report*, wenn er einer Gruppe beitritt.
2. Ein Host sendet keinen *Report*, wenn er eine Gruppe verlässt. Empfängt er allerdings eine *Query* (siehe Punkt 3), so meldet er die Gruppe nicht mehr.
3. Um festzustellen, ob Hosts irgendwelchen Gruppen angehören, senden *Multicast-Router* regelmäßig IGMP-*Queries* an die so genannte »*all-hosts-group*« mit der Adresse 224.0.0.1.
4. Auf ein ankommendes IGMP-*Query* antwortet ein Host für jede Gruppe, die er noch angehört, mit einem IGMP-*Report*.

Der *Multicast-Router* speichert mit Hilfe der *Queries* und *Reports* eine Tabelle der eigenen Interfaces, die einen Host oder mehrere Hosts als Mitglieder von Multicast-Gruppen führen. Geht ein weiterzuleitendes Multicast-Datagramm bei einem Router ein, so leitet er das Datagramm auf die Schnittstellen weiter, über die die Hosts zu erreichen sind.

8.5.5 Resource Reservation Protocol

Das ***R**esource **R**eservation **P**rotocol* (**RSVP**) (RFC 2205) wurde ursprünglich eingeführt, um für verzögerungssensitive Daten wie Sprach- oder Videodaten sowie wichtige Steuerungs- und Signalisierungsinformationen in jedem Router zwischen Sender und Empfänger Mindestbandbreite, Verzögerungsobergrenzen und Pufferspeicherplatz fest zu allokieren, um für ausgewählte Dienste gewisse Dienstgüten (**Quality of Service - QoS** (RFC 1633)) garantieren zu können. Bei jeder Reservierung wird in einem *call admission*-Prozess geprüft, ob noch ausreichend Ressourcen in jedem Router zwischen Sender und Empfänger für die anfordernde Verbindung zur Verfügung stehen. Das impliziert prinzipiell auch eine Vermischung des verbindungslosen IP mit Mechanismen, die von virtuellen Verbindungen (***P**ermanent **V**irtual **C**hannel* (**PVC**) oder ***S**witched **V**irtual **C**hannel* (**SVC**)) in anderen Protokollen (z. B. ATM) bekannt sind. Für jede RSVP-Verbindung werden die angeforderten Ressourcen reserviert und stehen für den übrigen Verkehr nicht zur Verfügung. Mit einem *policy control*-Prozess wird anschließend geprüft, ob die vereinbarten QoS-Parameter eingehalten werden.

Quality of Service

Byte 1		Byte 2	Byte 3	Byte 4
Version (4 Bit)	Flags (4 Bit)	Message Type (8 Bit)	RSVP Checksum (16 Bit)	
Send TTL (8 Bit)		reserved	RSVP Length (16 Bit)	
RSVP Message Objects				

Abb. 8.5-9: Format eines RSVP-Pakets

Die Struktur eines RSVP-Pakets ist in Abb. 8.5-9 dargestellt. Die Elemente und Funktionen der RSVP-Objekte sind ausführlich in RFC 2205 dargestellt. Grundsätzlich kann RSVP auf IPv4- und IPv6-Netzen arbeiten. Wichtig ist, dass RSVP immer in Einwegrichtung (*simplex*) betrieben wird und ein anfordernder Sender die Reservierung nur von sich in Richtung Empfänger vornimmt. Der Empfänger muss den gleichen Reservierungsvorgang in umgekehrter Richtung durchführen, wobei durch die verbindungslose Struktur von IP nicht sichergestellt ist, dass die Pakete in Rückrichtung auf dem gleichen Weg laufen.

Interessant ist RSVP im Zusammenhang mit ATM-Netzen, da hier anhand der ATM-Charakteristiken *Quality of Services* schon inherent sind. Wesentliche Aufgabe der Protokollimplementierungen ist hier die Anpassung (*mapping*) der RSVP-QoS auf die entsprechenden ATM-QoS. Leider existieren heute dafür herstellerspezifische Implementierungen, was aber durch die Standardisierung innerhalb der IETF im Zusammenhang mit *Multiprotocol Label Switching* vereinheitlicht wird.

8.5.6 Multiprotocol Label Switching

Seit Jahren diskutieren die Hersteller von IP-Equipment um den ersten Standard für das ***M**ulti**p**rotocol **L**abel **S**witching* (**MPLS**). Der jahrelange Streit um die Vorherrschaft in der IP *Switching*-Technologie, eine der wichtigsten Technologien für zukünftige IP-Netzwerke, zeigt, wie wichtig diese Technologie für die Hersteller geworden ist, und verunsichert zunehmend Anwender, die bislang nur auf spezifische Implementierungen zurückgreifen können. MPLS wird als die wichtigste Methode angesehen, die Technologien IP und ATM funktional zu einer Netzschicht zusammenzufassen. Das bedeutet, dass die OSI-Schichten 2 (*paket switching*) und 3 (*IP routing*) zu einer insgesamt leichteren Schicht zusammengefasst werden und direkt auf Übertragungssnetze einschließlich SDH- und Wellenlängen-Multiplexnetze (***W**avelength **D**ivision **M**ultiplexing* - **WDM**) zugreifen.

Grundsätzlich geht es darum, so genannte IP-*Flows* zu identifizieren, also aufeinander folgende IP-Datagramme, die zu einer Applikationssitzung zwischen definiertem Sender und Empfänger gehören, und für diese einen virtuellen Pfad (*virtual paths*) zu bestimmen, in dem ATM-ähnliche QoS-Parameter garantiert werden können. Für die Umsetzung werden **Integrated Switch Router** (**ISR**) eingesetzt, die je nachdem, ob der Hersteller starke Produkte in ATM- oder Router-Technik entwickelt, die jeweiligen Vorteile und Nachteile zu überbrücken versuchen. Meistens bestehen die ISR aus:

Integrated Switch Router

- einer schnellen Switching-Hardware, ähnlich ATM-Zell-Switches. Die Adressierungsbereiche für ATM und MPLS dürfen jedoch nicht überlappen,
- einem Switch-Controller, der die Switching-Hardware analog einem ATM-Switch steuert und mit anderen MPLS-Switches zusammenarbeitet,
- einem expliziten oder virtuellen IP-Router, der für die Umsetzung (*mapping*) von IP-Adressen auf ATM-Adressen oder die Adaption von virtuellen Pfaden (*virtual paths*) an IP-Adressen verantwortlich ist.

Vorteil des MPLS

Der **Vorteil des MPLS** liegt in der sehr schnellen Übertragung von Nutzdaten über OSI-Schicht 2, ohne dass für jedes IP-Datagramm eines IP-Flows die IP-Adresse bearbeitet werden muss. Stattdessen wird zu Beginn eines IP-Flows auf OSI-Schicht 2 der virtuelle Pfad durch das Netz festgelegt und alle IP-Datagramme werden durch diesen virtuellen Pfad übertragen. In MPLS werden so in Zukunft die hoch performanten Eigenschaften des *Switching* auf OSI-Schicht 2 mit den extrem robusten Eigenschaften des IP-Routing auf OSI-Schicht 3 miteinander verschmolzen.

8.5.7 IP-Routing

Eine wesentliche Aufgabe des IP-Protokolls ist die Weiterleitung der IP-Datagramme vom Sender zum Empfänger, welcher singulär sein kann (*unicast*) oder aus mehreren adressierten Hosts bestehen kann (*multicast*). Anhand der *destination address* im IP-Datagramm-Header muss ein Router bestimmen, auf welchem Port das IP-Datagramm weitergeleitet (*forwarding*) werden muss. Ein Router kennt dabei nur den nächsten Router in Richtung Empfänger, den sogenannten *next*

hop. Allgemein nennt man das in TCP/IP-Netzen eingesetzte Routingverfahren auch *hop-by-hop*. Die Konstruktion der dazu erforderlichen Routing-Tabellen und die dafür eingesetzten Protokolle und Algorithmen beeinflussen wesentlich die Leistungsfähigkeit eines IP-Netzes.

Kommt ein IP-Datagramm an einem Router an, wird in der Routing-Tabelle der Eintrag gesucht, der anzeigt, in welche Richtung das IP-Datagramm mit seiner spezifischen *destination address* als Host-, Netz- oder Defaultadresse weiterzuleiten ist. Die Erstellung der Routing-Tabelle kann statisch (feste Tabelleneinträge) oder dynamisch (aktualisierbare Tabelleneinträge) erfolgen, wobei beim **statischen Routing** Netzveränderungen und Fehler nur schwer zu berücksichtigen sind. Deshalb wird statisches Routing in aller Regel nur für kleine Netze mit einfacher Topologie eingesetzt. Beim **dynamischen Routing** werden durch Steuerungsmeldungen kontinuierlich Informationen über die nächsten Nachbarn (*next hop*), die Verfügbarkeit von Verbindungen und die automatische Anpassung im Fehlerfall (*backup*) sowie die Überlastung (*congestion*) einzelner Strecken gesammelt und ausgewertet.

statisches Routing

dynamisches Routing

Beispiel:
Empfängt ein Router einen Rahmen, so extrahiert er das Datagramm aus seinem Rahmen und untersucht den *Header* des IP-Datagramms. Er verarbeitet das Headerfeld, dekrementiert den TTL-Wert und errechnet die Headerprüfsumme von Neuem. Der Router betrachtet die Zieladresse des Datagramms und entscheidet, ob sie eine lokale Adresse ist. Befindet sich die Zieladresse im lokalen Netz, so wendet der Router ARP (siehe Abschnitt 8.4.2) an, um die Adresse der Netz-Interface-Schicht des Zielhosts zu bestimmen und dann das Datagramm verkapselt in einen Rahmen der Netz-Interface-Schicht direkt zum Zielhost zu verschicken.

Wenn sich das Ziel nicht lokal in einem Netz zum Router befindet, so leitet der Router das Datagramm zu einem anderen Router weiter. Diese Prozedur wird so lange weiter fortgeführt, bis das Datagramm das Zielnetz erreicht. Jeder Router aktualisiert sowohl den TTL-Wert als auch die Headerprüfsumme. Dazwischenliegende Router können Werte für *Identification* und *Segmentation Offset* ändern, wenn es notwendig ist, ein Datagramm zwischen Sender und Empfänger zu fragmentieren.

Allgemein werden bei **Routing-Algorithmen** entsprechend den graphentheoretischen Grundlagen den Strecken und Distanzen im Netz »Kosten« zugeordnet, die den Ressourcenverbrauch wie belegte Bandbreite oder Laufzeit messen [KAP95]. Routing-Protokolle versuchen, die Kosten im Netz zu minimieren. Bei TCP/IP-Netzen wird zwischen zwei Klassen von Algorithmen für Routing-Protokolle unterschieden, dem ***D**istance **V**ector **A**lgorithm* (**DVA**) und dem ***L**ink **S**tate **A**lgorithm* (**LSA**). Man unterscheidet interne (***I**nterior **G**ateway **P**rotocol* - **IGP**), externe (***E**xterior **G**ateway **P**rotocol* - **EGP**) und **Multicast-Routing-Protokolle**. In Tabelle 8.5-3 sind einige gebräuchliche Routing-Protokolle zusam-

Routing-Algorithmen

IGP
EGP
Multicast-Routing-Protokolle

mengefasst.

Tab. 8.5-3: *Auswahl einiger gebräuchlicher Routing-Protokolle*

Protokoll	Name / Bedeutung	Anwendung
OSPF	Open Shortest Path First	IGP
IS-IS	Intermediate System to Intermediate System	IGP
RIP	Routing Information Protocol	IGP
IGRP	Interior Gateway Routing Protocol	IGP
ERP	Exterior Routing Protocol	EGP
BGP	Border Gateway Protocol	EGP
IDRP	Inter Domain Routing Protocol	EGP
DVMRP	Distance Vector Multicast Routing Protocol	Multicast
PIM	Protocol Independent Multicast	Multicast

Autonome Systeme

IP-Netze sind in sogenannte **Autonome Systeme** (***A****utonomous* ***S****ystem* - **AS** (RFC 1930)) segmentiert, wobei ein AS aus einer Anzahl von Routern mit gemeinsamer Verwaltung besteht, in der eigene Regeln und Routing-Protokolle festgelegt sind. Nach außen zu anderen AS verhält es sich wie ein geschlossenes System, in das andere Netzadministratoren nicht eingreifen können. Intern werden IGP-Protokolle eingesetzt; zur Trennung von anderen AS werden EGP-Protokolle verwendet. Durch die Trennung in AS und somit in viele Routing-Domänen wird der Umfang der Aktualisierungen beim dynamischen *Routing* auf ein verarbeitbares Maß reduziert.

Selbsttestaufgabe 8.5-2:

Wozu ist das TTL-Feld in einem IP-Datagramm da? Was bedeutet ein TTL-Wert von 1?

Selbsttestaufgabe 8.5-3:

Eine ICMP-Meldung enthält im Typfeld den Wert 3 und im Codefeld den Wert 1. Erklären Sie die Bedeutung der Meldung.

8.5.8 Internet Protocol Version 6

Dieses Kapitel stellt die wichtigsten Änderungen des ***I**nternet **P**rotocol **V**ersion **6*** (**IPv6**) gegenüber der Version IPv4 dar und gibt einen Überblick über die wesentlichen Eigenschaften des Protokolls. Insbesondere sind dies die neuen Adress- und Datenformate, die Unterstützung von Dienstgüte und Sicherheitsaspekte, sowie Strategien zur Migration von IPv4 nach IPv6.

8.5.8.1 Entwicklung

Der steigende Bedarf an IP-Adressen im industriellen, aber auch im privaten Bereich führte in den letzten Jahren zu einem Engpass im IP-Adressraum. Zeitgleich fand das ***C**lassless **I**nter **D**omain **R**outing* (**CIDR**) mit einem gestiegenen Aufwand für die Verwaltung des Routing immer mehr Verbreitung. In den frühen neunziger Jahren erkannte die ***I**nternet **E**ngineering **T**ask **F**orce* (**IETF**) diese Probleme, so dass man sich mit dem ***R**equests **f**or **C**omments* (**RFC**) 2460 für eine neue Version des Internet Protokolls entschied, dem ***I**nternet **P**rotocol **V**ersion **6*** (**IPv6**).

Das neue Internet Protocol Version 6 – auch ***I**nternet **P**rotocol **N**ext **G**eneration* (**IPng**) genannt – wurde sowohl für den Einsatz in Hochgeschwindigkeitsnetze (z. B. Gigabit Ethernet) als auch für den Einsatz in Netze mit niedriger Bandbreite (z. B. Wireless LAN, Bluetooth, usw.) konzipiert.

Neuerungen bei IPv6

IPv6 besitzt im Gegensatz zu IPv4 folgende Merkmale:

Erweiterte Adressierung: IPv6 vergößert die IP-Adresslänge von 32 Bit auf 128 Bit, um die Anzahl der Ebenen der Adressierungshierarchie zu erweitern, eine größere Anzahl von adressierbaren Knoten und eine einfache Autokonfiguration von Adressen zu ermöglichen. Bei einer Länge von 128 Bit ergeben sich $2^{128} = 3,4 \cdot 10^{38}$ Adressen. Auf die Erdoberfläche bezogen bedeutet dies, dass IPv6 je Quadratmillimeter 667 Billiarden Adressen bereit stellt.

Vereinfachtes Header-Format: Einige im IPv4-Header enthaltene Felder fallen weg. Dadurch werden die Verarbeitungskosten der Datenpakete gesenkt und die Bandbreitekosten für den IPv6-Header begrenzt.

Erweiterungs-Header: Die Kodierung der Headeroptionen wurde geändert und erlaubt damit eine effizientere Weiterleitung, keine Längenbeschränkung der Optionen und die Flexiblität, neue Optionen einzuführen.

Flusskennung: Die neue Version bietet die Möglichkeit, Pakete zu kennzeichnen, die zu einem Datenfluss gehören. Dies können Pakete sein, für die der Anwender eine spezielle Behandlung (z. B. Echtzeit-Dienst) oder eine über den Standard hinausgehende Dienstgüte (***Q**uality **o**f **S**ervice* - **QoS**) verlangt.

Sicherheitsaspekte: IPv6 unterstützt eine Basismenge von Funktionalitäten, die Authentizität, Integrität und Vertraulichkeit gewährleisten.

8.5.8.2 Das Protokoll: IPv6

IPv6-Basis-Header
IPv6-Erweiterungs-Header

IPv6 ändert das Format der IP-Datagramme, deren Header aus zwei Teilen besteht: **IPv6-Basis-Header** und optionale **IPv6-Erweiterungs-Header**. Abb. 8.5-10 zeigt den Aufbau des Basis-Header.

Abb. 8.5-10: Format des IPv6-Header

Die Funktionen der einzelnen Felder des IPv6-Basis-Header sind:

- **Version** (4 Bit) gibt die Version des IP-Protokolls an: Version 6.
- **Traffic Class** (4 Bit) wird zur Kennzeichnung und Unterscheidung zwischen verschiedenen Verkehrsklassen oder Prioritäten von IPv6-Datagrammen eingesetzt. Für Anwendungen ohne Echtzeit-Anforderungen, wie News, E-Mail, Datenbankzugriff usw. werden die Prioritäten 0 bis 7 und für Anwendungen mit Echtzeit-Anforderungen werden die Prioritäten 8 bis 15 benutzt (siehe Abschnitt 8.5.8.5).
- **Flow Label** (24 Bit) wird zur Identifikation von IP-Flows, also aufeinanderfolgenden IP-Datagrammen, die alle zur gleichen Applikation eines Anwenders gehören, eingesetzt, um diese gesondert zu behandeln. Der Wert des Flow Label-Feldes wird durch eine Zufallszahl gebildet und enthält den Wert Null, wenn keine gesonderte Behandlung vorgesehen ist. Ein typisches Beispiel für einen IP-Flow ist ein Audio- oder Videodatenstrom einer Videokonferenz (siehe Abschnitt 8.5.8.5).
- **Payload Length** (16 Bit) bestimmt im Gegensatz zu IPv4 nicht die Länge des Datagramms, sondern die Länge des Pakets nach dem Header (maximal 65536 Byte).
- **Next Header** (16 Bit) spezifiziert den Typ des Header nach dem IPv6-Basis-Header. IPv6 setzt zwar dieselben Werte wie im IPv4-Protokoll-Feld ein (Code 6 für TCP und Code 17 für UDP), sieht aber zusätzlich die Nutzung von Erweiterungs-Headern vor, die im Feld *Next Header* nur dann mit einbezogen werden, wenn sie gebraucht werden.
- **Hop Limit** (8 Bit) beschreibt die Lebensdauer des Pakets. Er wird an jedem Knoten, der das Paket weiterleiten soll, um eins dekrementiert. Bei Null wird das Paket verworfen.

- **Source Address** (128 Bit) ist die IP-Adresse des Senders.
- **Destination Address** (128 Bit) ist die IP-Adresse des Empfängers.

8.5.8.3 Erweiterungs-Header

Die Optionen in IPv4, die eine gesonderte Bearbeitung des Datagramms in Routern oder im Endknoten erfordern, werden bei IPv6 durch sogenannte Erweiterungs-Header ersetzt. Bis auf wenige Ausnahmen werden diese Erweiterungs-Header nicht in den Routern entlang eines Pfades bearbeitet, sondern nur in den Endsystemen ausgewertet.

Die Erweiterungs-Header werden zwischen dem IPv6-Basis-Header und dem *Payload* (Nutzdatenteil) platziert und sind optional (Abb. 8.5-11).

Abb. 8.5-11: IPv6-Datagramm mit mehreren Erweiterungs-Header

Aus Abbildung Abb. 8.5-11 ist ersichtlich, dass ein IPv6-Datagramm beliebig viele Erweiterungs-Header enthalten kann. Allerdings besteht die Einschränkung, dass jede Erweiterung höchstens einmal in einem Datagramm vorkommen darf. Um die Leistungsfähigkeit für die Bearbeitung der nachfolgenden Erweiterungs-Header zu steigern, ist jeder dieser Erweiterungs-Header ein Vielfaches von 8 Byte lang.

Jeder Erweiterungs-Header gibt den Typ (*Next Header*) und den Abstand (*Header Extension Length*) zum Nachfolgenden an. Der *Next Header* wird als Codewert in RFC 1700 definiert, wobei diese in den meisten Fällen für IPv4 und IPv6 übereinstimmen (siehe Tabelle 8.5-4).

Tab. 8.5-4: *Codes für Erweiterungs-Header*

Code	Typ des Header
0	Hop-By-Hop Option
1	ICMP für IPv4
2	IGMP für IPv4
4	IPv4-Einkapselung
6	TCP
17	UDP
43	Routing Header
44	Fragment Header
50	Encapsulation Security Payload
51	Authentication Header
59	No Next Header
60	Destination Option

Bei der Verkettung der Erweiterungs-Header ist eine bestimmte Reihenfolge zu beachten, die nach RFC 2460 wie folgt festgelegt ist:

IPv6-Basis-Header,

Hop-by-Hop Option Header,

Destination Option Header[8],

Routing Header,

Fragment Header,

Authentication Header,

Encapsulation Security Payload Header,

Destination Option Header,

Upper Layer Header.

Hop-by-Hop Option Header

Der **Hop-by-Hop Option Header** wird, falls vorhanden, von allen Knoten, die ein IPv6-Datagramm auf dem Weg von der Quelle zum Ziel durchläuft, ausgewertet. Dieser Erweiterungs-Header wird vor allem eingesetzt, um Management- oder Debugging-Funktionen durchzuführen, bestimmte Verkehrsströme zwischen 2 Endknoten mit Hilfe des ***R**esource Re**s**er**v**ation **P**rotocols* (**RSVP**, siehe Abschnitt 8.5.5) gesondert zu behandeln oder um Pakete zu übertragen, die größer als 64 KB sind. Abb. 8.5-12 zeigt den Aufbau eines Hop-By-Hop Option-Header, wobei *Next Header* den Typ nach Tabelle 8.5-4 und *Header Extension Length* den Abstand zum nachfolgenden Extension-Header angibt. *Option Type* kennzeichnet die Option nach RFC 2460 und *Option Data Length* gibt die Anzahl der Bytes im Feld *Option Data* an.

Abb. 8.5-12: Hop-by-Hop-Header

Routing Header

Der **Routing Header** wird für *Source Routing* in IPv6 verwendet. Er dient zur Festlegung einer Reihe von Router bzw. eines Topologiebereichs, die auf dem Weg eines Datagramms zum Zielknoten passiert werden sollen. Router, die in der Liste der zu besuchenden Router stehen, müssen diesen Erweiterungs-Header bearbeiten. Zwei Varianten des *Routing Header* sind in RFC 2460 aufgeführt; zurzeit ist aber nur die Variante Routing-Typ 0 von Bedeutung, wobei die andere Variante für eventuelle Erweiterungen vorgesehen ist. Abb. 8.5-13 zeigt den Aufbau eines *Routing Header*. Im Routing Type-Feld wird die Art der Bearbeitung festgelegt. Der Wert null gibt

8 Der Destination Option Header ist der einzige Erweiterungs-Header, der in einem IP-Header zweimal vorkommen darf. Steht er vor dem Routing Header, so muss er von den im Routing Header aufgelisteten Routern bearbeitet werden, andernfalls muss er nur vom Zielknoten ausgewertet werden.

an, dass die in der Adressliste aufgeführten Knoten auf jeden Fall besucht werden müssen. *Segments Left* beinhaltet die Anzahl der Knoten, die das Datagramm auf dem Weg zum Ziel noch durchlaufen muss. Address[1] bis Address[n] stellt die Adressliste dar.

Abb. 8.5-13: Routing Header (Typ 0)

Die Fragmentierung von IPv6-Datagrammen wird im Gegensatz zu IPv4 mit dem **Fragment Header** realisiert, wenn das Datagramm zu groß für die **MTU** (***M**aximum **T**ransfer **U**nit*) des Netzes ist. Allerdings findet die Fragmentierung nur am Quellknoten statt, d. h. die zwischen Sender und Empfänger liegenden Router sind an der Fragmentierung nicht beteiligt. Der Sender kann entweder die garantierte, minimale MTU von 1280 Byte oder die während des Path MTU Discovery-Prozesses (RFC 1981) ermittelte Mindest-MTU im Pfad nutzen.

Fragment Header

Beispiel: Path MTU Discovery-Prozess
Nach der unteren Abbildung sendet Host A ein IPv6-Datagramm der Länge $3000\,Byte$ zu Host B. Das Datagramm passiert über den Router R_1 den *Link*$_1$ (Verbindung zwischen Host A und Router R_1 mit der Link-MTU von $4464\,Byte$). Am Router R_1 angekommen, wird es über den *Link*$_2$ (ebenfalls mit der Link-MTU von $4464\,Byte$) bis zum Router R_2 übertragen. Am Router R_2 wird nun die ICMP-Nachricht *message too big* erzeugt und zu Host A gesendet, da die Link-MTU von *Link*$_3$ kleiner als die Datagrammgröße ist ($1500\,Byte < 3000\,Byte$). Auf der Grundlage der in der ICMP-Nachricht *message too big* enthaltenen Link$_3$-MTU von $1500\,Byte$ wird nun die *Path MTU* (Pfad-MTU) von $1500\,Byte$ angenommen und das ursprüngliche Datagramm von Host A auf die Größe von $1500\,Byte$ fragmentiert und erneut zum Host B gesendet. Nachfolgende Datagramme können nun – falls notwendig – mit der bekannten *Path MTU* fragmentiert werden.

Bei der Fragmentierung der IPv6-Datagramme ist zwischen dem nicht fragmentierbaren Teil (IPv6-Basis-Header und falls vorhanden *Hop-by-Hop Header*, *Destination Option* und *Routing Header*)[9] und dem fragmentierbaren Teil des Datagramms (der Rest des IPv6-Datagramms) zu unterscheiden. Abb. 8.5-14 zeigt schematisch die Fragmentierung eines IPv6-Datagramms mit fragmentierbarem und nicht fragmentierbarem Bereich.

Abb. 8.5-14: Fragmentierung

9 Da diese Erweiterungs-Header von Router bearbeitet werden, dürfen diese nicht fragmentiert werden.

Abb. 8.5-15 stellt den Aufbau eines *Fragment Header* dar. Dabei gibt das Fragment Offset-Feld den *Offset* des jeweiligen Fragments bezogen auf den Datagrammanfang in Vielfachen von 8 Bytes an. Die Felder *Reserved* und *Res* sind reserviert und werden vor der Übertragung mit dem Wert Null initialisiert. Ist das M-Feld (*More Fragment Bit*) gesetzt, so folgen noch weitere Fragmente. Andernfalls handelt es sich um das letzte Fragment. Das Identification-Feld kennzeichnet die zusammenhängenden Fragmente eines Datagramms eindeutig.

Next Header (8 *Bit*)	Reserved (8 *Bit*)	Fragment Offset (13 *Bit*)	Res (2 *Bit*)	M (1 *Bit*)
Identification				

Abb. 8.5-15: Fragment Header

Authentication Header

Encapsulating Security Header

IPv6 bietet über zwei integrierte Erweiterungs-Header – **Authentication Header** und **Encapsulating Security Header** – Sicherheitsdienste an, die Authentizität, Integrität und Vertraulichkeit gewährleisten. Für die Nutzung der Sicherheitsdienste müssen sich die Kommunikationspartner untereinander über die *Security Association* abstimmen. Eine *Security Association* besteht aus der Kombination einer Empfängeradresse und eines ***S**ecurity **P**arameters **I**ndex* (**SPI**), der unter anderem folgende Parameter beschreibt:

Verschlüsselungs- und Authentifizierungsalgorithmen,

Schlüssel für Authentifizierung/Verschlüsselung,

Lebensdauer der Schlüssel,

Lebensdauer der *Security Association*,

Senderadresse und

Sicherheitsstufe der Kommunikation z. B. »streng geheim«, »geheim«, usw.

Die Verfahren zur Verwaltung der Schlüssel werden im ***I**nternet **S**ecurity **A**ssociation and **K**ey **M**anagement **P**rotocol* (**ISAKMP**, RFC 2408) und das Protokoll zum Austausch der Schlüssel im ***I**nternet **K**ey **E**xchange* (**IKE**, RFC 2409) beschrieben und beruhen auf dem von W. Diffie und M. E. Hellmann 1976 vorgeschlagenen Verfahren zum Austausch von Schlüsseln über unsichere Strecken.

In Abb. 8.5-16 ist der *Authentication Header* dargestellt, der die Integrität und Authentizität der IP-Datagramme sichert.

Abb. 8.5-16: Authentication Header

Das Authentication Data-Feld enthält Authentifizierungsdaten, deren Inhalt vom verwendeten Sicherheitsalgorithmus abhängt. Aus Interoperabilitätsgründen muss jede IPv6-Implementierung den **M**essage **D**igest **5** (**MD5**)-Algorithmus unterstützen. *Sequence Number* dient dazu, dass Pakete mit identischen Daten nicht wiederholt gesendet werden können, indem jedesmal bei der Berechnung der *Authentication Data* die sich ändernde *Sequence Number* mit einbezogen wird. Dadurch können so genannte Replay-Attacken aufgedeckt werden, bei denen Angreifer die Pakete kopieren und zu einem späteren Zeitpunkt erneut senden könnten.

Die Vertraulichkeit der IP-Datagramme wird durch den ***E**ncapsulation **S**ecurity **P**ayload* (**ESP**) *Header* gesichert, um einen abhörsicheren Datenaustausch zu ermöglichen. Hierzu wird der *Payload* des IP-Datagramms oder die Pakete der Transportschicht (z. B. User Datagramm Protocol- oder Transmission Control Protocol-Pakete) in ein ESP eingekapselt und dann verschlüsselt. Aus Abb. 8.5-17 wird deutlich, dass neben dem Payload Data-Feld, das die Nutzlast enthält, die Felder *Padding*, *Pad Length* und *Next Header* auch verschlüsselt werden. IPv6 überlässt die Wahl des Verschlüsselungsverfahrens den Anwendern; aus Gründen der Interoperabilität muss allerdings der ***D**ata **E**ncryption **S**tandard* (**DES**) mit ***C**ipher **B**lock **C**haining* (**CBC**) unterstützt werden.

Abb. 8.5-17: Encapsulation Security Payload Header

8.5.8.4 IPv6-Adressierung

In IPv6 wurde die Länge der IP-Adressen auf 128 Bit vergrößert und ihre Struktur für ein optimales Routing und einen flexiblen Aufbau von Adresshierarchien überarbeitet. Die IPv6-Adressen können in Unicast-, Anycast und Multicast-Adressen klassifiziert werden. Eine Unicast-Adresse identifiziert einen einzelnen Netzanschluss eines Routers oder eines Hosts. Anycast-Adressen dagegen identifizieren Gruppen von Netzanschlüssen. Ein Datagramm, dass an eine Anycast-Adresse verschickt wurde, wird nur an ein Mitglied dieser Gruppe zugestellt. Eine Multicast-Adresse schließlich stellt einen Identifikator für eine Gruppe von Netzanschlüssen dar, bei der ein Datagramm im Gegensatz zu einer Anycast-Adresse an alle Mitglieder der Gruppe geliefert wird.

Die Adresstypen werden nach RFC 2373 im so genannten ***F**ormat **P**refix* (**FP**) festgelegt, das aus den führenden Bits einer IPv6-Adresse besteht. Die Tabelle 8.5-5 zeigt die Zuordnung der Präfixe zu Adresstypen.

Tab. 8.5-5: *Einteilung der IPv6-Adressen nach Typen*

Adresstyp	Präfix (binär)	Teil des Addressraums
Reserviert (IPv4-Kompatibilität)	0000 0000	1/256
Nicht vergeben	0000 0001	1/256
NSAP-Adressen	0000 001	1/128
IPX-Adressen	0000 010	1/128
Nicht vergeben	0000 011	1/128
Nicht vergeben	0000 1	1/32
Nicht vergeben	0001	1/16
Aggregatable Global Unicast-Adressen	001	1/8
Nicht vergeben	010	1/8
Nicht vergeben	011	1/8
Nicht vergeben	100	1/8
Nicht vergeben	101	1/8
Nicht vergeben	110	1/8
Nicht vergeben	1110	1/16
Nicht vergeben	1111 0	1/32
Nicht vergeben	1111 10	1/64
Nicht vergeben	1111 110	1/128
Nicht vergeben	1111 1110 0	1/512
Link-Local Unicast-Adressen	1111 1110 10	1/1024
Site-Local Unicast-Adressen	1111 1110 11	1/1024
Multicast-Adressen	1111 1111	1/256

Beispiel:
Mit der Erweiterung der Adresslänge auf 128 Bit wurde eine neue Darstellung für IPv6-Adressen eingeführt, mit der sich lange Zahlenreihen kompakter darstellen lassen und somit die Lesbarkeit erhöhen.

Bei der neuen Darstellungsform werden acht 16-Bit-lange Blöcke gebildet und jeweils mit einem Doppelpunkt voneinander getrennt. Da sich jede Hexadezimalziffer durch 4 Bits darstellen lässt, werden die 16 Bits in jedem Block durch 4 Hexadezimalzahlen ersetzt (siehe Abbildung unten).

Eine weitere Vereinfachung in der Darstellung von IPv6-Adressen ergibt sich, indem eine Nullgruppe »0000« mit zwei aufeinanderfolgenden Doppelpunkten gekürzt wird. Die Adresse

$CDEF : 89AB : 0000 : 7100$ wird vereinfacht zu

$CDEF : 89AB :: 7100.$

Dabei ist zu beachten, dass in einer Adresse zwei aufeinanderfolgende Doppelpunkte aus Eindeutigkeitsgründen nur einmal vorkommen darf.

Aggregatable Global Unicast-Adressen

Bei IPv6 kommen den Aggregatable Global Unicast-Adressen nach RFC 2374 eine besondere Rolle zu, da sie die von IPv4 her bekannten öffentlichen Adressen weitgehend ersetzen. Für die IP-Paketzustellung und eine Adressaufteilung nach Providern ist der Adressbereich nach Abb. 8.5-18 in **Hierarchiestufen** unterteilt.

Unterteilung in Hierarchiestufen

Abb. 8.5-18: Aggregatable Global Unicast-Adresse

Das **FP** (***F**ormat **P**refix*)-Feld enthält für Aggregatable Global Unicast-Adressen den Wert 001. Das **T**op-**L**evel **A**ggregation **Id**entifier (**TLA ID**)-Feld dient zur Kennzeichnung der obersten Hierarchiestufe eines Netzbetreibers (z. B. Provider, die auf nationaler und internationaler Ebene tätig sind), wobei das Res-Feld für

zukünftige Anwendungen reserviert ist. Die nächst niedrige Hierarchiestufe eines Netzbetreibers – wie z. B. regionale Teilnetze eines großen Providers – wird durch das ***N**ext-**L**evel **A**ggregation **I**dentifier* (**NLA ID**)-Feld gekennzeichnet. Private Netzbetreiber, wie Hochschulen oder Firmen, werden schließlich durch das ***S**ite-**L**evel **A**ggregation **Id**entifier* (**SLA ID**)-Feld identifiziert. Das letzte Adressfeld *Interface ID* beschreibt das Interface des Knotens und kann lokale oder globale Eindeutigkeit besitzen.

Beispiel:
Eine mögliche Variante der Erzeugung der Interface-Kennung ist die Codierung einer ***E**xtended **U**nique **I**dentifier* (**EUI**)-64-Adresse auf der Grundlage einer 48 Bit langen ***M**edia **A**ccess **C**ontrol* (**MAC**)-Adresse. Bei diesem Verfahren wird nach Bit 23 einer MAC-Adresse ein 16-Bit langes Bitmuster eingeschoben und zusätzlich das Bit 6 invertiert (siehe Abbildung unten).

Bit 0- 23 : von der IEEE vergebene Herstellerkennung (OUI – Organisation Unique Identifier)

Bit 24- 47: vom Hersteller vergebene Bits

Link-Local und Site-Local Unicast-Adressen

IPv6 sieht neben den globalen Unicast-Adressen auch Adressen vor, die für den lokalen Einsatz bestimmt sind: Link-Local und Site-Local Unicast-Adressen. Während Link-Local Unicast-Adressen sich auf ein einzelnes physiches Netz beschränken und bei der automatischen Adresskonfiguration eingesetzt werden können, beziehen sich Site-Local Unicast-Adressen auf einen Standort (*Site*) bzw. auf eine Organisation ohne Anbindung an das Internet. Datagramme mit Site-Local Unicast-Adressen werden – im Gegensatz zu denen mit Link-Local Unicast-Adressen – von Routern weitergeleitet.

Abb. 8.5-19 zeigt den Aufbau der beiden lokalen Unicast-Adressen. Beide Adressen beinhalten die *Interface ID*, wobei die Site-Local Unicast-Adresse eine *Subnet*

ID enthält, die das Netz innerhalb der *Site* kennzeichnet. Soll die *Site* eine Anbindung zum Internet bekommen, so wird lediglich das Päfix der Site-Local Unicast-Adressen und der bisher mit Null gefüllte Teil durch die vom Provider vergebenen Adressteile ersetzt.

Abb. 8.5-19: a) Link-Local Unicast-Adresse
b) Site-Local Unicast-Adresse

Eingebettete IPv4-Adressen

IPv6-Übergangsstrategien enthalten für Hosts und Router eine Technik, um IPv6 Pakete dynamisch über IPv4-Netze zu tunneln. IPv6-Knoten, die diese Technik nutzen, bekommen für ihre Schnittstellen spezielle Unicast-Adressen – IPv4-compatible IPv6-Adressen – zugewiesen, die in den unteren 32 Bits eine IPv4-Adresse enthalten (Abb. 8.5-20).

Abb. 8.5-20: Format einer IPv4-compatible IPv6-Adresse

Eine weitere Variante einer IPv6-Adresse, die eine eingebettete IPv4-Adresse enthält, ist die so genannte IPv4-mapped IPv6-Adresse. Sie wird zur Adressierung von IPv4-Knoten benutzt, die kein IPv6 unterstützen.

Abb. 8.5-21: Format einer IPv4-mapped IPv6-Adresse

Anycast-Adressen

Anycast-Adressen werden aus dem Unicast-Adressraum vergeben und benutzen deshalb dasselbe Format wie Unicast-Adressen. Aus diesem Grund können Anycast-Adressen syntaktisch nicht von Unicast-Adressen unterschieden werden. Wenn eine Unicast-Adresse zu mehr als einer Schnittstelle zugeordnet wird, fasst man diese als eine Anycast-Adresse auf. Wird der Schnittstelle eines Knotens eine Anycast-Adresse zugewiesen, so muss der Knoten noch explizit konfiguriert werden, so dass er die zugewiesene Anycast-Adresse von der Unicast-Adresse unterscheiden kann. Eine Anycast-Adresse darf im Unterschied zu einer Unicast-Adresse nicht als Senderadresse in einem IPv6-Datagramm verwendet werden.

Beispiel:
Anycast-Adressen können dazu eingesetzt werden, Datagramme über ein Subnetz eines bestimmten Netzanbieters zu versenden, indem im Routing Header die Anycast-Adressen des Netzanbieter angegeben werden.

Weiterhin können Anycast-Adressen für replizierte Server verwendet werden, die die gleichen Internetdienste anbieten. Bei Überlastung oder Ausfall eines bestimmten Servers kann dann der replizierte Server die Dienste anbieten.

Multicast-Adressen

Für eine Gruppenkommunikation im Internet sind Multicast-Adressen notwendig. Typische Anwendungen für Multicasting Dienste sind z. B. Audio- und Videokonferenzen, *Mailgroups*, usw. Die erweiterten Anwendungsfelder von IPv6-Multicast umfassen auch Steuerungsaufgaben, die bei IPv4 notwendigen *Broadcast*s zum Auffinden von Diensten oder zur Umsetzung von Adressen ersetzen. Abb. 8.5-22 zeigt den Aufbau einer IPv6-Multicast-Adresse.

Abb. 8.5-22: Multicast-Adresse

Die ersten drei Bits im Flag-Feld sind reserviert und müssen auf Null gesetzt sein; das T-Bit im Flag-Feld gibt an, ob es sich um eine temporär zugeordnete Multicast-Adresse (T = 1) oder um eine durch die ***I**nternet **A**ssigned **N**umbering **A**uthority* (**IANA**) zugewiesene Adresse (T = 0) handelt, deren Zuordnung nicht zeitlich beschränkt ist. Im Scope-Feld wird die Reichweite, d. h. der Gültigkeitsbereich, der Multicast-Gruppe festgelegt. Die *Group ID* kennzeichnet schließlich die jeweilige Gruppe innerhalb des jeweiligen Gültigkeitsbereichs.

8.5.8.5 Quality of Service

Durch die Kennzeichnung von Prioritäten und die Anzeige von IP-Flows wird beim *Internet Protocol Version 6* die Verkehrssteuerung und die Gewährleistung von Dienstgüten (***Q**uality **o**f **S**ervice* - **QoS**) wesentlich vereinfacht. Die eintreffenden Datagramme, die ein Router im Sinne der QoS-Anforderungen speziell zu behandeln hat, werden dabei durch die Felder *Flow Label* und *Traffic Class* des IPv6-Basis-Header gekennzeichnet. Damit werden Anwendungen unterstützt, die ein gewisses Maß an stetigem Durchsatz, eine minimal zulässige Verzögerung oder *Jitter* erfordern.

Flow Labels

Unter einem *Flow* (Fluss) versteht man eine Folge von Datagrammen, die vom Sender zum Empfänger übertragen werden (*Unicast* oder *Multicast*) und für die eine Reservierung bestimmter Resourcen (wie Bandbreite, Puffer, usw.) und eine Sonderbehandlung bei der Verarbeitung in den Routern gefordert wird. Die spezielle Behandlung für Datagramme eines Flusses kann den Routern explizit mit einem Kontrollprotokoll wie dem ***R**esource **R**e**s**er**v**ation **P**rotocol* (**RSVP**, siehe Abschnitt 8.5.5) oder durch direkt in den Datagrammen enthaltenen Informationen (z. B. in den *Hop-By-Hop Option Header*) mitgeteilt werden.

Zwischen einem Sender und einem Empfänger können gleichzeitig mehrere *Flows* existieren und zusätzlich noch Datagramme übertragen werden, die nicht zu einem *Flow* gehören. Ein *Flow* wird eindeutig durch die Kombination einer Quelladresse mit einem von Null verschiedenen Flow Label-Wert identifiziert. Unterstützt ein Host oder Router keine Flow-Funktion, so wird der Flow Label-Wert von der Quelle mit Null initialisiert, wenn ein Datagramm versendet werden soll oder der Flow Label-Wert ignoriert, wenn ein ankommendes Datagramm verarbeitet werden soll.

Flow Label-Werte werden immer von der Quelle eines *Flows* gesetzt und sollten generell zufällig und uniform gewählt werden, um sie als Schlüssel in den Hash-Tabellen der Router einsetzen zu können. Alle Datagramme, die zum selben *Flow* gehören, müssen auch die gleiche Quelladresse, die gleiche Zieladresse und den gleichen Flow Label-Wert besitzen. Eventuell vorhandene *Hop-by-Hop Header* oder *Routing Header* müssen ebenfalls identisch sein. Router oder mögliche Empfänger von *Flows* können diese Bedingung überprüfen und bei einer Verletzung der Regel eine ICMP-Fehlermeldung senden.

Beispiel:
Router können auch ohne die bereits erwähnten Kontrollprotokolle oder die Informationen im Hop-by-Hop Header in den Flow-Behandlungszustand übergehen. Trifft ein Datagramm mit einem für den Router noch unbekannten und von Null verschiedenen Flow Label-Wert ein, so führt er die erforderlichen Verarbeitungsschritte durch und bestimmt z. B. das Ausgangs-Netz-Interface, aktualisiert das *Hop Limit* im IPv6-Header oder entscheidet, wie das Datagramm nach seinem Traffic Class-Wert in der Warteschlange bearbeitet werden soll.

Der Router speichert nun das Resultat dieser Verarbeitungsschritte und nimmt die Informationen in den *Cache* auf. Dabei wird der Flow Label-Wert als Schlüssel zum Hashing des Cachespeichers genutzt. Nachfolgende Datagramme des Flow, die die selbe Quelladresse und den selben Flow Label-Wert besitzen, werden nach den Informationen im *Cache* behandelt, ohne alle Felder des Header nochmal bearbeiten zu müssen.

Traffic Class

Das Traffic Class-Feld wird zur Kennzeichnung und Unterscheidung zwischen Verkehrsklassen oder Prioritäten von IPv6-Datagrammen eingesetzt. Dabei werden Datagramme von Anwendungen mit unterschiedlichen Anforderungen an das Netz (z. B. möglichst geringe Verzögerung bei Telnet oder möglichst hoher Durchsatz für *File Transfer* mit **FTP** - ***F**ile **T**ranser **P**rotocol*, usw.) bei der Durchleitung durch das Netz in den Routern nach der Traffic Class behandelt. Nach Tabelle werden für Anwendungen ohne Real-Zeit-Anforderungen die Prioritäten (Traffic Class-Werte) 0 bis 7 und für Anwendungen mit Real-Zeit-Anforderungen die Prioritäten 8 bis 15 benutzt.

Priorität	Verkehr
0	Nicht charakterisierter Verkehr
1	Füllverkehr (z. B. Netnews)
2	Verkehr unabhängiger Datenpakete (z. B. eMail)
3	Reserviert
4	Verkehr abhängiger Datenpakete von größeren Datenblöcken (z. B. FTP, HTTP, NFS)
5	Reserviert
6	interaktiver Verkehr (z. B. Telnet, X-Windows, Datenbankzugriff)
7	Internet Steuerverkehr (z. B. SNMP)
8	Videoverkehr mit hoher Wiedergabetreue
...	
15	Audioverkehr mit niedriger Wiedergabetreue

Beispiel:
Kommt es zu einer Überlastsituation in einem Router, der Datagramme mit Echtzeit-Anforderungen verarbeitet, so werden zuerst die Datagramme verworfen, die eine niedrige Priorität besitzen.

8.5.8.6 Übergangsstrategien

Das entscheidende Ziel der Übergangsstategien ist, die Kommunikation zwischen IPv4- und IPv6-Hosts zu ermöglichen. Die Übergangsstrategien sollten dabei eine einfache und inkrementelle Migration von IPv4 nach IPv6 mit möglichst geringen Abhängigkeiten ermöglichen.

Die IPv6-Übergangsstrategien beruhen auf folgenden Eigenschaften:

- Einzelne IPv4-Host oder -Router können auf IPv6 aufgerüstet werden, ohne dabei gleichzeitig andere Hosts oder Router aufrüsten zu müssen. Neue IPv6-Hosts oder -Router können nach und nach installiert werden. Dieser Prozess wird als inkrementelle Migration bezeichnet.
- Die Voraussetzung für das Aufrüsten von Hosts auf IPv6 ist, dass zuerst die ***D**omain **N**ame **S**erver* (**DNS**) auf IPv6 umgestellt werden müssen, damit diese IPv6-Adressen bearbeiten können. Das Aufrüsten der Routern ist allerdings keine Voraussetzung.
- Werden IPv4-Hosts oder -Router auf IPv6 aufgerüstet, können diese ihre alten Adressen weiterbenutzen, so dass ihnen keine neuen Adressen durch Administratoren zugewiesen werden müssen.
- Übergangsstrategien nutzen eingebettete Adressen.
- Zu Beginn der Übergangsphase werden alle auf IPv6 umgestellten Hosts und Router doppelte Protokollstacks (IPv4 und IPv6) besitzen.
- Die so genannte Tunnel-Technik wird eingesetzt, wenn z. B. zwei IPv6-Knoten über bestehende IPv4-Netze kommunizieren sollen. Hierbei werden IPv6-Datagramme beim Eintreten in IPv4-Infrastrukturen in IPv4-Datagramme gekapselt und über einen oder mehrere IPv4-Router an den nächsten IPv6-fähigen Knoten übertragen.
- Header-Übersetzungstechniken erlauben die Einführung von Routing-Topologien für das Routen mit IPv6 und den Einsatz von Hosts, die nur IPv6 unterstützen. Der Einsatz dieser Technik kommt in der späten Phase des Übergangs.

IPv4- und IPv6-Knoten sollen in einem gemeinsamen Netz existieren und kommunizieren können. Gehen die IPv4-Adressen aus, so stellen die Übergangsstrategien die Kommunikation zwischen IPv4- und IPv6-Hosts für eine unbestimmte Zeit sicher. Dieses Feature schützt letztendlich die Investitionen, die in IPv4 gesteckt wurden.

Selbsttestaufgabe 8.5-4:

Aus welchem Grund führte man IPv6 ein? Für welche Netze wurde IPv6 konzipiert?

Selbsttestaufgabe 8.5-5:

Beschreiben Sie die Unterschiede zwischen Aggregatable Global Unicast-, Link-local Unicast- und Site-local Unicast-Adresssen.

Selbsttestaufgabe 8.5-6:

Welches Format besitzen Anycast-Adressen?

8.6 TCP/IP-Protokolle der Transportschicht

Die Transportschicht, die das Verbindungsglied zwischen den Anwendungen und dem Kommunikationsnetz darstellt, beinhaltet unter anderem zwei wichtige und voneinander unabhängige Protokolle – das ***T**ransmission **C**ontrol **P**rotocol* (**TCP**) für eine gesicherte, fehlerkorrigierende Verbindung, die Flusskontrolle mit einschließt und das ***U**ser **D**atagram **P**rotocol* (**UDP**) für eine ungesicherte Verbindung mit Fehlererkennung aber ohne Fehlerkorrektur und Flusskontrolle. Beide Protokolle stellen die Ende-zu-Ende-Konnektivität zwischen zwei Endsystemen her.

8.6.1 User Datagram Protocol (UDP)

Das ***U**ser **D**atagram **P**rotocol* (**UDP**) benutzt für die Nachrichtenübermittlung das in der Schicht darunterliegende ***I**nternet **P**rotocol* (**IP**), wobei es dessen Eigenschaften wie Unzuverlässigkeit und Verbindungslosigkeit übernimmt. UDP ist ein einfaches Protokoll ohne Flusskontrolle, das keine Garantie für die Auslieferung der Nachrichten oder die richtige Reihenfolge bei der Auslieferung bietet. Diese Aufgaben können von den höheren Schichten übernommen werden.

UDP ist für fehlertolerante Anwendungen als Transportdienst gut geignet und wird z. B. bei der Übertragung von Echtzeitdaten wie Audio- und Videoströmen von Videokonferenzen eingesetzt, wo Fehlerkontrolle, -korrektur oder ein erneutes Anfordern von verlorengegangenen Paketen zu lange dauern würde.

Durch die Beschränkung auf wenige Funktionen ergibt sich ein einfacher und kompakter Aufbau der UDP-Nachricht nach RFC 768 (siehe Abb. 8.6-1).

Abb. 8.6-1: UDP-Nachricht

UDP-Header

Port-Nummer

Der **UDP-Header** besteht aus 4 Zwei-Byte-Feldern: *Source Port*, *Destination Port*, *Segment Length* und *Checksum*. **Port-Nummer** identifizieren Prozesse, die auf vernetzten Systemen laufen. Protokolle der Transportschicht beziehen sich nicht auf Netzschnittstellen wie es bei den Protokollen der Vermittlungsschicht der Fall ist, sondern auf Ports und damit auf Prozesse. Ports werden in der Regel vom Betriebssystem des Rechners verwaltet, das auch dafür sorgt, dass ankommende Daten gepuffert werden und somit keine Daten verloren gehen, bevor der zugehörige Prozess empfangsbereit ist.

Source Port
Destination Port

Die Felder **Source Port** und **Destination Port** haben jeweils eine Länge von 16 Bit, so dass insgesamt $2^{16} = 65536$ Ports möglich sind. Die Ports werden nach Abb. 8.6-2 in drei Bereiche unterteilt. Die *Well-Known Ports* werden von der ***I**nternet **A**ssigned **N**umbers **A**uthority* (**IANA**) zugewiesen und kontrolliert. Sie sind in der Regel für Systemprozesse vorgesehen. Die Anwendungshersteller können für ihre Anwendungen Port-Nummern aus dem Bereich der *Registered Ports* registrieren lassen. Die *Dynamic* oder *Private Ports* unterliegen keinerlei Kontrolle oder Beschränkung und lassen sich nach Belieben einsetzen.

Port-Typ	Port-Nummer
Well-Known Ports	0 - 1023
Registered Ports	1024 - 49151
Dynamic / Private Ports	49152 - 65535

Abb. 8.6-2: Port-Bereiche

Die Serversoftware läuft über ihre spezielle Port-Nummer, die Clientsoftware dagegen braucht nicht immer mit dem selben Port zu operieren. Vielmehr sollte jede Clientsitzung auf einem anderen Port (aus dem Wertebereich der *Dynamic / Private Ports*) laufen. Dies erlaubt einem Host, mehrere Instanzen einer Software oder eines Prozesses gleichzeitig einzusetzen z. B. mehrere Telnet- oder FTP-Sitzungen zwischen zwei gleichen Hosts.

Length

Das **Length**-Feld (16 Bit) gibt die Gesamtlänge der UDP-Nachricht in Byte an (maximal $2^{16} = 65536$ Byte). Die minimale Größe der Nachricht beträgt 8 Byte, was lediglich der Länge des Header entspricht.

Checksum

Pseudo-Header

Im Gegensatz zum *Internet Protocol* deckt die Prüfsumme (**Checksum**) bei UDP sowohl den Header als auch die Nutzdaten ab. Zur Berechnung wird ein so genannter **Pseudo-Header** herangezogen, der die Felder *Source Address*, *Destination Address* (IP-Adressen der Quelle und des Ziels), *Protocol* (Angabe des Protokolltyps, für UDP=17), *Unused* und *Length* (Längenangabe wie im UDP-Header) beinhaltet (siehe Abb. 8.6-3). Die Prüfsumme wird dabei über die gesamte UDP-Nachricht inklusiver Pseudo-Header gebildet und erlaubt dem Host die Überprüfung, ob die Daten unbeschädigt am gewünschten Ziel angekommen sind.

Abb. 8.6-3: Pseudo-Header

8.6.2 Transmission Control Protocol (TCP)

Im Gegensatz zu UDP stellt das *Transmission Control Protocol* den Applikationen einen zuverlässigen und verbindungsorientierten Transportdienst zur Verfügung und gewährleistet den gesicherten Austausch von Nutzdaten zwischen Rechnern in paketvermittelnden Netzen.

virtuelle Verbindung

Für den gesicherten Austausch von Daten wird eine **virtuelle Verbindung** (*Virtual Circuit*) zwischen Sender und Empfänger aufgebaut und Mechanismen zur Verfügung gestellt, mit denen die Lieferung, die Reihenfolge und die Vollständigkeit der Daten garantiert werden. Darüber hinaus bietet TCP verschiedene Techniken, um die Performance einer Verbindung zu steigern und eine Flusssteuerung, mit der sich der Absender dynamisch an die Geschwindigkeit des Empfängers anpassen kann.

8.6.2.1 Zuverlässiger Transport von Daten

Puffer

Bei Anwendungsprogrammen (*User Processes*) unterschiedlicher Hosts, die große Datenmengen übertragen sollen, werden die Daten als Bytestrom betrachtet. Diese Daten werden zunächst in einen **Puffer** geschrieben, um dann aus den Puffer-Daten TCP-Segmente (TCP-Pakete) zu bilden, die über die virtuelle Verbindung an den Empfänger gesendet werden. Empfängerseitig werden die Nutzdaten aus den Segmenten wieder in einen Puffer geschrieben, woraus ein Bytestrom gebildet wird, welcher an das Anwendungsprogramm weitergeleitet wird (siehe Abb. 8.6-4).

Abb. 8.6-4: Bildung von TCP-Segmenten aus den Puffer-Daten

Für den zuverlässigen Transport von Nutzdaten ist es wichtig, dass die Daten vom Empfänger unverfälscht und in der Reihenfolge des Sendens empfangen werden. Die Integrität der TCP-Segmente wird sichergestellt, indem die zu sendenden Segmente (genauer die Bytes im Puffer) fortlaufend nummeriert werden. Damit kann

der Empfänger den Verlust oder Duplikate eines Segments erkennen und somit entweder das verlorengegangene Segment erneut anfordern oder das Duplikat verwerfen. Treffen Segmente in falscher Reihenfolge ein, so werden diese mit Hilfe der Nummerierung vor der Weiterverarbeitung sortiert. Der korrekte Empfang von Segmenten muss vom Empfänger quittiert werden. Geschieht dies nicht innerhalb eines vorgegebenen Zeitintervalls (**Timeout-Intervall**), so werden die Segmente nochmals gesendet.

Timeout-Intervall

Über die Prüfsumme (siehe Abschnitt 8.6.2.2), die bei TCP obligatorisch ist, werden beschädigte Segmente vom Empfänger erkannt, jedoch nicht quittiert. Damit wird indirekt ein erneutes Versenden angefordert.

8.6.2.2 Der TCP-Header

Abb. 8.6-5: TCP-Header

Der TCP-Header umfasst ohne die Felder *Options + Padding* 20 Byte. Die ersten beiden Felder **Source Port** und **Destination Port** enthalten Port-Nummern, welche sender- und empfängerseitige Prozesse identifizieren und wie bei UDP in die Bereiche *Well-Known Ports*, *Registered Ports* und *Dynamic/Private Ports* aufgeteilt sind. Der Anwenderprozess, der die Verbindung initialisieren will, wählt dabei als *Source Port* eine beliebige Port-Nummer aus dem Bereich der *Dynamic/Private Ports* aus. Der *Destination Port* enthält die Port-Nummer des Anwenderprozesses im Zielrechner und ist in der Regel ein *Well-Known Port*. Tabelle 8.6-1 listet die Ports einiger bekannter Standarddienste auf.

Source Port
Destination Port

Tab. 8.6-1: *TCP- und UDP-Port-Belegungen*

Name	Port	Port-Typ	Beschreibung
FTP	21	TCP	File Transfer Protocol
Telnet	23	TCP	
SMTP	25	TCP	Simple Mail Transfer Protocol
DNS	53	UDP	Domain Name Service
WWW	80	TCP	World Wide Web
POP3	110	TCP	Post Office Protocol Version 3
NTP	123	UDP	Network Time Protocol
SNMP	161	UDP	Simple Network Management Protocol

Jedes Byte eines Datenstroms ist nummeriert und beginnt mit einer beliebigen Nummer, welche vom sendenden Host gewählt wird. TCP-Verbindungen arbeiten im Duplex-Modus, d. h. die Daten werden zur selben Zeit in beiden Richtungen übertragen. Jeder Host wählt eine beliebige Zahl zwischen 0 und $2^{32} - 1$ als Startpunkt für die Nummerierung der Bytes seiner eigenen Datenströme. Die **Sequence Number** (Sequenznummer) im TCP-Header stellt dabei die Nummer dar, die der sendende Host für das erste Byte im aktuellen Segment gewählt hat. Die *Sequence Number* ist ein 32 Bit-Wert, der auf Null zurückgesetzt wird, sobald er $2^{32} - 1$ erreicht hat.

Sequence Number

Acknowledgment Number

Die **Acknowledgment Number** (Bestätigungsnummer) dient der Bestätigung von empfangenen Datensegmenten. Diese Nummer wird vom Empfänger gesetzt, um dem Sender mitzuteilen, bis zu welchem Byte die Daten korrekt angekommen sind. Empfängt der Sender innerhalb eines Timeout-Intervalls keine Bestätigung der gesendeten Daten, so werden diese erneut gesendet.

Beispiel 8.6-1:
2 Segmente mit je 30 Byte Nutzdaten sollen übertragen werden. Hierzu wird zuerst das Segment 1 mit den durchnummerierten Bytes 1 bis 30 losgeschickt (siehe Abbildung unten). Das letzte TCP-Byte, das den Empfänger erreicht, hat die Nummer 30. Daher wählt der Empfänger in seiner Bestätigungsnachricht den Wert 31 für die *Acknowledgment Number*. Dieser Wert stellt die Nummer des nächsten Byte dar, das von der Gegenseite gesendet werden muss. Nach dem Erhalt der Bestätigung schickt der Sender das Segment 2 mit den Bytes 31 bis 60. Erhält der Sender innerhalb eines Timeout-Intervalls keine Bestätigung mit der *Acknowledgment Number* 61, so sender er das Segment 2 erneut.

Das Feld **Header Length** gibt die Länge des TCP-Headers an. Dieses Feld ist erforderlich, da die Länge des Optionsfeldes variabel ist. Die Länge des Headers ist ein Vielfaches von 4 Byte und ist maximal 60 Byte groß. **Header Length**

Der TCP-Header enthält 6 **Flag Bits**, die einzeln oder gleichzeitig gesetzt sein können. Sie werden eingesetzt, um Verbindungen auszuhandeln und diese zu verwalten: **Flag Bits**

- **URG:** Ist dieses Bit auf 1 gesetzt, so zeigt der *Urgent Pointer* auf die dringenden Daten eines Segments.
- **ACK:** Das ACK-Bit gibt an, ob die *Acknowledgement Number* im Header gültig ist. Ist das ACK-Bit auf 0 gesetzt, so muss die *Acknowledgement Number* ignoriert werden.
- **PSH:** Um den Rechenaufwand gering zu halten, hält TCP normalerweise die eingehenden Segmente zurück, bis der Puffer gefüllt ist, um sie dann an die Applikation weiterzuleiten. Bei einigen Anwendungen (wie z. B. Telnet) darf TCP die eingehenden Daten jedoch nicht zurückhalten. In solchen Fällen wird das PSH-Bit auf 1 gesetzt.
- **RST:** Dieses *Flag Bit* zeigt an, dass ein Fehler aufgetreten ist und die Verbindung zurückgesetzt werden muss. Es wird auch in einer Antwort auf eine Verbindungsanforderung eingesetzt, um mitzuteilen, dass diese zurückgewiesen wurde.
- **SYN:** Das SYN-Bit wird während der Verbindungsaufbauphase auf 1 gesetzt, um anzuzeigen, dass gerade die Synchronisation der Sequenznummer stattfindet.

- **FIN:** Das FIN-Bit wird zum Beenden einer Verbindung eingesetzt, wenn der Sender keine weiteren Daten mehr zu übertragen hat. Das Segment mit gesetztem FIN-Bit muss quittiert werden.

Window Size

Mit der Angabe der Fenstergröße (**Window Size**), die der Flusskontrolle bei TCP dient, steuert der Zielrechner den an ihn gesendeten Datenstrom im Quellrechner (siehe Abschnitt 8.6.2.4). Die Ermittlung der optimalen Fenstergröße gehört zu den wichtigsten Aufgaben einer TCP-Implementierung. Die maximale Fenstergröße ist auf 65535 Byte beschränkt.

Prüfsumme

Die TCP **Prüfsumme** wurde bereits erwähnt. Sie wird über das gesamte Segment und dem Pseudo-Header berechnet.

Urgent Pointer

Der **Urgent Pointer** wird nur dann verwendet, wenn das URG-Flag-Bit gesetzt ist. Er wird benutzt, um die Sequenznummer des letzten Byte für dringende Daten zu ermitteln.

TCP-Option

Die gängigste **TCP-Option** ist die Option für die die maximale Segmentgröße (***M**aximum **S**egment **S**ize* - **MSS**), die in der Regel von beiden Seiten der Verbindung im ersten ausgetauschten Segment spezifiziert wird. Hosts können die Fragmentierung der TCP-Segmente verhindern, indem sie der Gegenseite die maximale Segmentgröße mitteilen, die sie empfangen möchten.

Padding

Padding stellt sicher, dass die TCP-Segmentlänge ein Vielfaches von 32 Bit beträgt.

8.6.2.3 Virtual Circuits

Virtual Circuit

Ein **Virtual Circuit** verbindet zwei Prozesse auf unterschiedlichen Hosts und verhält sich wie eine leitungsgebundene Verbindung. Man unterscheidet wie bei den Durchschalteverbindungen zwischen Verbindungsaufbauphase, Verbindungsphase und Verbindungsabbauphase. Wenn ein Prozess eine TCP-Verbindung mit einem anderen Prozess initiiert, so handeln beide Prozesse das Öffnen einer Verbindung aus. Jeder muss der Teilnahme an der Verbindung zustimmen. Es gibt jedoch keine Garantie, dass eine TCP-Verbindung aufgebaut und aufrechterhalten werden kann.

Verbindungsaufbauphase

Um eine TCP-Verbindung zwischen zwei Kommunikationspartnern aufzubauen, wird in der Verbindungsaufbauphase das so genannte Drei-Wege-Handshake-Verfahren eingesetzt. Hierbei werden Steuerinformationen zwischen den beiden Seiten ausgetauscht, die sicherstellen, dass der Endpunkt der Verbindung existiert und die Daten angenommen werden können.

Abb. 8.6-6: Schritte der Verbindungsaufbauphase

Für den Aufbau einer Verbindung werden die drei Segmente **SYN A** (***Syn**chronisation **A***), **SYN B** (***Syn**chronisation **B***) und **ACK** (**Ack**nowledge) ausgetauscht (siehe Abb. 8.6-6):

1. Host A sendet Host B das Segment SYN A und fordert Host B dazu auf, der Verbindungsanforderung nachzukommen. Gleichzeitig wird Host B über die Sequenznummer von Host A informiert (167). Das SYN-Flag-Bit ist gesetzt, um anzuzeigen, dass die Verbindung gerade synchronisiert wird.

2. Als Bestätigung für den Erhalt des Segments SYN A sendet Host B nun das Segment SYN B. Bei diesem Segment ist das ACK-Flag-Bit gesetzt und das Acknowledgment Number-Feld enthält die um eins inkrementierte Eröffnungssequenznummer von Host A ($167 + 1 = 168$). Auch Host B wählt eine Eröffnungssequenznummer und nimmt sie in das Sequence Number-Feld auf (498). Weil die Synchronisation nicht abgeschlossen ist, ist das SYN-Flag-Bit noch auf 1 gesetzt.

3. Im letzten Schritt muss Host A noch den Erhalt von Segment SYN B bestätigen und sendet hierfür ein weiteres Segment (ACK). Da die Synchronisation abgeschlossen ist und damit beide Teilnehmer die Eröffnungssequenznummer der Gegenseite kennen, wird das SYN-Flag-Bit auf 0 gesetzt. Dabei wird die Sequenznummer um eins erhöht (499), das ACK-Flag-Bit gesetzt und der Wert 499 in das Feld *Acknowledge Number* aufgenommen.

Mit dem Eintreffen des Segments ACK bei Host B ist die Verbindungsaufbauphase abgeschlossen; beide Kommunikationspartner haben ihre Eröffnungssequenznummer mit dem Partner ausgetauscht und sich somit synchronisiert.

Verbindungsphase

Sobald eine Verbindung aufgebaut ist, beginnen beide Seiten mit der Übertragung der Nutzdaten in Segmenten. Während die Segmente mit den Nutzdaten in einer Richtung transportiert werden, werden gleichzeitig Kontrolldaten in die entgegengesetzte Richtung übertragen, um den Erhalt eines jeden Segments zu bestätigen. Dabei verschickt der Kommunikationspartner bei einfachen Protokollen das nächste Segment nur dann, wenn die Bestätigung des zuletzt gesendeten Segments eingetroffen ist.

Verbindungsabbauphase

Auch in der Verbindungsabbauphase wird zum Beenden einer TCP-Verbindung das so genannte Drei-Wege-Handshake-Verfahren eingesetzt. Die beiden an der Kommunikation beteiligten Teilnehmer teilen in dieser Phase ihren Wunsch zum Beenden der Verbindung mit und bestätigen jeweils den Wunsch des Partners (siehe Abb. 8.6-7).

1. Hat Host A keine Daten mehr zu verschicken, so sendet er zunächst das FIN A-Segment mit gesetztem FIN-Bit.

2. Als Bestätigung für den Erhalt des Segments FIN A sendet Host B nun das Segment ACK A. Nachdem Host A dieses Segment erhalten hat, ist die Verbindung von Host A nach Host B beendet; damit können auch keine Segmente mit Nutzdaten von Host A nach Host B versendet werden.

   Sobald auch Host B keine Daten zu übertragen hat, sendet er das FIN B-Segment mit gesetztem FIN-Bit. Nach RFC 793 sind *Sequence Number* und *Acknowledgment Number* des FIN B-Segments identisch mit denen des ACK A-Segments.

3. Als Bestätigung für den Erhalt des FIN B-Segments sendet Host A nun ACK B. Mit dem Eintreffen des Segments ACK B bei Host B ist die Verbindungsabbauphase abgschlossen und die TCP-Verbindung abgebaut, so dass Resourcen wie z. B. Puffer freigegeben werden können.

Abb. 8.6-7: Schritte der Verbindungsabbauphase

8.6.2.4 TCP-Fenstermechanismus

Der TCP-Fenstermechanismus bietet durch die Angabe der *Window Size* im TCP-Header zum einen die Möglichkeit der Flussteuerung an den Endpunkten einer TCP-Verbindung und zum anderen ein effizientes Übermitteln der Daten. Die Fenstergröße gibt an, wie viele Bytes – beginnend ab dem *Acknowledgement Number* – der Zielrechner in seinem Aufnahmepuffer noch aufnehmen kann. Diese Datenmenge kann vom Sender versendet werden, ohne vom Empfänger auf eine Bestätigung zu warten.

Besteht die Gefahr, dass der Empfänger mehr Daten empfängt als er speichern kann, so wird die Fenstergröße verkleinert. Ist der Puffer gefüllt, so gibt der Empfänger eine Fenstergröße von Null an, um so die senderseitige Übertragung zu stoppen, bis genügend Platz im Puffer vorhanden ist.

Beispiel 8.6-2:
Die Prozesse auf Host A und Host B sind durch einen *Virtual Circuit* verbunden. In der Verbindungsaufbauphase wurde jeweils von beiden Teilnehmern eine MSS von 1044 Byte ausgehandelt. Der Prozess auf Host B soll nun 5120 Byte zu Host A versenden. Allerdings hat der Prozess auf Host A zum Empfang der Daten nur 3072 Byte an freiem Pufferplatz zur Verfügung.

Prozess A gibt zunächst durch das Versenden eines Segments mit *Window Size* = 3072 an Prozess B seinen freien Pufferplatz bekannt (siehe Abbildung unten). Hierauf verschickt Prozess B drei Segmente mit jeweils 1044 Byte - 20 Byte = 1024 Byte (MSS - Größe des TCP-Header) an Nutzdaten. Bei Empfang der einzelnen Segmente verringert sich der freie Pufferplatz um jeweils 1024 Byte, so dass in ACK 1 bis 3 (Bestätigungsmeldung für den Empfang von Segment 1 bis 3) die *Window Size* jeweils auf 2048, 1024 und auf 0 heruntergesetzt wird. Durch die Angabe der *Window Size* von 0 in ACK 3 weiss der Sender nun, dass Prozess A zunächst keine weitere Daten mehr wünscht, bis er die *Window Size* erhöht.

Prozess A liest irgendwann 2048 Byte aus dem Puffer – leert also den Puffer um 2048 Byte – und sendet die zuletzt gesendete Bestätigungsnachricht ACK 3 mit der geänderten *Window Size* von 2048 ab. Der Sender kann nun zwei weitere Segmente (Segment 4 und Segment 5) direkt hintereinander abschicken. Da bis zur Ankunft der beiden Segmente 4 und 5 Prozess A keine Daten aus dem Puffer gelesen hat, verringert sich der freie Pufferplatz zunächst auf 1024 Byte und dann auf 0 Byte. Aus diesem Grunde ist die *Window Size* bei ACK 5 Null.

Im Unterschied zu den herkömmlichen Sliding-Window-Mechanismen erlaubt der TCP-Fenstermechanismus wie im obigen Beispiel dargestellt, zusätzlich die Veränderung der Fenstergröße zur Flusssteuerung. Für den TCP-Fenstermechanismus müssen am Sender drei Zeiger bereitgestellt werden (siehe Abb. 8.6-8).

Abb. 8.6-8: Sliding-Window von TCP mit 3 Zeigern

Der Zeiger Z1 markiert den linken Rahmen des aktuellen Fensters, der die bereits übertragenen und bestätigten Bytes von den noch zu bestätigenden Bytes trennt. Zeiger Z3 markiert den rechten Rahmen des aktuellen Fensters, der das höchste Byte in der Sequenz definiert, das übertragen werden darf, bevor weitere Bestätigungen eingehen. Innerhalb des aktuellen Fensters kennzeichnet Zeiger Z2 schließlich die Grenze, die die bereits übertragenen von den noch nicht übertragenen Bytes trennt. Da die meisten TCP-Implementationen die Bytes im Fenster ohne Verzögerung senden, bewegt sich diese Grenze innerhalb des Fensters in der Regel sehr schnell von links nach rechts.

Beispiel 8.6-3:
Im Folgenden wird die Fensterverschiebung und die Zeiger während der Datenübertragung von Prozess B nach Prozess A dargestellt. Dabei wird angenommen, dass Prozess A immer die gleiche *Window Size* ankündigt, da er die eintreffenden Daten aus dem Puffer sofort ausliest, bevor weitere neue Daten eintreffen. Aus diesem Grund ist die Größe des aktuellen Fensters bei Prozess B konstant.

Die Abbildung unten zeigt die Fensterverschiebung in Abhängigkeit der eintreffenden Bestätigungen am Senderprozess B für ein dreiphasiges Szenario. In der ersten Phase wurden alle Daten versendet. Da für diese Daten keine Bestätigung eingetroffen ist, sind Zeiger Z2 und Z3 gleich.

In Phase 2 kommt eine Bestätigung für den Empfang von Byte 3 und 4 an. Daraufhin verschiebt sich das Fenster mit den beiden Zeigern Z1 und Z3 um 2 Byte nach rechts. Die zwischen Zeiger Z2 und Zeiger Z3 eingegrenzten Bytes können nun in ein Segment verpackt und zum Empfänger verschickt werden.

In der letzten Phase sind Zeiger Z2 und Z3 gleich, da alle Daten verschickt wurden.

1. Phase:
Alle Daten im aktuellen Fenster versendet. Warten auf Bestätigungen.

2. Phase:
Bestätigung für den Empfang von Byte 3 und 4 trifft ein. Fenster verschiebt sich um 2 Stellen nach rechts.

3. Phase:
Ein Segment mit Byte 10 und 11 als Nutzdaten wird versendet.

8.6.2.5 TCP-Performance-Merkmale

Das TCP-Protokoll beinhaltet vier wichtige Performance-Merkmale, um die die ursprüngliche Version des TCP-Protokolls ergänzt wurde [LOP99]:

- *Slow Start*
- *Congestion Avoidance*
- *Fast Retransmit*
- *Fast Recovery*

Slow Start

Slow Start behandelt das Problem der Regulierung der Übertragungsgeschwindigkeit von Segmenten, indem beobachtet wird, wie schnell die Gegenseite die Segmente bestätigt, die bereits versendet wurden. Falls die Bestätigungen schnell ankommen, so wird der *Transmission Window*[10] vergrößert; treffen sie dagegen langsam ein, so wird der *Transmission Window* verkleinert.

Congestion Avoidance

Congestion Avoidance wird gewöhnlich in Verbindung mit dem *Slow Start* implementiert. Er bietet einen Mechanismus mit dem verloren gegangene Pakete behandelt werden. *Congestion Avoidance* schraubt die Übertragungsgeschwindigkeit herunter, wenn der Sender einen Hinweis bekommt, dass Segmente auf dem Übertragungsweg verloren gegangen sind.

Fast Retransmit

Doppelte Bestätigungen können durch verlorene Segmente oder durch Segmente, die in falscher Reihenfolge geliefert wurden, entstehen. Falls ein Segment verloren geht, warten einfache TCP-Implementationen bis der Timer abläuft, bevor das fehlende Segment erneut versendet wird. Im Falle eines **Fast Retransmit** kann die Wartezeit vermieden werden: Wenn drei mal hintereinander (oder öfter) doppelte Bestätigungen empfangen werden, so geht der TCP-Prozess davon aus, dass ein TCP-Segment fehlt. Das Segment wird erneut gesendet, ohne auf den Ablauf des Timers zu warten.

Fast Recovery

Der Empfang von doppelten Bestätigungen bedeutet unter Umständen auch, dass irgendwo entlang der Route ein Verkehrsstau aufgetreten ist. Für den Fall, dass der Fast Retransmit-Algorithmus aktiv ist, schreibt der **Fast Recovery**-Algorithmus vor, dass TCP-Implementationen zuerst mit dem Congestion Avoidance-Algorithmus beginnen.

Selbsttestaufgabe 8.6-1:

Welches der beiden Protokolle, TCP und UDP, ist verbindungsorientiert? Welches der beiden Protokolle würden Sie vorziehen, wenn Sie eine Echtzeit-Sprachübertragung realisieren wollen? Begründen Sie Ihre Antwort.

10 *Transmission Window* ist eine Fenstergröße, die auf der Grundlage der Rundlaufzeit (**RTT** – ***R**ound **T**rip **T**ime*) und der *Window Size* bestimmt wird.

9 Agenten im Internet

9.1 Eigenschaften von Agenten

Agenten bieten ein mächtiges Werkzeug zur Erstellung asynchroner verteilter Netzanwendungen. Dabei spielt die eigentliche Topologie des Netzs nur eine untergeordnete Rolle. Agenten können sowohl in einem lokalen (LAN), einem regionalen (MAN) oder einem weltumspannenden Netz (WAN) wie dem Internet eingesetzt werden. Der Begriff des Agenten ist durch eine Reihe von Forschungsdisziplinen beeinflußt. Sowohl Disziplinen der Informatik wie Kommunikations- und Rechnernetze, künstliche Intelligenz, verteilte Intelligenz, Entscheidungstheorie, aber auch geisteswissenschaftliche Disziplinen wie Psychologie prägen den Begriff des Agenten [BUR97], [BRZA et al. 98]. Aufgrund seiner interdisziplinären Bedeutung kann jedoch nur schwer eine einheitliche, allen Disziplinen genügende Definition getroffen werden. Daher soll hier ein Agent über die Summe seiner Eigenschaften charakterisiert werden. Es ist leicht auszumachen, welche Eigenschaften den einzelnen Disziplinen entstammen (siehe Abb. 9.1-1).

Definition anhand Eigenschaften

Abb. 9.1-1: Eigenschaften eines Agenten [BRZA et al. 98].

Autonomes Handeln

Ein Agent ist ein Programm, das in fremdem Auftrag autonom handelt. Ist der Agent mobil, so kann er seinen Ausführungsort innerhalb seiner Umgebung, dem Netz, wechseln, indem er zu verschiedenen Rechnern reist. Unter der Kommunikation soll die Interaktion mit anderen Agenten verstanden werden, während Kooperation die auf ein gemeinsames Ziel ausgerichtete Kommunikation bedeutet. Die Forderung nach einem individuellen Verhalten soll durch das Attribut Charakter zum Ausdruck kommen.

Lernfähigkeit und Verhalten

Ein Agent ist lernfähig, wenn er aus bestehendem Wissen weiteres Wissen schlußfolgern kann. Sein Verhalten ist proaktiv, wenn der Agent auf sich ändernde Bedingungen seiner Umgebung reagiert, indem er ein Abbild des Zustandsraums der Umgebung nutzt. Sein Verhalten ist reaktiv, wenn er anhand einer Menge von Regeln auf sich ändernde Bedingungen seiner Umgebung reagiert.

Je nach Ausprägung dieser Eigenschaften spricht man von *kooperierenden Agenten*, *intelligenten Agenten* oder als deren Vereinigungsmenge '*Smart Agents*'. Während bei den kooperierenden Agenten das Augenmerk auf den Eigenschaften Kommunikation, Kooperation und mit Abstrichen auch Autonomie liegt, zeichnen sich intelligente Agenten durch ein proaktives bzw. reaktives Verhalten aus. Solche Agenten sind lernfähig und verfügen über einen Charakter. Smart Agents bilden als die Vereinigungsmenge aus kooperierenden und intelligenten Agenten autonome, durch das Netz wandernde Programme, die auf Änderungen ihrer Umgebung reagieren, eigenständig handeln und einen eigenen Charakter ausbilden. Durch ihre Fähigkeit zu lernen, leiten sie aus der Umgebung Wissen ab, das sie nutzen, um neues Wissen über die Umgebung zu folgern. Smart Agents sind erst in Ansätzen realisiert.

Dieser Beitrag behandelt die Klasse der kooperierenden Agenten. Kooperierende Agenten, nachfolgend kurz Agenten genannt, dienen als Basis zur Erstellung verteilter Anwendungen innerhalb einer dynamischen und heterogenen Netzumgebung. Die folgenden Punkte machen ihren Nutzen deutlich:

Nutzen

- Agenten helfen, die Netzlast zu reduzieren, indem sie zu den gewünschten Daten reisen, die sie zur Erfüllung einer Aufgabe benötigen. Die Übertragung u. U. großer Datenmengen durchs Netz entfällt, da nur noch die gefilterten Ergebnisdaten zu transportieren sind

Reduzierung der Netzlasten ...

- Agenten helfen die Netzkosten ihres Eigentümers zu reduzieren. Dieser kann die Netzverbindung seines Rechners während der Tätigkeit des Agenten auf einem fremden Rechner aufheben. Der Agent begibt sich nach der Beendigung seiner Tätigkeiten in eine Wartestellung, erkennt einen späteren Verbindungsaufbau zu seinem Heimatrechner und kehrt daraufhin zurück.

... und der Netzkosten

- Die Tätigkeit eines Agenten wird von einer plötzlichen Netzunterbrechung nicht gestört, auch dann nicht, wenn der Agent gerade selbst migriert. In diesem Fall setzt der Agent auf seinem letzten konsistenten Zustand auf und wiederholt den Reisevorgang zu einem späteren Zeitpunkt.

Kontinuität

Anwendung

Denkbare Anwendungsbeispiele für Agenten liegen im Bereich des Electronic Commerce: Agenten treten als Anbieter, Käufer, Verkäufer und Vermittler von Waren bzw. Diensten auf und bewegen sich dabei gegebenenfalls von Ort zu Ort.

Weitere Anwendungsmöglichkeiten sind in den Bereichen Informationsbeschaffung, Unterstützung von Gruppenarbeit und personalisierter Dienste anzusiedeln. Ein Beispiel hierfür ist die prototypisch realisierte, agentenbasierte Kommunikationsplattform Campus.de [REU98] für Studenten der FernUniversität Hagen. Sie unterstützt den Informationsaustausch und die Gruppenbildung zwischen Studenten einer virtuellen Universität bei der Planung und Durchführung ihres Studiums.

Im Bereich der Fernwartung von Systemkomponenten ist der vom österreichischen Ozondatenverbund genutzte odvAgent anzusiedeln. Der odvAgent bereist eine Gruppe von Rechnern und überprüft diese auf eventuelle Fehlerzustände

In den aufgeführten Beispielen werden Agenten genutzt, um an Daten bzw. Informationen auf fremden Rechnern zu gelangen oder diese zu verwalten. Ein Agent

kann aber auch auf einen fremden Rechner reisen, um dessen Rechenleistung zu nutzen [RURE et al. 99] . Es ist ebenfalls denkbar, Agenten mittels Divide-et-Impera-Verfahren zur Code-Parallelisierung rechenintensiver Algorithmen zu nutzen

9.2 Arbeitsumgebung mobiler Agenten

Arbeitsumgebung

Eine notwendige Vorraussetzung für die Tätigkeit eines Agenten auf einem Rechner ist eine geeignete Arbeitsumgebung. Die Arbeitsumgebung ist ein Programm, das möglichst permanent aktiviert ist und nur im Falle des Ausschaltens des Rechners beendet wird. Bis auf die Phasen des Reisens befindet sich ein Agent ständig in der Kontrolle einer Arbeitsumgebung. Sie kann mehreren Agenten unterschiedlicher Herkunft gleichzeitig als Aufenthaltsort dienen. Soll ein Rechner für einen Agenten erreichbar sein, so muß er über eine derartige Umgebung verfügen.

Aufgaben

Im einzelnen umfassen die Aufgaben einer Arbeitsumgebung:

- das Starten heimischer Agenten
- das Beherbergen eigener und fremder Agenten
- die Synchronisation der Tätigkeiten verschiedener Agenten
- die Kommunikation mit reisenden heimischen Agenten
- den Schutz des Rechners und seines Dateisystems vor böswilligen Agenten

Fungieren Arbeitsumgebungen als Dienstanbieter, dann sollten sie für an- und abreisende Agenten möglichst häufig empfangs- und sendebereit sein. Dieser Docking-Mechanismus und das Starten heimischer Agenten sind ihre Haupttätigkeit.

Synchronisation

Da die Arbeitsumgebung i. d. R. gleichzeitig mehreren Agenten als Aufenthaltsort dient, ist eine Synchronisation der Tätigkeiten verschiedener Agenten erforderlich. Einige Arbeitsumgebungen gewichten die Tätigkeiten einzelner Agenten und vergeben entsprechend dieser Gewichtung Rechnerresourcen an Agenten.

Oftmals ist eine Kommunikation der Arbeitsumgebung mit ihren reisenden Agenten notwendig, um eine rechnerübergreifende Kommunikation zwischen Agenten zu ermöglichen.

Sollen Arbeitsumgebungen und Agenten in einem offenen System eingesetzt werden und ist eine vorgeschaltete Kontrolle des Agenten vor Eintritt in das Systen ausgeschlossen, dann müssen die Arbeitsumgebungen Mechanismen bieten, die den Schutz des Rechners und seines Dateisystems gewährleisten.

Agentensystem

Ein System, daß aus einer Menge von Arbeitsumgebungen und Agenten besteht, heißt Agentensystem. Das Agentensystem kann in einem Netz mit Computern verschiedener Bauart und verschiedener Betriebssysteme aufgespannt werden. Die Anpassung des Agentensystems an solch ein heterogenes Rechnernetz wird durch den Einsatz geeigneter plattformunabhängiger Programmiersprachen wie *Java*, *Telescript* oder *Tcl* (Tool Command Language) unterstützt. Mit Hilfe entsprechender betriebssystemspezifischer Interpreter wird die Plattformunabhängigkeit eines

hierauf aufsetzenden Agentensystems erreicht. Agentensysteme, die mit plattformabhängigen Programmiersprachen wie *C/C++*, *Perl*, *Smalltalk* oder *Pascal* realisiert sind, müssen über systemeigene Interpreter verfügen.

Ein Agentensystem kann sowohl *zentral* als auch *dezentral* organisiert sein. Bei einer zentralen Topologie erfolgt die Verwaltung der Zustände des Agentensystems durch eine ausgezeichnete Instanz. Diese muß permanent aktiviert sein. Arbeitsumgebungen, die sich an- oder abmelden, tun dies bei dieser Instanz. Sie benachrichtigt alle weiteren im Agentensystem aktiven Arbeitsumgebungen über einen geänderten Zustand innerhalb des Agentensystems

9.3 Basistechnologien

Die nachfolgend beschriebenen Basistechnologien bilden neben der unidirektionalen Nachrichtenübertragung (*Message Passing*) die Grundlage zur Modellierung derjenigen Eigenschaften eines Agenten, die der Disziplin Kommunikations- und Rechnernetze zuzuordnen sind (siehe Abb. 9.1-1). Sowohl die Kommunikation als auch die Mobilität des Agenten setzen zuunterst auf der Socketkommunikation auf, bei der durch die Vergabe zweier fester Nummern eine Verbindung zwischen zwei Rechnern etabliert wird. Auf der nächst höheren Ebene kommen dann die Technologien *Remote Procedure Call*, *Agent Remote Procedure Call* bsw. *Remote Programming* zum Tragen.

9.3.1 Kommunikationskonzepte

Mobiler Code wird als die Fähigkeit bezeichnet, die Verknüfpungen zwischen Code und der Umgebung, in der er ausgeführt wird, dynamisch zu ändern [CA97]. Mobiler Code ist kein vollständig neues Konzept, er wurde erstmalig in Zeiten von einzelnen Hochleistungscomputern (Mainframes) eingesetzt. Dabei wurden Stapelverarbeitungsprogramme geschrieben, übertragen und ausgeführt oder die Sprache Postscript zur Druckausgabe verwendet. Die folgenden Abbildungen (Abb. 9.3-1 bis Abb. 9.3-4) zeigen die Entwicklung von der Client/Server Architektur hin zu Mobilen Agenten. Ein Host ist eine Rechnereinheit, auf der Programme (Codes) ausgeführt werden können. Ressourcen können im Allgemeinen als Speicherplatz, CPU-Zeit oder Daten auf einem Host interpretiert werden.

- **Client-Server Paradigma:** Dieses Paradigma ist gut bekannt und wird vielfach genutz. Resourcen und Code befinden sich auf Host B. Host A fragt die Ausführung eines Dienstes auf Host B an und dieser bearbeitet die Anfrage indem er den den entsprechenden Code auf seinem System ausführt und auf die vorhandenen Resourcen zugreift. Danach übermittelt Host B die berechneten Ergebnisse an Host A.

Abb. 9.3-1: Client Server

- **Remote Evaluation:** Host A besitzt den Code, aber verfügt im Gegensatz zu Host B nicht über die notwendigen Resourcen. Daher sendet Host A seinen Code an Host B, der den Code ausführt, auf seine Resourcen zugreift und dann das Ergebnis der Berechnung an Host A zurücksendet.

Abb. 9.3-2: Remote Evaluation

- **Code on Demand:** Host A besitzt die notwendigen Resourcen, aber verfügt nicht über den Code um sie zu bearbeiten. Daher fragt er Host B nach dem Code und führt ihn nach der Übermittlung bei sich aus.

Abb. 9.3-3: Code on Demand

- **Mobiler Agent:** Host A besitzt den Code und hat die Fähigkeit diesen auf Host B zu übertragen, damit dieser dann auf die lokalen Resourcen von Host B zugreifen kann. Dieses Konzept unterscheidet sich signifikant von den anderen drei Beispielen, da hier der gesamte Code inklusiver seiner azzoziierten Daten transferiert werden.

Abb. 9.3-4: Mobile Agent

9.3.2 Remote Procedure Call

Eine Kommunikation mittels *Remote Procedure Call* (RPC) erfolgt nach dem Client-Server-Prinzip. Ein Rechner - der Client - benötigt einen Dienst $D_i, \quad i \in 1, ..., n$ auf einem anderen Rechner - dem Server. Alle Botschaften, die mittels eines standardisierten Protokolls verschickt werden, sind entweder *reply* oder *request* Botschaften. Der Client sendet dem Server den Prozeduraufruf und die notwendigen Eingabeparameter PClie der auf Server-Seite verfügbaren Prozedur. Der Server führt die Prozedur aus, indem er sie u. U. mit weiteren Parametern PServ versorgt, und antwortet mit der Übertragung der Ergebnisdaten (siehe Abb. 9.3-5).

reply- oder request-Botschaften

Abb. 9.3-5: Kommunikation mittels RPC

9.3.3 Remote Programming

Technische Basis für mobile Agenten

Remote Programming (RP) bildet die technische Basis für das mobile Agenten-Paradigma. Im Gegensatz zum klassischen RPC oder dem ARPC erfolgt die Kommunikation zwischen zwei Objekten in RP nicht mittels Austausch von *reply* und *request* Botschaften. Stattdessen schickt der Client das aufzurufende Programm selbst als 'Anfrage' zum Server, so daß es dort ausgeführt werden kann. Der 'entfernte Prozeduraufruf' wird damit zu einem 'lokalen Prozeduraufruf auf einem entfernten Rechner'.

Abb. 9.3-6: Kommunikation mittels RP.

Die auf Client-Seite verfügbaren erforderlichen Eingangsparameter PClie werden zusammen mit dem Programm übertragen. Durch dieses Vorgehen ist man nicht länger auf feste Prozeduraufrufe beschränkt, wie sie im Vorfeld der Kommunikation mittels RPC feststehen müssen. Stattdessen ist ein allgemeiner Mechanismus zur Übertragung von Prozeduren und deren Ausführung auf einem fremden Rechner zu definieren. Die Netzlast wird reduziert, da die Interaktion zwischen Client und Server nicht exclusiv über das Netz stattfindet, sondern einzelne Netztransaktionen ausreichen. Insbesondere bei der Bearbeitung einer Familie von Diensten $D_1, ..., D_n$, wird die globale Kommunikation stark reduziert und durch eine lokale Kommunikation auf Server-Seite ersetzt (siehe Abb. 9.3-6).

Innerhalb der Agententechnologie werden $D_1, ..., D_n$, zusammen mit PClie als mobiler Agent aufgefaßt. In der Abbildung ist bewußt auf die Rückführung des Agenten zum Client verzichtet worden, da ein Agent i.a. eine Reiseroute zu mehreren Rechnern verfolgt, bevor er zum Client zurückkehrt. Ein Agent, der mittels RP wandert und auf m Stationen jeweils n Dienste ausführt, benötigt hierfür $m + 1$ Netztransaktionen. Unter der Nutzung von RPC fallen $m(2n)$ Netztransaktionen an.

Remote Procedure Call (RPC)

9.3.4 Vor- und Nachteile von Mobilen Agenten

Mobile Agenten haben die einzigartige Fähigkeit, sich selbst von einem Host über das Netz zu einem anderen Host zu übertragen. Zusätzlich können Sie autonom agieren und ihre Aufgaben im Auftrag ihrer Besitzer ausführen. Damit haben sie im Gegensatz zu traditionellen Ansätzen einige Vorteile. [LA99] fasst folgende Vorteile von mobilen Agenten zusammen:

- **Verringerung der Netzlast:** Traditionelle Kommunikationsprotokolle (z. B. Challenge/Response, Client/Server) erfordern mehrfache Interaktionen bis ein einzelner Auftrag komplett abgearbeitet ist. Applikationen benötigen häufig einen extrem hohen Datendurchsatz der selbst in Hochgeschwindigkeitsnetzen zu einer signifikanten Netzlast führt. Wie in [DAS03] gezeigt ist, können Mobile Agenten dramatisch die Geschwindigkeit und Effizienz verbessern, wenn in einem verteilten und heterogenen Umfeld auf viele Daten zugegriffen wird.
- **Minimierung der Latenzzeit:** Jedes Echtzeit-System (z. B. Netzüberwachung, Netzmanagement oder Fernsteuerung von Systemen) benötigt auch bei dynamisch verändernden Systemumgebungen ein schnelles Antwortverhalten. Langsames Antwortverhalten ist für Echtzeitsysteme kaum tragbar und führt häufig zu Systemproblemen und im schlimmsten Fall sogar zu einem Systemausfall. Konventionelles Netz-Management basiert häufig auf SNMP (***S**imple **N**etwork **M**anagement **P**rotocol*) und wird zentralisiert betrieben. Obwohl eine zentralisierte Verwaltung dem Administrator den Vorteil verschafft, das gesamte Netz von einem einzelnen Ort zu verwalten, ist gerade dieser Ort fehleranfällig und bei starker Systemauslastung und hohem Datenverkehr im Netz ein Flaschenhals. Ein Mobiler Agent ermöglicht eine bessere Lösung, da er lokal auf den entfernten Systemen ausgeführt wird und Überwachungen durchführen oder Maßnahmen autonom einleiten kann wenn ein Problem auftritt.

Simple Network Management Protocol

- **Asynchrone, autonome und dynamische Berechnung:** Mobile Agenten haben die Fähigkeit, autonom auf Systemänderungen zu reagieren. Zusätzlich können sie mit anderen Agenten interagieren um ihre Ziele zu erreichen.
- **Robustheit und Fehlertoleranz:** Da Mobile Agenten die Fähigkeit haben, dynamisch auf sich ändernde Umgebungssituationen zu reagieren, können mit ihnen robuste und fehlertolerante verteilte Systeme aufgebaut werden. Wenn ein Hostsystem heruntergefahren wird, muss dies allen auf diesem System laufenden Agenten mitgeteilt werden, um ihnen die Möglichkeit zu geben auf einen anderen Host zu migrieren, um dort ihre Arbeit fortzusetzen.

Aus dieser Zusammenfassung sind die herausragenden Vorteile Mobiler Agenten ersichtlich. Leider gibt es eine Reihe von Nachteilen und Problemen, die gelöst werden müssen bevor Applikationen mit Mobilen Agenten Verbreitung und Akzeptanz finden. Aus [VI04] und [MI99b] ist eine Liste zusammengestellt, die die Gründe für die bisherigen Vorbehalte gegen den Einsatz von Mobilen Agenten zusamemnfasst:

- **Schwierigkeiten in der Konzeption von Systemen:** Eine Ausführungsumgebung für Mobile Agenten in dem auch Test- und Entwicklungsumgebungen

berücksichtigt sind, hat, bezogen auf Systemkomponenten und einer Verteilung in einer heterogenen Umgebung, eine hohe Komplexität.

- **Schwierigkeiten in der Realisation von Sicherheitsaspekten:** Eine der größten Schwächen bezieht sich auf Sicherheitsaspekte, die in zwei Gruppen unterteilt werden können: Einen Host vor bösartigen Agenten schützen und einen Agenten vor bösartigen Hosts schützen.
- **Schwierigkeiten in der Konzeption von Kommunikationsinfrastrukturen:** Um mit anderen Agenten (die sich auch auf einem anderen System befinden können) oder anderen Systemen zu kommunizieren ist eine Kommunikationsinfrastruktur notwendig. Dazu ist es notwendig, dass alle ausgetauschten Daten in einem Format vorliegen, dass von allen Kommunikationspartnern in der gleichen Weise verstanden wird. Will ein Agent mit einem anderen Host oder einem anderen Agenten kommunizieren, muss ferner der aktuelle Standort des Kommunikationspartners bekannt sein.
- **Keine Killerapplikation:** Alle Anwendungsfälle die Mobile Agenten implementieren könnten sind bereits mit traditionellen Ansätzen realisiert worden. Eine Notwendigkeit von Mobilen Agenten ist bisher nicht gegeben.
- **Konzept ähnlich zu Würmern:** Ein Mobiler Agent kann selbstständig den Transfer seines Codes und seiner Daten zu einem anderen Host initiieren. Dieses Verhalten kann sehr leicht mit einem Angriff eines Wurmes verwechselt werden.

9.4 Mobilität und Kommunikation

Mobilitäts- und Kommunikationsmechanismen aller prototypisch realisierten Agentensysteme lassen sich auf die vorgestellten Basistechnologien zurückführen. Während die Mobilität des Agenten in jedem Fall mit Hilfe von RP realisiert wird, wird zur Kommunikation zwischen Agenten sowohl Message Passing als auch RPC (ARPC), im Einzelfall sogar ebenfalls RP genutzt. Im folgenden sollen die beiden Agenteneigenschaften Mobilität und Kommunikation näher betrachtet werden.

9.4.1 Mobilität

Die Überführung eines Agenten von der Arbeitsumgebung eines Rechners in die Arbeitsumgebung eines weiteren Rechners kann in die Teilprozesse *Persistenzbildung*, *Migration* und *Externalisation* untergliedert werden.

Persistenzbildung

Unter der *Persistenzbildung* von Objekten versteht man die Möglichkeit, Objekte aus dem Programm, in dem sie geschaffen wurden, auszugliedern, sie permanent zu speichern und bei Bedarf dem Programm wieder zur Verfügung zu stellen. Nach der Ankunft eines Agenten legen viele Systeme eine persistente Spiegelstruktur des Agenten in einem ausgezeichneten Verzeichnis auf dem gegenwärtigen Arbeitsrechner an.

Migration

An die Persistenzbildung schließt sich der Teilprozeß der *Migration* an. Unter der Migration ist das geeignete Verpacken der Objektdaten des Agenten, der Transport zu einem anderen Ort sowie das dortige Auspacken der Objektdaten zu verstehen. Hierbei wird zwischen starren und ausführbaren Objektdaten unterschieden. Starre Objektdaten eines Agenten beinhalten Informationen über das Profil des Agenten, seine Zwischenzustände, Eingangsparameter, Teilergebnisse etc.. Ausführbare Objektdaten enhalten die prozeduralen Eigenschaften des Agenten. Bei der Übertragung beider Objekttypen finden RP und innerhalb Java-basierter Systeme die Objekt-Serialisierung ihren Einsatz.

Bezüglich des Migrationszeitpunktes kann zwischen einer schwachen und einer starken Migration unterschieden werden. Während bei der *schwachen Migration* ein Agent seine Tätigkeit nur in bestimmten Programmzuständen unterbricht und seine gegenwärtige Arbeitsumgebung verläßt, kann bei der *starken Migration* der Agent zu jedem Zeitpunkt seiner Tätigkeit die Arbeitsumgebung wechseln.

Externalisation

Die sich der Migration anschließende *Externalisation* behandelt die Integration der ausführbaren Objektdaten des Agenten in eine Arbeitsumgebung. Die Integration geschieht zur Laufzeit und soll hier für ein Agentensystem, das in einer objektorientierten Sprache implementiert ist, vorgestellt werden. Bei der Externalisation müssen nicht notwendigerweise alle Objekt-Informationen integriert werden, wenn einige bereits im Zielsystem vorhanden sind. Die Externalisation ist in die Phasen *Laden*, *Definieren* und *Instantiieren* untergliedert. In der Phase des *Ladens* werden leere Klassen entsprechend der Spiegelstruktur des Agenten erstellt, sofern sie nicht schon als Systemklassen oder Klassen der Arbeitsumgebung vorliegen. Nachdem die entsprechenden Klassen eines migrierten Agenten geladen worden sind, werden diese Klassen als solche *definiert*. In die noch leeren Klassen wird der Binärcode der entsprechenden Datei aus dem Verzeichnis der Spiegelstruktur übertragen. Die Klasse ist jetzt instantiierbar. Mit der *Instantiierung* dieser Klassen ist der Binärcode der Klasse integriert und ausführbar.

Transaktionskonzept

Während Migration und Externalisation technisch notwendige Schritte bei der Überführung eines Agenten in eine andere Arbeitsumgebung sind, erscheint der Teilprozeß der *Persistenzbildung* nicht unmittelbar notwendig. Ein Grund für die Persistenzbildung ist die Sicherung der Objektdaten des Agenten und ihre Ablage für einen zukünftigen Gebrauch. Für den Fall des unvorhersehbaren Terminierens eines Rechners bzw. der Arbeitsumgebung des Agenten sorgt die Persistenzbildung dafür, daß die Agenten fremder Eigentümer und ihre Ergebnisdaten nicht verloren gehen. Bei einem Neustart der Umgebung können die Agenten ihre Tätigkeit auf dem gegenwärtigen Arbeitsrechner wieder aufnehmen und später, ohne Verlust von Ergebnisdaten, zu ihrem Heimatrechner zurückkehren. Die Persistenzbildung schafft damit die Grundlage eines *Transaktionskonzeptes* für die Migration eines Agenten. Zu jedem Zeitpunkt muß eindeutig bestimmbar sein, auf welchem Rechner sich der Agent gerade befindet. Gerade in der Phase der Migration des Agenten ist festzulegen, welchem Rechner der Agent zuzuordnen ist. Nur so können Agenten, die sich auf der Reise befinden, durch andere Instanzen lokalisiert werden. Die nachfolgend eingeführte globale Kommunikation, aber auch Verwal-

tungsmechanismen für Agentensysteme mit einer dezentralen Topologie setzen das eindeutige Lokalisieren mobiler Agenten voraus.

9.4.2 Kommunikation

Komplexe Problemstellungen können häufig nur durch mehrere miteinander kooperierende Agenten gelöst werden. Dabei widerspricht der autonome Charakter eines Agenten seiner Kooperationsfähigkeit nur scheinbar. So zeichnet sich ein Agent primär durch das eigenständige Lösen einer Aufgabe aus, die Existenz anderer Agenten im System interessiert ihn hierbei nicht. Erst wenn er bei der Bearbeitung seiner Aufgabe feststellt, diese nicht allein lösen zu können, kontaktiert er weitere Agenten.

Direkte und indirekte Kommunikation

Wollen Agenten *direkt* miteinander kommunizieren, so müssen sie über den Namen, die Fähigkeiten und den möglichen Aufenthaltsort des potentiellen Kommunikationspartners informiert sein. Ferner muß bekannt sein, welche Kommunikations-Protokolle die Partner beherrschen. Diese Art der Kommunikation eignet sich für eine Gruppe von Agenten, deren Eigenschaften untereinander bekannt sind und dies daher mit den *weißen Seiten* der Telekom vergleichbar ist. Agenten, die weder Name, Fähigkeiten noch Aufenthaltsort des potentiellen Kommunikationspartners kennen, müssen über eine zwischengeschaltete Instanz vermittelt werden. Ihre Kommunikation verläuft *indirekt*; diese Situation entspricht den *gelben Seiten* der Telekom.

9.4.2.1 Direkte Kommunikation

Die Hauptforderung an einen direkten Kommunikationsmechanismus für Agenten ist die Möglichkeit des Informationsaustausches. Eine zweite wichtige Forderung ist die Gewährleistung eines Höchstmasses an Autonomie für den einzelnen Agenten. Ein Kommunikationsmechanismus, der die Eigenschaften *Orts-Transparenz*, *Zeit-Transparenz*, *Unabhängigkeit gegenüber dem Kooperationsprotokoll* und *Kompatibilität gegenüber der Realisierungssprache* besitzt, erfüllt diese Forderung.

Orts-Transparenz

Ein Kommunikationsmechanismus ist orts-transparent, wenn es unerheblich ist, an welchem Ort sich die einzelnen Kommunikationspartner befinden. Die Agenten sollen sowohl lokal auf einem Rechner, als auch *global* auf unterschiedlichen Rechnern kommunizieren können. Bei der globalen Kommunikation sind im Vorfeld Konzepte zur *Lokalisierung* und anschließenden *Blockierung* kommunikations-williger Agenten innerhalb des Agentensystems erforderlich. Vor der eigentlichen Kommunikation muß der Sender den Aufenthaltsort des Empfängers ausfindig machen. Kann dann eine Socketverbindung zur Kommunikation zwischen zwei Agenten etabliert werden, darf keiner der beiden seinen gegenwärtigen Arbeitsrechner verlassen. Erst wenn einer von beiden dem Anderen das Kommunikationsende quittiert, können die Agenten ihre Migration fortsetzen.

Zeit-Transparenz

Ein Kommunikationsmechanismus nennt man zeit-transparent, wenn es unerheblich ist, in welchen Zeitabständen die Kommunikationspartner aufeinander reagieren. Bei der *synchronen* Kommunikation reagiert der Empfänger einer Information direkt nach dem Erhalt einer Information, während bei der *asynchronen* Kommunikation zeitliche Verzögerungen innerhalb der Kommunikation durchaus erwünscht sind. Insbesondere kommt es in diesem Fall zu einer Ablage von Kommunikationseinheiten an hierfür vorgesehenen Orten. Der empfangende Agent entscheidet selber, zu welchem Zeitpunkt er auf eine Anfrage oder Botschaft reagiert. Die synchrone Kommunikation ist mit dem *Telefonieren* vergleichbar, während die asynchrone Kommunikation dem *Briefkastenprinzip* der Post entspricht.

Ein Kommunikationsmechanismus für Agenten sollte eine syntax- und semantikunabhängige Ebene im Hinblick auf das zwischen den Agenten gewählte Kooperationsprotokoll bilden. Auf der nächst höher gelegenen Ebene können dann Agentensprachen wie *KQML* (Knowledge Query Manipulation Language), *TKQML* der Tcl-Variante von KQML, *KIF* (Knowledge Interchange Format) u.s.w. zur Kooperation eingesetzt werden. In allen diesen Fällen finden RPC bzw. ARPC als Basistechnologien ihren Einsatz. Für die direkte Kommunikation zwischen Agenten ist ebenfalls die Basistechnologie RP denkbar, wobei ausführbare Objektdaten als Kommunikationseinheiten zwischen den Agenten zur Übertragung und Ausführung gelangen.

Kompatibilität

Ein Kommunikationsmechanismus sollte kompatibel sein, d. h. er sollte auch die Kommunikation zwischen Agenten aus verschiedenen Agentensystemen und Realisierungssprachen unterstützen. Agenten eines in Java implementierten Systems, wie z. B. das nachfolgend beschriebene AAPI-Paket, sollten die Fähigkeit besitzen, mit Agenten eines Systems, welches z. B. in Telescript oder Agent Tcl implementiert ist, zu kommunizieren.

9.4.2.2 Indirekte Kommunikation

Trader, Exporteur und Importeur

Die indirekte Kommunikation zwischen Agenten erfolgt über eine zwischengeschaltete Instanz. Solch eine Instanz nennt man *Trader* und ist ein Objekt, mit dessen Hilfe Dienste vermittelt werden [ISO95] . Trading kann als eine Erweiterung des klassischen Client/Server-Paradigmas verstanden werden, wobei die Erweiterung aus der Trader-Komponente und ihrer Einbindung in den Kommunikationsprozeß besteht. Der Trader tritt als Dienstvermittler zwischen *Exporteur* und *Importeur* auf. Innerhalb eines Agentensystems handelt es sich bei einem Exporteur um einen (Server)-Agenten, der einen Dienst zur Verfügung stellt, und bei einem Importeur um einen (Client)-Agenten, der einen Dienst entgegennimmt und nutzt.

Abb. 9.4-1: Prinzip der Dienstvermittlung [ISO97] .

Nachfolgend soll die in Abb. 9.4-1 skizzierte Wechselwirkung zwischen Trader, Importeur und Exporteur erläutert werden:

- Ein Trader akzeptiert *Dienstangebote* von Exporteuren. Das Dienstangebot beschreibt dabei die Charakteristik des Dienstes, den der Exporteur anbieten möchte. Ein Trader nimmt dieses Dienstangebot entgegen und verwaltet es in seiner zentralen oder dezentralen Datenbank
- Ein Trader akzeptiert an ihn gestellte *Dienstanfragen* seitens eines Importeurs. Dabei beschreibt die Dienstanfrage die Anforderungen des Importeurs an den gesuchten Dienst.
- Auf eine Dienstanfrage hin durchsucht der Trader seine Datenbank, um der Dienstanfrage genügende Dienstangebote zu finden. Diese werden dem Importeur als Ergebnis der Dienstvermittlung übergeben.
- Der Importeur kann, nachdem ihm der Trader passende Dienstangebote vermittelt hat, mit den Exporteuren der Dienstangebote interagieren und den Dienst nutzen. Dies geschieht über den *Dienstaufruf* und die *Diensterbringung*.

Dienstaufruf/Diensterbringung

Die indirekte Kommunikation mittels Trader erlaubt es einem Importeur, dem Client-Agenten, durch einen Trader zu seiner Laufzeit einen Dienst zu suchen und zu nutzen. Dabei muß der Exporteur des gesuchten Dienstes dem Importeur nicht zuvor bekannt sein. Während der Export eines Dienstes, die Dienstanfrage und die Dienstvermittlung über direkte und i. d. R. globale Kommunikationsmechanismen realisiert sind, kann der Dienstaufruf und die Diensterbringung innerhalb von Agentensystemen auch die Migration von Importeur zu Exporteur oder umgekehrt mit anschließender lokaler, direkter Kommunikation bedeuten. Kriterien für die Migrationsrichtung sind dann die Plattformunabhängigkeit von $D_1, ..., D_n$, sowie die Verfügbarkeit und Mächtigkeit von P_{Clie} und P_{Serv} (siehe Abb. 9.3-6).

9.4.3 Migration von Mobilen Agenten

Der vom Mobilen Agenten ausgelöster Migrationsprozess unterbricht die Ausführung des Agenten auf dem besuchten Host. Der Mobile Agent wird komprimiert und über einen Kommunikationskanal zu einem anderen Host verschickt um dort seine Arbeit wieder aufzunehmen. In [ER01] wird die Migrationsfähigkeit von Mobilen Agenten mit drei verschiedenen Aspekten klassifiziert: Die Sicht des Programmierers, die Sicht des Agenten und Aspkete der Systemumgebung. Aus Sicht des

Programmierers kann der Migrationsprozess in zwei verschiede Bereiche unterteilt werden. Eine starke Migrationsfähigkeit impliziert dass die Daten und der Ausführungsstatus eines Mobilen Agenten zusammen mit seinem Code migriert werden und die Arbeit auf dem neuen Host direkt mit der nächsten Instruktion fortgesetzt wird. Schwache Migrationsfähigkeit liegt dann vor, wenn nur die Daten zummen mit dem Agenten transferiert werden können, die Ausführung kann nicht an der nächsten Stelle fortgesetzt werden, jedoch wird eine vorgegebene Methode ausgeführt mit der der Agent wieder aktiviert wird. Aus Sicht des Agenten muss die günstigste Übertragungsstrategie gewählt werden um seinen Code, die Daten und den Ausführungsstatus an den nächsten Host zu übertragen. Die Übertragungsstrategie bestimmt den Zeitpunkt der Migration und wählt den Zielhost aus. Dabei ist es auch möglich, dass sich der Agent dupliziert und zu mehreren Zielen wandert. Der letzte Aspekt ist die tatsächliche Datenübermittlung. Abb. 9.4-2 zeigt die verschiedenen Schritte während einer Migration.

Starke Migrationsfähigkeit

Schwache Migrationsfähigkeit

Abb. 9.4-2: Migrationsaspekte

9.5 Arbeiten auf dem Gebiet mobiler Agenten

Stellvertretend für andere Arbeiten zum Thema mobiler Agenten soll ein Querschnitt über bestehende Systeme vorgestellt werden. Das *Aglet*-Konzept von IBM ist ein Beispiel für ein kommerziell entwickeltes Agentensystem. Verbreiteter sind jedoch Arbeiten die der Forschung einzelner Universitäten entstammen. Ein Beispiel hierfür ist das am Dartmouth College entwickelte Agentensytem *Agent Tcl*. Die Systeme *Mole* und *JAE* dokumentieren den Forschungsstand an deutschen Hochschulen.

Die meisten vorgestellten Systeme basieren auf der Sprache Java. Lediglich das Agentensystem Agent Tcl nutzt die Sprache Tcl .

9.5.1 Aglets

Das IBM-Aglet-Konzept [IBM96] ist eine Erweiterung der mobilen Java Applets. Wie bei einem Applet migriert der Bytecode der Agletklasse durch das Netz. Im Gegensatz zu Applets wird jedoch auch der Zustand eines Aglets übertragen. Damit kann das Aglet seine Arbeit in einer neuen Umgebung ausgehend von seinem letzten Zustand aufnehmen. Folgende Ereignisse im Leben eines Aglets legen seinen Zustand fest: Erschaffung, Duplizierung, Reise, Aktivierung, Deaktivierung und Terminierung.

Zustand eines Aglets

Entsprechend den verschiedenen Zuständen, in denen sich ein Aglet befinden kann, wird dem Aglet-Anwender eine Schnittstelle geboten, die eine Überarbeitung des Aglets in bestimmten Phasen seines Lebens ermöglicht.

Das Migrationsverhalten eines Aglets ist schwach und erfolgt mit Hilfe der Objekt-Serialisierung aus dem *Java Development Kit*. Der Bytecode der Agletklasse wird über das *Java-ClassLoader*-Konzept auf dem Zielrechner verfügbar gemacht.

Kommunikation zwischen Aglets

Die Kommunikation zwischen Aglets kann direkt sowohl lokal als auch global erfolgen. In beiden Fällen werden Stellvertreter, sog. *AgletProxies*, geschaffen, über die die Kommunikation stattfindet. Als Kommunikationsmechanismen sind RPC und Message Passing vorgesehen. Während mit der Hilfe von RPC ein Agent eine öffentliche Methode eines anderen Agenten aufrufen kann, wird über das Message Passing ein datenorientierter Kommunikationsmechanismus zum Transfer von Daten zwischen Prozessen geboten. In beiden Fällen dient der AgletProxy dem Aglet bei der Kommunikation als Schutz vor unbefugter Manipulation. Die Unterstützung einer indirekten Kommunikation sieht das Konzept zunächst nicht vor. Eine Verwaltung des Systems erfolgt über eine zentrale Instanz.

Prinzipal

Privilegien

Ein Schutz einzelner Komponenten des Agentensystems erfolgt über das Definieren von Prinzipalen und ihren Privilegien [KADA et al. 98] . Ein Prinzipal ist eine Entität, die authentifizierbar ist und Zugang zum Agentensystem hat oder Teil desselben ist. Prinzipale können Aglets, Agleterzeuger, Agletbesitzer, Arbeitsumgebungen etc. sein. Privilegien sind Rechte die den einzelnen Prinzipalen zugestanden werden. Dabei können Prinzipale ihre Privilegien untereinander beschränken. Sieht eine Arbeitsumgebung für ein Aglet restriktivere Operationen vor als der Agleterzeuger, so muß das Aglet dies akzeptieren. Andernfalls wird seine Arbeit unterbunden.

Dieses Modell ist unter Nutzung der Java Komponenten *Verifier* und *Security-Manager* realisiert. Während der Verifier jeden eingehenden externen Programmcode auf sein korrektes Format untersucht und so sicherstellt daß die Applikation zur Laufzeit in keinen unerwünschten Zustand übergeht, können mit Hilfe des Security Managers objekt-, klassen- oder threadweise Restriktionen bzgl. einzelner Operationen vergeben werden. Die Virtual Machine, der Java Bytecode Interpreter, kann dann zur Ausführungszeit einzelner Programmteile prüfen, ob die gewünschte Operation mit den vergebenen Rechten konform geht. Damit bietet das IBM-Aglet

Sicherheitskonzept den ausführenden Arbeitsumgebungen Schutz vor Agenten mit einer fremden Identität.

9.5.2 D'Agents

Das System D'Agents geht auf das am Dartmouth College entwickelte System *Agent Tcl* zurück. Die aktuelle Version D'Agent 2.1 erlaubt es dem Programmierer Mobile Agenten in verschiedenen Sprachen (z. B. Tcl, Java and Scheme) zu erstellen. Es wird eine hochperfomate, multi-threaded Architektur bereitgestellt, die eine starke Mobilität (in Tcl und Java) ermöglicht und durch verschiedene Sicherheitsmechanismen den Host gegen bösartige Agenten schützt.

Abb. 9.5-1: Protokollstack D'Agents

Die unterste Ebene im Protokollstack (Abb. 9.5-1) beschreibt den Transportmechanismus über TCP/IP. Darüber nimmt ein Serverprozess ankommende Agenten entgegen, authentifiziert die Identität des Besitzers und stellt dem Agenten definierte Ressourcen zur Verfügung. Danach werden die Komponenten des Agenten (z. B. Code, Daten und Ausführungsstatus) ausgepackt und in die Ausführungsumgebung injiziert. Hier wird der Agent mit allen zugehörigen Komponenten verwaltet bis dieser eine Migration zu einem anderen Host übernimmt. Jeder Server stellt den Agenten dazu Dienste zur Kommunikation und Migration bereit. Eine gemeinsam verwendete C/C++ Bibliothek implementiert die D'Agent Funktionalitäten, für jede unterstützte Sprache wird für die Agenten selbst eine Ausführungsumgebung bereitgestellt. In der zweiten Ebene steht die virtuelle Ausführungsumgebung für jede Sprache (z. B. Scheme, Java, oder Tcl), in der Stubs den Agenten die Ausführung von Funktionen in der gemeinsamen Bibliothek erlauben, ein Statusmodul, das die Mobilität des Agenten unterstützt, und ein Sicherheitsmodul, das die angebotenen Resourcen verwaltet. D'Agent unterstützt die starke Mobilität für die Sprachen Tcl und Java und migriert damit den Stack und lokale Variable.

9.5.3 Mole

Mobiles Agentensystem

Mole [STBA et al. 96] , ist der Prototyp eines mobilen Agentensystems, welches seit Juni 1996 verfügbar ist und an der Universität Stuttgart entwickelt wird. Stationäre *System-Agenten* verwalten Ressourcen und Dienste einer Arbeitsumgebung und stellen diese mobilen User-Agenten zur Verfügung. Die Arbeitsumgebungen für System- oder *User-Agenten* sind Places, wobei jeder Rechner mehrere Places ausführen kann. Dieses Dienstvermittlungskonzept erfüllt jedoch nur eingeschränkt die Anforderungen an eine indirekte Kommunikation, da ein mobiler Agent (Client-Agent) immer erst einen *Place* aufsuchen muß um dessen lokal verfügbare Dienste zu erfragen. Eine tatsächliche Vermittlung findet nicht statt.

Bei der direkten Kommunikation unterstützt Mole sowohl die lokale als auch die globale Kommunikation. Entsprechend dem IBM-Aglet-Konzept sind als Kommunikationsmechanismen RPC und Message Passing vorgesehen. Während RPC einer lokalen, synchronen Inter-Agenten-Kommunikation dient, wird über den Message Passing-Mechanismus eine globale asynchrone Inter-Agenten-Kommunikation erreicht. Die Verwaltung der globalen Inter-Agenten-Kommunikation erfolgt zentral. Da es in dem Agentensystem keine Komponente gibt, die über den Aufenthaltsort jedes Agenten informiert ist, legt ein stationärer *Mailbox*-Agent alle eingehenden Nachrichten ab und leitet diese bei Anfrage eines mobilen Agenten weiter.

In Mole werden Agenten als Objektgruppen aufgefaßt. Objektgruppen dürfen keinerlei Referenzierungen auf fremde Objekte haben. Dieses Insel-Konzept vereinfacht den zur Überführung eines Agenten notwendigen Persistenzmechanimus. Auch Mole verfolgt eine schwache Migration.

Mole-Sicherheitskonzepte

Mole konzentriert sich bzgl. seiner Sicherheitskonzepte auf den Schutz eines Agenten vor böswilligen Arbeitsumgebungen. Unter dem Begriff *Time limited Blackbox Protection* [HOH98] wird die Spezifikation eines Agenten über eine Umrechnung in einen ausführbaren Agenten (*Blackbox*) überführt, der für ein bestimmtes Zeitintervall weder lesbar noch modifizierbar ist. Die Umrechnung besorgt ein Vermischungsalgorithmus (*Mess-up Algorithm*) der Daten, Anweisungen und Kontrollfluß-Elemente eines für den Menschen deutbaren Binärcodes derart anordnet und mit irreführenden Bezeichnungen versieht, daß in begrenzter Zeit keinerlei Interpretation des Agenten möglich ist. Um eine langfristige Analyse eines auf diese Weise entstellten Agenten zu unterbinden werden zur Laufzeit des Agenten notwendige Variablen und Anweisungen von einer vertrauenswürdigen Arbeitsumgebung angefordert. Ein, an eine synchronisierte Uhr gekoppeltes Verfallsdatum macht den Agenten nur für einen vorgegebenen Zeitraum ausführbar.

Dieses Schutzkonzept soll zukünftig implementiert werden und innerhalb des Systems Mole die Agenten vorrangig vor Leseangriffen schützen. Einer Reihe weiterer denkbarer Angriffe bleibt der Agent auch jetzt noch ausgesetzt.

Da dieser Ansatz die Umsetzung einer synchronisierten Uhr erfordert und auf jeder Station des Agenten die Interaktion mit einer vertrauenswürdigen Umgebung not-

wendig ist, erscheint er schwer realisierbar und restriktiv. Ein weiterer Nachteil ist seine Applikationsabhängigkeit. Es ist für jede Spezifikation eines Agenten neu zu entscheiden welche Anweisungen zur Laufzeit integrierbar sind und zu welchem Zeitpunk diese anzufordern sind.

9.5.4 AAPI-Paket

Das Agent Application Programming Interface (AAPI)-Paket ist eine Erweiterung der Java-Klassenbibliothek und wird seit 1997 an der Fernuniversität Hagen entwickelt [WES98] . Es unterstützt eine schwache Migration mobiler Agenten. Zur Kommunikation der Agenten werden dem Entwickler lokale synchrone und asynchrone Kommunikationsmechanismen zur Verfügung gestellt. Eine indirekte Kommunikation erfolgt mit Hilfe der Trader-Komponente Contact. Im Gegensatz zu den anderen vorgestellten Systemen wird mit dem AAPI-Paket ein dezentrales Agentensystem aufgespannt.

Repressiver und präventiver Schutz

Das AAPI-Paket schützt die Arbeitsumgebung für Agenten und ihren Rechner vor böswilligen oder fehlerhaft programmierten Agenten, indem es präventiv sicherheitskritische Operationen mobiler Agenten unterbindet. Weiter schützt es präventiv einen Agenten vor böswilligen Arbeitsumgebungen, indem es die Einsicht aller für die gegenwärtige Ausführung irrelevanten Daten verhindert. Es schützt ihn repressiv, indem es evtl. Manipulationen des Agenten und seiner mitgeführten Daten erkennbar macht. Unterschieden werden hierbei die statischen und die dynamischen Objekte des Agenten. Veränderbare dynamische Objekte wie die Route [WESC et al. 99] oder die mobilen Daten eines Agenten werden durch den Einsatz kryptographischer Verfahren geschützt, konstante Objekte hingegen, wie das Profil des Agenten oder sein Binärcode werden repressiv durch den Einsatz kryptographischer Protokolle geschützt.

Entwicklungswerkzeug für AAPI-Agenten

Mit der Agent Local Help Application (ALOHA) existiert ein Entwicklungswerkzeug für AAPI-Agenten. ALOHA soll den Agentenprogrammierer weitgehend von Detailkenntnissen des AAPI-Pakets befreien und so für kürzere Entwicklungszeiten bei der Erstellung mobiler Agenten sorgen. Neben einem Individualisierungs-Editor für Agenten, mit dem auf intuitive Weise das Erstellen verschiedener Agententypen unterstützt wird, kann mit Hilfe eines Migrations-Simulators das Reiseverhalten eines Agenten rechnerlokal nachgestellt werden. Gleichzeitig enthält ALOHA einen Docking-Mechanismus, über den individualisierte Agenten in einem Depot gesammelt und empfangen werden können. ALOHA entbindet den Agentenprogrammierer damit zusätzlich von der Erstellung einer Arbeitsumgebung für Agenten.

9.5.5 JAE

Technologie in drahtloser Umgebung

JAE (Java Agent Environment) [PAKÜ et al. 97] ist ein an der technischen Hochschule Aachen konzipiertes Agentensystem. Das Augenmerk dieses Systems liegt in der Integration von Agententechnologien in drahtloser Umgebung mit Agenten-

technologien der Internet-Umgebung. Denkbare mobile Systeme in diesem Anwendungsfeld sind das Mobiltelefon, der PDA (persönlicher digitaler Assistent) sowie das Notebook.

Unterschieden wird in JAE zwischen *Agenten-Servern*, *mobilen Agenten* und *stationären Service-Agenten*. *Agenten-Server* sind die Arbeitsumgebungen eines Agenten auf einem Rechner. *Service-Agenten* sind auf einem Rechner beheimatet und führen bei Anfrage eines mobilen Agenten Dienstleistungen für diesen aus. Die *mobilen Agenten* selber können sich von Agenten-Server zu Agenten-Server bewegen.

Ein Prototyp dieses Agentensystems deckt die Bereiche Zugriff auf Internetdienstleistungen und Mobilität des Agenten ab. Über Service-Agenten kann ein leichter Zugriff auf verschiedene Internet-Dienste erfolgen.

JAE bietet zur Zeit keine Sicherheitskonzepte, weder zum Schutz der Arbeitsumgebung noch zum Schutz des Agenten vor böswilligen Arbeitsumgebungen.

9.6 Überblick

Die vorgestellten Systeme haben das konzeptionelle Stadium verlassen und befinden sich in der Phase prototypischer Realisierung. Während Agent Tcl und JAE einen Schwerpunkt auf Mechanismen zur Migration von Agenten bei temporären Netzverbindungen legen, überzeugt der IBM-Ansatz durch die Einteilung eines Agentenlebens in einzelne Lebensphasen und die dadurch erwachsende Anwender-Schnittstelle zur Agentenindividualisierung.

Schutz der Arbeitsumgebung

Ist der Schutz der Arbeitsumgebung vor potentiell böswilligen Agenten in den Systemen IBM-Aglets und Agent Tcl gewährleistet, so bleiben Sicherheitsdienste zum Schutz des Agenten vor böswilligen Arbeitsumgebungen in diesen Systemen unbehandelt. Im System Mole soll ein Ansatz zum Schutz des Agenten gegenüber Leseangriffen integriert werden. Das AAPI-Paket verfolgt Sicherheitsdienste zum Schutz der mobilen Daten und der Route eines Agenten. Leseangriffe auf den Binärcode können zwar nicht verhindert werden, jedoch kann im Falle eines funktionsmodifizierenden Angriffs das Senden des korrekten Binärcodes erzwungen werden.

	Aglets	D'Agents	Mole	AAPI-Paket	JAE
schwache Migration	+	+	+	+	+
Lebenszyklus-API	+			+	
direkte lokale Komm.	+	+	+	+	+
direkte globale Komm.	+	+	+		+
indirekte Kommunikation		?	(+)	+	+
Sicherheit Agent			(+)	(+)	
Sicherheit Rechner	+	+		+	
Temporäre Verbindung		+		+	+
dezentrale Verwaltung				+	
plattformunabhängig	+		+	+	+

+ : = ist Eigenschaft des Agentensystems.
(+) : = ist eingeschränkte Eigenschaft des Agentensystems.
? : = keine Aussage.

Abb. 9.6-1: Leistungsumfang existierender mobiler Agentensysteme.

Fähigkeit der direkten Kommunikation

Die Fähigkeit einer direkten lokalen bzw. globalen Kommunikation, gleichwohl ob synchron oder asynchron, ist in fast allen vorgestellten Systemen vorhanden bzw. vorgesehen. Einzig das AAPI-Paket bietet keinen Mechanismus zur globalen direkten Kommunikation. Eine indirekte Kommunikation unterstützt in vollem Umfang nur das Agentensystem JAE. Für Agent Tcl konnte diesbezüglich keine Aussage getroffen werden.

Alle Systeme unterstützen die schwache Migration der Agenten und die meisten Systeme legen eine zentrale Struktur mit einem oder mehreren Servern zugrunde. Lediglich das AAPI-Paket verfolgt einen dezentralen Ansatz.

Der Leistungsumfang der vorgestellten Systeme ist in Abb. 9.6-1 dargestellt. Da sich alle Agentensysteme in der Entwicklung befinden, soll die Tabelle nur als Anhaltspunkt dienen.

Lösungshinweise zu Selbsttestaufgaben

Lösung zu Selbsttestaufgabe 1.3-1:

1. Ein Protokoll besteht aus den Regeln (einschließlich den zeitlichen Bedingungen), nach denen der logische Meldungsaustausch zwischen zwei gleichen Schichten von Kommunikationssystemen abgewickelt wird.

2. Beim Sender sowie beim Empfänger werden die Schichten 1-4 durchlaufen. Zusätzlich wird in jeder Vermittlungsstelle der Protokollstack der Schichten 1-3 abgearbeitet. Insgesamt kommt man damit bei 2 Vermittlungsstellen auf 20 Schichten .

Lösung zu Selbsttestaufgabe 1.4-1:

Am Eingang des Leitungscodieres liegt die zufällige Binärfolge

$$a_n = \texttt{1110 0001 0101}$$

an. Die Codierung mittels AMI-Codierer bzw. CMI-Codierer ist im Bild zu sehen.

Lösung zu Selbsttestaufgabe 1.4-2:

1. Berechnung des Kontrollwortes mittels modulo-2 Arithmetik

   - $2^n \cdot M = 2^5 \cdot M =$ `101000110100000`
   - $\frac{2^5 \cdot M}{G}$:

```
101000110100000 : 110101 = 1101010110
110101
 111011
 110101
   111010
   110101
     111110
     110101
       101100
       110101
        110010
        110101
           1110
```

   Die Division liefert:

   - den Quotienten $Q =$ `1101010110`
   - den Rest $R =$ `01110`
   - die gesicherte Bitfolge lautet somit: $T =$ `101000110101110`

2. Mittels Polynomschreibweise

   - $2^n \cdot M = 2^5 \cdot (x^9 + x^7 + x^3 + x^2 + 1) = (x^{14} + x^{12} + x^8 + x^7 + x^5)$
   - Bei der Division in Polynomschreibweise ist zu beachten, daß auch hier die modulo2-Arithmetik zu verwenden ist.

$$
\begin{array}{l}
(x^{14}+x^{12}+x^{8}+x^{7}+x^{5}) : (x^{5}+x^{4}+x^{2}+1) = x^{9}+x^{8}+x^{6}+x^{4}+x^{2}+x^{1} \\
\underline{(x^{14}+x^{13}+x^{11}+x^{9})} \\
\qquad x^{13}+x^{12}+x^{11}+x^{9}+x^{8}+x^{7}+x^{5} \\
\qquad \underline{(x^{13}+x^{12}+x^{10}+x^{8})} \\
\qquad\qquad x^{11}+x^{10}+x^{9}+x^{7}+x^{5} \\
\qquad\qquad \underline{(x^{11}+x^{10}+x^{8}+x^{6})} \\
\qquad\qquad\qquad x^{9}+x^{8}+x^{7}+x^{6}+x^{5} \\
\qquad\qquad\qquad \underline{(x^{9}+x^{8}+x^{6}+x^{4})} \\
\qquad\qquad\qquad\qquad x^{7}+x^{5}+x^{4} \\
\qquad\qquad\qquad\qquad \underline{(x^{7}+x^{6}+x^{4}+x^{2})} \\
\qquad\qquad\qquad\qquad\qquad x^{6}+x^{5}+x^{2} \\
\qquad\qquad\qquad\qquad\qquad \underline{(x^{6}+x^{5}+x^{3}+x^{1})} \\
\qquad\qquad\qquad\qquad\qquad\qquad x^{3}+x^{2}+x^{1}
\end{array}
$$

3. Die Umsetzung dieser Polynome in die binäre Schreibweise liefert die selben Ergebnisse für Q, R und T wie bereits oben errechnet.

Lösung zu Selbsttestaufgabe 1.6-1:

Es gibt folgenden Zustände:

- Zustand 0 = keine Nachricht unquittiert
- Zustand 1 = 1 Nachricht unquittiert
- Zustand 2 = 2 Nachrichten unquittiert
- Zustand 3 = 3 Nachrichten unquittiert

Die Zustandsübergänge sind im Bild zu sehen, wobei

- N = Nachricht wird gesendet
- Q = Quittung wird gesendet

Lösung zu Selbsttestaufgabe 2.1-1:

1. Wie die entsprechenden Namen schon andeuten, wird bei der verbindungsorientierten Nachrichtenübermittlung für die Dauer der Übertragung zwischen den Endeinrichtungen eine Verbindung aufgebaut, die für den Transfer mit ihrer gesamten Bandbreite und ohne Unterbrechung zur Verfügung steht. Bei der verbindungslosen Datenübertragung werden die Daten dem Transportnetzwerk mit einer Adresse übergeben und dieses kümmert sich um die Beförderung. Dabei können gleichartige Nachrichten je nach Auslastung des Netzes durchaus unterschiedliche Wege nehmen und verschiedene Durchlaufzeiten aufweisen.
2. X.25 ist ein paketorientiertes Protokoll, X.21 wird bei der verbindungsorientierten Durchschaltevermittlung eingesetzt.
3. Die Datenübertragung nach X.21 kann in die Phasen Verbindungsaufbau, Datenübermittlung und Verbindungsabbau untergliedert werden.
4. Zuordnung der Zustände zu den Verbindungsphasen

- Verbindungsaufbau: Verbindungsanforderung, Wahlaufforderung, Wählzeichen, Verbindung im Aufbau, B-Tln empfangsbereit (Zustände 1-6)
- Datenübertragung (Zustand 7)
- Verbindungsabbau: Auslöseanforderung, Auslösebestätigung (Zustände 8-9)

Lösung zu Selbsttestaufgabe 2.2-1:

1. Unter der Transparenz eines Netzes versteht man die Möglichkeit alles an Daten zu übertragen, was durch die physikalische Definition erlaubt ist. Man könnte sich beispielsweise vorstellen, daß einige 8-Bit Zeichen als spezielle Steuercodes verwendet werden und somit für die Datenübertragung nicht eingesetzt werden können, da eine Unterscheidung nicht möglich ist. Genau diese Einschränkung ist bei einem transparenten Netz nicht vorhanden.

2. Bei sporadisch auftretenden Daten besteht das Problem darin, daß eine Synchronisation mitgesendet wird, damit der Empfänger erkennt, wann ein Datenpaket übertragen wird. Für diese Synchronisation muß ein spezielles Codewort reserviert werden, was dem Prinzip des transparenten Netzes zunächst widerspricht. Die Lösung dieses Problems ist durch das Zero Insertion Verfahren erreicht, das in alle Wörter, in denen fünf oder mehr Einser hintereinander auftreten, nach fünf Einsen eine Null einfügt.

3. Betrachtet man die Gesamtzeit, die für die Übertragung einer Nachricht benötigt wird, so beträgt der Anteil, der für das Senden von Nutzdaten verwendet wird im Durchschnitt $\frac{t_p}{t_c}$. Vergrößert man nun die Paketlänge,

so wird für $t_p \gg t_q$ der Quotient $\frac{t_p}{t_c} \approx 1$; d. h., daß durch Vergrößern der Paketlänge der Durchsatz erhöht werden kann.

Ein Quittierungsverfahren wird in der Regel dann eingesetzt, wenn man keinen fehlerfreien Kanal zur Verfügung hat und damit rechnen muß, daß einige Pakete fehlerhaft oder überhaupt nicht ankommen. Daher ist die Verwendung sehr großer Pakete nicht praktikabel, da die Wahrscheinlichkeit, daß ein Fehler auftritt mit der Paketlänge steigt. Das Ziel also ist es, ein optimales, an die Übertragungsstrecke angepaßtes Verhältnis zu wählen.

Lösung zu Selbsttestaufgabe 2.4-1:

1. X.21 wird auf Schicht 1, HDLC auf Schicht 2 und X.25 auf Schicht 3 angesiedelt. Diese Hierarchie ist ein Beispiel, wie auf Basis der normalen Leitungsvermittlung Paketvermittlung betrieben werden kann. Man kann sich das X.25-Netz vereinfacht als ein Netz von Transportknoten vorstellen, die alle über normale Telefonleitungen miteinander verbunden sind und über diese ihre Daten übertragen. Es ist denkbar hier sowohl Festverbindungen zu installieren, als auch Wählverbindungen vorzusehen, die nur bei Bedarf aufgebaut werden.

2. Die lokale Numerierung erlaubt eine Beschränkung auf verhältnismäßig geringe Anzahl erlaubter Nummern. Bei einer netzweit eindeutigen Numerierung müßten zum einen diesbezüglich globale Abspachen getroffen werden, die den Protokolloverhead unnötig erhöhen, zum andern müßte in jedem Paketkopf der Platz für eine LCN-Codierung reserviert werden, die der maximalen Anzahl von Nummern im Netz entspricht.

Lösung zu Selbsttestaufgabe 3.1-1:

Die Eigenschaften und Merkmale von HSLANs und MANs sind:

- geographische Ausdehnung von 10 bis mehreren 100 km, häufig sogar unbeschränkte Ausdehnung
- Höhere Datenraten als IEEE-LANs: 100 bis mehrere Gbit/s
- Diensteintegration: Integration von asynchronem (paketvermitteltem) und isochronem (leitungsvermitteltem) Verkehr
- Unterstützung von Echtzeitanwendungen durch Integration von isochronem Verkehr
- Der Bitstom ist in Zellen bzw. Slots aufgeteilt
- Typische Vertreter: FDDI, FDDI II, DQDB, ATM-Ring, CRMA, CRMA II, ATM-Ring

Lösung zu Selbsttestaufgabe 3.2-1:

Für 100 Terminals erhalten wir folgende Ankunftsrate λ für das Gesamtsystem:

$$\lambda = 100 \cdot 2 \cdot \frac{1}{60} Pakete/s = \frac{10}{3} Pakete/s$$

Die Paketlänge umfaßt 96 Byte = 768 bit. Die Übertragungszeit t_p für ein Paket beträgt somit:

$$t_p = \frac{768bit}{64bit} s = 12 \cdot 10^{-3} s = 12ms$$

Für den normierten Durchsatz $\overline{D} = t_p \cdot D$ erhalten wir:

a) Pure Aloha:

$$\lambda \cdot P \cdot \tau = \frac{10}{3} \cdot 12 \cdot 10^{-3} = 40 \cdot 10^{-3}$$
$$\overline{D} = \lambda \cdot P \cdot \tau \cdot e^{-2\lambda \cdot P \cdot \tau} = 0,04 \cdot e^{-0,08} = 3,69\%$$

b) Slotted Aloha:

$$\lambda \cdot P \cdot \tau = \frac{10}{3} \cdot 12 \cdot 10^{-3} = 40 \cdot 10^{-3}$$
$$\overline{D} = \lambda \cdot P \cdot \tau \cdot e^{-\lambda \cdot P \cdot \tau} = 0,04 \cdot e^{-0,04} = 3,84\%$$

Lösung zu Selbsttestaufgabe 3.3-1:

a) Pro Station wird eine Verzögerung von einem Bit verursacht. Dies entspricht einer Verzögerung von $0,25\mu s$. Bei 100 Stationen sind dies zusammen $25\mu s$. Die Monitorstation verursacht eine Verzögerung von 24 Bit, d. h. $6\mu s$. Die Signallaufzeit beträgt $42\mu s$. Somit verweilt ein Bit $73\mu s$ im Ring. Die Tokenumlaufzeit beim Ring beträgt somit $(73 + 6)\mu s = 79\mu s$.

b) Eine Meldung mit 1024 Byte Daten und 6 Byte Adresslänge besteht insgesamt aus (1024+21) Byte = 1045 Byte. Sie verweilt $(\frac{1045 \times 8}{4} + 73)\mu s = 2,163ms$ auf dem Ring.

Lösung zu Selbsttestaufgabe 3.4-1:

a) Die Slots eines gegenläufigen Doppelrings mit *Slot Reuse* können im Mittel viermal in einem Umlauf wiederverwendet werden: Bei gleichverteilt sendenden Stationen auf dem Ring ist die maximale Entfernung zwischen zwei Stationen in beiden Richtungen jeweils eine halbe Ringlänge. Die mittlere Entfernung zwischen 2 Stationen beträgt also ein Viertel der Ringlänge, so daß die Slots im Mittel viermal in einen Umlauf freigegeben werden.

Durch die auf beiden Ringen jeweils im Mittel 4-fach wiederverwendbaren Slots beträgt die theoretische Kanalkapazität auf beiden Ringen zusammen insgesamt das 8-fache der nominellen Übertragungsgeschwindigkeit.

b) Beim CRMA-II kann eine Nachricht, nachdem ihre Übertragung begonnen wurde, in jedem Fall zu Ende gesendet werden, während im ATM-Ring eine Station von einer Nachricht nur so viele Zellen senden darf, bis ihr zuvor festgelegtes Guthaben erschöpft ist.

Im CRMA-II werden die gegebenenfalls zur Übertragung einer Nachricht nötigen zusätzlichen *gratis slots* von der Station selbst erzeugt, während die ankommenden besetzten Slots verzögert werden. Die Verallgemeinerung eines solchen Verfahrens wird auch *Buffer Insertion*-Verfahren genannt, da ein zusätzlicher Puffer in den Datenstrom eingefügt wird.

Beim ATM-Ring beruht der Medienzugriff auf dem *Slotted Ring*-Prinzip: Auf dem Ring kreisen Slots in festen Abständen; im ATM-Ring handelt es sich hierbei um die ATM-Ring Zellen. Auf diese Zellen dürfen die Stationen nur solange zugreifen, bis ihr zuvor festgelegtes Guthaben erschöpft ist.

c) Beim CRMA-II sind zur Übertragung einer 53 Byte langen ATM-Zelle 17 ADUs, also 68 Byte notwendig, der zusätzliche Overhead beträgt demnach 28,3 Prozent.

Im ATM-Ring entsprechen die Zellen genau dem ATM-Zellenformat, es ist demnach zumindest in Bezug auf den Transport der Daten kein zusätzlicher Overhead erforderlich.

Lösung zu Selbsttestaufgabe 4.2-1:

Abb. 4.2-23: PCM-Sprachübertragung

t_L ist die Laufzeit, τ die Schutzzeit. Man kann aus der obigen Abbildung die folgende Gleichung ablesen

$$n \cdot 125\mu s = 2 \cdot t_L + 2 \cdot \frac{(2 + 10 \cdot n)\, Bit}{256\, kbit/s} + 2 \cdot \tau,$$

wobei $t_L = 6\mu s/km \cdot l$, $\tau = 5\mu s$

$$\begin{aligned} n \cdot 125 &= 2 \cdot 6 \cdot l/km + 2 \cdot \tfrac{2+10 \cdot n}{256} \cdot 10^3 + 2 \cdot 5, \\ 125 \cdot n &= 12 \cdot l/km + 78,125 \cdot n + 25,625, \\ 46,875 \cdot n &= 12 \cdot l/km + 25,625. \end{aligned}$$

Man löst die letzte Gleichung nach *l/km* auf,

$$l/km = \frac{46,875}{12} \cdot n - \frac{25,625}{12}$$

Es ergibt sich die folgende Wertetabelle:

n	1	2	3	4	5	6	7	8
l/km	1,8	5,7	9,6	13,5	17,4	21,3	25,2	29,1

Abb. 4.2-24: maximal überbrückbare Leitungslänge

Bei einer Verzögerung von 0,5 *ms* ist *n* = 4. Daraus ergibt sich die maximal-überbrückbare Leitungslänge *l* = 13,5 *km*.

Lösung zu Selbsttestaufgabe 4.2-2:

Die einzelnen Synchronworte sind:

PCM30 (RK)	0 0 1 1 0 1 1
PCM120 (RK)	1 1 1 1 0 1 0 0 0 0
Token Ring (SFD)	1 0 1 0 1 0 1 1
HDLC (*Flag*)	0 1 1 1 1 1 1 0
ISDN (Barker)	$+ + + - - - + - - + -$

Für die Berechnung der Autokorrelationsfunktion betrachten wir jeweils

den Erwartungswert

$$E\{\mathbf{s}(t) \cdot \mathbf{s}(t+\tau)\}$$

zu diskreten Zeitpunkten $\tau = 0, 1, 2, \ldots$ usw.. Dazwischen verläuft die Autokorrelationsfunktion linear.

Um die Berechnungsschritte zu verdeutlichen, skizzieren wir für PCM 30(RK) die Funktionen $s(t)$ und $s(t+\tau)$ für $\tau = 0, 1, 2, \ldots, 5$ untereinander. Den Wert der Autokorrelationsfunktion erhalten wir, in dem wir, Ergodizität voraussetzend, das Integral

$$R(\tau) = \lim_{T \to \infty} \frac{1}{2T} \int_{-T}^{+T} s(t) \cdot s(t+\tau) dt$$

bilden (vgl. KT I, Kapitel 3).

Das Integral kann für diskrete τ durch das Abzählen der Flächendeckung zwischen *s(t)* und $s(t+\tau)$ direkt ermittelt werden. Anschließend normieren wir *R*(0) = 1, um einen Vergleich zwischen den verschiedenen Autokorrelationsfunktionen zu ermöglichen.

Für die PCM Rahmenkennung erhalten wir:

Abb. 4.2-25: PCM Rahmenkennung

Mit $a^2 = \frac{1}{2}$ erhalten wir

$$\begin{aligned}
R(0) &= 1 \\
R(T) &= \tfrac{1}{2} \\
R(2T) &= \tfrac{1}{4} \\
R(3T) &= \tfrac{1}{2} \\
R(4T) &= \tfrac{1}{4} \\
R(5T) &= 0
\end{aligned}$$

Die Autokorrelationsfunktion für die PCM30-Rahmenkennung sieht somit wie folgt aus:

Abb. 4.2-26: Autokorrelationsfunktion für die PCM30-Rahmenkennung

Wir können nun die Berechnung beschleunigen, in dem wir lediglich die Symbolfolgen s(0), s(t), s(2T) usw. verschoben untereinander schreiben und s(0) mit s(T), s(2T), s(3T) usw. ausmultiplizieren, um daraus R(T), R(2T) usw. zu ermitteln.

Für die PCM120 Rahmenkennung erhalten wir:

s(0)	1 1 1 1 0 1 0 0 0 0	*5R(0) = 5*
s(T)	1 1 1 1 0 1 0 0 0 0	*5R(T) = 3*
s(2T)	1 1 1 1 0 1 0 0 0 0	*5R(2T) = 3*
s(3T)	1 1 1 1 0 1 0 0 0 0	*5R(3T) = 2*
s(4T)	1 1 1 1 0 1 0 0 0 0	*5R(4T) = 1*
s(5T)	1 1 1 1 0 1 0 0 0 0	*5R(5T) = 1*
s(6T)	1 1 1 1 0 1 0 0 0 0	*5R(6T) = 0*

Die Autokorrelationsfunktion für die PCM120 Rahmenkennung sieht somit wie folgt aus:

Abb. 4.2-27: Autokorrelationsfunktion für die PCM120-Rahmenkennung

Für das HDLC Flag `0 1 1 1 1 1 1 0` erhalten wir entsprechend:

$$
\begin{aligned}
6R(0) &= 6 \\
6R(T) &= 5 \\
6R(2T) &= 4 \\
6R(3T) &= 3 \\
6R(4T) &= 2 \\
6R(5T) &= 1 \\
6R(6T) &= 0
\end{aligned}
$$

und somit die Autokorrelationsfunktion

Abb. 4.2-28: Autokorrelationsfunktion

Für den Token Ring *Starting Frame Delimiter* erhalten wir:

$$
\begin{aligned}
5R(0) &= 5 \\
5R(T) &= 1 \\
5R(2T) &= 3 \\
5R(3T) &= 1 \\
5R(4T) &= 2 \\
5R(5T) &= 1 \\
5R(6T) &= 1 \\
5R(7T) &= 1 \\
5R(8T) &= 0
\end{aligned}
$$

und somit die Autokorrelation

Abb. 4.2-29: Autokorrelationsfunktion

Für den beim ISDN U_{k0} verwendeten Barkercode $+++---+--+-$ erhalten wir:

$$
\begin{aligned}
11R(0) &= 11 \\
11R(T) &= 0 \\
11R(2T) &= -1 \\
11R(3T) &= 0 \\
11R(4T) &= -1 \\
11R(5T) &= 0 \\
11R(6T) &= -1 \\
11R(7T) &= 0 \\
11R(8T) &= -1 \\
11R(9T) &= 0 \\
11R(10T) &= -1 \\
11R(11T) &= 0 \\
11R(12T) &= 0
\end{aligned}
$$

und somit die Autokorrelationsfunktion:

Abb. 4.2-30: Autokorrelationsfunktion

Das sicherste Synchronwort ist die Barkerfolge, weil die Spitze der Autokorrelationsfunktion weit über den Überschwingern liegt; die Detektion wird damit einfach. Beim Token Ring erreicht die R(2T) 60 % des Wertes von R(0). Die Suche des Synchronwortes wird deshalb durch Codeverletzung im Startdelimiter (im Header) unterstützt.

Lösung zu Selbsttestaufgabe 4.4-1:

Ähnlich wie Schicht-2-Rahmen in bestimmte Felder aufgeteilt sind, sind auch die Schicht-3-Nachrichten nach einem einheitlichen Muster aufgebaut.

Abb. 4.4-11: Aufbau der Schicht3-Nachrichten

Die Nachrichten bestehen immer aus einer ganzen Anzahl von Oktetten und beginnen immer mit einem festen Nachrichtenkopf, bestehend aus Protokolldiskriminator, gefolgt von der Referenznummer, gefolgt vom Nachrichtentyp.

Dem Nachrichtentyp folgen vorgeschriebene oder wahlweise Nachrichtenelemente.

Lösung zu Selbsttestaufgabe 4.5-1:

Zu den Vorteilen der Zentralkanalsignalisierung gegenüber kanalindividuellen Verfahren zählt

- Signalisierung simultan zur Übermittlung von Nutzinformation bei bereits aufgebauter Nutzkanalverbindung möglich,
- kurze Verbindungaufbauzeit in Folge der 64-kbit/s-Signalilsierungskanäle,
- nahezu unbegrenzter Vorrat an Signalisierungselementen (Nachrichten, Nachrichtenparameter), verbunden mit
- hoher Flexibilität gegenüber neuen Anforderungen, u. a. durch die Einführung neuer Dienste und Dienstmerkmale,
- rechnerfreundliche Struktur der Signalisierungselemente,
- trotz höheren Grundaufwands im Vergleich mit kanalgebundenen Verfahren kostengünstiger, da die zentralisierten Signalisiereinrichtungen für viele Nutzkanalverbindungen verwendet werden,
- Nutzung des Signalisierungsnetzes für weitere Anwendungen außerhalb der Signalisierung (z. B. Betriebs- und Wartungsinformationen zwischen den Vermittlungsstellen und Wartungszentren),
- gesicherte Übermittlung der Signalisierungselemente.

Lösung zu Selbsttestaufgabe 5.1-1:

Typische Merkmale des ATM-Verfahrens sind:

- kurze Nachrichtenpakete (genannt Zellen) mit einer Länge von 53 Byte, bestehend aus einem Zellkopf mit 5 Byte und einem Informationsfeld mit 48 Byte.
- Bildung von virtuellen Verbindungen zwischen zwei Teilnehmern,
- Verlagerung aller Funktionen für den Transport der Zellen durch das Netz in die Schicht 2.
- Das ATM-Verfahren ist von der physikalischen Übertragung der Zellen unabhängig,
- und es werden unterschiedlichste Dienste, wie Daten-, Text- , Sprach-, Audio-, Standbild- und Bewegtbildübertragung integriert.

Lösung zu Selbsttestaufgabe 5.1-2:

Für den Anwender ergeben sich zwei wesentliche Vorteile durch die variable Bitrate:

- die Anpassung der Bitübermittlungsrate der Verbindung an die Schwankungen der Datenquelle d. h. ökonomische Nutzung der verfügbaren Bandbreite und
- die Unabhängigkeit der Endgeräteschnittstelle von der Bitrate des Dienstes, die wiederum eine von der Schnittstelle unabhängige und daher rasche Einführung neuer Dienste und spätere Anpassung an neue Komprimierungsverfahren ermöglicht.

Lösung zu Selbsttestaufgabe 5.4-1:

Die Bandbreiteneffizienz und die Nettobitrate bei der Übertragung von ATM Zellen über einen STM-1 Rahmen mittels VC-4 läßt sich folgendermaßen berechnen. Ein STM-1 Rahmen besteht aus insgesamt 2430 Byte und hat eine Rahmendauer von $125\,\mu s$. Dies entspricht einer Übertragungsrate von $155,520\,Mbit/s$. Für die Nutzlast eines VC-4 ergegeben sich $261 \times 9\,Byte = 2349\,Byte$ und somit ist die Nutzlast eines C-4 $2340\,Byte$. Dadurch ergibt sich eine Bandbreiteneffizienz von

$$\frac{2340 \cdot \frac{48}{53} Byte}{2430\,Byte} = 87,2\%$$

Die Nettobitrate berechnet sich zu

$$155,520\,Mbit/s \cdot 87,2\% = 135,63\,Mbit/s\,.$$

Lösung zu Selbsttestaufgabe 5.4-2:

a. Die Übertragung von ATM-Zellen über einen STM-4 Rahmen mittels des VC-4-4C ist folgendermaßen:

   Ein STM-4 Rahmen der SDH hat eine Rahmendauer von $125\,\mu s$ und besteht aus 9 Reihen, 270 Spalten à 4 Byte davon werden 9 Spalten à 4 Byte für den SOH des STM-4 Rahmens verwendet. Der Rahmen umfaßt also 9720 Byte, wobei 324 Byte für den SOH des STM-4 Rahmens verwendet werden. Dies ergibt eine Übertragungsrate von $622,080\,Mbit/s$. Innerhalb des STM-4 Rahmens wird der virtuelle Container VC-4-4C benutzt. Der AU-4 Pointer des STM-4 SOH zeigt auf das 1. Byte des VC-4-4C. Der VC-4-4C wird für den C-4-4C benutzt. Der Container C-4-4C enthält dann 9 Reihen und 260 Spalten à 4 Byte = 9360 Byte. Die Ausrichtung der ATM-Zellen erfolgt auf Bytegrenzen. Die Länge des C-4-4C ist kein ganzzahliges Vielfaches der ATM-Zellenlänge, so daß sich eine ATM-Zelle über zwei C-4-4C Container erstrecken kann.

b. Für die Bandbreiteneffizienz und die Nettobitrate bei dieser Übertragungsweise ergibt sich: Die Größe eines STM-4 Rahmens beträgt 9720 Byte. Der C-4-4C umfaßt 9360 Byte. Dadurch ergibt sich eine Bandbreiteneffizienz von

$$\frac{9360 \cdot \frac{48}{53} Byte}{9720 \text{ Byte}} = 87,2\%$$

Die Nettobitrate berechnet sich zu

$$622,080 \text{ Mbit/s} \cdot 87,2\% = 542,45 \text{ Mbit/s}$$

Lösung zu Selbsttestaufgabe 5.4-3:

Das HEC Feld einer PLOAM Zelle vom Typ F3 wird wie folgt berechnet: Die ersten 32 Bit des Zellkopfes

`0000 0000 0000 0000 0000 0000 0000 1001`

werden zunächst in ein Polynom von maximalem Grad 31 konvertiert:

$$M = x^3 + 1$$

Anschließend wird M mit x^8 multipliziert: $M' = M \cdot x^8 = x^{11} + x^8$. M' wird nun durch das Generatorpolynom $G = x^8 + x^2 + x + 1$ modulo 2 dividiert:

$$\begin{array}{lll} x^{11} + x^8 & : (x^8 + x^2 + x + 1) = x^3 + 1 & \\ \underline{x^{11} + x^5 + x^4 + x^3} & & \\ \quad x^8 + x^5 + x^4 + x^3 & & \\ \quad \underline{x^8 + x^2 + x + 1} & & \\ \qquad x^5 + x^4 + x^3 + x^2 + x + 1 & & \end{array} \qquad 5.4\text{-}2$$

Als Rest der Polynomdivision ergibt sich $R = x^5 + x^4 + x^3 + x^2 + x + 1$ und somit die Bitfolge der CRC Summe zu `0011 1111`. Auf die CRC Summe wird die Bitfolge `0101 0101` modulo 2 addiert:

$$\begin{array}{l} 0011\ 1111 \\ \underline{0101\ 0101} \\ 0110\ 1010 \end{array} \qquad 5.4\text{-}3$$

In das HEC Feld der ATM-Zelle wird somit der Wert `0110 1010` eingetragen.

Lösung zu Selbsttestaufgabe 6.1-1:

GSM-Netze haben eine zellulare Netzstruktur, bei der jede Zelle eine räumliche Ausdehnung von bis zu etwa 35 km hat. Durch die räumliche Aufteilung in Zellen ist es möglich, die verfügbaren Frequenzen nach gewissen Schutzabständen zeitgleich mehrfach zu verwenden.

Lösung zu Selbsttestaufgabe 6.2-1:

GSM unterscheidet zwischen Sprach- und Datendiensten. Hierbei ist auch eine Unterteilung zwischen Basis- und Zusatzdienste möglich, wobei Basisdienste nochmals in Träger- und Teledienste zerfallen. Zu den bekanntesten Telediensten gehört der Kurznachrichtendienst SMS (***S**hort **M**essage **S**ervice*).

Lösung zu Selbsttestaufgabe 6.3-1:

In einem **HLR** (***H**ome **L**ocation **R**egister*) als "Heimatdatenbank" werden die Daten der Teilnehmer verwaltet, die Kunden des Funknetzeigentümers sind. In einem **VLR** (***V**isitor **L**ocation **R**egister*) dagegen werden die Kennungen derjenigen Teilnehmer gespeichert, die sich aktuell in dem betreffenden Bereich (*Location Area*) aufhalten, aber eigentlich woanders „beheimatet“ sind.

Die **MSC**s (***M**obile **S**witching **C**enter)* bilden die eigentlichen Funkvermittlungsstellen und steuern die BS. Außerdem stellen sie auch die Übergänge zu anderen Netzen wie ISDN dar.

Lösung zu Selbsttestaufgabe 6.4-1:

Für den GSM 900 stehen für den *Down*- und *Uplink* jeweils 124 Teilbänder zur Verfügung. GSM 1800 dagegen stellt für den *Down*- und *Uplink* jeweils 374 Teilbänder bereit.

Lösung zu Selbsttestaufgabe 6.4-2:

Die Daten, die während einer Periode in einem Zeitschlitz von $0,577\,ms$ übertragen werden nennt man *Burst*. Da ein TDMA-Rahmen aus 8 Zeitschlitzen besteht, passen hier genau 8 *Bursts* hinein.

Lösung zu Selbsttestaufgabe 6.5-1:

Der Wechsel der Funkverbindung während eines bestehenden Gespräches wird als *Handover* bezeichnet. Beim Intra-Zell-Handover wird eine neue Funkverbindung zur gleichen Basisstation (BTS) aufgebaut, wenn etwa Störungen im benutzten Kanal vorliegen. Beim Inter-Zell-Handover wird eine neue Funkverbindung zu einer neuen Basisstation aufgebaut. Dies tritt z. B. ein, wenn sich Mobilstationen aus dem Sende- bzw. Empfangsbereich einer Basisstation entfernen und in den Bereich einer anderen Basisstation bewegen.

Lösung zu Selbsttestaufgabe 8.1-1:

TCP/IP bietet den Vorteil, dass es in heterogenen Netzen eingesetzt werden kann, d. h. für TCP/IP sind die Unterschiede der Netze auf physikalischer Ebene oder in der Topologie belanglos.

Lösung zu Selbsttestaufgabe 8.2-1:

Das TCP/IP-Modell besteht aus 4 Schichten:

Anwendungsschicht,

Transportschicht,

Internet-Schicht (Vermittlungsschicht),

Netz-Interface-Schicht.

Das *Internet Protocol* ist ein Protokoll der Vermittlungsschicht.

Lösung zu Selbsttestaufgabe 8.5-1:

Nach Abb. 8.5-5 liegt die IP-Adresse $225.192.136.110$ zwischen $224.0.0.0$ und $239.255.255.255$. Demnach handelt es sich um eine Klasse D IP-Adresse (Multicastadresse).

Lösung zu Selbsttestaufgabe 8.5-2:

Der Wert im TTL-Feld gibt die Anzahl der Netzknoten an, die ein Datagramm passieren darf. Diese Einschränkung verhindert, dass Datagramme im Netz unendlich lange zirkulieren, weil das Ziel z. B. nicht existiert oder nicht erreichbar ist. Bei einem TTL-Wert von eins darf das Datagramm nur einen Knoten passieren. Beim zweiten Knoten wird es aus dem Netz entnommen und verworfen.

Lösung zu Selbsttestaufgabe 8.5-3:

Das Typfeld spezifiziert die Art der ICMP-Meldung. Nach Tabelle 8.5-1 entspricht ein Wert von 3 im Typfeld einer *destination unreachable*-Fehlermeldung. Das Codefeld beinhaltet weitergehende Informationen über den Inhalt einer Meldung – eine genaue Beschreibung des Fehlers. Ein Wert von 1 im Codefeld bedeutet nach Tabelle 8.5-2, dass der Host nicht erreichbar ist.

Lösung zu Selbsttestaufgabe 8.5-4:

Der steigende Bedarf an IP-Adressen im industriellen, aber auch im privaten Bereich führte in den letzten Jahren zu einem Engpass im IP-Adressraum. Zeitgleich fand das ***C****lassless* ***I****nter* ***D****omain* ***R****outing* (**CIDR**) mit einem gestiegenen Aufwand für die Verwaltung des Routing immer mehr Verbreitung.

Aus diesen Gründen führte man das ***I****nternet* ***P****rotocol* ***V****ersion* ***6*** (**IPv6**) ein, das sowohl für den Einsatz in Hochgeschwindigeitsnetze als auch für den Einsatz in Netze mit niedriger Bandbreite konzipiert wurde.

Lösung zu Selbsttestaufgabe 8.5-5:

Aggregatable Global Unicast-Adressen sind im gesamten Internet gültig, haben also eine globale Eindeutigkeit. IPv6 sieht neben den globalen Unicast-Adressen auch Adressen vor, die für den lokalen Einsatz bestimmt sind: Link-Local und Site-Local Unicast-Adressen. Während Link-Local Unicast-Adressen sich auf ein einzelnes physisches Netz beschränken und bei der automatischen Adresskonfiguration eingesetzt werden können, beziehen sich Site-Local Unicast-Adressen auf einen Standort (*Site*) bzw. auf eine Organisation ohne Anbindung an das Internet.

Lösung zu Selbsttestaufgabe 8.5-6:

Anycast-Adressen werden aus dem Unicast-Adressraum vergeben und benutzen deshalb dasselbe Format wie Unicast-Adressen. Aus diesem Grund können Anycast-Adressen syntaktisch nicht von Unicast-Adressen unterschieden werden.

Lösung zu Selbsttestaufgabe 8.6-1:

TCP ist ein verbindungsorientiertes Protokoll; UDP ist verbindungslos. Für die Realisierung einer Echtzeit-Übertragung ist die Schnelligkeit der Übertragung ein wesentliches Kriterium. Zudem sind verloren gegangene Pakete tolerierbar. Bei TCP wird jedes Segment quittiert. Dies wird zu einem Flaschenhals im Bezug auf die Schnelligkeit einer Echtzeitübertragung. Aus diesem Grund ist bei der Wahl zwischen TCP und UDP für eine Echtzeit-Sprachübertragung das *User Datagram Protocol* vorzuziehen.

Übungsaufgaben

Übungsaufgaben zu Kapitel "Offene Systeme"

Aufgabe 1: Ordnen Sie den folgenden Diensten und Funktionen die korrekte Schicht im OSI-Modell zu: **10 Pkt**

Dienst	Schicht
Festlegung der lokalen (systeminternen) Darstellung für eine Sitzung	
Notspeisung	
File Transfer	
Verwaltung von Sitzungen	
Quelle und Senke für die Kommunikation	
Sicherung der Teilstrecken	
Wegesuche	
Optimierung des Kommunikationsnetzes	
Ende-zu-Ende Sicherung	
Message Handling	
Umsetzung zwischen lokalen und neutralen Darstellungen während einer Sitzung	
Ersatzroutenbestimmung	
Auf- und Abbau von Ende-zu-Ende Verbindungen	
Flußkontrolle auf Teilstrecken	
Leitungscodierung	
Aktivieren und Deaktivieren physikalischer Strecken	
Flußregelung zwischen Endsystemen	
Wortweise Multiplexbildung	
Auf- und Abbau von Sitzungen zwischen Anwenderinstanzen	
Physikalische Bitübertragung	

Aufgabe 2: MMS43 (Modified Monitored Sum 43) ist ein 4B3T („4 Bipolar 3 Ternary“) Blockcode, bei dem 4 Binärzeichen in 3 Ternärzeichen umgesetzt werden. Hierbei wird zwischen 4 Codetabellen (S1 - S4) umgeschaltet. In einer Variante lauten die Codetabellen: **10 Pkt**

	S_1	S_2	S_3	S_4
0001	0 - + 1	0 - + 2	0 - + 3	0 - + 4
0111	- 0 + 1	- 0 + 2	- 0 + 3	- 0 + 4
0100	- + 0 1	- + 0 2	- + 0 3	- + 0 4
0010	+ - 0 1	+ - 0 2	+ - 0 3	+ - 0 4
1011	+ 0 - 1	+ 0 - 2	+ 0 - 3	+ 0 - 4
1110	0 + - 1	0 + - 2	0 + - 3	0 + - 4
1001	+ - + 2	+ - + 3	+ - + 4	- - - 1
0011	0 0 + 2	0 0 + 3	0 0 + 4	- - 0 2
1101	0 + 0 2	0 + 0 3	0 + 0 4	- 0 - 2
1000	+ 0 0 2	+ 0 0 3	+ 0 0 4	0 - - 2
0110	- + + 2	- + + 3	- - + 2	- - + 3
1010	+ + - 2	+ + - 3	+ - - 2	+ - - 3
1111	+ + 0 3	0 0 - 1	0 0 - 2	0 0 - 3
0000	+ 0 + 3	0 - 0 1	0 - 0 2	0 - 0 3
0101	0 + + 3	- 0 0 1	- 0 0 2	- 0 0 3
1100	+ + + 4	- + - 1	- + - 2	- + - 3

Nach dem jeweiligen Codewort ist die Codetabelle (S) angegeben, die als nächstes verwendet wird.

Die „Running Digital Sum“ (RDS) eines Codes ist die Summe der codierten Ternärwerte, wobei folgende Zuordnung gilt:

- $0 = 0$
- $+ = +1$
- $- = -1$

3 Pkt a) Codieren Sie das Signal

$a_n = 011000110100010110110101001011101010011 1$

in MMS43, beginnend mit dem Zustand S1. Welchen Wert hat die RDS nach dem letzten codierten Bit ?

5 Pkt b) Welchen Wert kann die RDS beim MMS43 maximal annehmen, wenn beim Beginn der Codierung RDS = 0 im Zustand S_1 gilt ? Kann die RDS negativ werden? Begründen Sie diese Aussagen.

2 Pkt c) Muß der Decodierer wissen in welchen Zustand die Codierung beginnt? Begründen Sie Ihre Aussage!

10 Pkt **Aufgabe 3**: Modellieren Sie das Alternating Bit Protocol für den Sender aus Beispiel 1.6-1 als Petrinetz. Lösen Sie dabei die folgenden Teilaufgaben:

Bemerkung: *Die Lösung enthält zusätzlich zum Text eine Aninmation.*

1,5 Pkt a) Welche Meldungen gelangen an das Alternating Bit Protocol des Senders und

von welcher Schicht kommen diese?

b) Welche Meldungen werden vom Alternating Bit Protocol des Senders erzeugt und an welche Schicht gehen diese? **1,5 Pkt**

c) Zeichnen Sie das Petrinetz für das Alternating Bit Protocol für den Sender ! **7 Pkt**

Übungsaufgaben zu Kapitel "Die X.21 und X.25 Protokolle"

Aufgabe 4: Eine Datenendeinrichtung DEE soll eine Schicht 3 Verbindung nach dem X.25 Protokoll zu einer zweiten Station mit folgenden auszuhandelnden Leistungsmerkmalen für beide Richtungen aufbauen: **10 Pkt**

- Durchsatzklasse: 4800 bit/s
- Paketgröße: 256 Byte
- Fenstergröße: 4

Vereinfacht sollen für die DEE's Adressen mit 4 Dezimalstellen angenommen werden; die Adresse der gerufenen DEE sei `1234`; die der rufenden DEE `5678`. Die Übertragung soll unter Verwendung von mit 4 bit binär codierten Dezimalstellen (BCD-Code) der Adressen, Modulo 8 Zählung und ohne Ende-zu-Ende Quittierung erfolgen. Das GFI Feld hierfür lautet `0001`.

a) Stellen Sie die Schicht 3 Meldung `CALL REQUEST` für die Anforderung dieser Verbindung in der Form nach Abb. 2.4-6 binär codiert dar. Darin soll die rufende DEE auch ihre eigene Adresse angeben. Nehmen Sie außerdem an, daß die virtuelle Verbindung zur DÜE durch die logische Gruppennummer `0` und die logische Kanalnummer `255` gegeben sei. Das optionale User Data Feld soll nicht benutzt werden. Die hier benötigten Leistungsmerkmale werden im Facility Feld durch einen Facility Code und weitere Parameter folgendermaßen angegeben: **5 Pkt**

	8	7	6	5	4	3	2	1
Facility Code	0	0	0	0	0	0	1	0
Durchsatzklasse	X	X	X	X	X	X	X	X

In den Bits 8 bis 5 wird die Durchsatzklasse für die Übertragung von der gerufenen DEE aus, in den Bits 4 bis 1 für die Richtung von der rufenden DEE aus binär codiert angegeben. Es können folgende Klassen angegeben werden:

3	75 bit/s
4	150 bit/s
5	300 bit/s
6	600 bit/s
7	1200 bit/s
8	2400 bit/s
9	4800 bit/s
10	9600 bit/s
11	19200 bit/s
12	48000 bit/s
13	64000 bit/s

Tab. 1: *Paketgröße*

	8	7	6	5	4	3	2	1
Facility Code	0	1	0	0	0	0	1	0
	0	0	0	0	X	X	X	X
	0	0	0	0	X	X	X	X

In den Bits 4 bis 1 des 1. Oktetts der Parameter wird die Paketgröße für die Übertragung von der gerufenen DEE aus angegeben, in den entsprechenden Bits des zweiten Oktetts für die Übertragung von der rufenden DEE aus. Die Paketgröße wird binär codiert als Logarithmus zur Basis 2 der Paketgröße in Bytes angegeben. Erlaubt sind Werte von 4 bis 12, entsprechend 16 bis 4096 Bytes. Es sollen jeweils 256 Bytes verwendet werden.

Tab. 2: *Fenstergröße*

	8	7	6	5	4	3	2	1
Facility Code	0	1	0	0	0	0	1	1
	0	X	X	X	X	X	X	X
	0	X	X	X	X	X	X	X

In den Bits 7 bis 1 des 1. Oktetts der Parameter wird die Fenstergröße für die Übertragung von der gerufenen DEE aus, in den Bits 7 bis 1 des zweiten Oktetts für die Richtung von der rufenden DEE aus binär angegeben. Der Wert 0 ist nicht erlaubt.

4 Pkt b) Die Nachricht soll in einem Schicht 2 Informationsrahmen übertragen werden. Die Schicht 2 Adresse der rufenden DEE sei A (`00000011`), die der nächsten DÜE sei B (`00000001`). Stellen Sie den vollständigen Schicht 2 Rahmen mit der CALL REQUEST Meldung in der Form gemäß folgendem Bild dar. Setzen Sie dabei als FCS die Folge (`1000001110010111`), die mit Hilfe eines Rechners ermittelt wurde, ein. Beachten Sie folgende Annahmen bei der Erstellung des Steuerfeldes: Die letzte ausgesandte Schicht 2 Meldung hatte die Sendefolgenummer N(S)=3. Mit der Übertragung des I-Rahmens soll gleichzeitig der Empfang einer Nachricht der DÜE mit der Sendefolgenummer 6 quittiert werden. Das Poll Bit soll gesetzt werden. Beachten Sie zudem

folgende Konvention für die Übertragungsreihenfolge der Bits in Schicht 2 Rahmen:

1 … 8	1 … 8	1 … 8	1 … 16	1 … 8
Flag	Adresse	Steuerfeld	FCS	Flag

Die Bits des Info Feldes werden in der Reihenfolge übertragen, wie sie von höheren Schichten übergeben werden. Bei Meldungen der Schicht 3 des X.25 findet eine byteweise Übertragung, jeweils mit dem niederwertigstem Bit beginnend, statt. Das erste Byte enthält das GFI Feld.

c) Kennzeichnen Sie in dem zu b) erstellten Schicht 2 Rahmen die Stellen, nach denen eine 0 eingefügt werden muß, um nach dem Zero-Insertion Verfahren Transparenz zu erzielen, durch Pfeile. **1 Pkt**

Aufgabe 5: Betrachten Sie das Go-Back-n Protokoll als Beispiel für ein Automatic Repeat Request (ARQ) System. Ein Sender S sendet hierbei Pakete, die durch eine Sendefolgenummer gekennzeichnet werden, an einen Empfänger E. Für jedes korrekt empfangene Paket schickt E eine Quittierung mit der Sendefolgenummer des entsprechenden quittierten Pakets an S. Ist bei E ein Paket i nicht korrekt empfangen worden, so verwirft E alle weiteren ankommenden Pakete mit einer Sendefolgenummer $> i$ und wartet auf den korrekten Empfang des Pakets i, d. h. der Empfänger E hat die Fensterbreite 1. S darf höchstens n Pakete ohne Erhalt einer Quittierung senden. Sind die Pakete mit den Sendefolgenummern 1 bis $i-1$ korrekt übertragen und quittiert worden und hat S die Pakete mit den Sendefolgenummern i bis $i+n-1$ ohne Erhalt einer positiven Quittierung für Paket i gesendet, wiederholt er den Sendeprozeß ab Paket i. **12 Pkt**

Es soll angenommen werden, daß alle Pakete dieselbe Zeit zur Übertragung benötigen. Deshalb wird als Zeiteinheit die Übertragungsdauer für ein Paket genommen. Die Wahrscheinlichkeit, daß ein Paket nicht korrekt bei E angekommen ist, sei p und unabhängig von der Störung anderer Pakete. Die Quittierungen von E nach S kommen ohne Störung an. Der Sender S befindet sich im saturierten Zustand, d. h. er hat immer Pakete zu senden. Außerdem wird angenommen, daß die Übertragung von $n-1$ Paketen länger dauert als die Signallaufzeit vom Empfänger E zum Sender S und zurück.

Die Zufallsvariable **X** bezeichne die Zeit bis zur erfolgreichen Übertragung eines Pakets, nachdem das vorhergehende Paket erfolgreich übertragen wurde. Wird ein Paket gleich beim ersten Sendeversuch korrekt übertragen, so wird **X** mit 1 realisiert. Ist für ein Paket genau eine Wiederholung der Übertragung nötig, so wird **X** mit $n+1$ realisiert.

4 Pkt a) Zeichnen Sie für das folgendes Sende/Empfangs-Szenario für das oben erläuterte Protokoll mit Parameter $n = 4$ in ein Raum-Zeit Diagramm:

- Die Pakete 1 bis 4 werden korrekt übertragen und quittiert.
- Das Paket 5 wird bei der ersten Übertragung fehlerhaft und bei der zweiten Übertragung korrekt übertragen und quittiert.
- Die Pakete 6 bis 10 werden korrekt übertragen und quittiert.

Zeichen Sie das Raum-Zeit Diagramm bis zum Zeitpunkt des Eintreffens der Quittierung von Paket 10 bei S.

2 Pkt b) Wie groß ist die Wahrscheinlichkeit $P\{\mathbf{X} = 1 + k \cdot n\}$ für beliebiges $n \geq 1$ und $k \geq 0$?

6 Pkt c) Berechnen sie die Mittelwerte $E[\mathbf{X}]$und $E[\mathbf{X}^2]$!

19 Pkt **Aufgabe 6**: Wir betrachten das Stop-and-Wait ARQ-Verfahren aus Abb. 2.2-2. Ein Nachrichtenpaket P besteht beim Stop-and-Wait ARQ-Verfahren aus einer Steuerinformation der Länge L_s und einer Nutzinformation der Länge L_n. Die Quittungen ACK und NAK haben die Länge L_q. Die Signallaufzeit beträgt in einer Richtung l. Reaktionszeiten der Stationen seinen vernachlässigbar klein. Die Übertragungsrate sei $R = \frac{1}{\tau}$. Die Sendestation habe stets Nachrichten zu übertragen.

Die Wahrscheinlichkeit, daß ein Bit auf dem Übertragungskanal verfälscht wird, sei q_b. q sei die Wahrscheinlichkeit, daß das Nachrichtenpaket P oder die Quittierung durch Bitfehler in der Übertragung verfälscht wird.

3 Pkt a) Geben Sie eine allgemeine Formel für q an.

3 Pkt b) Geben Sie allgemeine Formeln für die folgenden Größen an:

- t_p: Übertragungsdauer für ein Nachrichtenpaket P
- t_q: Übertragungsdauer für eine Quittung.
- t_c: Zykluszeit: (Zeit zwischen zwei Nachrichten)

1 Pkt c) Es sei t die Zeit, die erforderlich ist, ein Nachrichtenpaket P erfolgreich abzusenden. Geben Sie eine allgemeine Formel für den Mittelwert $E\{t\}$an.

2 Pkt d) Geben Sie allgemeine Formeln für den Durchsatz D und dem auf die Übertragungsrate $R = \frac{1}{\tau}$ genormten Durchsatz $\overline{D}$ an.

7 Pkt e) Es seinen nun folgende Parameter gegeben:

$$L_s = 6Byte$$
$$L_n = 1069Byte$$
$$L_q = 6Byte$$
$$l = 5,5ms$$
$$R = \frac{1}{\tau} = 64kBit/s$$
$$q_b = 10^{-5}$$
$$L_s = 6Byte$$

Berechnen Sie daraus, die Größen q, t_p, t_q,t_c, $E\{t\}$, D und $\overline{D}$.

f) Wie verhält sich der Durchsatz D, wenn die Länge L_n der Nutzinformation auf einen sehr klein bzw. sehr groß Wert festgesetzt wird. **3 Pkt**

Übungsaufgaben zu Kapitel "LANs und MANs"

Aufgabe 7: 1. Fragen zu Ethernet (1-persistent CSMA/CD) **15 Pkt**

a) Welcher Mechanismus wird bei Ethernet eingeleitet, wenn eine sendende Station eine Kollision erkennt? **4 Pkt**

b) Warum wird bei der Standardisierung von Ethernet eine Mindestpaketlänge vorgeschrieben ? **1 Pkt**

c) Wir betrachten nun in den weiteren Aufgaben den folgenden bidirektionalen Bus: **4 Pkt**

- Zwei Stationen A und B im Abstand von $l = 2500m$
- Signalausbreitungsgeschwindigkeit von $\nu = \frac{1km}{5\mu s}$
- Übertragungsrate des Mediums von $R = \frac{1}{\tau} = 10Mbit/s$

Skizzieren Sie für ein Sendeszenario zwischen Station A und B ein Raum-Zeit Diagramm, das die Notwendigkeit für eine Mindestpaketlänge $t_{P,min}$ aufzeigt, wobei $t_{P,min}$ die Mindestdauer einer Paketübertragung in Sekunden ist.

d) Geben Sie eine allgemeine Formel für $t_{P,min}$ und P_{min} an, wobei P_{min} die Mindestpaketlänge in Bit ist. **3 Pkt**

e) Berechnen Sie aus den angegebenen Daten die Werte für $t_{P,min}$ und P_{min} und vergleichen Sie die errechneten Werte mit den Werten aus dem Standart für Ethernet. **3 Pkt**

Aufgabe 8: Betrachtet wird das non-persistant slotted CSMA-CD Verfahren bei der folgenden Netzkonfiguration: **8 Pkt**

- $n = 100$ Terminals
- $\lambda_S = 12$ Pakte pro Minute und Terminal
- Paketlänge $P = 1000Byte$
- Übertragungsrate: $R = \frac{1}{\tau} = 10Mbit/s$

Berechnen Sie daraus die folgende Werte:

1 Pkt a) Ankunftsrate des gesamten System: λ

1 Pkt b) Übertragungsdauer t_P für ein Paket.

2 Pkt c) Den Durchsatz D.

4 Pkt d) Wie verhält sich der Durchsatz D, wenn man die Paketankunftsrate λ_S an den Stationen von 12 auf 30 Pakete pro Minute erhöht. Wie erklären Sie sich diese Veränderung von D.

18 Pkt **Aufgabe 9**: Gegeben sei ein Token Ring, an dem die Stationen äquidistant am Ring angeordnet sind. Der Ring ist durch die folgenden Parameter charakterisiert:

- Anzahl der Stationen: $n = 100$
- Abstand der benachbarten Stationen: $l_i = 200m$
- Übertragungsraten: $R_1 = \frac{1}{\tau_1} = 10Mbit/s$ bzw. $R_2 = \frac{1}{\tau_2} = 1Gbit/s$
- Länge eines *free* bzw. *busy* Token: $P_T = 3Byte$
- Länge eines Pakets: $P_P = 1000Byte$
- Ausbreitungsgeschwindigkeit des Übertragungssignals: $\nu = \frac{1km}{5\mu s}$
- keine Bitverzögerung in den Stationen
- Es treten keine Übertragungsfehler auf
- Jede Station sendet beim Erhalt eines *free* Token genau ein Paket, d. h. alle Stationen befinden sich im saturierten Zustand, Abfertigungsdisziplin: 1-limited.

Folgende Arbeitsweisen stehen für den Ringbetrieb zur Verfügung:

- SPM (=**S**ingle **P**acket **M**ode)

  Erhält eine Station i einen *free* Token, so wird der Token als *busy* Token weitergegeben und das Paket gesendet. Hat die sendende Station ihr gesendetes Paket komplett erhalten, so sendet sie einen *free* Token ab und gibt die Sendeberechtigung an die nächste Station ($i + 1$) ab.
- MPM (=**M**ulti **P**acket **M**ode)

  Erhält eine Station i einen *free* Token, so wird der Token als *busy* Token weitergegeben, das Paket gesendet und gleich im Anschluß wird ein *free* Token an die nächste Station ($i + 1$) abgegeben.

4 Pkt a) Zeichnen Sie ein Raum-Zeit Diagramm für den SPM und eines für den MPM Modus und zeichnen Sie in die Diagramme die folgenden Größen ein:

- t_V: virtuelle Übertragungszeit für ein Paket
- t_T: Übertragungszeit für einen Token
- t_P: Übertragungszeit für ein Paket
- λ_S: Signallaufzeit nach Station i nach $(i+1)$
- λ_R: Signallaufzeit um den Ring

Unter der virtuellen Übertragungszeit t_V versteht man diejenige Zeit, die von der Ankunft eines *free* Token an der Station i bis zur Ankunft eines *free* Token an der Station $(i+1)$ vergeht.

b) Geben Sie allgemeine Formeln für die virtuelle Übertragungszeit t_V und der Zykluszeit t_Z für die Betriebsmodi SPM und MPM an. Dabei ist die Zykluszeit t_Z, diejenige Zeit, die von der Ankunft eines *free* Token an der Station i bis zur nächsten Ankunft eines *free* Token an der Station i vergeht. **4 Pkt**

c) Geben Sie allgemeine Formeln für die Größe t_V an, falls in jeder Station eine Bitverzögerung von k Bit entsteht (Für den SPM und MPM Modus). **2 Pkt**

d) Unter der Annahme, daß keine Bitverzögerung in den Stationen entsteht, sind die folgenden Berechnungen für die oben genannten Werte durchzuführen. Berechnen Sie μ_S und μ_R **1 Pkt**

e) Berechnen Sie t_T, t_P, t_V, t_Z und die Auslastung $\rho = \frac{t_P}{t_V}$ für die beiden Modi SPM und MPM für die Übertragungsrate R_1. **2 Pkt**

f) Berechnen Sie t_T, t_P, t_V, t_Z und die Auslastung $\rho = \frac{t_P}{t_V}$ für die beiden Modi SPM und MPM für die Übertragungsrate R_2. **2 Pkt**

g) Welche der beiden Modi würden Sie einsetzten, wenn der Ring mit der Übertragungsrate R_2 betrieben wird. **1 Pkt**

h) Es werden nun die 4 verschiedenen Abfertigungsdisziplinen beim Erhalt eines *free* Token betrachtet: **2 Pkt**

- *1-limited:* Eine Station darf maximal einen Rahmen senden.
- *k-limited:*Eine Station darf maximal k Rahmen senden.
- *gated:*Eine Station i darf alle Pakete, die sich zum Ankunftszeitpunkt des *free* Token an der Station i in der Paketwarteschlange der Station i befinden, senden. Pakete, die nach der Ankunft des *free* Token in der Paketwarteschlange ankommen, können erst im nächsten Zyklus übertragen werden.
- *exhaustive:* Eine Station i darf alle Pakete, die sich ab dem Ankunftszeitpunkt des *free* Token an der Station i in der Paketwarteschleife befinden, senden. Auch Pakete, die nach der Ankunft des *free* Token an der Station i zur Übertragung eintreffen, während die Station i noch sendet, werden im selben Übertragungszyklus übertragen.

Bei welchen der vier Abfertigungsdisziplinen kann eine maximale Zykluszeit t_Z garantiert werden?

10 Pkt **Aufgabe 10**: In der Norm für die physikalische Schicht des FDDI sind u. a. folgende Werte festgelegt:

- maximale Faserlänge: $200km$
- maximale Anzahl von Stationen: 1000
- maximaler Abstand zwischen 2 Stationen: $2km$

Berechnen Sie unter der Annahme einer Lichtlaufzeit von $5\frac{\mu s}{km}$

2 Pkt a) die zeitliche und physikalische Ausdehnung eines Rahmens maximaler Länge auf dem Ring,

1 Pkt b) die zeitliche und physikalische Ausdehnung eines Tokens minimaler Länge (11 Byte) auf dem Ring,

1 Pkt c) die maximale Laufzeit zwischen 2 Stationen.

2 Pkt d) Ein typisches FDDI-Szenario besteht aus einem Doppelring mit $100km$ Ringlänge und 500 angeschalteten Stationen. Berechnen Sie die Tokenumlaufzeit ohne Last unter der Voraussetzung, daß die Verzögerung je Station $600ns$ beträgt.

4 Pkt e) Die *Target Token Rotation Time* TTRT in einem FDDI-Ring betrage $100ms$. Bei der Ankunft des Tokens an einer Station mißt die Station die *Token Rotation Time* TRT zu $40ms$. Wieviele Bytes asynchronen Verkehrs darf die Station senden, wenn sie mit Rahmen maximaler Länge arbeitet und der letzte Rahmen immer vollständig gesendet werden darf?

8 Pkt **Aufgabe 11**: Gegeben sei eine Station eines DQDB-Busses mit 2 Prioritätsklassen für asychronen Verkehr. Priorität 0, die niedrigere Priorität, arbeitet mit dem Requestcounter RC_0, dem Countdowncounter CC_0 und dem Requestbit REQ_0 des ACF-Feldes. Priorität 1, die höhere Priorität, arbeitet entsprechend mit RC_1, CC_1 und REQ_1.

Die momentanen Zählerstände sind in der folgenden Skizze dargelegt:

Wie in Beispiel 3.4-1 sei sei Bus A der Bus in Senderichtung. Folgende Slots mit den Informationen B (B = Busy) , REQ_0 und REQ_1 im ACF kommen in der angegebenen Reihenfolge an den Bussen (A und B) vorbei:

Bus A B=	Bus B $REQ_0 =$	Bus B REQ_1=	ankommende Meldungen in der Station
0	0	0	
1	0	1	
0	1	0	
			Meldung der Priorität 1 kommt an
1	1	0	
0	0	1	
0	0	1	
1	0	0	
			Meldung der Priorität 0 kommt an
0	0	0	
0	1	0	
0	1	1	
0	1	0	
0	0	0	

Vervollständigen Sie wie in Beispiel 3.4-1 die Zählerstände im folgenden Schema und geben sie darin an, wann die beiden Meldungen, die an der Station als Sendewunsch ankommen, abgeschickt werden können:

Bus A B=	Bus B $REQ_0 =$	Bus B $REQ_1 =$	Meldungen und Aktionen in der Station	$RC_0 =$	$RC_1 =$		$CC_0 =$	$CC_1 =$
				2	2		0	0
0	0	0						
⋮	⋮	⋮	⋮	⋮	⋮		⋮	⋮

Übungsaufgaben zu Kapitel "ISDN - Das diensteintegrierende Kommunikationsnetz"

2 Pkt **Aufgabe 12**: Im ISDN wollen drei verschiedene Stationen auf den Signalisierungskanal (D-Kanal) zugreifen. Welche Station setzt sich bei diesem Zugriff durch? Zur Lösung der Aufgabe kann Animation 4.2-12 verwendet werden.

- Station A: `01101`
- Station B: `11011`
- Station C: `11110`

15 Pkt **Aufgabe 13**: Geben Sie den Rahmen an der S_0 Schnittstelle mit den jeweiligen Bitwerten ($+$, 0, $-$) entsprechend Abb. 4.2-4 bis Abb. 4.2-7 für folgende Situation an.

Der Basisanschluß ist aktiviert, ein B-Kanal (B1) wird genutzt, der zweite B-Kanal ist frei. Der D-Kanal in Richtung NT $\rightarrow$ TE ist frei. Den D-Kanal in Richtung TE $\rightarrow$ NT versuchen zwei Endgeräte gleichzeitig zu belegen, einer mit der Folge

`01011`

und der andere mit der Folge

`00111`

Im B1 Kanal (NT $\rightarrow$ TE) wird zweimal der Wert

`11100011`

übertragen. Im B1-Kanal (TE $\rightarrow$ NT) wird erst der Wert

`00001111`

und dann der Wert

`00001110`

übertragen.

6 Pkt **Aufgabe 14**: Bilden sie folgenden Ablauf einer ISDN-Telefonverbindung auf den Nachrichtenfluß und die Ereignisse in Schicht 3 der Benutzer-Netz-Schnittstelle TE-A - ET-A ab

Verwenden und erklären Sie hierbei die Nachrichten: `ALERT`, `CONN`, `DISC`, `INFO`, `REL`, `REL COM`, `SETUP`, `SETUP ACK`.

1 Pkt a) Der rufende Teilnehmer A hebt den Hörer seines Telefons ab und hört den Wählton. Dieser Ton fordert ihn auf, die Nummer des gewünschten Anschlusses zu wählen.

b) Der Teilnehmer A gibt die Nummer des Teilnehmers B ein. Sobald die erste Ziffer eingegeben ist, wird der Wählton abgeschaltet. **1 Pkt**

c) Nachdem alle Ziffern gewählt sind, wartet er auf den Rufton. Er weiß, daß es bei Teilnehmer B klingelt, sobald der Rufton zu hören ist. **1 Pkt**

d) Hebt Teilnehmer B den Hörer seines Telefons ab, sind die beiden verbunden und können ein Gespräch führen. Teilnehmer A muß ab diesem Zeitpunkt für das Gespräch bezahlen. **1 Pkt**

e) Am Ende des Gesprächs legen beide den Hörer wieder auf. Im Netz wird die Verbindung daraufhin abgebaut. **2 Pkt**

Aufgabe 15: Skizzieren Sie den Nachrichtenfluß und bestimmen Sie die Aktivitäten in Schicht 3 zwischen TE und ET des gerufenen Teilnehmers beim Aufbau einer ankommenden Telefonverbindung im ISDN sowie beim Abbau durch den gerufenen Teilnehmer. **11 Pkt**

Aufgabe 16: Wieviele unterschiedliche Typen von Nachrichtenelementen von Schicht-3-Nachrichten des ISDN lassen sich allein aufgrund der Betrachtung der Formate codieren? Welchen Gewinnfaktor erzielt man theoretisch durch die Einführung des Nachrichtenelementes *Shift*? Wird dieser im Moment voll ausgeschöpft? **5 Pkt**

Aufgabe 17: In einem digitalem ISDN-Vermittlungssystem ist das Dienstmerkmal **Anklopfen** vorhanden. Im vereinfachten Beispiel sind an der Telefonanlage *V* drei Teilnehmer *TE1*, *TE2* und *TE3* angeschlossen. Allen Teilnehmern stehen 2 B-Kanäle sowie ein D-Kanal zur Verfügung, die Rufnummemrn der Teilnehmer sind entsprechend `RUF1`, `RUF2` und `RUF3`. **13 Pkt**

Folgender Ablauf soll dargestellt werden:

- Teilnehmer *TE2* baut erfolgreich eine Sprachverbindung zu *TE3* auf.

- Teilnehmer TE1 versucht mit Blockwahl einen Ruf zu *TE2* aufzubauen, dabei erhält *TE2* eine Inband-Tonsignalisierung und zusätzlich zeigt das Display folgenden Text an: `A klopft an`.
- *TE1* erhält einen Rufton und im Display die Meldung `Anklopfton bei B`
- *TE2* legt auf und die Verbindung zu *TE3* wird abgebaut
- Die Telefonanlage *V* baut beginnend bei *TE2* die Verbindung zwischen *TE1* und *TE2* auf.
- *TE1* und *TE2* sind verbunden.
- *TE2* legt auf und die Verbindung wird abgebaut.

10 Pkt a) Zeichnen Sie in das vorbereitete Diagramm den gesamten Schicht 3 Signalisierungsablauf ein.

Verwenden Sie hierbei die Nachrichten: `ALERT`, `CALL PROCEEDING`, `CONN`, `CONN ACK`, `DISC`, `INFO`, `REL`, `REL COM`, `SETUP`, `SETUP ACK`. Informationen können mit der Nachricht `NOTIFY` übermittelt werden.

Lösungshinweis: Das Leistungsmerkmal "Anklopfen" kann auf verschiedene Weise realisiert werden. Die Realisierung in diesem vereinfachten Beispiel muss nicht mit einer in bestehenden Netzen implementierten Lösung übereinstimmen.

1 Pkt b) Auf welchem Kanal wird die Inband-Tonsignalisierung an den Teilnehmer übertragen?

2 Pkt c) An der Telefonanlage V ist nun ein weiterer Teilnehmer TE4 angeschlossen. Teilnehmer TE3 hat nun zusätzlich das Leistungsmerkmal "Rufweiterschaltung zu TE4 wenn belegt" aktiviert. Damit entsteht zwischen diesem Leistungsmerkmal und dem Leistungsmerkmal "Anklopfen" ein Konflikt. Welcher Konflikt ist dies und wie wird er sinnvoll behoben?

4 Pkt **Aufgabe 18**: Beschreiben Sie den Anschluß herkömmlicher X.25-Endeinrichtungen ans ISDN!

Übungsaufgaben zu Kapitel "B-ISDN: Die ATM-Technik"

6 Pkt **Aufgabe 19**:

2 Pkt a) Welche Aufgaben hat das HEC-Feld im Kopf der ATM-Zelle?

2 Pkt b) Wie funktioniert der Mechanismus der Zellkopfehlerüberwachung einer ATM-Zelle?

2 Pkt c) Wie funktioniert der Mechanismus der Zellgrenzerkennung einer ATM-Zelle?

3 Pkt **Aufgabe 20**: Beschreiben Sie die Aufgaben der PLOAM-Zellen detailliert!

3 Pkt **Aufgabe 21**: Bestimmen sie das HEC-Feld des ATM-Zellkopfes für eine ATM-Zelle mit der folgenden Zellkopfbelegung!

Byte 1	Byte 2	Byte 3	Byte 4
0000 1000	0011 0000	0011 0011	0010 0101

12 Pkt **Aufgabe 22**:

3 Pkt a) Nennen Sie die Anforderungen und Funktionen der Anpassungsschicht vom Typ AAL 2!

2 Pkt b) Nennen Sie die Anforderungen an die Anpassungsschicht vom Typ AAL3/4!

c) Beschreiben Sie die Aufteilung der Anpassungsschicht vom Typ AAL 3/4 in Teilschichten und deren Aufgaben! **2 Pkt**

d) Wie und mit welchem AAL Typ würden Sie die Übertragung von herkömmlichen Sprachdaten (Sprachproben von $8bit$ jede $125\mu s$) über ein ATM-Netzwerk durchführen? **5 Pkt**

Übungsaufgaben zu Kapitel "GSM"

Aufgabe 23: **6 Pkt**

Gegeben ist der folgende Ausschnitt einer GSM-Systemarchitektur:

a) Erklären Sie die einzelnen Komponenten. **2 Pkt**

b) Welche beiden logischen Kanalgruppierungen gibt es und wie werden die logischen Kanäle auf physikalische Kanäle abgebildet? Geben Sie an, auf welchen logischen Kanälen der *Access Burst*, *Frequency Correction Burst* und *Synchronisation Burst* jeweils gesendet werden. **2 Pkt**

2 Pkt c) Logische Kanäle können verschiedene Senderichtungen haben. Tragen Sie in die Tabelle die Senderichtungen der logischen Kanäle ein.

Logischer Kanal	Senderichtung MS BTS
TCH	
BCCH	
FCCH	
SCH	
FACCH	

mit $\longrightarrow$: Von MS nach BTS
$\longleftarrow$: Von BTS nach MS
$\longleftrightarrow$: Beide Richtungen

8 Pkt **Aufgabe 24**:

Eine Basisstation kommuniziert mit 2 Mobilstationen MS_1 und MS_2. Die Entfernung zu MS_1 beträgt $6\,km$ und die Entfernung zu MS_2 beträgt $16\,km$. Hierbei kommt Adaptive Rahmenausrichtung zum Einsatz. Ferner sei eine Funkwellenausbreitungsgschwindigkeit von $300000\,km/s$ und die Dauer eines Bits bei GSM von $3,69\,\mu s$ angenommen.

2 Pkt a) Was versteht man unter einer Adaptiven Rahmenausrichtung und wann wird sie eingesetzt?

1 Pkt b) Wie lange braucht ein Signal, um von der Basisstation zu MS_1 bzw. MS_2 zu gelangen?

2 Pkt c) Wieviel soll der TA-Wert für die einzelnen Mobilstationen betragen, um die Signallaufzeit und somit die Überlappung der *Bursts* auszugleichen?

Lösungshinweis: Berücksichtigen Sie dabei, dass das Signal von der Basisstation zur Mobilstation und wieder zurrück laufen muss.

3 Pkt d) Leiten Sie aus den beiden vorhergehenden Aufgabenteilen eine Formel ab, die die Entfernung der Mobilstation zur Basisstation in Abhängigkeit von TA angibt. Welche Entfernungsänderung hat eine Erhöhung des TA-Wertes um eins zur Folge? Zeigen Sie, wie sich aus der Formel der maximale Radius von ca. $35\,km$ einer Zelle bei GSM ergibt.

Lösungshinweis: Für TA sind nur ganzzahlige Werte zugelassen.

Aufgabe 25: Eine *Mobile Station* passiert über einen Zeitraum hinweg verschiedene Zellen und *Location Areas*. Dabei wird zeitweise ein Gespräch geführt. **6 Pkt**

a) Was versteht man unter einem *Handover*? Nach welchen Zellübergängen findet ein *Handover* statt und um welchen *Handover*-Typ handelt es sich dabei jeweils? **3 Pkt**

b) In welchen Zellen finden *Location Updates* statt? **2 Pkt**

Lösungshinweis: Beachten Sie dabei, dass es sich bei einem *Location Update* um eine Hintergrund-Prozedur handelt, die im Gegensatz zum *Handover* bei nicht aktiven Teilnehmern läuft.

c) Welche GSM spezifische Kennungen werden dem Teilnehmer bei einem *Location Update* neu zugewiesen? **1 Pkt**

Übungsaufgaben zu Kapitel "Wireless Local Area Networks"

Aufgabe 26: Im IEEE Standard nach 802.11b sollen mehrere Funknetze auf gleichem Raum parallel betrieben werden. Wieviele Funknetzte können ohne gegenseitige Störung betrieben werden? **3 Pkt**

Aufgabe 27: Nennen und erklären Sie kurz vier physikalische Eigenschaften, die eine Übertragung in einem Funkkanal erschweren können! **4 Pkt**

7 Pkt **Aufgabe 28**: Zeigen Sie, dass folgende Aussage für beide Fälle korrekt ist:

$$\int_0^T e^{j\cdot 2\pi\cdot\omega_1 t}\cdot e^{-j\cdot 2\pi\cdot\omega_2 t}dt = \begin{cases} konst \text{ falls} & \omega_1 = \omega_2 \\ 0 & \text{für benachbarte } \omega \end{cases}$$

Übungsaufgaben zu Kapitel "Internet-Protokolle"

7 Pkt **Aufgabe 29**:

2 Pkt a) Ordnen Sie folgende IP-Adressen in die jeweiligen Adressklassen ein:

IP-Adresse	Klasse
194.168.12.21	
129.116.134.17	
241.235.94.115	
225.121.210.20	
126.168.12.21	

2 Pkt b) Gegeben ist die IP-Adresse $196.168.77.179$ und die Subnetzmaske $255.255.255.240$. Berechnen Sie den Host- und den Subnetzanteil. Wie viele IP-Adressen stehen im Subnetz für die Adressierung zur Verfügung?

3 Pkt c) Wie lautet die Adresse des Subnetzes aus Aufgabenteil b? Nennen Sie die Broadcast-Adresse, die kleinste und die größte IP-Adresse der Hosts in diesem Subnetz.

8 Pkt **Aufgabe 30**: Ein Datagramm hat eine Länge von $2300\,Byte$ und enthält einen Header von $20\,Byte$. Während der Übertragung erreicht das Datagramm eine Verbindung mit einer MTU von $1200\,Byte$.

1 Pkt a) In wie viele Fragmente wird das ursprüngliche Datagramm aufgeteilt?

2 Pkt b) Wie muss das DF-Bit im ursprünglichen Datagramm gesetzt sein, damit eine Fragmentierung stattfinden kann? Welche Werte haben die MF-Bits in den Fragmenten?

2 Pkt c) Wie groß sind die einzelnen Fragmente?

2 Pkt d) Geben Sie den *Fragment Offset* der Fragmente an.

e) Woran erkennt der Empfänger, dass die beiden Fragmente zusammengehören? **1 Pkt**

Aufgabe 31: **9 Pkt**

a) Für welche Aufgaben werdn das ***I**nternet **C**ontrol **M**essage **P**rotocol* (**ICMP**) und das ***I**nternet **G**roup **M**essage **P**rotocol* (**IGMP**) eingesetzt? **2 Pkt**

b) ICMP und IGMP werden zwar der Vermittlungsschicht zugeordnet, ihre Pakete werden jedoch vorher in ein anderes Protokoll-Paket der Vermittlungsschicht gekapselt, bevor sie der Netz-Interface-Schicht übergeben werden. Um welches Protokoll-Paket handelt es sich dabei? In diesem Protokoll-Paket wird durch ein bestimmtes Feld angezeigt, welches Protokoll im Datenfeld angesprochen wird. Wie heißt das Feld und wie lauten die Werte für ICMP und IGMP? **3 Pkt**

c) Wie viele Bytes beträgt ein IP-Datagramm mit einem gekapselten ICMP- bzw. IGMP-Paket? **2 Pkt**

d) Ein IP-Datagramm, dessen DF-Bit auf 1 gesetzt ist, erreicht während der Übertragung eine Verbindung, bei der eine Fragmentierung erforderlich ist. Hierbei kommt es zu einer Fehlermeldung. Welchen ICMP-Typ und ICMP-Code enthält die Fehlermeldung? **2 Pkt**

Aufgabe 32: **17 Pkt**

a) Welche Neuerungen bietet IPv6 im Vergleich zu IPv4? **6 Pkt**

b) Vergleichen Sie den IPv4-Header mit dem Basis-Header von IPv6. Welche Felder sind weggefallen und welche Felder wurden bei IPv6 durch neue ersetzt? **4 Pkt**

c) Für welche Zwecke können Erweiterungs-Header eingesetzt werden? Nennen Sie drei Beispiele. **3 Pkt**

d) Es wird ein IP-Datagramm mit Erweiterungs-Header nach der folgenden Abbildung aufgebaut. **4 Pkt**

IPv6-Basis-Header	Nutzdaten	Fragment Header	Routing Header	Hop-by-Hop Option Header

Erklären Sie, wieso der hier dargestellte Aufbau nicht stimmt und korrigieren Sie ihn.

9 Pkt **Aufgabe 33**:

3 Pkt a) Beschreiben Sie den Unterschied zwischen Unicast-, Anycast- und Multicast-Adressen.

2 Pkt b) Adresstypen können durch so genannte Präfixe unterschieden werden. Die Tabelle 8.5-5 listet die einzelnen Präfixe und die Adresstypen auf und gibt die jeweilige Belegung des Adressraums an. Berechnen Sie auf der Grundlage dieser Tabelle, wie viel Prozent des Adressraumes noch nicht vergeben sind.

4 Pkt c) Für eine Aggregatable Global Unicast-Adresse soll die *Interface ID* auf der Grundlage der ***M**edia **A**ccess **C**ontrol* (**MAC**)-Adresse mit ***E**xtended **U**nique **I**dentifier* (**EUI**)-64 codiert werden. Die MAC-Adresse in hexadezimaler Schreibweise lautet 00-0b-cd-22-68-97. Führen Sie die erforderlichen Rechenschritte durch und geben Sie das Ergebnis in Hexadezimalzahlen an.

10 Pkt **Aufgabe 34**: Es sollen zwei Internet-Anwendungen betrachtet werden:

1. E-Mail
2. Videostreaming (die Übertragung von Videodaten in Echtzeit).

Beantworten Sie die folgenden Fragen:

1. Ergänzen Sie die folgende Tabelle, indem Sie den folgenden Diensten und Funktionen die richtige Schicht im TCP/IP-Modell zuordnen:

	Netz-Interface-Schicht	Internet-Schicht	Transportschicht		Anwendungsschicht
			TCP	UDP	
Routing der E-Mail					
E-Mail Client					
Display des Videostreams					
Ende-zu-Ende Verbindung (E-Mail)					
Ende-zu-Ende Verbindung (Videostream)					

2. Stellen Sie die E-Mail Kommunikation anhand des TCP/IP-Modells graphisch dar (Datenfluss zwischen den Schichten).

3. Weches Transportprotokoll würden Sie zur Übertragung der jeweiligen Anwendung einsetzen? Nennen Sie jeweils zwei Ihnen bekannten Anwendungen, die TCP beziehungsweise UDP nutzen.

4. Wie viele IP-Datagramme werden zur Übertragung der jeweiligen Anwendung benötigt? Gehen Sie von den folgenden Annahmen aus:

- Größe der E-Mail = 6000 Bytes (Nutzdaten), Größe des Videostreams = 10.000 Bytes (Nutzdaten)
- IP setzt jeweils auf Ethernet auf
- Die maximale Nutzdatengröße, (*Maximum Transfer Unit - MTU*) beträgt 1500 Bytes (Ohne Ethernet-Header)
- TCP-Header = 20 Bytes, UDP-Header = 8 Bytes, IP-Header = 20 Bytes. Vernachlässigen Sie jeweils den Applikations-Header.

Aufgabe 35: *TCP* **6 Pkt**

Host A und Host B haben eine TCP-Verbindung aufgebaut und einen *Transmission Window* von 20 Bytes ausgehandelt. Host A möchte nun Daten mit einer Größe von 50 Bytes an Host B senden. Erklären Sie die einzelnen Schritte der Datenübertagung mit den jeweiligen Bestätigungen.

Übungsaufgaben zu Kapitel "Agenten im Internet"

Aufgabe 36: Was ist ein Mobiler Agent? **2 Pkt**

Aufgabe 37: Erklären Sie, wie ein schädlicher Host einen Mobilen Agenten angreifen kann. **3 Pkt**

Aufgabe 38: Was ist der Unterschied zwischen starker und schwacher Migrationsfähigkeit? **2 Pkt**

Aufgabe 39: Fassen Sie Vor- und Nachteile Mobiler Agenten zusammen. **3 Pkt**

Lösungen zu Übungsaufgaben

Lösung zu Aufgabe 1:

Zuordnung der Schichten im OSI-Modell.

Dienst	Schicht
Festlegung der lokalen (systeminternen) Darstellung für eine Sitzung	6
Notspeisung	1
File Transfer	7
Verwaltung von Sitzungen	5
Quelle und Senke für die Kommunikation	7
Sicherung der Teilstrecken	2
Wegesuche	3
Optimierung des Kommunikationsnetzes	3
Ende-zu-Ende Sicherung	4
Message Handling	7
Umsetzung zwischen lokalen und neutralen Darstellungen während einer Sitzung	6
Ersatzroutenbestimmung	3
Auf- und Abbau von Ende-zu-Ende Verbindungen	4
Flußkontrolle auf Teilstrecken	2
Leitungscodierung	1
Aktivieren und Deaktivieren physikalischer Strecken	1
Flußregelung zwischen Endsystemen	4
Wortweise Multiplexbildung	2
Auf- und Abbau von Sitzungen zwischen Anwenderinstanzen	5
Physikalische Bitübertragung	1

Lösung zu Aufgabe 2:

a) Die Codierung ist in folgender Tabelle dargestellt:

Signal a_n	0110	0011	0100	0101	1011	0101	0010	1110	1010	0111
Zustand	S_1	S_2	S_3	S_3	S_2	S_2	S_1	S_1	S_1	S_2
Codierung c_n	-++	00+	-+0	-00	+0-	-00	+-0	0+-	++-	-0+
RDS	1	2	2	1	1	0	0	0	1	1

Codierung der Folge a_n mittles MMS43:

$c_n = -++00+-+0-00+0--00+-00+-+++--0+$

$RDS = 1$

b) Bei MMS43 gilt $0 \leq RDS \leq 3$, d. h. der maximale Wert der RDS ist 3 und sie kann nie negativ werden. Dies Aussagen kann man aus der Zustandsdarstellung des MMS43 Codes ablesen.

An den Übergängen sind die Änderungen der RDS markiert. Man erkennt, daß die Zustände genau den Werten der RDS entsprechen:

- S_1: RDS = 0
- S_2: RDS = 1
- S_3: RDS = 2
- S_4: RDS = 3

Hieraus folgen die Behauptungen sofort.

c) Da die 64 verwendeten ternären Codeworte alle verschieden sind, kann die Decodierung ohne zusätzliche Zustandsinformationen durchgeführt werden. Innerhalb eines Zustandes sind alle Codeworte unterschiedlich. Jedem Ternärwort ist eindeutig ein Binärwort zuzuordnen.

Lösung zu Aufgabe 3:

a) Die Meldung `Send_req` wird von der darüberliegenden Schicht an das Alternating Bit Protocol geleitet und die Meldungen `ACK1` und `ACK0` werden von der darunterliegenden Schicht geliefert.

b) Die Meldungen `Data0` und `Data1` werden an die darunterliegnde Schicht und die Meldung `Send_cnf` wird an die darüberliegende Schicht weitergeleitet.

c) Im Bild ist ein mögliches Petrinetz für die Modellierung des Alternating Bit Protocol für den Sender dargestellt. Eine Marke zeigt immer den aktuellen Zustand (`Bereit1`, `Warten0`, `Bereit2`, `Warten1`) an.

Animation 1: Petrinetz

Lösung zu Aufgabe 4:

a) Schicht-3-Meldung CALL REQUEST

8							1	
0	0	0	1	0	0	0	0	GFI, Gruppennummer 0
1	1	1	1	1	1	1	1	Kanalnummer 255
0	0	0	0	1	0	1	1	Pakettyp (vgl. Abb. 2.4-4)
0	1	0	0	0	1	0	0	Adresslängen
0	0	0	1	0	0	1	0	gerufene Adresse 1234
0	0	1	1	0	1	0	0	
0	1	0	1	0	1	1	0	rufende Adresse 5678
0	1	1	1	1	0	0	0	
0	0	0	0	1	0	0	0	Facility length 8
0	0	0	0	0	0	1	0	Durchsatzklasse (Facility Code)
1	0	0	1	1	0	0	1	9 - 4800 bit/s
0	1	0	0	0	0	1	0	Paketgröße (Facility Code)
0	0	0	0	1	0	0	0	je 2^8 Bytes = 256 Bytes
0	0	0	0	1	0	0	0	
0	1	0	0	0	0	1	1	Fenstergröße (Facility Code)
0	0	0	0	0	1	0	0	4 (von der gerufenen DEE)
0	0	0	0	0	1	0	0	4 (von der rufenden DEE)

b)

Flag	Adresse	N(S) P/F N(R)	
01111110	10000000	0 001 1 111	000001000 11111 111

11 010000	00100010	01001000	00101100	01101010

00011110	00010000	01000000	10011001	01000010

00010000	00010000	11000010	0010000	00100000

FCS	Flag
1000001110010111	01111110

Schicht-3-Meldungen werden im Informationsrahmen von Schicht-2-Rahmen (s. o.) eingebettet. Im Adressfeld muß die Adresse der DÜE angegeben werden, weil I-Rahmen als Befehle verwendet werden. Die Null im LSB des Control Feldes kennzeichnet den I-Rahmen. Mit N(R) = 7 werden empfangene Nachrichten bis einschließlich N(S) = 6 quittiert. Mit N(S) = 4 wird die eigene Sendefolgenummer angegeben.

c) Siehe Pfeile in Bild von Teilaufgabe 2.

Lösung zu Aufgabe 5:

a) Betrachtet wird das folgende Sende/Empfangs-Szenario für das in der Aufgabenstellung erläuterte Protokoll mit Parameter $n = 4$.

- Die Pakete 1 bis 4 werden korrekt übertragen und quittiert.
- Das Paket 5 wird bei der ersten Übertragung fehlerhaft und bei der zweiten Übertragung korrekt übertragen und quittiert.
- Die Pakete 6 bis 10 werden korrekt übertragen und quittiert.

Das Raum-Zeit-Diagramm bis zum Zeitpunkt des Eintreffens der Quittierung von Paket 10 bei S:

b) Zufallsvariable $\mathbf{X}$: Zeit bis zur erfolgreichen Übertragung eines Pakets, dabei wird die Zeit in Einheiten der Übertragungsdauer für ein Paket gemessen.

p: Wahrscheinlichkkeit für die falsche bzw. gestörte Übertragung eines Pakets, wobei Quittierungen ohne Störung ankommen.

Dann ist: $P\{\mathbf{X} = 1 + k \cdot n\} = p^k(1-p), k = 0, 1, ...$

c) Die Berechnung der Mittelwerte:

$$
\begin{aligned}
E[\mathbf{X}] &= \sum_{k=0}^{\infty}(1 + k \cdot n) \cdot (1-p) \cdot p^k \qquad 1 \\
&= (1-p) \cdot \left(\sum_{k=0}^{\infty} p^k + n \cdot p \sum_{k=0}^{\infty} k \cdot p^{k-1} \right) \\
&= (1-p)\left(\frac{1}{1-p} + \frac{n \cdot p}{(1-p)^2} \right) \\
&= 1 + \frac{n \cdot p}{1-p}
\end{aligned}
$$

$$
\begin{aligned}
E[\mathbf{X}^2] &= \sum_{k=0}^{\infty}(1 + k \cdot n)^2 \cdot (1-p) \cdot p^k \\
&= \sum_{k=0}^{\infty} p^k + 2 \cdot n \cdot p \sum_{k=0}^{\infty} k \cdot p^{k-1} + n^2 \cdot p \sum_{k=0}^{\infty} k^2 \cdot p^{k-1} \\
&= \big(1-p\big) \cdot \left(\frac{1}{1-p} + \frac{2 \cdot n \cdot p}{(1-p)^2} + \frac{n^2 \cdot p \cdot (1+p)}{(1-p)^3} \right) \\
&= 1 + \frac{2 \cdot n \cdot p}{1-p} + \frac{n^2 \cdot p \cdot (1+p)}{(1-p)^2}
\end{aligned}
$$

Mit Hilfe von

$$\sum_{k=0}^{\infty} k \cdot p^k = \frac{p}{(1-p)^2} \qquad 2$$

$$\sum_{k=0}^{\infty} k^2 \cdot p^{k-1} = \frac{(1-p)^2 - 2 \cdot p \cdot (1-p)}{(1-p)^4} = \frac{1+p}{(1-p)^3}$$

Lösung zu Aufgabe 6:

a) Die allgemeine Formel für q lautet:

$$\begin{aligned} q &= P\{mind.\, ein\, Bitfehler\, in\, Nachrichtenpaket\, oder\, Quittung\} \qquad 3 \\ &= 1 - P\{kein\, Bitfehler\, in\, Nachrichtenpaket\, oder\, Quittung\} \\ &= 1 - \left(1 - q_b\right)^{L_s + L_n + L_q} \end{aligned}$$

b) Die allgemeinen Formeln lauten:

$$\begin{aligned} t_p &= \frac{L_s + L_n}{R} = (L_s + L_n) \cdot \tau \qquad 4 \\ t_q &= \frac{L_q}{R} = L_q \cdot \tau \\ t_c &= t_p + 2 \cdot l + t_q \end{aligned}$$

c) Nach Gl. 2.2-6 ist $E\{t\}$ gegeben durch:

$$E\{t\} = \frac{1}{1-q} \cdot t_c$$

d) Die allgemeinen Formeln für D und $\overline{D}$ lauten:

$$\begin{aligned} D &= \frac{L_n}{E\{t\}} \qquad 5 \\ \overline{D} &= \frac{D}{R} = D \cdot \tau \end{aligned}$$

e) Die Berechnungen führen zu den folgenden Ergebnissen:

$$q = 0,0828$$
$$t_p = \frac{(6Byte + 1069Byte)}{64000bit/s} = 0,1343s = 134,3ms$$
$$t_q = \frac{6Byte}{64000bit/s} = 0,00075s = 0,75ms$$
$$t_c = 0,1461s = 146,1ms$$
$$E\{t\} = 0,1593s = 159,3ms$$
$$D = 53676bit/s = 53,676kbit/s$$
$$\overline{D} = 0,839 = 83,9\%$$

f) Der Durchsatz D nimmt genau für einen bestimmten Wert für die Länge L_n der Nutzinformation sein Maximum an. Dies liegt bei den obigen Parametern bei $L_{n,max} \approx 1069Byte$. Ist $L_n < L_{n,max}$ bzw. $L_n > L_{n,max}$, so sinkt der Durchsatz D. Im folgendem Bild ist der Verlauf von D in Abhängigkeit von L_n angegeben.

Lösung zu Aufgabe 7:

a) Bemerkt eine sendende Station durch Abhören des Kanals (LWT = **L**isten **W**hile **T**alking), daß das gesendete Paket durch ein anderes gestört wird, so bricht die sendende Station die Übertragung des Pakets ab und sendet das sogenannte *JAM*-Signal, durch welches sichergestellt wird, daß alle Stationen am Bus registrieren, daß eine Kollision stattgefunden hat.

Die an der Kollision beteiligten Stationen, befinden sich nun in dem sogenannten Contention Zustand.

Nach einem Backoff-Verfahren (im Ethernet Standard: *Truncated Binary Exponential Backoff Algorithm*) wird nach einer gewissen, zufallsabhängigen Wartezeit ein erneuter Übertragungversuch für das Paket gestartet.

b) Die Mindestpaketlänge ist zur sicheren Kollisionserkennung für eine sendende Station notwendig.

c) Sendezenario zwischen A und B unter der Voraussetzung $t_P \leq t_{P,min}$

Bezeichnungen:

- $t_{P,min}$: Mindestdauer einer Paketübertragung
- $\mu = \frac{l}{\nu}$: Signallauflaufzeit eines Signals von Station A nach B

$t_0 = 0$	Paket P1 wird von A nach B gesendet
$t_1 = t_P$	Paket P1 wird komplett gesendet. A geht von korrekter Übertragung aus
$t_2 = \mu$	Paket P2 wird von B nach A gesendet. Kollision mit P_1.

d) Die allgemeinen Formeln für $t_{P,min}$ und P_{min}, wobei P_{min} die Mindestpaketlänge in Bit ist, lauten:

$$t_{P,min} = 2\frac{l}{\nu} = 2\mu$$
$$P_{min} = t_{P,min} \cdot R = \frac{t_{P,min}}{\tau} = \frac{2\mu}{\tau} = \frac{2l}{\nu \cdot \tau}$$

e) Für die Werte für $t_{P,min}$ und P_{min} ergibt sich:

$$\begin{aligned} t_{P,min} &= 2\frac{l}{\nu} = 2\mu = 2 \cdot 2500m \cdot \frac{5 \cdot 10^{-6}s}{1000m} \\ &= 25 \cdot 10^{-6}s = 25\mu s \\ P_min &= t_{P,min}R = \frac{t_{P,min}}{\tau} \\ &= 25\mu s \cdot 10 \cdot 10^6 bit/s = 250bit = 31,25Byte \end{aligned}$$

Der Wert für P_{min} im Ethernet Standard beträgt: $P_{min} = 64Byte$

Lösung zu Aufgabe 8:

a) Ankunftsrate λ des gesamten Systems:

$$\lambda = n \cdot \lambda_S = 100 \cdot \frac{12}{60} \cdot \frac{Pakete}{s} = 20\frac{Pakete}{s} = 160kbit/s$$

b) Übertragungsdauer t_P für ein Paket:

$$t_P = \frac{P}{R} = \frac{1000 \cdot 8 \cdot bit}{10 \cdot 10^6 \cdot bit/s} = 800\mu s$$

c) Durchsatz D für $\lambda_S = 12$ Pakete pro Minute und Terminal

$$\begin{aligned} D &= \frac{\lambda \cdot e^{-\lambda\tau}}{1 + P - P \cdot e^{-\lambda\tau}} \\ \lambda \cdot \tau &= 160000\frac{bit}{s} \cdot \frac{1}{10 \cdot 10^6 bit/s} = 0,016 \\ e^{-\lambda\tau} &= 0,984 \\ D &= \frac{157460,37bit/s}{127,98} = 1230,33bit/s \end{aligned}$$

d) Durchsatz D für $\lambda_S = 30$ Pakete pro Minute und Terminal

Trotz steigender Ankunftsrate λ_S an den Stationen, sinkt der Durchsatz D von $1230,33\frac{bit}{s}$ auf $1221,27\frac{bit}{s}$. Grund: Durch die steigende Ankunftsrate λ_S an den Stationen steigt die Wahrscheinlichkeit für Kollisionen von Pakten auf dem Bus.

Lösung zu Aufgabe 9:

a) Das Raum-Zeit Diagramm für den SPM und den MPM Modus mit den folgenden Größen:

- t_V: virtuelle Übertragungszeit für ein Paket
- t_T: Übertragungszeit für einen Token
- t_P: Übertragungszeit für ein Paket
- λ_S: Signallaufzeit nach Station i nach $(i+1)$
- λ_R: Signallaufzeit um den Ring

1. SPM-Modus:

2. MPM-Modus:

b) Für die virtuelle Übertragungszeit t_V und der Zykluszeit t_Z für die Betriebsmodi SPM und MPM ergeben sich die folgenden allgemeinen Formeln:

- SPM-Modus:

$$t_{V,SPM} = t_T + t_P + \mu_R + \mu_S$$
$$t_{Z,SPM} = (t_T + t_P + \mu_R + \mu_S)n$$

- MPM-Modus:

$$t_{V,MPM} = t_T + t_P + \mu_S$$
$$t_{Z,MPM} = (t_T + t_P + \mu_S)n$$

c) Allgemeine Formeln für die Größen t_V, falls in jeder Station eine Bitverzögerung von kBit entsteht:

- SPM-Modus:

$$t_{V,SPM} = t_T + t_P + \mu_R + \mu_S + n \cdot k \cdot \tau$$

- MPM-Modus:

$$t_{V,MPM} = t_T + t_P + \mu_S + k \cdot \tau$$

d) Für die Werte von μ_S und μ_R ergibt sich:

$$\mu_S = \frac{200m}{1000m} \cdot 5\mu s = 1\mu s$$
$$\mu_R = 100\mu s$$

e) Mit der Übertragungsrate $R_1 = \frac{1}{\tau_1} = 10Mbit/s$ ergibt sich:

$$t_{T,R_1} = \frac{3 \cdot 8bit}{10Mbit/s} = 2,4\mu s$$
$$t_{P,R_1} = \frac{1000 \cdot 8bit}{10Mbit/s} = 800\mu s$$

1. SPM-Modus:

$$t_{V,SPM,R_1} = 903,4\mu s$$
$$\rho_{SPM,R_1} = 88,6\%$$
$$t_{Z,SPM,R_1} = 90,34ms$$

2. MPM-Modus

$$t_{V,MPM,R_1} = 803,4\mu s$$
$$t_{Z,MPM,R_1} = 80,34ms$$
$$\rho_{MPM,R_1} = 99,6\%$$

f) Mit der Übertragungsrate $R_2 = \frac{1}{\tau_2} = 1Gbit/s$ ergibt sich:

$$t_{T,R_2} = \frac{3 \cdot 8bit}{1Gbit/s} = 0,024\mu s$$
$$t_{P,R_2} = \frac{1000 \cdot 8bit}{1Gbit/s} = 8\mu s$$

- SPM-Modus:

$$t_{V,SPM,R_2} = 109,0\mu s$$
$$t_{Z,SPM,R_2} = 10,9ms$$
$$\rho_{SPM,R_2} = 7,33\%$$

- MPM-Modus

$$t_{V,SPM,R_2} = 9,024mus$$
$$t_{Z,SPM,R_2} = 0,9024ms$$
$$\rho_{SPM,R_2} = 88,65\%$$

g) Wenn der Ring mit der Übertragungsrate R_2 betrieben wird, muß der MPM Modus zur Übertragung der Pakete gewählt werden (Dies entspricht dem Betrieb der Stationen im FDDI Protokoll).

h) Ledglich für die Abfertigungsdisziplinen 1-limited und k-limited kann eine maximale Zykluszeit t_Z garantiert werden. Bei der Abfertigungsdisziplin exhaustive kann es sogar passieren, daß die Zykluszeit t_Z unendlich wird.

Lösung zu Aufgabe 10:

a) Die maximale Rahmenlänge im FDDI beträgt $4500Byte$.

$$t_{max} = \frac{4500 \cdot 8bit}{100\frac{Mbit}{s}} = 360\mu s$$
$$l_{max} = \frac{4500 \cdot 8bit}{100\frac{Mbit}{s} \cdot 5\frac{\mu s}{km}} = 72km$$

b) Die minimale Länge eines Token beträgt $11Byte$.

$$l_{min} = \frac{11 \cdot 8bit}{100\frac{Mbit}{s}} = 0,88\mu s$$
$$l_{min} = \frac{11 \cdot 8bit}{100\frac{Mbit}{s}} \cdot \frac{1}{5\frac{\mu s}{km}} = 176m$$

c) Die maximale Laufzeit zwischen 2 Stationen beträgt

$$T_{max} = 2km \cdot 5\frac{\mu s}{km} = 10\mu s$$

d) Minimale Tokenumlaufzeit = Zeit für das Aussenden eines Tokens + Verzögerungszeit der Stationen + Umlaufzeit

$$T_{u,min} = \frac{11 \cdot 8bit}{100\frac{Mbit}{s}} + 500 \cdot 600ns + 100km \cdot 5\frac{\mu s}{km} = 800,88\mu s$$

e) Die *Token Holding Time* beträgt:

$$THT = TTRT - TRT = 100ms - 40ms = 60ms$$

In dieser Zeit können $100\frac{Mbit}{s} \cdot 60ms = 6 \cdot 10^6 bit$ Informationen übertragen werden, bzw. $\frac{6 \cdot 10^6}{4500 \cdot 8} = 166,7Rahmen$. Da der letzte Rahmen ganz übertragen wird, dürfen $167Rahmen$ bzw. $751500Byte$ gesendet werden.

Lösung zu Aufgabe 11:

Bus A Busy	Bus B REQ_0	Bus B REQ_1	Meldungen und Aktionen in der Station	RC_0	RC_1		CC_0	CC_1
				2	2		0	0
0	0	0		1	1		0	0
1	0	1		2	2		0	0
0	1	0		2	1		0	0
			Meldung Priorität 1 kommt an	2	1	→	0	1
1	1	0		3	1		0	1
			Request 1 auf Bus B, $REQ_1 = 1$					
0	0	1		3	1		0	0
0	0	1		3	1		0	0
			Meldung Priorität 1 senden					
1	0	0		3	1		0	0
			Meldung Priorität 0 kommt an	3	1	→	3	0
0	0	0		2	0		2	0
			Request 0 auf Bus B, $REQ_0 = 1$					
0	1	0		2	0		1	0
0	1	1		3	0		1	0
0	1	0		3	0		0	0
0	0	0		2	0		0	0
			Meldung Priorität 0 senden					

Lösung zu Aufgabe 12:

Station C.

Lösung zu Aufgabe 13:

Im D-Kanal (TE → NT) setzt sich der Teilnehmer mit dem niedrigeren Wert

`00111`

durch. Dieser Wert wird auch in dem Echo-Kanal (E-Bit) gespiegelt. Das Aktivierungsbit A ist auf 1 gesetzt. Die jeweils zweite Codeverletzung ist in der Skizze gekennzeichnet. Die L-Bits werden so gesetzt, daß die Gleichstromfreiheit jeweils erreicht wird.

Lösung zu Aufgabe 14:

Der für den Verbindungsauf- und -abbau notwendige Datenaustausch wird in der Benutzer-Netz-Schnittstelle des ISDN im Zeichengabekanal (D-Kanal) durchgeführt. Der Nachrichtenaustausch zwischen dem Telefon (TE) und der Teilnehmervermittlungsstelle (TE) erfolgt für den beschriebenen Ablauf in folgenden Schritten:

a)

$$\mathtt{TE} \xrightarrow{\mathtt{SETUP}} \mathtt{ET}$$

Der Anstoß zum Verbindungsaufbau erfolgt durch das Telefon (TE), indem die Schicht-3-Instanz die Nachricht `SETUP` zum Netz (ET) sendet. In der `SETUP`-Nachricht ist die Angabe Telefondienst enthalten.

$$\mathtt{TE} \xleftarrow{\mathtt{SETUP\ ACK}} \mathtt{ET}$$

Die Schicht-3-Instanz des Netzes bestätigt den Empfang der (gültigen) `SETUP`-Nachricht, indem sie eine `SETUP ACK` (*Setup Acknowledge*) Nachricht zum Endgerät sendet. Diese enthält die Angabe des B-Kanals, an den sich das Telefon anschalten kann. Gleichzeitig wird auch dem B-Kanal der Wählton übermittelt.

b)

$$\mathtt{TE} \xrightarrow{\mathtt{INFO}} \mathtt{ET}$$

Teilnehmer A gibt über eine Tastatur die Rufnummer von Teilnehmer B ein; von der Schicht-3-Instanz des Telefons werden daraufhin `INFO`-Nachrichten (Information) zum Netz gesendet. Sobald im Netz die erste `INFO`-Nachricht empfangen wurde, schaltet es den Wählton wieder ab. In ISDN wird eine Verbindung zum gewählten Anschluß hergestellt und dort gerufen.

c)

$$\mathtt{TE} \xleftarrow{\mathtt{ALERT}} \mathtt{ET}$$

Sobald es beim gerufenen Telefon klingelt, wird dies mittels der Nachricht `ALERT` (*Alerting*) angezeigt; auf dem B-Kanal wird der Rufton übermittelt.

d)

$$\mathtt{TE} \xleftarrow{\mathtt{CONN}} \mathtt{ET}$$

$$\mathtt{TE} \xrightarrow{\mathtt{CONNACK}} \mathtt{ET}$$

Nachdem Teilnehmer B seinen Hörer abgenommen hat, wird dieser Vorgang der Schicht-3-Instanz des Telefons von Teilnehmer B mit der `CONN`-Nachricht (*Connection*) angezeigt.

Beide Telefone sind nun miteinander verbunden; über den B-Kanal erfolgt der Austausch der Sprachinformation.

e)

$$\mathtt{TE} \xleftarrow{\mathtt{DISC}} \mathtt{ET}$$

Legt der B-Teilnehmer den Hörer wieder auf, dann wird das Telefon des A-Teilnehmers mittels einer `DISC`-Nachricht (*Disconnect*) darüber informiert; Der A-Teilnehmer wird aufgefordert, die Verbindung aufzulösen. Innerhalb des ISDN ist die Nutzkanalverbindung bereits abgebaut; der B-Kanal auf der Anschlußleitung steht jedoch noch zur Verfügung.

$$\mathtt{TE} \xrightarrow{\mathtt{REL}} \mathtt{ET}$$

Teilnehmer A legt nun ebenfalls auf; die Schicht-3-Instanz sendet daraufhin eine `REL`-Nachricht (*Release*) zum Netz und schaltet sich vom B-Kanal ab.

$$\mathtt{TE} \xleftarrow{\mathtt{REL\ COM}} \mathtt{ET}$$

Das Netz schaltet sich vom B-Kanal ab und sendet eine `REL COM` -Nachricht zur Schicht-3-Instanz des Telefons. Damit ist netzseitig die Schicht-3-Transaktion beendet. Nach dem Empfang dieser Nachricht beendet die Schicht-3-Instanz des Telefons die Transaktion ebenfalls. Zwischen den Schicht-3-Instanzen des Telefons und des Netzes besteht nun keine Kommunikationsbeziehung mehr.

Lösung zu Aufgabe 15:

a. Der gerufene Anschluß wird vom Vorliegen eines Verbindungswunsches durch senden der Nachricht `SETUP` informiert. Die Nachricht `SETUP` enthält zum Beispiel folgende Informationen:

   - geforderter Übermittlungsdienst (hier: Sprache)
   - zu verwendender B-Kanal

- Rufnummer des A-Teilnehmers
- geforderter Dienst (hier: Telefondienst)

b. Jedes am passiven Bus angeschaltete Endgerät überprüft die bei der `SETUP`-Nachricht enthaltene Information um festzustellen, ob eine Verbindungsanforderung vorliegt (Kompatibilitätsprüfung). Ist eine Anforderung „Telefonie“ enthalten, dann fängt ein freies Telefon an zu klingeln und sendet eine `ALERT`-Nachricht zum Netz. Diese zeigt an, daß ein kompatibles Endgerät am Bus angeschaltet ist, das die Verbindung annehmen kann. Die erste `ALERT-Nachricht` wird vom Netz zum rufenden Anschluß weitergegeben.

c. Sobald am gerufenen Anschluß der Hörer eines Telefons abgenommen wird, sendet die Schicht-3-Instanz dieses Endgerätes die Nachricht `CONN` zum Netz.

d. Die Schicht-3-Instanz des Telefons muß nun warten, bis vom Netz die Nachricht `CONN ACK` empfangen wird. Erst wenn diese Nachricht empfangen wurde, darf sich das Endgerät an den in der `SETUP`-Nachricht angegebenen B-Kanal anschalten.

   Die Teilnehmervermittlungsstelle schaltet den B-Kanal am gerufenen Anschluß mit dem Aussenden der Nachricht `CONN ACK` durch (und sendet eine `CONN-Nachricht` zum rufenden Anschluß).

e. Legt der gerufene Teilnehmer den Hörer wieder auf, dann übermittelt die Schicht-3-Instanz des Telefons eine DISC-Nachricht und schaltet sich vom B-Kanal ab. (Analog zu Aufgabe 14.)

f. Analog zu Aufgabe 14.

g. Analog zu Aufgabe 14.

Lösung zu Aufgabe 16:

Zur Bestimmung der Anzahl der möglichen unterschiedlichen Typen von Nachrichtenelementen wird die Anzahl der Bit für den *Info Element Identifier* herangezogen. Dies sind 3 Bit für Einbyte-Nachrichtenelemente und 7 Bit für Mehrbyte-Nachrichtenelemente.

Ohne Codesatzumschaltung ergeben sich also $2^3 = 8$ Einbytenachrichtenelemente und $2^7 = 128$ Mehrbytenachrichtenelemente. Insgesamt also 136 verschiedene Typen.

Das Informationselement *Shift* ist ein Oktett lang und kann an beliebiger Stelle auftreten. Es besitzt in allen Codesätzen denselben *Info Element Identifier*, damit zwischen den Codesätzen hin und her geschaltet werden kann. Im Prinzip stehen hierfür die 4-Bit *Info Element Contents* zur Verfügung. Also ergibt sich der Gewinnfaktor g zu

$$g = 2^4 = 16.$$

(Hierbei werden die verschiedenen „Shift“s als unterschiedliche Typen bewertet.) Im Moment sind nur 4 Codesätze definiert, d. h. der Gewinn wird nur zu einem Viertel ausgeschöpft.

Einbyte Nachrichtenelement

8	7	6	5	4	3	2	1
1	Info Element Identifier			Info Element Contents			

Mehrbyte Nachrichtenelement

8	7	6	5	4	3	2	1
0	Info Element Identifier						
Length of Contents							
Info Element Contents							

Lösung zu Aufgabe 17:

a)

b) Der Ton wird in den Sprechkanal eingefügt und damit über den B-Kanal übertragen

c) Der Konflikt zwischen beiden Leistungsmerkmalen ist die Reihenfolge. Ein Leistungsmerkmal muss priorisiert werden. Es macht Sinn, die Rufweiterschaltung zu priorisieren da dadurch ein bereits stattfindendes Gespräch ungestört weitergeführt werden kann.

Lösung zu Aufgabe 18:

Beim Anschluß herkömmlicher X.25-Endeinrichtungen ans ISDN beschränkt sich das ISDN Grundsystem auf die Funktion eines transparenten Zubringers von der ISDN-Teilnehmervermittlungsstelle über 64-kbit/s-Fernverbindungen zu speziellen Paketvermittlungseinrichtungen, die entweder innerhalb (integrierte Lösung) oder außerhalb des ISDN (Netzübergangslösung) angeordnet sein können. In diesem Fall dienen ISDN Benutzer- und Zwischenamtssignalisierung nur zum Herstellen der Zubringerverbindung, so daß für diese Übermittlungsdienste weitere, dienstspezifische Protokolle im Nutzkanal (im *Slot*) abgewickelt werden müssen.

Lösung zu Aufgabe 19:

a) Das fünfte Byte im ATM-Zellkopf wird als HEC (*Header Error Control*) bezeichnet und dient einerseits zur Fehlerentdeckung oder -korrektur, zum anderen zur Unterstützung der Zellgrenzenerkennung. Für die Fehlererkennung oder korrektur wird ein zyklischer Code (CRC - *Cyclic Redundancy Check*) mit dem Generatorpolynom $x^8 + x^2 + x + 1$ verwendet. Dem errechneten CRC-Wert wird die Folge `0101 0101` addiert, um die Synchronisation zu unterstützen. Dieser Wert wird schließlich in das HEC-Feld eingetragen und übertragen.

b) Im Bild ist das Zustandsdiagramm für die Fehlerüberwachung wiedergegeben.

c) Die Funktion der Zellgrenzenerkennung in ATM-Netzen ist so ausgelegt, daß sie vom verwendeten Übertragungssystem unabhängig wird. Das vorgesehene Verfahren beruht auf der Korrelation zwischen dem HEC-Wert und der Zellkopfinformation und wird gewöhnlich in Hardware implementiert um ausreichende Schnelligkeit zu gewährleisten. Die Synchronisation erfolgt in drei Schritten:

- Zu Beginn der Synchronisation wird die Suche nach einem gültigen HEC-Oktett Bit für Bit durchgeführt (*Hunt-Zustand*).
- Die Verifikation des gefundenen Oktetts als HEC-Wert erfolgt Zelle für Zelle (*Presynch-Zustand*) und wird z. B. sechs Mal wiederholt.

- Nach Abschluß der Verifikation wird in den Synch-Zustand gewechselt, die ATM-Schnittstelle ist nun synchronisiert. Erst bei mehrfachem Empfang (z. B. sieben Mal) inkorrekter HEC-Werte wird eine erneute Synchronisationsphase eingeleitet.

Lösung zu Aufgabe 20:

Die Realisierung der Schicht-1-Aufgaben erfordern vielfältige Betriebs-, Verwaltungs und Wartungsfunktionen. Diese werden als PLOAM-Funktionen (*Physical Layer Operating, Administration and Maintenance Functions*) bezeichnet. Für die Erbringung dieser Funktionen müssen verschiedene PLOAM-Meldungen über die Übertragungsstrecke (zwischen Schicht-1-Instanzen) ausgetauscht werden. Sie unterstützen beispielsweise Überwachungsfunktionen der physikalischen Schicht oder übertragen Störungsmeldungen. Je nach verwendetem Übertragungssystem können die Meldungen außerhalb der ATM-Transportstrecke (z. B. im PCM-Rahmen oder SDH Overhead) oder in besonderen Zellen (PLOAM-Zellen) im ATM-Zellstrom eingebettet werden. Die PLOAM-Zellen werden durch festgelegte, reservierte Bitmuster im Zellkopf identifiziert. Die PLOAM Zellen werden innerhalb der Schicht 1 erzeugt und verwendet und werden nicht an die übergeordneten ATM-Schicht weitergeleitet. Beispiel für die Verwendung vom PLOAM-Zellen: Bei der Übertragung von ATM-Zellen nach dem Übertragungsprinzip des reinem ATM werden die ATM-Zellen auf einer bitsynchronen Übertragungsstrecke lückenlos aneinander gereiht. Nach maximal 26 ATM-Zellen der ATM-Schicht werden PLOAM-Zellen zur Fehler- und Leistungsüberwachung der Verbindungsstrecke eingefügt.

Lösung zu Aufgabe 21:

Das HEC Feld dieser Zelle wird wie folgt berechnet: Die ersten 32 Bit des Zellkopfes `0000 1000 0011 0000 0011 0011 0010 0101` werden zunächst in ein Polynom vom maximalen Grad 31 konvertiert:

$$M = x^{27} + x^{21} + x^{20} + x^{13} + x^{12} + x^{9} + x^{8} + x^{5} + x^{2} + 1$$

Anschließend wird M mit x^8 multipliziert:

$$M' = M \cdot x^8 = x^{35} + x^{29} + x^{28} + x^{21} + x^{20} + x^{17} + x^{16} + x^{13} + x^{10} + x^{8}$$

M' wird nun durch das Generatorpolynom $G = x^8 + x^2 + x + 1$ modulo 2 dividiert:

$$\left(x^{35} + x^{29} + x^{28} + x^{21} + x^{20} + x^{17} + x^{16} + x^{13} + x^{10} + x^{8}\right) : \left(x^{8} + x^{2} + x + 1\right) = x^{27} + x^{19} + x^{11} + x^{9} + x^{8} + x^{4} + x^{2}$$

$$\underline{x^{35} + x^{29} + x^{28} + x^{27}}$$

$$x^{27} + x^{21} + x^{20} + x^{17} + x^{16} + x^{13} + x^{10} + x^{8}$$

$$\underline{x^{27} + x^{21} + x^{20} + x^{19}}$$

$$x^{19} + x^{17} + x^{16} + x^{13} + x^{10} + x^{8}$$

$$
\begin{array}{l}
\underline{x^{19} + x^{13} + x^{12} + x^{11}} \\
\quad x^{17} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} \\
\quad \underline{x^{17} + x^{11} + x^{10} + x^{9}} \\
\qquad x^{16} + x^{12} + x^{9} + x^{8} \\
\qquad \underline{x^{16} + x^{10} + x^{9} + x^{8}} \\
\qquad\quad x^{12} + x^{10} \\
\qquad\quad \underline{x^{12} + x^{6} + x^{5} + x^{4}} \\
\qquad\qquad x^{10} + x^{6} + x^{5} + x^{4} \\
\qquad\qquad \underline{x^{10} + x^{4} + x^{3} + x^{2}} \\
\qquad\qquad\quad x^{6} + x^{5} + x^{3} + x^{2}
\end{array}
$$

Als Rest der Polynomdivision ergibt sich $R = x^6 + x^5 + x^3 + x^2$ und somit die Bitfolge der CRC-Summe zu `0110 1100`. Auf die CRC-Summe wird die Bitfolge `0101 0101` modulo 2 addiert:

$$
\begin{array}{l}
01101100 \\
\underline{01010101} \\
00111001
\end{array}
$$

In das HEC-Feld der ATM-Zelle wird somit der Wert `0011 1001` eingetragen.

Lösung zu Aufgabe 22:

a) Die Anforderungen an die Anpassungsschicht vom Typ 2:

- Unterstützung der Dienstklasse B (verzögerunssensitive, verbindungsorientierte Datenübertragung mit variabler Bitrate (*VBR-Services*)).
- Bearbeitung von SDUs variabler Länge innerhalb der Anpassungsschicht (Segmentieren und Zusammensetzen/Reassemblieren)
- Übertragung von Zeitinformationen zwischen Sender und Empfänger zur Handhabung der Zeitverzögerungen durch das ATM-Netzwerk.
- Fehlermeldungen an höhere Schichten.

Dies wird erreicht durch die folgenden Funktionen:

- Taktrückgewinnung für Audio- und Videosignale mit variabler Bitrate durch Einfügen eines Zeitstempels in die CS-PDU.
- Zellfolgeüberwachung,
- und *Forward Error Correction* (FEC).

b) Die Anforderungen an die Anpassungsschicht vom Typ 3/4:

- Unterstützung der Diensteklassen C und D, d. h. VBR-Services ohne Zeitrelation zwischen Quelle und Ziel (verbindungsorientiert oder verbindungslos).
- Bearbeitung von SDUs mit variabler Länge innerhalb der Anpassungsschicht. Da das Datenformat der Konvergenzteilschicht (CS - *Convergence Sublayer*) länger sein kann als das Payload-Feld der SAR Dateneinheit PDUs (SAR - *Segmentation and Reassembley Sublayer*), muß die CS-Dateneinheit auf mehrere SAR-Dateneinheiten aufgeteilt werden.
- Multiplex- und Demultiplexfunktionen bei der verbindungslosen Datenübertragung (Diensteklasse D): Im Rahmen des Verbindungsaufbaus wird die Anzahl der CPCS-Verbindungen, die über eine ATM-Verbindung abgewickelt werden, definiert. Damit ist es möglich, mehrere CPCS-Verbindungen über eine ATM-Verbindung zu multiplexen.
- Erkennung von Bitfehlern in SAR-Rahmen und Fehlermeldung an die höhere Teilschicht CS: Mittels CRC-10 Prüfsumme ($G = x^{10}+x^9+x^5+x^4+x+1$) über den SAR-Rahmen (Header und Payload) können Bitfehler im SAR-Rahmen entdeckt werden. Die fehlerhaften Zellen können optional an die CPCS-Teilschicht weitergereicht werden.

c) Die Anpassungsschicht vom Typ AAL 3/4 teilt sich auf in die Konvergenz-Teilschicht (CS - *Convergence Sublayer*) und die darunterliegende Segmentierung und Reassemblierung (SAR - *Segmentation and Reassembley*). Die Konvergenzteilschicht des AAL 3/4 ist wiederum unterteilt in die anwendungsspezifische Konvergenzteilschicht (SSCS - *Service Specific Convergence Sublayer*) und der darunterliegenden allgemeinen Konvergenzschicht (CPCS - *Common Part Convergence Sublayer*). Die Teilschicht CPCS reicht zur Unterstützung von Anwendungen der Dienstklasse D aus. Für Anwendungen der Diensteklasse C werden häufig zusätzliche anwendungsspezifische SSCS Funktionen benötigt.

d) Sprachübertragung über ein ATM-Netzwerk: Dieser Dienst ist der Diensteklasse A (verbindungsorientiert, zeitsensitiv und konstante Bitrate) zuzuordnen. Die Anforderungen der Diensteklasse A werden durch AAL-Typ 1 unterstützt. Bei der Sprachübertragung wird der Adaptionsschicht AAL 1 alle $125\mu s$eine Sprachprobe bestehend aus 8 bit zur Verfügung gestellt. Die Adaptionsschicht sammelt diese bis eine ATM-Zelle gebildet werden kann. Der Payload eines Übertragungsrahmen der SAR-Teilschicht kann 47 Byte aufnehmen. Es werden also 47 Sprachproben gesammelt, in das Informationsfeld des SAR-Rahmens geschrieben, mit einer Sendefolgenummer versehen und der ATM-Schicht übergeben. Mit Hilfe der Sequenznummer werden die produzierten Zellen durchnummeriert, dadurch kann der Empfänger den vollständigen Empfang und die Reihenfolge der Zellen erkennen. Die Sprachproben erhalten somit neben den Verzögerungen im ATM-Netzwerk durch das Warten in der SAR Teilschicht eine zusätzliche Verzögerung durch das Auffüllen der Dateneinheiten der AAL-1 Schicht.

Es ist bei dieser Methode der Sprachübertragung zu beachten, daß 47 Sprachproben eine Länge von $5,875ms$ haben. Hätten die ATM-Zellen größere Informationsfelder, würden entsprechend größere Verzögerungszeiten verursacht werden. Bei der Übertragung von Sprachinformation beispielsweise würde ein Echo erzeugt werden, das nur mit aufwendigen Echokompensatoren beseitigt werden könnte. Kürzere Zellen oder ATM-Zellen mit nur teilweise gefüllten Informationsfeldern verbessern die Übertragung bezüglich Verzögerung, vermindern aber die Effizienz der Übertragung. Die Zusammenhänge von Paketierungsverzögerung beim Erzeugen der ATM-Zellen und der Effizienz der Übertragung haben zur Festlegung der ATM-Zelle auf insgesamt 53 Byte mit einem 48 Byte langen Informationsfeld geführt.

Lösung zu Aufgabe 23:

a) Die Komponenten aus der Abbildung sind:

MS: *Mobile Station* – Mobiles Endgerät.

BTS: Die *Base Transceiver Station* stellt die Sende- und Empfangsstation dar. Sie bildet also den reinen Funkteil einer Base Station (BS) für das Senden und Empfangen.

BSC: Der *Base Station Controller* ist der Steuerungsteil einer BS und kann mehr als eine BTS steuern. BTS und BSC bilden also zusammen eine BS – die sogenannte Funkfeststation.

MSC: Das *Mobile Switching Center* stellt die Funkvermittlungsstelle dar und dient somit zur Verbindungssteuerung. Einem MSC ist mindestens ein Base Station Controller zugeordnet.

GMSC: Das *Gateway Mobile Switching Center* stellt den Übergang zu anderen Netzen wie z. B. ISDN dar.

b) Es gibt zwei große logische Kanalgruppierungen: Verkehrskanäle und Signalisierkanäle. Die Zuordnung der logischen Kanäle auf physikalische Kanäle findet auf der Ebene der Mehrfachrahmen statt. Ein Mehrfachrahmen wird aus 26 bzw. 51 TDMA-Rahmen gebildet.

Der *Access Burst* wird auf dem RACH, *Frequency Correction Burst* auf dem FCCH und der *Synchronisation Burst* auf dem SCH gesendet.

c)

Logischer Kanal	Senderichtung MS BTS
TCH	⟷
BCCH	←
FCCH	←
SCH	←
FACCH	⟷

mit → : Von MS nach BTS
← : Von BTS nach MS
⟷ : Beide Richtungen

Lösung zu Aufgabe 24:

a) Die Teilnehmer eines Mobilfunksystems können sich an beliebigen Orten mit unterschiedlichen Entfernungen zur Basisstation aufhalten. Dadurch unterscheidet sich die Laufzeit der Signale zwischen den Endgeräten und der Basisstation. Dies führt dazu, dass sich die *Bursts*, die von Mobilstationen mit unterschiedlicher Entfernung zur Basisstation gesendet werden, stark überlappen. Um dieses Problem der Überlappung zu beheben, wird Adaptive Rahmenausrichtung eingesetzt, bei der die Basisstation jeder Mobilstation über den SACCH einen Code (*Timing Advance* Wert) übermittelt, der angibt, um wieviele Bitdauern der Antwort-*Burst* vorverlegt werden muss.

b) Bei einer Funkwellenausbreitungsgschwindigkeit von $300000\,km/s$ braucht das Signal

$$\frac{1}{300000\,km/s} \cdot 6\,km \;=\; 20\,\mu s,$$

um von BS zu MS_1 zu gelangen und

$$\frac{1}{300000\,km/s} \cdot 16\,km \;\approx\; 53,33\,\mu s,$$

um von BS zu MS_2 zu gelangen.

c) Da das Signal von der Basisstation zur Mobilstation und wieder zurück laufen muss, muss für MS_1:

$$2 \cdot 20\,\mu s \;=\; 40\,\mu s$$

und für MS_2:

$$2 \cdot 53,33\,\mu s \;=\; 106,66\,\mu s$$

ausgeglichen werden. Der TA-Wert gibt die Anzahl der Bit-Dauer an, die vorverlegt werden müssen, um die Signallaufzeit (oder Überlappung der *Bursts*) auszugleichen. Da nun die Dauer eines Bits bei GSM $3,69\,\mu s$ beträgt, ergeben sich folgende TA-Werte:

$$40\,\mu s \cdot \frac{1}{3,69\,\mu s} \;\approx\; 11 \text{ für } MS_1 \text{ und}$$

$$106,66\,\mu s \cdot \frac{1}{3,69\,\mu s} \;\approx\; 29 \text{ für } MS_2.$$

d) Die Formel, die die Entfernung der Mobilstation zur Basisstation in Abhängigkeit vom TA-Wert liefert, lautet:

$$S = \frac{TA \cdot c \cdot t_{\text{bit}}}{2}$$

mit c: Funkwellenausbreitungsgschwindigkeit

mit t_{bit}: Dauer eines Bits bei GSM $= 3,69\,\mu s$

Setzt man die Konstanten c und t_{bit} ein, so erhält man:

$$S = TA \cdot 554\,m.$$

Die Erhöhung des TA Wertes um eins hat eine Entfernungsänderung von $554\,m$ zur Folge.

Der maximale Wert für TA beträgt 63. Daraus ergibt sich eine maximale Entfernung von:

$$S = TA \cdot 554\,m = 63 \cdot 554\,m = 34,902\,km.$$

Dies entspricht dem maximalen Zellradius von ca. $35\,km.$

Lösung zu Aufgabe 25:

a) Als *Handover* bezeichnet man den Wechsel der Funkverbindung während eines Gesprächs. Dabei sollte der *Handover* vom Teilnehmer nicht bemerkt werden, d. h. es sollten z. B. keine Störgeräusche entstehen.

Da ein *Handover* während eines Gespräches stattfindet, muß in unserem Fall die gestrichelte Route des Teilnehmers in Betracht gezogen werden. Hierbei wird spätestens nach dem Übergang in eine neue Zelle ein *Handover* durchgeführt:

1. *Handover* nach dem Übergang von Zelle f_{110} zu Zelle f_{75},
2. *Handover* nach dem Übergang von Zelle f_{75} zu Zelle f_{145},
3. *Handover* nach dem Übergang von Zelle f_{145} zu Zelle f_{17},
4. *Handover* nach dem Übergang von Zelle f_{17} zu Zelle f_{201}.

Bei dem hier auftretenden *Handover* handelt es sich um Inter-Zell-Handover.

b) Während eines Gespräches wird kein *Location Update* durchgeführt. Nach dem Übergang von Zelle f_{23} zu Zelle f_{110} empfängt die Mobilstation über den *Broadcast Channel* eine neue LAI und bemerkt somit, dass sie sich in einer neuen *Location Area* befindet (LA_1). Hierauf leitet sie ihre erste *Location Update* ein. Der zweite *Location Update* wird in der Zelle f_{110} noch einmal eingeleitet, da der Timer für das periodische *Location Update* spätestens nach 24 h abläuft, der von der Basisstation vorgegeben wird. Der dritte *Location Update* findet erst in Zelle f_{201} statt, da erst hier seit dem Eintritt in LA_2 das Gespräch beendet wird.

c) Bei einem *Location Update* werden dem Teilnehmer von der VLR zwei GSM spezifische Kennungen neu zugewiesen – die ***M**obile **S**tation **R**oaming **N**umber* (**MSRN**) und die ***T**emporary **M**obile **S**ubscriber **I**dentity* (**TMSI**).

Lösung zu Aufgabe 26:

Da sich die eizelnen Kanäle im Frequenzbereich überlappen, können nur 3 Kanäle ohne Überlappung gleichzeitig betrieben werden: z. B. Kanal 1, 6 und 11.

Lösung zu Aufgabe 27:

Folgende pyhsikalische Eigenschaften erschweren die Übertragung in einem Funkkanal:

- **Abschattung**: Wenn sich zwischen dem Sender und Empfänger größere Objekte befinden wird der Empfänger durch das Objekt abgeschattet. Je höher die Frequenz des Sendesignals, desto größer der Effekt.
- **Dämpfung**: Mit größer werdender zurückgelegter Weglänge L der elektromagnetischen Welle steigt die Dämpfung D des Signals an. In einer Freiraumübertragung ist $D \sim L^2$
- **Mehrwegeausbreitung**: Das Signal erreicht den Empfänger nicht nur auf dem direkten Weg, sondern auf unterschiedlichen Pfaden. Hervorgerufen wird dieser Effekt durch *Relfexion*, *Refraktion*, *Streuung* und *Beugung*. Mehrwegeausbreitung führt beim Empfänger zu Intersymbolinterferenzen. Eine Verringerung der Feldstärke ist grundsätzlich möglich, beispielsweise durch destruktive Überlagerung von elektromagnetischen Wellen beim Empfänger.
- **Interferenzen**: Andere Qullen wie Basis- und Mobilstationen aber auch Funksysteme bzw. elektronische Geräte (Mikrowelle, TV) können Störungen verursachen und damit Interferenzen hervorrufen.

Lösung zu Aufgabe 28:

$$\int_0^T e^{j2\pi\omega_1 t} \cdot e^{-j2\pi\omega_2 t} dt = \int_0^T e^{j2\pi(\omega_1-\omega_2)t} dt$$

$$= \frac{1}{j2\pi(\omega_1-\omega_2)} \cdot e^{j2\pi(\omega_1-\omega_2)t}\Big|_0^T$$

$$= \frac{e^{j2\pi(\omega_1-\omega_2)T}}{j2\pi(\omega_1-\omega_2)} - \frac{1}{j2\pi(\omega_1-\omega_2)} = \frac{1}{j2\pi(\omega_1-\omega_2)} \cdot (e^{j2\pi(\omega_1-\omega_2)T} - 1)$$

für $\omega_1 = \omega_2$

$$\frac{(e^{j2\pi\cdot 0\cdot T} - 1)}{j2\pi \cdot 0} \rightarrow \frac{0}{0}$$

nach l' Hopital und Ableitung nach $(\omega_1 - \omega_2)$

$$\lim_{(\omega_1-\omega_2)\rightarrow 0} \frac{(e^{j2\pi(\omega_1-\omega_2)T} - 1)'}{(j2\pi(\omega_1-\omega_2))'} = \lim_{(\omega_1-\omega_2)\rightarrow 0} \frac{j2\pi T \cdot e^{j2\pi(\omega_1-\omega_2)T}}{j2\pi} = T$$

für $\omega_2 = \omega_1 + \frac{1}{T}$

$$\frac{1}{j2\pi(\omega_1-\omega_1-\frac{1}{T})} \cdot (e^{j2\pi(\omega_1-\omega_1-\frac{1}{T})\cdot T} - 1) = \frac{-T(e^{-j2\pi} - 1)}{j2\pi} = 0$$

Lösung zu Aufgabe 29:

a) Die IP-Adressen werden nach der folgenden Tabelle den Klassen zugeordnet:

IP-Adresse	Klasse
194.168.12.21	C
129.116.134.17	B
241.235.94.115	E
225.121.210.20	D
126.168.12.21	A

b)

IP-Adresse	196.168.77.179	11000100	10101000	01001101	10110011	AND
Subnetzmaske	255.255.255.240	11111111	11111111	11111111	11110000	
Subnetzanteil		11000100	10101000	01001101	10110000	

IP-Adresse	196.168.77.179	11000100	10101000	01001101	10110011	AND NOT
Subnetzmaske	255.255.255.16	11111111	11111111	11111111	11110000	
Hostanteil		00000000	00000000	00000000	00000011	

4 Bit

Es stehen insgesamt $2^4 = 16$ IP-Adressen für die Adressierung im Subnetz zur Verfügung.

c) Die kleinste Adresse $196.168.77.176$ stellt die Adresse des Subnetzes und die größte Adresse $196.168.77.191$ stellt die Broadcast-Adresse im Subnetz dar. Die restlichen Adressen werden für Hosts verwendet:

IP-Adresse des Subnetzes	196.168.77.176	11000100 10101000 01001101 10110000
Broadcast-Adresse	196.168.77.191	11000100 10101000 01001101 10111111
Host mit kleinster IP-Adresse	196.168.77.177	11000100 10101000 01001101 10110001
Host mit größter IP-Adresse	196.168.77.190	11000100 10101000 01001101 10111110

Lösung zu Aufgabe 30:

a) Das Datagramm wird in zwei Fragmente aufgeteilt.

b) Damit eine Fragmentierung stattfinden kann, muss das DF-Bit im ursprünglichen IP-Datagramm gesetzt sein ($DF = 1$).

Da das IP-Datagramm in zwei Fragmente aufgeteilt wird, sind die MF-Bits der Fragmente unterschiedlich gesetzt:

Fragment 1: $MF = 1$: Es folgt ein weiteres Fragment.

Fragment 2: $MF = 0$: Es folgt kein weiteres Fragment.

c) Die Nutzdaten, die tansportiert werden müssen, haben eine Größe von $2280\,Byte$. Die Fragmente haben die folgende Größe:

Fragment 1: $1176\,Byte$ (ohne Header),

Fragment 2: $2280\,Byte - 1176\,Byte = 1104\,Byte$ (ohne Header).

d) Die *Fragment Offsets* der beiden Fragmente betragen:
Fragment 1: *Fragment Offset* $= 0$,

Fragment 2: *Fragment Offset* $= \frac{1176\,Byte}{8\,Byte} = 147$.

e) Beide Fragmente besitzen denselben Wert im *Identification*-Feld des IP-Headers.

Lösung zu Aufgabe 31:

a) Das *Internet Control Message Protocol* wird zur Übertragung von Fehlermeldungen und Netzinformationen eingesetzt. Die Daten des ICMP werden oft direkt an Managementapplikationen höherer Schichten weitergeleitet.

Das *Internet Group Management Protocol* wird zur Verwaltung von Gruppenadressen eingesetzt, wobei alle Rechner einer Multicastgruppe auf die Multicast-Adresse dieser Gruppe reagieren.

b) ICMP und IGMP werden zwar der Vermittlungsschicht zugeordnet, ihre Pakete werden jedoch vorher in ein IP-Datagramm gekapselt, bevor sie der Netz-Interface-Schicht übergeben werden.

Das 8 Bit lange *Protocol Identity*-Feld des IP-Datagramms gibt an, welches Protokoll im Datenfeld angesprochen wird. Wird ein ICMP-Paket übertragen, so beträgt der Wert in diesem Feld 1; bei einem IGMP-Paket beträgt der Wert 2.

c) Die Größe eines IP-Datagramms mit einem ICMP- oder IGMP-Paket im Nutzdatenteil des Datagramms beträgt für beide Fälle $28\,Byte$.

d) Da das DF-Bit auf 1 gesetzt ist, kommt es während der Übertragung zu einer Fehlermeldung, wenn eine Fragmentierung erforderlich ist. Die Fehlermeldung wird mit einem ICMP-Paket verschickt, das im ICMP-Typfeld den Wert 3 (*destination unreachable*) und im ICMP-Codefeld den Wert 4 (*fragmentation necessary but don't fragment bit is set*) enthält.

Lösung zu Aufgabe 32:

a) IPv6 besitzt im Gegensatz zu IPv4 folgende Merkmale:

Erweiterte Adressierung: IPv6 vergößert die IP-Adresslänge von 32 Bit auf 128 Bit, um die Anzahl der Ebenen der Adressierungshierarchie zu erweitern, eine größere Anzahl von adressierbaren Knoten und eine einfache Autokonfiguration von Adressen zu ermöglichen.

Vereinfachtes Header-Format: Einige in IPv4-Header enthaltene Felder fallen weg. Dadurch werden die Verarbeitungskosten der Datenpakete gesenkt und die Bandbreitekosten für den IPv6-Header begrenzt.

Erweiterungs-Header: Die Kodierung der Headeroptionen wurde geändert und erlaubt damit eine effizientere Weiterleitung, keine Längenbeschränkung der Optionen und die Flexiblität, neue Optionen einzuführen.

Flusskennung: Die neue Version bietet die Möglichkeit, Pakete zu kennzeichnen, die zu einem Datenfluss gehören. Dies können Pakete sein, für die der Anwender eine spezielle Behandlung (z. B. Echtzeit-Dienst) oder eine über den Standard hinausgehende Dienstgüte (***Q**uality **o**f **S**ervice* - **QoS**) verlangt.

Sicherheitsaspekte: IPv6 unterstützt eine Basismenge von Funktionalitäten, die Authentizität, Integrität und Vertraulichkeit gewährleisten.

b) Diverse Felder des IPv4-Header wie *Header Length*, *Header Checksum*, *Type of Service* und *Identification* wurden eliminiert (durchgestrichene Felder in der folgenden Abbildung). Felder, die Informationen zur Fragmentierung (*Flags*, *Fragment Offset*) und zu Optionen enthielten (schraffierte Felder in der Abbildung), fielen zwar auch weg, aber diese Informationen werden bei IPv6 in verschiedenen Erweiterungs-Header codiert.

Die *Total Length*-, *Time to Live*-, *Protocol*-Felder werden durch Traffic Class, Payload Length, Hop Limit und Next Header bei IPv6 ersetzt (schattierte Felder in der Abbildung).

c) Der *Routing Header* wird für *Source Routing* in IPv6 verwendet. Er dient zur Festlegung einer Reihe von Routern bzw. eines Topologiebereichs, die auf dem Weg des Datagramms zum Zielknoten passiert werden sollen.

Die *Hop-by-Hop Option Header* werden vor allem eingesetzt, um Management- oder Debugging-Funktionen durchzuführen, bestimmte Verkehrsströme zwischen 2 Endknoten mit Hilfe des ***R**esource Re**s**er**v**ation **P**rotocols* (**RSVP**) gesondert zu behandeln oder um Pakete zu übertragen, die größer als 64 KB sind.

Authentication Header und *Encapsulation Security Payload Header* sind zwei Erweiterungs-Header, die Authentizität, Integrität und Vertraulichkeit gewährleisten.

d) Die Erweiterungs-Header werden zwischen dem IPv6-Basis-Header und dem *Payload* (Nutzdatenteil) platziert und nicht nach dem Payload. Außerdem muss bei der Verkettung der Erweiterungs-Header eine bestimmte Reihenfolge eingehalten werden, die nach RFC 2460 wie folgt festgelegt ist:

IPv6-Basis-Header,

Hop-by-Hop Option Header,

Destination Option Header,

Routing Header,

Fragment Header,

Authentication Header,

Encapsulation Security Payload Header,

Destination Option Header,

Upper Layer Header.

Der korrekte Aufbau des IP-Datagramms wäre:

IPv6-Basis-Header	Hop-by-Hop Option Header	Routing Header	Fragment Header	Nutzdaten

Lösung zu Aufgabe 33:

a) Die IPv6-Adressen können in Unicast-, Anycast und Multicast-Adressen klassifiziert werden. Eine Unicast-Adresse identifiziert einen einzelnen Netzanschluss eines Routers oder eines Hosts. Anycast-Adressen dagegen identifizieren Gruppen von Netzanschlüssen. Ein Datagramm, dass an eine Anycast-Adresse verschickt wurde, wird nur an ein Mitglied dieser Gruppe zugestellt. Eine Multicast-Adresse schließlich stellt einen Identifikator für eine Gruppe von Netzanschlüssen dar, bei der ein Datagramm im Gegensatz zu einer Anycast-Adresse an alle Mitglieder der Gruppe geliefert wird.

b) Zur Berechnung werden die Teile des Adressraumes, die für Adressen vergeben wurden oder reserviert sind, aufaddiert:

Reserviert (IPv4-Kompatibilität)	$\frac{1}{256}$	
NSAP-Adressen	$\frac{1}{128}$	
IPX-Adressen	$\frac{1}{128}$	
Aggregatable Global Unicast-Adressen	$\frac{1}{8}$	
Link-local Unicast-Adressen	$\frac{1}{1024}$	
Site-local Unicast-Adressen	$\frac{1}{1024}$	
Multicast-Adressen	$\frac{1}{256}$	
Summe	$\frac{154}{1024}$	$\approx 0{,}15$

Damit sind $100\% - 15\ \% = 85\%$ des Adressraumes noch nicht vergeben.

c)

Lösung zu Aufgabe 34:

1. Zuordnung der Dienste und Funktionen im TCP/IP-Schichtenmodell (Abb. 2)

	Netz-Interface-Schicht	Internet-Schicht	Transportschicht TCP	Transportschicht UDP	Anwendungsschicht
Routing der E-Mail			X		
E-Mail Client					X
Display des Videostreams					X
Ende-zu-Ende Verbindung (E-Mail)			X		
Ende-zu-Ende Verbindung (Videostream)				X	

Abb. 2: Zuordnung der Dienste bzw. Funktionen

2. In Abb. 3 ist eine E-Mail Kommunikation anhand des TCP/IP-Modells graphisch dargestellt. Das Videostreaming gestaltet sich ähnlich nur mit dem Unterschied, dass hier als Transportprotokoll UDP eingesetzt wird.

Abb. 3: Datenfluss zwischen den Schichten

3. a. Zur Übertragung der E-Mail wird das TCP-Protokoll und zur Übertragung der Videodaten das UDP-Protokoll eingesetzt.

   Bei der Übertragung von Videodaten werden einzuhaltende zeitliche Anforderungen an den zu übertragenden Datenstrom gestellt. Fehlerhaft übertragene oder verlorengegangene Pakete können meist nicht mehr verwertet werden. Eine Wiederholung derselben (wie dies bei TCP der Fall ist) wäre sinnlos (ausserdem sind bei der Übertragung von Videodaten Verluste tolerierbar). Obwohl das UDP-Protokoll keine Zuverlässigkeit hinsichtlich der Datenübertragung garantieren kann, eignet es sich dennoch besser für die Übertragung von Daten, die mit einer zeitlichen Restriktion versehen sind. Unter Zuhilfenahme anderer Protokolle und/oder darüberliegender Applikationsschichten, können die Echtzeitbedingungen erfüllt werden.

   b. Zwei Applikationen, die TCP nutzen: ***H**yper**T**ext **T**ransfer **P**rotocol* (**HTTP**), ***F**ile **T**ransfer **P**rotocol* (**FTP**).

   Zwei Applikationen, die UDP nutzen: ***D**omain **N**ame **S**ervice* (**DNS**), ***S**imple **N**etwork **M**anagement **P**rotocol* (**SNMP**).

4. Die Größe der MTU beträgt 1500 Bytes (ohne Ethernet-Header). Es sind zwei Fälle zu unterscheiden:

a. E-Mail Kommunikation: Für die E-Mail Kommunikation wird TCP als Transportprotokoll eingesetzt: Der E-Mail Header soll vernachlässigt werden, daraus folgt es können maximal 1500 Bytes - 20 Bytes (TCP-Header) - 20 Bytes (IP-Header) = 1460 Bytes Nutzdaten übertragen werden. Die Anzahl der Datagramme beträgt demnach:

$$6000/1460 = 4,109$$

Es werden 5 Datagramme übertragen.

b. Videostreaming: Es können maximal 1500 Bytes - 20 Bytes - 8 Bytes = 1472 Bytes Nutzdaten pro Datagramm übertragen werden. Die Anzahl der Datagramme beträgt:

$$10.000/1472 = 6,793$$

Es werden 7 Datagramme übertragen.

Lösung zu Aufgabe 35:

Die einzelnen Schritte der Datenübertragung sind wie folgt:

- Host A startet mit dem Senden der Daten von Byte 1 bis Byte 20.
- Hierauf wartet dieser auf die Bestätigung über den Erhalt der 20 Bytes von Host B.
- Antwortet Host B nicht mit einer Bestätigung über den Erhalt der 20 Bytes, so wartet Host A bis ein gewisses Timeout-Interval abgelaufen ist und sendet die Daten erneut.
- Bestätigt Host B den Erhalt der 20 Bytes, so sendet Host A die nächsten 20 Bytes mit der Nummer 21 bis 40.
- Hierauf wartet Host A erneut auf eine Bestätigung von Host B für die Bytes 21 bis 40.
- Nach Erhalt der Bestätigung für die Bytes 21-40 verschickt Host A die letzten 10 Bytes mit der Nummern 41-50.
- Sobald Host B diese Daten erhalten hat, sendet er seine letzte Bestätigung. Die zuletzt verschickten Daten sind zwar kleiner als die *Window Size*, es wurde aber im TCP-Header mitgeteilt, dass sie 10 Bytes groß sind. Aus diesem Grunde erwartet Host B auch keine weiteren Daten mehr.

Lösung zu Aufgabe 36:

Ein Mobiler Agent ist ein Programm, dass autonom agieren kann und selbstständig zu anderen Hosts in einem heterogenen Netz migrieren kann umd die Aufgaben im Auftrag seines Besitzers zu erfüllen.

Lösung zu Aufgabe 37:

Ein schädlicher Host kann die Dienste und Daten eines Mobilen Agenten angreifen. Im Fall eines Datenangriffs können Daten mofiziert oder gelöscht werden. Ein Dienstangriff kann einem Denial-of-Service Angriff ähneln, z. B. indem das migrieren auf einen anderen Host unterbunden wird oder indem eigene Resourcen nicht zur Verfügung gestellt werden.

Lösung zu Aufgabe 38:

Eine starke Migrationsfähigkeit impliziert dass die Daten und der Ausführungsstatus eines Mobilen Agenten zusammen mit seinem Code migriert werden und die Arbeit auf dem neuen Host direkt mit der nächsten Instruktion fortgesetzt wird. Schwache Migrationsfähigkeit liegt dann vor, wenn nur die Daten zummen mit dem Agenten transferiert werden können, die Ausführung kann nicht an der nächsten Stelle fortgesetzt werden, jedoch wird eine vorgegebene Methode ausgeführt mit der der Agent wieder aktiviert wird.

Lösung zu Aufgabe 39:

Voteile: Verringerung der Netzlast; Minimierung der Latenzzeit; Asynchrone, autonome und dynamische Berechnung; Robustheit und Fehlertoleranz

Nachteile: Schwierigkeiten in der Konzeption von Systemen; Schwierigkeiten in der Realisation von Sicherheitsaspekten; Schwierigkeiten in der Konzeption von Kommunikationsinfrastrukturen; keine Killerapplikation; Konzept ähnlich zu Würmern

Literaturverzeichnis

Literaturverzeichnis zu Kapitel 1

[BER] D. Bertsekas, R. Gallager: *Data Networks*, 2nd Edition, Prentice Hall, 1992.

[DAY] J. D. Day, H. Zimmermann: *The OSI Reference Model*, Proc. IEEE, Vol. 71, No. 12 , Dec. 1993.

[HAL] F. Halsall: *Data Communications, Computer Networks, Open Systems*, 4. Edition, Addison Wesley, 1996.

[KAD] F. Kaderali: *Digitale Kommunikationstechnik I*, Vieweg Verlag, 1991.

[STA03] W. Stallings: *Data & Computer Communication*, 7. Edition, Prentice Hall, 2003.

[SCW] M. Schwartz: *Telecommunication networks*, Addison-Wesley Verlag, 1987.

[TAN] A. S. Tanenbaum: *Computer Networks*, Prentice Hall, 4. Edition, 2003.

[WET01] D. Wetteroth: *OSI Model for Communications*, McGraw-Hill, 2003.

Literaturverzeichnis zu Kapitel 2

[BAR] R. Barnett, S. Maynard-Smith: *Packet Switched Networks*, Sigma Press, 1990.

[BER] D. Bertsekas, R. Gallager: *Data Networks*, Prentice-Hall, 1987.

[CON] D. Conrads: *Datenkommunikation*, Vieweg Verlag, 1993.

[FRE] J. Freer: *Computer Communications and Networks*, Plenum Press, 1988/96/98.

[GAR] A. Leon-Garcia, F. Widjaja: *Communication Networks*, McGraw Hill 1994/01.

[HAL] F. Halsall: *Data Communications, Computer Networks, Open Systems*, Addison-Wesley 1992/98.

[HAY] J. F. Hayes: *Modeling and Analysis of Computer Communications Networks*, Plenum Press, 1984.

[KAD] F. Kaderali: *Digitale Kommunikationstechnik II*, Vieweg Verlag, 1995.

[SCW] M. Schwartz: *Telecommunication Networks: Protocols, Modeling, and Analysis*, Addison-Wesley, 1987.

[SPR] J. D. Spragins, J. L. Hammond, K. Pawlikowski: *Telecommunications. Protocols and Design*, Addison-Wesley, 1991.

[STO] K. Stöttinger: *X.25 - Datenpaketvermittlung*, Datacom, 1991.

Literaturverzeichnis zu Kapitel 3.1 und 3.2

[BER] D. Bertsekas, R. Gallager: *Data Networks*, Prentice-Hall, 1992.

[CON] D. Conrads: *Datenkommunikation*, Vieweg Verlag, 1993.

[ELS] W. Elsässer: *ISDN und Lokale Netze*, Vieweg Verlag, 1995.

[FRE] J. Freer: *Computer Communications and Networks*, Plenum Press, 1988.

[GAR] A. Leon-Garcia, I. Widjaja: *Communication Networks*, McGraw Hill 1994/01.

[HAL1] F. Halsall: *Multimedia Communications*, Addison Wesley 2000/01.

[HAL2] F. Halsall: *Data Communications, Computer Networks, Open Systems*, Addison Wesley 1992/98.

[KAD] F. Kaderali: *Digitale Kommunikationstechnik II*, Vieweg Verlag, 1995.

[KYA] O. Kyas: *Fast Ethernet. Fast Token. Ring, Segment Switching*, DATACOM, 1995.

[SCW] M. Schwarz: *Telecommunication Networks: Protocols, Modeling, and Analsyis*, Addison-Wesley, 1987/97.

[STA] W. Stallings: *Local networks*, Macmillan Publishing Avenue, 1990.

Literaturverzeichnis zu Kapitel 3.3 und 3.4

[ALB] B. Albert, A. P. Jayasumana: *FDDI and FDDI-II: Architecture, Protocols, and Performance*, Artech House, 1994.

[BAD] A. Badach, E. Hoffmann, O. Knauer: *High Speed Internetworking. Grundlagen und Konzepte für den Einsatz von FDDI und ATM*, Addison-Wesley, 1994.

[BER] D. Bertsekas, R. Gallager: *Data Networks*, Prentice-Hall, 1987/92.

[CON] D. Conrads: *Datenkommunikation*, Vieweg Verlag, 1993.

[FRE] J. Freer: *Computer Communications and Networks*, Plenum Press, 1988.

[GAR] A. Leon-Garcia, I. Widjaja: *Communication Networks*, McGraw Hill, 1994/01.

[HAL1] F. Halsall: *Multimedia Communications*, Addison Wesley, 2000/01.

[HAL2] F. Halsall: *Data Communications, Computer Networks, Open Systems*, Addison Wesley, 1992/98.

[HEI] J. Heinänen: *Review of Backbone Technologies*, Computer Networks and ISDN Systems 12, 1991.

[JAI] R. Jain: *FDDI Handbook*, Addison-Wesley, 1994.

[JTC] JTC1/SC6/WG6:*Specification of the ATM-Ring Protocol*, JTC, 1990.

[KAD] F. Kaderali: *Digitale Kommunikationstechnik II*, Vieweg Verlag, 1995.

[KES] G. C. Kessler, D. A. Train: *Metroplolitan Area Networks: Concepts, Standards, and Services*, McGraw-Hill, Inc., 1991.

[MIC] H. W. Michael, W. J. Cronin, K. F. Pieper: *FDDI: An introduction to fiber distributed data interface*, Digital Press, 1993.

[PRY] M. De Prycker: *Asynchronous transfer mode - solutions for broadband ISDN*, Ellis Horwood, 1993.

[ROS1] F. E. Ross: *An overview of FDDI*, IEEE, J. Selected Areas in Communication, Vol. 7, No. 7, Sept. 1991.

[ROS2] F. E. Ross: *Overview of FFOL Computer Communications*, Vol. 15, No. 1, Jan. 1992.

[SEM] G. Semaan: *Isochronism and Asychronism: Methods and Performance of HSLANs*, Computer Networks and ISDN Systems 21, 1991.

[SHA] A. Shah, G. Ramakrishnan: *FDDI: A High Speed Network*, Prentice Hall, 1994.

[SPO] D. L. Spohn: *Data Network Design*, McGraw-Hill, Inc., 1993.

[SPR] J. D. Spragins, J. L. Hammond, K. Pawlikowski: *Telecommunications. Protocols and Design*, Addision-Wesley, 1991.

[ZAR] P. Zafiropulo: *On LANs and MANs: An Evolution from Mbit/s to Gbit/s*, Proc. EFOC LAN 1990, München.

[ZIT] M. Zitterbart: *Hochleistungskommunikation Band I: Technologie und Netze*, Oldenbourg Verlag, 1995.

Literaturverzeichnis zu Kapitel 4

[AMB] W. D. Ambrosch, A. Maker, B. Sasscer: *The Intelligent Network*, Springer Verlag, 1989.

[BAN] G. Bandow et. al.: *Zeichengabesysteme*, L.T.U. Vertriebsgesellschaft, 1992/95.

[BER] D. Bertsekas, R. Gallager: *Data Networks*, Prentice-Hall, 1987/92.

[BOC] P. Bocker: *ISDN, das diensteintegrierende, digitale Nachrichtennetz; Konzepte, Verfahren, Systeme*, Springer Verlag, 1986/90.

[HEL] H. J. Helgert: *ISDN - Architectures, Protocols, Standards*, Addison-Wesley, 1991.

[KAD] F. Kaderali: *Digitale Kommunikationstechnik II*, Vieweg Verlag, 1995.

[KAH] P. Kahl: *ISDN, das neue Fernmeldenetz der Deutschen Bundespost Telekom*, R. v. Decker's Verlag, 1992/95.

[KAN] A. Kanbach, A. Körber: *ISDN, die Technik*, Hüthig Verlag, 1990/98.

[KES] G. C. Kessler: *ISDN - Concepts, Facilities and Services*, McGraw Hill, 1993/95.

[VER] P. K. Verma: *ISDN Systems - Architecture, Technology and Applications*, Prentice Hall, 1990.

Literaturverzeichnis zu Kapitel 5

[GRE] A. Grebe: *Reservierungsverfahren in regionalen Gbit/s Netzen - Leistungsbewertung und Optimierung*, Dissertation im Fachbereich Elektrotechnik der FernUniversität Hagen, 1996

[HÄN] R. Händel, M. N. Huber: *Integrated Broadband Networks*, Addison Wesley, 2. Auflage, 1994

[ITU] ITU-T Recommendations der G-Serie, I-Serie und Q-Serie: *ATM-Forum: ATM User-Network Interface Specification Version 3.1*, Prentice Hall, 1995

[I371-93] ITU-T Recommendation: *Traffic control and congestion control in B-ISDN*, Nov. 93

[I432-94] ITU-T Recommendation: *B-ISDN user-network interface - Physical layer specification*, Jan. 94

[KAD] F. Kaderali: *Digitale Kommunikationstechnik II*, Vieweg Verlag, 1995

[KYA] O. Kyas: *ATM-Netzwerke*, Datacom-Verlag, 1994

[ONV] R. O. Onvural, Y. Viniotis: *Asynchronous Transfer Mode Networks*, Plenum Press, 1993

[PRY] M. de Prycker: *Asynchronous Transfer Mode - Solution for Broadband ISDN*, 2. Auflage, Ellis Horwood, 1993

[SAM] F. Sammartino: *The ATM Forum - An official supplement from the ATM Forum Data Communications*, Oct. 1992

[SIE] G. Siegmund: *ATM - Die Technik des Breitband-ISDN*, R. v. Decker, 2. Auflage, 1994

[STA] W. Stallings: *ISDN and Broadand ISDN with Frame Relay and ATM*, Prentice Hall Macmillian, 1995

[BLA] U. Black: *ATM Volumen 1, Foundation for Broadband Networks*, Prentice Hall, 1995

[RAT] E. Rathgeb: *ATM – Infrastruktur für die Hochleistungskommunikation*, Springer Verlag, 1997

Literaturverzeichnis zu Kapitel 6

[DUQ] Manuel Duque-Anton: *Mobilfunknetze*, Vieweg, 2002.

[EVB] J. Eberspächer, H.-J. Vögel, C. Bettstetter: *GSM - Global System for Mobile Communication*, B. G. Teubner, 3. Auflage, 2001.

[HEI] Gunnar Heine: *GSM Networks*, Artech House Publishers, 1999.

[KAD] F. Kaderali: *Handbuch Unternehmensnetze*, Deutscher Wirtschaftsdienst, 2001.

[MEH] Asha K. Mehrotra: *GSM System Engineering*, Artech House Publishers, 1997.

[SAU] Martin Sauter: *Mobile Kommunikationssysteme*, Vieweg, 2004.

[WAB] Bernhard H. Walke: *Mobile Radio Networks*, Wiley, Second Edition, 2002.

[WAL] Bernhard Walke: *Mobilfunknetze und ihre Protokolle 1*, Teubner Verlag, 3. Auflage 2001.

Literaturverzeichnis zu Kapitel 7

[EA05] Jon Edney, William A. Arbaugh: *Real 802.11 Security: Wi-Fi Protected Access and 802.11i*, Addison Wesley, 2005

Literaturverzeichnis zu Kapitel 8.1 bis 8.5.7

[BLA99] Uyless Black: *Voice over IP*, Prentice Hall, 1999.

[CDE00] Douglas E. Comer: *Internetworking with TCP/IP: Principles, Protocols, and Architectures, 4 edition*, Prentice Hall, 2000.

[CDO04] Douglas E. Comer: *Computer Netzwerke und Internets mit Internet-Anwendungen*, 3. überarbeitete Auflage, Prentice Hall, 2004.

[CKR74] Vinton Cerf, Robert E. Kahn: *A Protocol for Packet Network Interconnections*, IEEE Transactions on Communications, Mai 1974.

[DOY98] Jeff Doyle: *CCIE Professional Development: Routing TCP/IP*, Macmillan Technical Publishing, 1998.

[GON99] Marcus Goncalves: *Voice over IP Networks*, McGraw-Hill, 1999.

[HEI96] Mathias Hein: *TCP/IP*, International Thomson Publishing, 1996.

[KAD01] Firoz Kaderali: *Handbuch Unternehmensnetze: Zukunftssicherer Einsatz und Betrieb heterogener DV- und TK-Netze*, Deutscher Wirtschaftsdienst, 2001.

[KAP95] Firoz Kaderali, Werner Poguntke: *Graphen, Netze, Algorithmen*, Vieweg Verlag, 1995.

[KURa] F. Kurose and K. W. Ross: *Computer Networking: A Top-Down Approach Featuring the Internet*, Addison-Wesley, 3 edition, 2004. Online version http://occ.awlonline.com/bookbind/pubbooks/kurose-ross1/#

[KURb] F. Kurose, K.W. Ross: *Computernetze: Ein Top-Down-Ansatz mit Schwerpunkt Internet*, Addison-Wesley, 2004.

[PED04] Larry L. Peterson, Bruce S. Davie: *Computernetze: eine systemorientierte Einführung*, Deutsche Ausgabe der 3. amerikanischen Auflage, Dpunkt Verlag, 2004.

[PIC93] David M. Piscitello, Lyman A. Chapin: *Open Systems Networking: Tcp/Ip and Osi*, Addison-Wesley, Professional Computing, 1993.

[PRW02] Walter E. Proebster: *Rechnernetze: Technik, Protokolle, Systeme, Anwendungen*, 2. überarbeitete Auflage, Oldenbourg, 2002.

[RAE98] Martin Raepple: *Sicherheitskonzepte für das Internet*, Dpunkt Verlag, 1998.

[SCH99] Heinz Schulte: *Telekommunikation*, Interest Verlag, 1999.

[STA03] William Stallings: *Data & Computer Communication*, 7 edition, Prentice Hall, 2003.

[TAN03] A. S. Tanenbaum: *Computer-Netzwerke*, 4. Auflage, Prentice Hall, 2003.

[WET01] Debra Wetteroth: *OSI Refernce Model for telecommunications*, McGraw-Hill Professional Publishing; 1st edition, 2001.

Literaturverzeichnis zu Kapitel 8.5.8 - 8.6

[BAH01] A. Badach, E. Hoffmann: *Technik der IP-Netze*, Hanser Verlag, 2001.

[BMA95] S. Bradner, A. Mankin: *RFC 1752, The Recommendation for the IP Next Generation Protocol*, Januar 1995.

[CED03] E. Douglas Comer: *TCP/IP: Konzepte, Protokolle und Architekturen*, mitp-Verag, 2003.

[CDD03] E. Douglas Comer, E. Ralph Droms: *Computer networks and internets*, Prentice Hall, 4. Auflage, 2003.

[DEC95] S. Deering, A. Conta: *ICMP for the Internet Protocol Version 6 (IPv6)*, Internet Draft, Januar 1995.

[DEE94] S. Deerin: *Simple Internet Protocol Plus (SIPP) Specification (128-bit address version)*, Internet Draft, Juli 1994.

[DEH95] S. Deering, R. Hinden: *Internet Protocol, Version 6 (IPv6) Specification*, Internet Draft, März 1995.

[DIP02] Hans P. Dittler: *IPv6: Technik, Anwendung, Migration*, Dpunkt Verlag, 2002.

[FUL92] V. Fuller: *RFC 1338, Supernetting an Address Assignment and Aggregation Strategy*, Juni 1992.

[HIN95] R. Hinden: *IP Version 6 Addressing Architecture*, Internet Draft, April 1995.

[HUI94] C. Huitema: *RFC-1715, The H Ratio for Address Assignment Efficiency*, November 1994.

[KAD01] Firoz Kaderali: *Handbuch Unternehmensnetze*, Deutscher Wirtschaftsdienst, 2001.

[LOP99] Pete Loshin: *IPv6: Clearly Explained*, Academis Press, 1999.

[POS81] J. Postel: *RFC 791*, Internet Protocol, September 1981.

[POS94] J. Postel: *RFC 1700*, Assigned Numbers, Oktober 1994.

[REL95] Y. Rekhter, T. Li: *An Architecture for IPv6 Unicast Address Allocation*, Internet Draft, März 1995.

[RELO95] Y. Rekhter, P. Lothberg P: *An IPv6 Global Unicast Address Format*, Internet Draft, März 1995.

[STE96] E. Thomas Steven: *IPng and the TCP/IP Protocols*, Wiley, 1996.

Literaturverzeichnis zu Kapitel 9

[LA99] D. B. Lange, M. Oshima: *Seven Good Reasons for Mobile Agents*, Communications of the ACM, 1999.

[GR99] R. S. Gray., G. Cybenko, D. Kotz, R. A. Peterson, D. Rus: *D'Agents: Applications and Performance of a Mobile Agent System*, Communications of the ACM, 1999.

[CA97] A. Carzaniga, G. P. Picco, G. Vigna: *Designing Distributed Applications with Mobile code paradigm*, Department of Computer Science of Dartmouth College, 2001.

[ER01] C. Erfurth, P. Braun, W. R. Rossak: *Migration Intelligence for Mobile Agents*, Artificial Intelligence and the Simulation of Behaviour (AISB) Symposium on Software mobility and adaptive behaviour. University of York, United Kingdom, pp. 81-88, 2001.

[DAS03] S. Das, K. Shuster, C. Wu: *Dynamic Information Retrieval Optimization Using Mobile Agents*, AAMAS, 2003.

[KA98] N. M. Karnik, A. R. Tripathi: *Design Issues in Mobile Agent Programming System*, Department of Computer Science, Univ. of Minnesota, 1998.

[MI99] D. S. Milojicic, F. Douglis, Y. Paindaveine, R. Wheeler, S. Zhou: *Process migration*, 1999.

[FU98] A. Fuggetta, G. P. Picco, G. Vigna: *Understanding Code Mobility*, IEEE Transactions on Software Engineering, 1998.

[VI04] G. Vigna: *Mobile Agents: Ten Reasons for Failure*, In the proceeding of MDM 2004, pp. 298-299, Berkeley, CA, 2004.

[PA01] H. Paulino: *An Overview of Mobile Agent Systems*, Technical Report Series DCC-02-1, 2001.

[BR00] P. Braun, C. Erfurth, W. Rossak: *An Introduction to the Tracy Mobile Agent System: Technical report*, Universität Jena, 2000.

[MI99b] D. S. Milojicic: *Mobile Agent Applications*, IEEE concurrency, 1999.

Index

AAL 201
AAL 1 225
AAL 2 226
AAL 3/4 226
AAL 5 228
AC 249
Access Burst 261
Access Point 304
ACF 113
ACM 172
Ad-Hoc Modus 305
Adaptive Frame Alignment 262
adaptive Gabel 142
Adresse 51
Adressenpriorität 80
Adressfeld 214
ADU 116
AGCH 265
algebraische Methoden 33
Aloha System 73
Anforderungen von Anwendungen 197
Anklopfen 183
ANM 172
Anrufliste 183
Anrufumleitung 183
Anrufweiterleitung 183
anwendungsorientierte Funktionen 3
Anwendungsprozeß 3
Anwendungsschicht 24, 311
Anzeige der Rufnummer 183
Application Layer 24
ARP 315
Asynchronous Balanced Mode 50
ATM over SDH 202
ATM Adaptation Layer 201
ATM Ring 118
ATM-Adaptionsschicht 201
ATM-Layer 201
ATM-Protokollmodell 200
ATM-Schicht 201
ATM-Verfahren 190
Außerband 160
AUA 118
Audiotex 186
Authentication Header 339
Autonome Systeme 332

B-ISDN-Referenzmodell 199
B-Kanal 127
Bündelfehler 272
BA 126
Backbone Network 103
Backoff-Verfahren 89
Backoff-Zeit 301
Barkerfolge 144
Basic Service Set 305
Basisanschluß 126
Basisdienste 244
BCCH 265
Beacon-Frame 303
Beacon-Frames 306
Benutzer-Netz-Schnittstelle 214
BER 197
BIB 163
Bit Error Rate 197

Bitübertragungsschicht 14
Bitfehlerrate 197
Bridge 102
Broadcast-Adresse 325
Brouter 102
BS 249
BSC 249
BSIC 276
BSN 163
BSSAP 256
BTS 249
Burst-Typen 260
Burstebene 196

C-Netz 241
CAC 234
Carrier Sense Multiple Access 83
CBCH 266
CBR-Dienste 196
CC 173
CCCH 265
CDMLA 70
CDV 197
CDVT 234
Cell Delay Variation 197
Cell Error Rate 197
Cell Insertion Rate 197
Cell Loss Rate 197
Cell Transfer Delay 197
Cheapernet 90
CI 276
CIC 169
CIR 197
Clear to send 302
Client 309
CLP-Bit 214
CLR 197
Codierung der Signalisierung 271
Codierung der Sprache 272
Connection Management 255
Constant Bit Rate Services 196
Contention Resolution 78
Contention Window 302
CR-Meldung 173
CRC 209
CRMA 112
CRMA-II 115
CSMA-Verfahren 83
CSMA/CD 67
CSMA/CD - Verfahren 83
CSMA/CR - Verfahren 83
CTD 197
Cyclic Reservation Multiple Access 112

D-Kanal 127, 136
D-Netz 241
Darstellungsschicht 23
DAS 104
Data Link Layer 18
Datagramme 23
DCCH 266
DCE 51
Deadlock-Freiheit 28
Demultiplexbildung 218
deskriptive Methoden 33
deterministischen Verfahren 66
deterministischer Zugriff 70
Dienste 175
Dienstzugangspunkt 9, 292

Direct Sequence Spread Spectrum 289
Distribution System 288
DLCI 181
Dotted Decimal Notation 322
DPC 169
DQDB 106
DTAP 256
DTE 51
Dummy Burst 261
Durchschaltevermittlung 22
dynamisches Routing 331

E-Netz 241
Echo-Kanal 136
Echokompensation 139
Echokompensationsverfahren 142
Einfügungsrate von fehlgeleiteten Zellen 197
EIR 249
Empfangsnummer 46
Encapsulating Security Header 339
Encapsulation 313
Ende-zu Ende-Funktionen 23
Ende-zu-Ende-Signalisierung 160
Entwicklung von lokalen Netzen 66
erweiterter endlicher Zustandsautomat 29
Estell 33
ET 129
Ethernet 88
ETHERNET 66
ETSI 126, 241, 242
EUI 64-Adresse 343
Extended Finite State Machine 29
Exterior Gateway Protocol 331

FACCH 266
Fairneß-Strategie 137
FCAPS 284
FCCH 265
FDDI 103
FDDI-II 106
FDMA 70, 259
Fehlertoleranz Resynchronisation 28
Fenster 21
Fenstergröße 45
FFOL 106
FIB 162
FISU 162
Flag 20, 162
Flußkontrolle 21
Fragment Header 337
Fragmentierung 303
Frame Relay 191
Free Phone 185
Frequency Correction Burst 261
Frequency Hopping Spread Spectrum 293
Frequency Hopping Spread Spectrum 289
FSK 94
FSN 162
Funknetz 252

Gateway 103
Gebührenübernahme 183
Gebührenanzeige 183
General Format Identifier 57
Generic Field Control 214
getaktete Aloha System 75
GFC 214

GFI 57
gleichverteilte Wiederholung zur Kollisionsauflösung 78
GMSC 250, 279
GMSK 258
GPRS 247
GSM 242
GSM-Datendienste 245
GSM-Netze 241
GSM-Trägerdienste 245

höhere Programmiersprachen 33
höhere Schichten 201
Handover 242, 279
HDLC 50
Header Error Control 209
HEC 209
Hidden-Station-Problem 302
higher layers 201
HLR 249
Hop-by-Hop Option Header 336
HSCSD 247
HSLAN 103
Hunt-Zustand 209

IA Nr. 5 39
IAM 172
IEEE LAN Reference Model 71
Im-Band 160
IMEI 276
IMSI 275
IMT 2000 241
IN 184
Infrastruktur Modus 305
Instanz 8
Integrated Switch Router 330
Intelligentes Netz 184
Inter-Zell-Handover 280
Interframespace 301
Interior Gateway Protocol 331
Interleaving 273
Internet 308
Internet-Schicht 311
Intra-Zell-Handover 279
IPv6-Basis-Header 334
IPv6-Erweiterungs-Header 334
ISDN 126
ISDN-Adresse 159
ISDN-Rufnummer 159
ISUP 256

Kapselung 313
Klassifizierung von lokalen Netzen 66
Kollisionsauflösung 77, 80
Kollisionsauflösungs-Strategie 78
Kommunikationssteuerungsschicht 23
Kompatibilität 4
Konferenzschaltung 183
Konformität 28
Konsistenz 28
Korrektheit 28
Kurzwahl 183

LAI 276
LAN 66
Landesweite, einheitliche Rufnummer 186
LAP 50
LAPB 50
LAPD 146
LCA 118
LCI 58

Leistungsfähigkeit 29
Leistungsmerkmalen 183
Leitungsabschluß 129
Leitungscodes 14
leitungsvermittelten Diensten im ISDN 176
Livelock-Freiheit 28
LLC 72
LMSI 276
Local Area Networks 66
Location Update 282
Location Area 251
Location Areas 249
Location-Registration 283
Logical Channel Identifier 58
Logische Kanäle 259, 264
Lokale Netze 66
lokalen Netze 23
Loopback-Adresse 325
LOTOS 34
LSSU 162
LT 129

MAC 72
MAC-Adresse 343
Makeln 183
MAN 69, 103
MAP 256
MCTD 197
Mean Cell Transfer Delay 197
Mehrfachrahmen 268, 269
mittlere Bitrate 196
Mobile Originated Call 277
Mobile Station 248
Mobile Terminated Call 279
Mobilfunk 241
Mobility Management 255
Mobilvermittlungsnetz 252
Module 4
Modulo m-Zählung 45
MS 248
MSC 249
MSISDN 275
MSRN 275
MSU 162, 169
MTP 161
MTU 314
Multicast 323, 330
Multicast-Klassen 327
Multimedia-Anwendungen 196
Multinumbering 245
Multiplexbildung 19, 218

Nachrichtenelement 154
Nachrichtentyp 152
negativen Quittierung 42
Network Allocation Vector 301
Network Layer 22
Netz-Interface-Schicht 310
Netz-Netz-Schnittstelle 214
Netz-Signalisierung 160
netzorientierte Funktionen 3
New Packet Mode 181
NNI 214
non-persistent CSMA 83
Normal Burst 260
Normal Response Mode 50
Notizblock 184
Notruf 244
NPC 234

NSP 161
NSS 250
NT 129

offene Systeme 6
OMC 250
OPC 169
Open Systems 6
OSI 4
OSI-Dienste 8
OSS 250

p-persistent CSMA 84
Paging 282, 283
Paket 20
Pakettyp 58
paketvermittelten Dienste im ISDN 177
Paketvermittlung 22
Parameterüberwachung 238
PCH 266
PCN 186
PCR 163
PDU 224, 225
Performanz 285
persistent CSMA 83
Petrinetze 31
physical layer 201
Physical Lacer Convergence Procedure 293
Physical Layer 14
Physical Medium Dependent 293
physikalische Schicht 201
PIN 186
Ping-Pong-Technik 139
Plausibilitätsprüfung 42
PLMN 251
PLOAM-Funktionen 208
PLOAM-Zellen 208
PM 204
Point Coordinator 303
positiv-negativ Quittierung 42
positive Quittierung 42
PRA 126
Presentation Layer 23
Presync-Zustand 209
Primärmeldungen 8
Primärmultiplexanschluß 126
Prioritätsregelung 136
Privater Informationsdienst 186
Protokoll 8
Protokolldiskriminator 152
Protokollvalidation 28
Protokollverifikation 28
Pseudo-Noise-Code 295
PT-Feld 214
Puffer 352
Pure Aloha 73

QoS 235
Quality of Service 329
quittieren 21
Quittierung 42

R-Schnittstelle 253
Rückruf 184
RACH 265, 277
Radio Ressource Management 255
Rahmenhierarchie 270
Referenznummer 152
reines ATM 202

REL 172
Repeater 102
Request to send 302
Requests for Comments 312
Reservierungsverfahren 111
RFC 312
RLC 172
Roaming 242
Route 12
Router 102, 309
Routing Header 336
Routing-Algorithmen 331
RVCI 118

S-Schnittstelle 253
SACCH 266
SAM 172
SAPI 146
SAR 225
SAS 104
Sationen 304
SCCP 161, 170, 256
SCH 265
Schicht 4
Schicht 2 18
Schicht 5 23
Schicht 1 14
Schicht 3 22
Schicht 4 23
Schicht 6 23
Schicht 7 24
Schichten im MSC 256
Schichtung im BSS 255
Schichtung im Endgerät 255
schnelle Paketvermittlung 187
Schnittstelle 3
SCR 235
SDCCH 266
SDH 202
SDL 34
SDLC 50
SDU 226
selektive Wiederholung 49
Sendenummer 46
sequenziert 21
Sequenzierung 45
Server 309
Service 0800 185
Service 0900 186
Service 180 186
Service Set Identification 305
Session Layer 23
Sicherheit 28
Sicherungsschicht 18
Signalisierebene 256
Signalisierkanal 160
Signalisiersystem Nr. 7 250
Signalisierung 160
SIM-Karte 243, 248
Simple Network Management Protocol 371
SIO 169
Sitzungsschicht 23
Slotted Aloha 75
Slotted Aloha mit einer endlichen Anzahl q von Quellen 78
Slottime 302
SLS 169
SMDS 191

SMS 245
Source Routing 102
Spaltungsalgorithmen für die Kollisionsauflösung 80
Sperren 184
Splitting Algorithms 80
SSS 249
StarLan 90
Start Delimiter 99
Stationsdienste 306
statisches Routing 331
Stealing Flags 261
Steuerfeld 52
stochastischen Verfahren 66
stochastischer Zugriff 69
Stop-and-Wait ARQ 42
Stromversorgung 18
Subnetzbildung 323
Synchronization Burst 261
Systemhierarchie 251

TA 129
TC 204
TCAP 161, 256
TCH 265
TCP/IP-Modell 310
TCP/IP-Protokollfamilie 310
TDMA 70, 259
TE 128
TEI 146
Teilnehmer-Signalisierung 160
Teledienste 176
Telefondienst 186
Televotum 186
Termination 28
TF 170
THT 106
time out 42
time out 42
Timeout-Intervall 353
Timing Advance 263
TMN 285
TMSI 275
Token Access 91
Token Bus 94
Token Ring 97
Transparenz eines Netzes 41
Transport Layer 23
Transportebene 256
Transportschicht 23, 311
TRT 106
truncated binary exponential backoff 89
TTRT 106

U-Schnittstelle 139
Überlast 234
Überlastkontrolle 234
Übermittlungsdienste 175
Übertragungsmedium 14
UMTS 241
UNI 214
Unicast 323, 330
UPC 214, 234
User Datagram Protocol (UDP) 350

Variable Bit Rate Services 196
Varianz der Zellübermittlungsverzögerung 197
VBR-Dienste 196
VCI 214

Verbindungsebene 196
Verbindungsendpunkte 9
verbindungslose Datenübermittlung 9
verbindungslosen Datenübermittlung 12
verbindungsorientierte Datenübermittlung 10
Verkehrskontrolle 234
Vermittlungsabschluß 129
Vermittlungsschicht 22
Virtual Circuit 352, 356
virtuelle Verbindung 59, 352
VLR 249
Vollständigkeit 28
VPI 214

Wahlwiederholung 184
WAN 103
WEP 305
Wiederholung mit fester Wahrscheinlichkeit zur Kollisionsauflösung 78
window size 45
Wired Equivalent Privacy 299

X.21 - Empfehlung 38
X.25 Protokoll 56

Zeichengabe 160
Zeitgetrenntlage 139
Zeitgetrenntlageverfahren 139
Zeitmultiplexverfahren 139
Zellübermittlungsverzögerung 197
Zellebene 196
Zellen 21
Zellfehlerrate 197
Zellverlustrate 197
zentraler Signalisierkanal 160
Zero Insertion 41
Zugangskontrolle 235
Zugriff auf den D-Kanal 136
Zugriff zum Übertragungsmedium 22
Zusatzdienste 244, 246
Zuteilungsstrategie 236
zyklische Codes 21