Einführung in Bluetooth

Heiko Holtkamp
(heiko@rvs.uni-bielefeld.de)

24.03.2003
überarbeitet am 05.11.2003

Referat zum Seminar
Bluetooth Grundlagen

Veranstalter
Prof. Peter Ladkin PhD, Marcel Holtmann
Technische Fakultät, AG RVS
Universität Bielefeld
Wintersemester 2002/03

Abstract

Bluetooth ist eine universelle Funkschnittstelle, die das 2,4-GHz-ISM-Band nutzt, um portable Geräte wie Mobiltelefone, PDAs, Notebooks etc. ohne Kabel miteinander zu verbinden. Bluetooth setzt auf so genannte drahtlose Ad-hoc Pikonetze; dies sind lokale Netze, die üblicherweise eine geringe Ausdehnung haben und keine Infrastruktur benötigen. Jedes Gerät kann simultan mit bis zu sieben anderen Geräten innerhalb eines Pikonetzes kommunizieren. Daneben besteht auch die Möglichkeit, dass ein einzelnes Gerät mehreren verschiedenen Pikonetzen angehören kann.

Dieser Text führt in die grundlegenden Konzepte von Bluetooth ein und stellt anhand des Bluetooth-Protokollstapels die wichtigsten Komponenten von Bluetooth dar. Der Schwerpunkt liegt dabei auf den so genannten Core-Protokollen, die den eher hardwarenahen Teil beschreiben. Die prinzipiellen Funktionsweisen wie das Frequenzsprungverfahren, Zeitmultiplex, ISM-Band und Pikonetz werden vorgestellt. Daneben befasst sich die Einführung mit den „höheren“ Protokollen zur Verbindungsverwaltung und -steuerung, dem Protokoll zur Diensterkennung und weiteren adaptierten Protokollen, sowie den Bluetooth-Profilen. Im Text werden an entsprechenden Stellen zahlreiche Hinweise auf vertiefende Literatur gegeben, um so den weiteren Einstieg in das Thema zu erleichtern.

Inhaltsverzeichnis

1 Einleitung

Anders als z.B. Wireless-LAN (WLAN), das speziell für die Funkvernetzung entwickelt wurde, oder DECT (Digital Enhanced Cordless Telecommunications), das hauptsächlich für Schnurlos-Telefone genutzt wird, eignet sich Bluetooth mit seiner einfach zu erweiternden Spezifikation für viele unterschiedliche Anwendungen. Ein Blick in die im Aufbau befindliche Bluetooth-Datenbank [ct2003b] der Zeitschrift c't – Magazin für Computertechnik aus dem Verlag Heise zeigt, dass mittlerweile allein auf dem deutschen Markt hunderte von Bluetooth-Geräten für verschiedene Anwendungen verfügbar sind.

Bluetooth war zunächst nur als eine Art universeller Kabelersatz für den Anschluss verschiedener Peripheriegeräte gedacht. Nach einer eher schleppenden Einführung von Bluetooth-Geräten sind nun mittlerweile viele Handys, Headsets und PDAs mit der Bluetooth-Funktechnik verfügbar. Aber auch Drucker, Tastaturen, Mäuse, Modems, ISDN- und DSL-Adapter, Digitalkameras, Camcorder etc. sind mittlerweile mit Bluetooth ausgerüstet.

Doch Bluetooth ist weit mehr als ein reiner Kabelersatz. Mit der zunehmenden Verbreitung kommen weitere Anwendungen hinzu, die erst über die einfach Funkverbindung mittels Bluetooth richtig Spaß machen: Kommunikation zwischen PDAs und Handys – zum Surfen oder zur Übertragung von Kontakten und Terminen (PIM – Personal Information Management); Übertragen von Bildern aus der Digitalkamera direkt auf einen Drucker ohne den Umweg PC etc. Der Einsatzbereich von Bluetooth weitet sich in die Richtung drahtloser LANs aus.

Dies ist zwar auch mit anderen drahtlosen Übertragungstechniken möglich, z.B. mit der weit verbreiteten Übertragung via Infrarot über die IrDA-(Infrared Data Association) Schnittstelle,, jedoch hat IrDA eine sehr geringe Reichweite und die Notwendigkeit einer direkten Sichtverbindung. Damit beschränkt sich die Anwendung üblicherweise auf reine Punkt-zu-Punkt Verbindungen, die nur zwei Geräte miteinander kommunizieren lässt. Bluetooth hingegen erlaubt die Verbindung von mehr als zwei Geräten und erfordert keinen direkten Sichtkontakt zwischen den Geräten.

Die Geschichte von Bluetooth[1] beginnt im 10. Jahrhundert (wenn auch diese geschichtlichen Ereignisse keinen direkten Einfluss auf die Entwicklung der Bluetooth-Funktechnik haben) mit Harald Gormsen, König von Dänemark und Einiger des Dänischen Reiches (Vereinigung von Dänemark und Norwegen über weite Teile). Wie üblich zur damaligen Zeit hatte Harald Gormsen einen Beinamen: Blåtand. Die direkte Übersetzung von Blåtand bedeutet nun nicht Blauzahn, sondern weist eher auf die dunkle Erscheinung Harald Gormsens hin; Blå war ein Wort für dunkel im Mittelalter. Auf dem berühmten Runenstein von Jelling (Dänemark), den Harald in Erinnerung an seine Eltern aufstellen ließ, werden Haralds Verdienste um die Einigung und Christianisierung Dänemarks hervorgehoben.

Tausend Jahre[2] später begann das norwegische Unternehmen Ericsson sich für die Verbindung von Mobiltelefonen und anderen Geräten wie z.B. PDAs ohne Kabel zu interessieren und startete Forschungsarbeiten unter dem Titel „Multi-Communicator Link“ [Haartsen1998]. Bald wurde das Projekt umbenannt, da ein Freund der Entwickler Wikinger-Fan war, und so wurde in Anspielung auf Harald Gormsen, der Dänemark und Norwegen ohne Kabel vereinte, der Projektname Bluetooth gewählt. Im Frühjahr 1998 bildeten die Unternehmen Ericsson, Intel, IBM, Nokia und Toshiba die Bluetooth-SIG (Special Interest Group, ein Konsortium) [Bluetooth2003a] mit dem Ziel, eine kostengünstige Lösung für eine drahtlose Netzwerktechnik zu

1 Zur Geschichte von Harald Gormsen siehe auch Microsoft Encarta Enzyklopädie [Microsoft2001]

2 Erste Ideen zu einem kostengünstigen Funkverfahren mit niedrigem Energieverbrauch wurden 1994 in einer von Ericsson initiierten Studie gesammelt; Anfang 1997 begannen erste Arbeiten zum „Multi-Communicator Link“ [Haartsen1998]

entwickeln. Viele weitere Firmen und Forschungseinrichtungen sind mittlerweile dem Konsortium beigetreten [Bluetooth2003a], mit dem gemeinsamen Ziel der Entwicklung von Mobiltelefonen, Laptops, Notebooks, Headsets, Tastaturen, Mäusen usw. mit eingebauter Bluetooth-Funktechnik.

Im Juli 1999 veröffentlichte die Bluetooth-SIG die Version 1.0(a)[3] der Bluetooth-Spezifikation. Kurz danach begann das IEEE-Standardisierungsgremium 802.15 (Wireless-Personal-Area-Networks Arbeitsgruppe, kurz WPAN) die Bluetooth-Spezifikation zu untersuchen und als Grundlage seiner Arbeit zu nutzen [Tanenbaum2003, IEEE2002b]. Die IEEE 802.15-Arbeitsgruppe standardisiert die Bitübertragungs- und Sicherungsschicht des Bluetooth-Protokollstapels; die restlichen Protokolle fallen nicht in ihre Verantwortlichkeit.

Seit dem Jahr 2001 erscheinen immer mehr Produkte auf dem Massenmarkt, die mit Bluetooth-Funkmodulen ausgestattet sind [Ahlers2001, Zivadinovic2003a]. Bluetooth setzt zum Boom an.

2 Bluetooth-Vernetzung

Bluetooth nutzt, wie Geräte nach dem 802.11b- oder 802.11g-Standard auch, das 2,4-GHZ-ISM-Band[4] [Haartsen1998, Tanenbaum2003, Bluetooth2003c]. Für das 2,4-GHz-ISM-Band ist keine Lizenzierung erforderlich. Jedoch unterscheiden sich sowohl das Medienzugriffsverfahren als auch die angebotenen Dienste grundlegend voneinander (für eine detaillierte Darstellung das IEEE 802.11-Standards sei an dieser Stelle auf [Tanenbaum2003] verwiesen).

Bluetooth nutzt das so genannte *Frequency Hopping Spread Spectrum (FHSS)* [Bluetooth2003c, Haartsen1998, Tanenbaum2003]. FHSS verwendet 79 Kanäle innerhalb des 2,4-GHz-ISM-Bandes von denen jeder 1 MHz breit ist. Ein Generator für Pseudozufallszahlen erzeugt die Folge von Frequenzen, auf die gewechselt wird. Verwenden alle Geräte den gleichen Startparameter und sind die gesamte Zeit über synchronisiert, wechseln die Geräte gleichzeitig auf die Frequenzen. Jede Frequenz ist nur für eine gewisse Zeitspanne aktiv. Die Zufallssteuerung von FHSS ist eine gute Möglichkeit, Frequenzen in einem nicht regulierten ISM-Band zuzuweisen. Daneben bietet es auch ein Minimum an Sicherheit, da ein Eindringling nicht weiß, zu welcher Frequenz gewechselt wird und auch nicht die Zeitspanne kennt, für die eine Frequenz gültig ist. Das Abhören einer Verbindung wird dadurch erschwert.

Die Grundeinheit eines Bluetooth-Systems ist ein *Pikonetz* (*Piconet*). Ein Pikonetz ist per Definition eine Ansammlung von Bluetooth-Geräten, welche aus einem *Master*-Knoten und bis zu sieben *Slave*-Knoten besteht. Ein solches Pikonetz ist in Abbildung 1 dargestellt

Zusätzlich zu den maximal sieben aktiven Slave-Knoten sind in Abbildung 1 noch weitere Geräte dargestellt. So genannte *geparkte Geräte* (P, parked) können nicht aktiv an der Kommunikation in einem Pikonetz teilnehmen. Geparkte Geräte haben keine Verbindung, sind jedoch bekannt und können innerhalb weniger Millisekunden reaktiviert werden. *Geräte in Bereitschaft* (SB, standby) nehmen nicht am Pikonetz teil. Eine detaillierte Darstellung der Verbindungsmodi folgt in Kapitel 6. Über 200 Geräte können geparkt werden. Der Grund für die maximale Anzahl von 8 aktiven Geräten pro Pikonetz liegt in der lediglich 3 Bit großen Adresse, welche Bluetooth für den Medienzugriff verwendet. Falls ein geparktes Gerät kom-

3 Die Version 1.0b der Bluetooth-Spezifikation wurde am 01.12.1999 veröffentlicht.

4 Die Bezeichnung ISM-Band steht für Industrial (Industrie), Scientific (Wissenschaft), Medical (Medizin)-Band. Die meisten Regierungen haben ISM-Bänder für den nichtlizensierten Gebrauch reserviert. ISM-Bänder werden bei schnurlosen Telefonen, Funkmäusen, funkgesteuerten Spielzeugen und einer Vielzahl von Haushaltsgeräten (z.B. Mikrowellengeräten) genutzt. Die Zuordnung von ISM-Bändern ist in den Ländern unterschiedlich, das 2,4-GHz-Band ist allerdings in den meisten Ländern verfügbar. (Vergleiche hierzu auch [Tanenbaum2003])

munizieren möchte, sich aber bereits sieben aktive Geräte in einem Pikonetz befinden, muss eine dieser Stationen geparkt werden, damit das bis dahin geparkte Gerät aktiviert werden kann.

Da Bluetooth FHSS verwendet ist es außerordentlich wichtig, dass die Geräte innerhalb eines Pikonetzes miteinander synchronisiert sind. Der erste Schritt der Bildung eines Pikonetzes ist

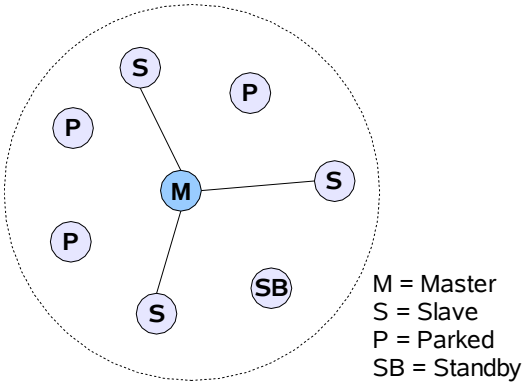

Abbildung 1: Ein Bluetooth-Pikonetz

deshalb, dass eine künftiges Master-Gerät seine Gerätekennung und den Wert seiner internen Uhr aussendet. Alle Bluetooth-Geräte haben prinzipiell die gleichen Fähigkeiten hinsichtlich der Vernetzung, jedes Gerät kann Master oder Slave werden. Dabei gilt, dass das Gerät, welches das Pikonetz einrichtet automatisch zum Master wird, alle weiteren nachfolgenden Stationen sind damit Slave. Die Frequenzsprungfolge der Geräte wird durch die Gerätekennung, eine weltweit eindeutige 48-Bit-Kennung, festgelegt. Die Sprungfolge wird durch die Uhr des Masters bestimmt. Nachdem ein Gerät also seine eigene Uhr an die des Masters angepasst hat, kann es prinzipiell am Pikonetz teilnehmen. Jedes aktive Gerät erhält eine 3 Bit große *Active Member Address (AMA)*, während alle geparkten Geräte eine 8 Bit große *Parked Member Address (PMA)* erhalten. Geräte, die sich im Standby befinden, erhalten keine Adresse.

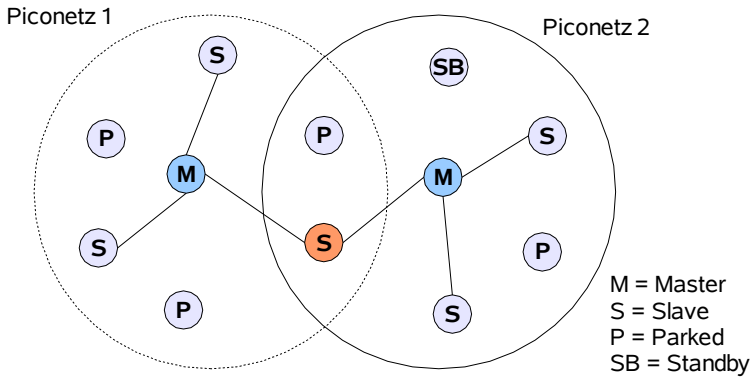

Abbildung 2: Ein Bluetooth-Scatternetz

Alle Geräte innerhalb eines Pikonetzes folgen der selben Sprungsequenz und teilen sich damit auch den gleichen 1 MHz-Kanal. Falls immer mehr Geräte einem Pikonetzt beitreten, sinkt damit der Datendurchsatz pro Gerät sehr schnell. Dies hat zur Einführung so genannter *Scatternetze (Scatternets - Streunetze)* geführt. Scatternetze werden allerdings noch nicht von allen Bluetooth-Geräten beherrscht und auch an der Spezifikation wird weiterhin gearbeitet [Wollert2002]. Abbildung 2 zeigt ein Bluetooth Scatternetz, das aus zwei Pikonetzen besteht. In dem Beispiel nimmt ein Gerät an beiden Pikonetzen teil. Jedes Pikonetz hat dabei seine eige-

ne Sprungfolge, welche durch den jeweiligen Master bestimmt wird. Alle Pikonetze teilen sich dabei die im 2,4-GHz-ISM-Band zur Verfügung stehende Bandbreite von rund 80 MHz.

Will ein Gerät an mehr als nur einem Pikonetz teilnehmen, so muss es sich jeweils auf das Pikonetz synchronisieren, an dem es zu einem bestimmten Zeitpunkt teilnehmen will. Ist ein Gerät ein Slave, muss es sich auf die Sprungfolge des Pikonetzes synchronisieren, an dem es teilnehmen will. Nach der Synchronisation ist es ein Slave im neuen Pikonetz, nimmt aber nicht mehr an dem alten Pikonetz teil. Bevor ein Gerät ein Pikonetz verlässt, informiert es den Master darüber, dass es nun für eine Zeit nicht mehr erreichbar sein wird. Die verbleibenden Geräte im Pikonetz können mit der Kommunikation fortfahren. Ein Master kann ebenso sein Pikonetz verlassen und zum Slave in einem anderen Pikonetz werden. Sobald der Master das Pikonetz verlässt, wird die Kommunikation in dem Pikonetz unterbrochen, bis der Master wieder in das Netz zurückkehrt. Ein Master kann allerdings nicht Master in einem zweiten Pikonetz werden, da dies zu einem identischen Verhalten bzw. identischer Synchronisation und Sprungfolge führen würde, und so aus zwei Pikonetzen ein Pikonetz macht. Die Kommunikation zwischen zwei Pikonetzen wird durch Geräte verwirklicht, die immer zwischen diesen Netzen hin und her springen.

3 Bluetooth-Protokollstapel

Die Bluetooth-Spezifikation [Bluetooth2003c, Bluetooth2003d] umfasst mittlerweile sehr viele Protokolle. Die Spezifikation in der derzeit aktuellen Version 1.1 besteht aus über 2000 Seiten, die nicht nur die Bluetooth-Protokolle, sondern auch eine Vielzahl von Anpassungen

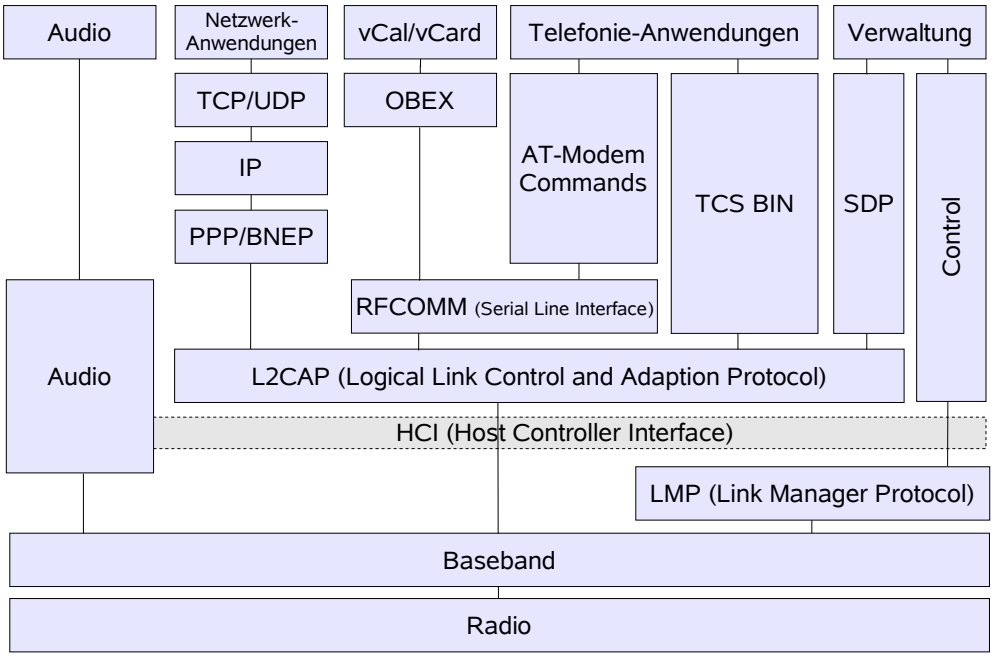

Abbildung 3: Bluetooth-Protokollstapel

(nach [Bluetooth2003c, Palo2003, Siep2001, Tanenbaum2003])

und Erweiterungen beschreiben. Die *Kernspezifikation (Core Specification)* [Bluetooth2003c] von Bluetooth beschreibt die Protokolle von der Bitübertragungsschicht bis zur Sicherungs-

schicht (nach dem OSI-Modells); die *Profilspezifikation (Profile Specification)* [Bluetooth2003d] enthält Protokolle und Funktionen zur Anpassung von Bluetooth an herkömmliche und neue Anwendungen.

Die Protokolle des Bluetooth-Standards können lose in Schichten eingeteilt werden (vgl. Abbildung 3). Die Strukturierung der Schichten folgt dabei weder dem OSI-Modell noch dem TCP/IP-Modell oder einem anderen bekannten Referenzmodell [Tanenbaum2003][5]. Die IEEE arbeitet aber bereits daran, Bluetooth besser in das 802-Modell einzupassen [Tanenbaum2003]. Abbildung 4 stellt die Beziehung des IEEE 802.15.1 Standards (Bluetooth WPAN[6]) zum OSI-Modell dar.

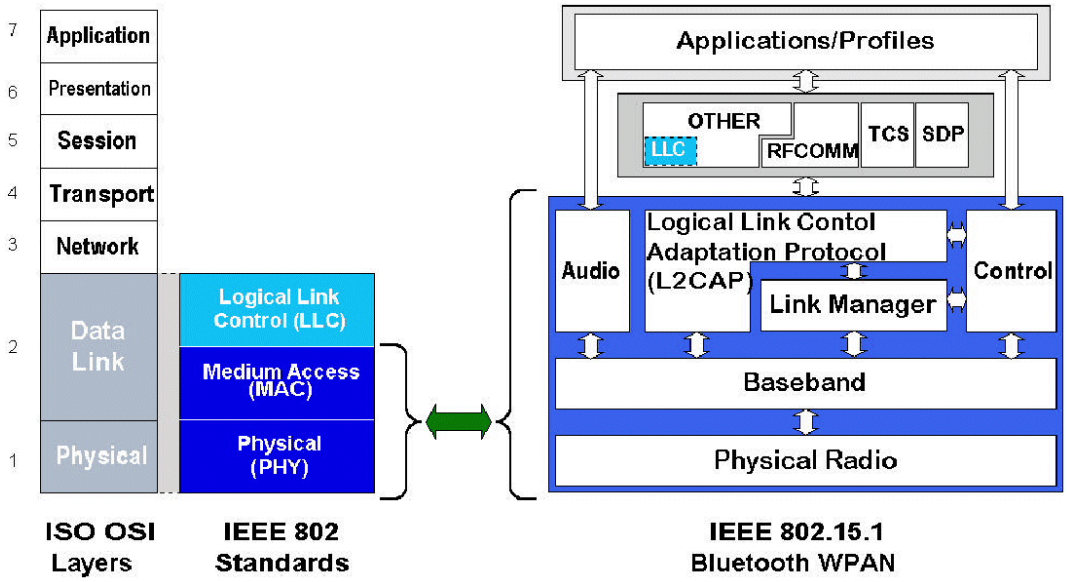

Abbildung 4: Beziehung zwischen IEEE 802.15.1 (Bluetooth WPAN) und dem OSI-Referenzmodell [Siep2001]

In den folgenden Kapiteln werden die grundlegenden Schichten bzw. Protokolle von Bluetooth näher beschrieben. Dies sind:

- *Funkschicht (Radio)*: Diese Schicht behandelt die Funkübertragung, genutzte Frequenzen, Modulationen und Sendeleistungen. Die Funkschicht entspricht im wesentlichen der Bitübertragungsschicht des OSI- bzw. 802-Modells. Die Schicht behandelt die Funkübertragung und Modulation der Signale.
- *Basisband (Baseband)*: Die Basisbandschicht beschreibt die Mechanismen zum Verbindungsaufbau, die Rahmenstruktur sowie das Zeitverhalten. Die Schicht entspricht in etwa der MAC-Teilschicht des 802-Standards (Sicherungsschicht des OSI-Modells), enthält aber auch Elemente der Bitübertragungsschicht.
- *Verbindungsverwaltung und Verbindungssteuerung (Link Manager Protocol, LMP; Logical Link Control and Adaption Protocol, L2CAP)*: Die nächste Schicht besteht aus Protokollen, die lose miteinander in Verbindung stehen. Der Link Manager steuert den Verbindungsaufbau und die Verbindungsverwaltung zwischen zwei Geräten, inklusive Sicherheits- und Authentifizierungsfunktionen. Das L2CAP ist ein Protokoll der Sicherungsschicht, das die höheren Schichten an die Fähigkeiten des Basisbandes anpasst und Über-

5 Für einen Überblick des OSI- und TCP/IP-Referenzmodells siehe auch [Holtkamp2003].
6 WPAN – Wireless Personal Area Networks; Arbeitsgruppe in IEEE 802 (http://www.ieee802.org/15/).

tragungsdetails (verbindungslos und verbindungsorientiert) verbirgt. Daneben bestehen die Audio- und Steuerungsprotokolle, die wie die Namen schon vermuten lassen, die Behandlung von Audio-Daten und Steuerungsdaten regeln. Audio-Anwendungen können z.B. direkt die Basisbandschicht nutzen, nachdem die Audiosignale entsprechend kodiert wurden.

- *Dienstfindung (Service Discovery Protocol, SDP)*: Das SDP dient zur Erkennung und Suche nach Diensten mit bestimmten Eigenschaften und der Beschreibung von Diensten innerhalb der Funkreichweite eines Bluetooth-Gerätes.

Die weiteren Schichten bestehen aus einer Mischung verschiedener Protokolle (die Schichten werden oft auch unter der Bezeichnung „Middleware-Schicht“ zusammengefasst [Tanenbaum2003]). Oberhalb des L2CAP befindet sich das Protokoll RFCOMM (Radio Frequency Communication), welches als Ersatz für ein Kabel eine Serielle Schnittstelle nach RS-232 emuliert (wie z.B. für den Anschluss von Tastatur, Maus, Modem etc. genutzt). Mit RFCOMM können (serielle) Anwendungen und Protokolle weiterhin über Bluetooth genutzt werden. Zur Steuerung von Telefon- bzw. Telefoniefunktionen dient TCS BIN (Telephony Control Protocol Specification – Binary), welches ein Bit-orientiertes Protokoll für die Sprach- und Datenverbindung zwischen Bluetooth-Geräten ist. Die HCI-Schnittstelle (Host Controller Interface) bietet Zugang zum Basisband und zur Verbindungssteuerung. Das HCI kann als die Schnittstelle zwischen der Hardware in einem Bluetooth-Gerät und der Software angesehen werden. In der Regel ist die Funktionalität der Funk- und Basisbandschicht sowie Teile des HCIs durch ein Bluetooth-Modul in Hardware realisiert. Viele weitere Protokolle wurden für den Bluetooth-Standard adaptiert („adopted protocols“) und finden sich im Protokollstapel wieder. Internet-Anwendungen können z.B. weiterhin TCP/IP über das Point-to-Point Protocol (PPP) oder das Bluetooth Network Encapsulation Protocol (BNEP) verwenden. Telefonieanwendungen können die bekannten AT-Kommandos wie bei einer herkömmlichen Modemverbindung nutzen. Zum Austausch von Kalenderinformationen und Visitenkarten (vCalender und vCard) kann das vom IrDA-Standard her bekannte Object Exchange Protocol (OBEX) verwendet werden. Audio-Anwendungen können direkt die Basisbandschicht nutzen, nachdem die Audio-Signale entsprechend kodiert wurden.

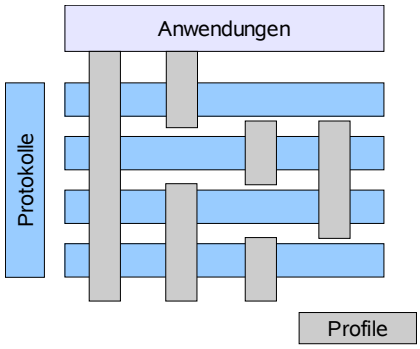

Abbildung 5: Bluetooth-Profile

Auf der obersten Schicht befinden sich die Anwendungen und *Profile*. Profile stellen „Standardlösungen“ für ein bestimmtes Nutzungsszenario dar. Die Bluetooth-Spezifikation [Bluetooth2003d] fasst 13 verschiedene Anwendungen, die als Profile bezeichnet werden, zusammen, die unterstützt werden sollen und stellt für jede quasi einen eigenen Protokollstapel bereit. Jedes Profil nutzt eine gewisse Auswahl an Protokollen. In Abbildung 5 ist der Zusammenhang von Profilen und Protokollen schematisch dargestellt. Profile sind vertikale Balken durch die Protokollarchitektur, deren Protokolle als horizontale Balken gezeichnet sind.

Eine detaillierte Beschreibung einzelner Schichten und Protokolle der Bluetooth-Spezifikation folgt in den nächsten Abschnitten.

4 Bluetooth-Funkschicht

Die Bluetooth-Spezifikation [Bluetooth2003c] beschreibt die Funkschicht mit gerade einmal knapp 15 Seiten. Hier werden lediglich die Trägerfrequenzen und die Sendeleistung festgelegt. Wie in vorhergehenden Kapiteln bereits gesagt, nutzt Bluetooth das 2,4-GHz-ISM-Band (2.400 – 2.4835 GHz) [Bluetooth2003c]. Das Band ist in 79 Kanäle[7] mit einem Abstand von je 1 MHz (f=2402 + k MHz, k=0,...,78) aufgeteilt [Bluetooth2003c]. Am unteren Ende gibt es ein sogenanntes *Lower Guard-Band* mit einer Breite von 2 MHz und am oberen Ende wird der Frequenzbereich durch ein 3,5 MHz breites *Upper Guard-Band* abgeschlossen. Die Schutzbänder sorgen für einen Abstand zu den darüber und darunter liegenden Bändern und vermeiden so Interferenzen zu anderen Frequenzbändern [Wollert2002]. Das 2,4-GHz-ISM-Band ist mit gewissen nationalen Einschränkungen praktisch weltweit lizenzfrei verfügbar.

Die Übertragung erfolgt mit einem kombinierten Frequenzsprung-/Zeitduplexverfahren (Frequency Hop/Time Division Duplex, FH/TDD), dem *Frequency Hopping Spread Spectrum (FHSS)* [Haartsen1998]. Ein Kanal ist in 625µs Intervalle, den sogenannten Slots, eingeteilt; jeder Slot nutzt eine andere (Hop)Frequenz. Dies führt zu einer Sprungrate von 1600 Sprüngen pro Sekunde. Die Zuordnung der Slots erfolgt nach dem TDD-Verfahren. Das bedeutet, dass Sender und Empfänger abwechselnd eine Sendeberechtigung haben.

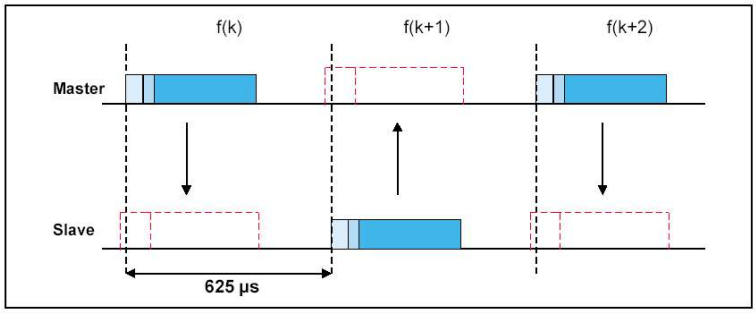

Abbildung 6: FH/TDD Verfahren [Bluetooth2003c]

Als Modulationsverfahren setzt Bluetooth *GFSK (Gaussian Frequency Shift Keying)* ein [Haartsen1998], eine spezielle Variante der *Frequenzumtastung*[8] *(Frequency Shift Keying, FSK)* ein, bei der ein Gauß'scher Tiefpassfilter der Frequenzumtastung zugefügt ist.

Bluetooth-Geräte sind in verschiedene *Leistungsklassen (power classes)* eingeteilt [Bluetooth2003c]. Den Herstellern ist es freigestellt, welche Leistungsklasse sie implementieren. Häufig ist aber vom Anwendungsprofil (siehe Kapitel 9) ein Rückschluss auf die Leistungsklasse möglich. Geräte, die von (kleinen bzw. schwachen) Akkus gespeist werden, z.B. Peripheriegeräte wie Tastatur, Maus etc. oder auch mobile Geräte wie z.B. Handys etc. sind in der

7 Für Länder mit einer Beschränkung des 2,4-GHz-Bandes, wie z.B. Frankreich, Spanien oder Japan [Haartsen1998], bietet Bluetooth ein spezielles Frequenzsprungverfahren. In Frankreich liegt der Frequenzbereich z.B. zwischen 2.4465 und 2.4835 GHz, die entsprechenden Bluetooth-Kanäle sind dort f=2454 + k MHz, k=0,...22 [Bluetooth2003c].

8 Die einfachste Variante der Frequenzumtastung ist die Binary FSK (BFSK), bei der der binären 1 die Frequenz f_1 und der binären 0 die Frequent f_2 zugewiesen wird. Eine einfache Implementierung der BFSK wechselt nun in Abhängigkeit der Daten zwischen zwei Oszillatoren mit bestimmten Frequenzen hin und her. Damit bei der Umschaltung keine plötzlichen Phasensprünge auftreten, können spezielle Frequenzmodulatoren eingesetzt werden, die eine kontinuierliche Phase erzeugen (auch Continuous Phase Modulation, CPM genannt). (vergleiche hierzu u.a. [Tanenbaum2003])

Regel mit einer Sendeleistung von 1 mW bis 2,5 mW (Leistungsklasse 3 und 2) ausgestattet . Andere (stationäre) Geräte wie z.B. Drucker, ISDN/DSL-Adapter, Modems etc. oder auch Notebooks verfügen häufig über eine Sendeleistung von 100mW (Leistungsklasse 1).

Leistungsklasse	Max. Leistung	Nominale Leistung	Minimale Leistung	Reichweite[9]
1	100 mW	N/A	1mW	100 – 150 m
2	2,5 mW	1mW	0,25mW	10 – 25 m
3	1 mW	N/A	N/A	10 m

Tabelle 1: Leistungsklassen von Bluetooth-Geräten (nach [Bluetooth2003c])

5 Bluetooth-Basisbandschicht

Die Funktionen der Basisbandschicht sind vielfältig, sie ist der MAC-Schicht nach dem 802-Modell (Sicherungsschicht nach dem OSI-Modell) am ähnlichsten (vgl. hierzu Abbildung 4). Die Basisbandschicht verwandelt den reinen Bitstrom in Rahmen, definiert physikalische Verbindungen und regelt den schnellen Wechsel der Frequenzen.

Kanaldefinition

Ein *Kanal (Channel)* ist dargestellt durch eine pseudo-zufällige Sprungsequenz zwischen den 79 bzw. 23 möglichen Frequenzen im Bluetooth-Band (siehe Kapitel 4). Jedes Gerät, das aktiv an einem Pikonetz teilnimmt, muss zur gleichen Zeit auf die gleiche Trägerfrequenz springen (die Frequenz f_i). Sobald ein Master Daten auf der Frequenz f_k gesendet hat, kann ein Slave auf der Frequenz f_{k+1} antworten. Dieses Schema ist in Abbildung 6 dargestellt. Üblicherweise beginnen die Übertragungen des Masters in den geraden Zeitschlitzen und die der Slaves in den ungeraden [Bluetooth2003c, Tanenbaum2003]. Diese Eigenschaft entspricht dem normalen Zeitmultiplex-Verfahren (hier TDD), bei dem der Master eine Hälfte der Zeitschlit-

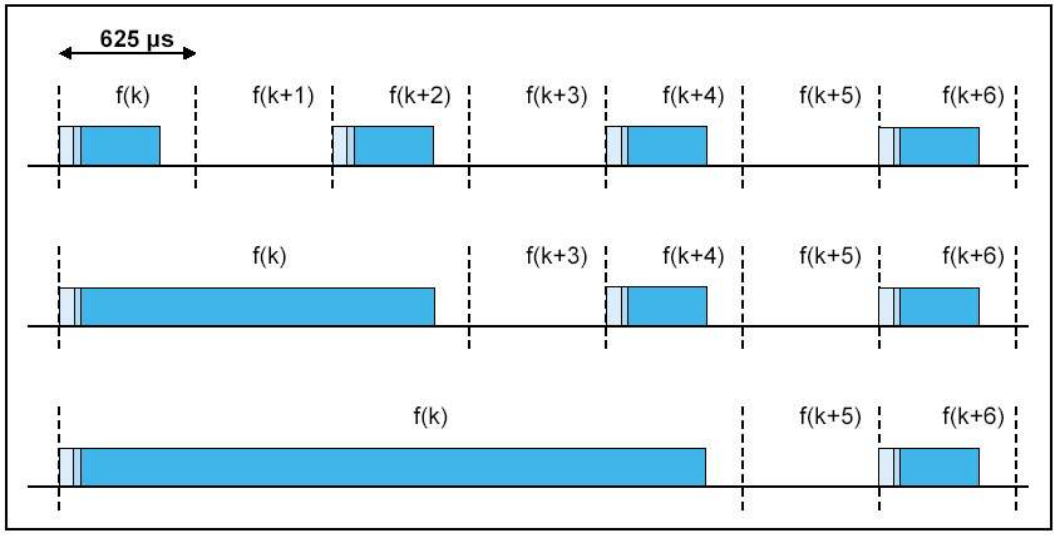

Abbildung 7: Multi-Slot Pakete / Frequenzwahl bei Multi-Slot Paketen

[Bluetooth2003c]

9 Angaben zu den Reichweiten aus [Zivadinovic2003b] und [Zivadinovic2003c]

ze verwendet und die Slaves die andere Hälfte. Bluetooth beschreibt neben Paketen, die einen Zeitschlitz belegen können, Pakete, die 3 oder 5 Zeitschlitze belegen können (*multi-slot packets*). Sobald ein Master oder Slave Pakete sendet, das 3 oder 5 Zeitschlitze lang ist, bleibt der Sender auf der gleichen Frequenz. Innerhalb eines Paketes findet kein Frequenzwechsel statt (siehe Abbildung 7). Nach der Übertragung des Pakets folgt der Wechsel auf die Frequenz, die durch die Sprungfolge (unabhängig vom Sendevorgang) vorgegeben ist (in Abbildung 7 folgt z.B. der Sprung nach dem Senden eines 3-Slot Paketes von der Frequenz f_k auf die Frequenz f_{k+3}). Grund für dieses Verhalten ist, dass eventuell nicht jede Station die Übertragung mitbekommen hat und deshalb nicht speziell auf die Übertragung von Daten in mehreren Zeitschlitzen reagieren kann. Alle Stationen, die nicht an der Übertragung beteiligt sind fahren deshalb immer mit der durch den Master vorgegebenen Sprungfolge fort.

Die Sprungsequenz innerhalb eines Pikonetzes wird durch den Master bzw. durch die Geräteadresse des Masters vorgegeben. Die Phase der Sprungsequenz ist durch die interne Uhr des Masters, die mit den anderen Geräten im Pikonetz synchronisiert ist, bestimmt.

Bluetooth-Geräteadressen

Jedem Bluetooth-Gerät ist eine weltweit eindeutige *Bluetooth-Geräteadresse (Bluetooth Device Address – BD_ADDR)* zugeordnet. Die Adresse ist vom IEEE 802-Standard abgeleitet [Bluetooth2003c]. Die 48-Bit großen Adressen sind in drei Teile untergliedert

- LAP – Lower Address Part, mit 24 Bit
- UAP – Upper Address Part, mit 8 Bit
- NAP – Non-significant Address Part, mit 16 Bit.

Die Felder LAP und UAP bilden den so genannten *signifikanten Teil (significant part)* einer Bluetooth-Geräteadresse.

Company Assigned	Company ID	
LAP	UAP	NAP

Abbildung 8: Bluetooth-Geräteadresse (nach [Bluetooth2003c])

Bluetooth-Paketformat

Abbildung 9 stellt die Struktur eines *Bluetooth-Pakets*[10] im Basisband dar. Ein Paket besteht in der Regel aus den folgenden drei Feldern: *Zugriffscode (Access Code)*, *Paketkopf (Header)* und den *Nutzdaten (Payload)*. Ein Paket kann aus lediglich dem Zugriffscode bestehen, aus dem Zugriffscode und einem Paketkopf, sowie aus Zugriffscode, Paketkopf und Nutzdaten.

Ein Paket beginnt mit einem *Zugriffscode (Access Code)*, der 72 Bit lang ist, falls ein Paketkopf folgt, andernfalls ist er 68 Bit lang. Der Zugriffscode wird zur Synchronisation und Identifizierung eines Pikonetzes, zur Geräteabfrage und zum Geräteruf genutzt. Es gibt drei verschiedene Zugriffscodetypen:

10 In der Literatur wird ein Bluetooth-Paket auch oft als Rahmen (analog zur Bezeichnung Rahmen bei der Sicherungsschicht nach dem OSI-Modell) bezeichnet (vgl. [Tanenbaum2003]). Ich verwende hier durchgängig die Bezeichnung Paket wie in der Bluetooth-Kernspezifikation [Bluetooth2003c]. Zur Unterscheidung von Paketen der Basisbandschicht und Paketen höherer Schichten ist es allerdings angebracht, bei der Basisbandschicht von Rahmen zu sprechen.

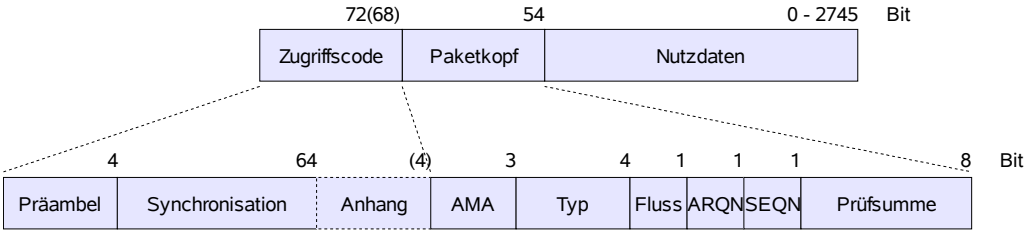

Abbildung 9: Bluetooth-Paket im Basisband

- *Channel Acces Code (CAC)*; dieser dient zur Synchronisation und Erkennung eines Pikonetzes. Der CAC wird mit jedem Paket, das über das Pikonetz versendet wird, geschickt.
- *Device Access Code (DAC)*; wird zum Übertragen bestimmter Kennungen während des Geräterufs genutzt (z.B. beim Paging).
- *Inquiry Access Code (IAC)*; wird zur Geräteabfrage verwendet.

Der Zugriffscode selbst besteht aus einer 4 Bit großen *Präambel (Preamble)*, einem 64 Bit großen *Synchronisationsfeld (Synchronization)* und einem 4 Bit großen *Anhang (Trailer)*, falls ein Paketkopf folgt. Die Präambel ist ein festgelegtes Muster von 4 Zeichen, dass nach folgendem Schema definiert ist: Falls das der Präambel folgende Synchronisationsfeld mit einer 0 beginnt[11], lautet die Präabel 0101, andernfalls lautet die Präambel 1010. Das 64 Bit große Feld zur Synchronisation wird von der 24 Bit großen LAP (siehe oben, Bluetooth-Geräteadressen) abgeleitet. Falls der Zugangscode zu einer Datenübertragung genutzt werden soll (also ein CAC), so wird der LAP der weltweit eindeutigen 48-Bit-Geräteadresse des Masters verwendet. Falls ein Gerät gerufen werden soll (also ein DAC), wird der LAP des gerufenen Gerätes verwendet. Will ein Bluetooth-Gerät andere, beliebige Geräte innerhalb eines Bereiches finden, nutzt es einen speziell reservierten LAP[12], dieser Zugriffscode wird als *GIAC – General Inquiry Access Code* [Bluetooth2003c] bezeichnet. Soll eine bestimmte Gruppe Bluetooth-Geräte gefunden werden werden hierfür weiterhin spezielle LAPs verwendet, dieser Zugriffscode heißt *DIAC – Dedicated Inquiry Access* [Bluetooth2003c]. Der Trailer wird dem Zugriffscode angefügt, sobald ein Paketkopf dem Access Code folgt. Der Trailer ist wie die Präambel ein fest definiertes Muster von vier Zeichen. Die Anhangssequenz ist entweder 1010, wenn das MSB[12] eine 0 ist, oder 0101, wenn das MSB eine 1 ist.

Der *Paketkopf (Header)* eines Bluetooth-Pakets ist 54 Bit lang und enthält Informationen über die Verbindung (Link Control – LC – Information). Der Header besteht aus sechs Feldern (siehe Abbildung 9) und umfasst die typischen Bestandteile nach der Schicht 2 des OSI-Modells: Adresse, Pakettyp, Flusssteuerung, Fehlerüberwachung und Prüfsumme. Die eigentliche Länge des Paketkopfes (d.h. die Summe der Bits im Header) beträgt 18 Bit. Der Header wird durch einen *Vorwärtsfehlerkorrekturmechanismus (Forward Error Correction)*[13] geschützt, da der Paketkopf wesentliche Verbindungsdaten enthält. Die Bits des Paketkopfes werden dreifach gesendet (*1/3 FEC*). Ein Empfänger kann dann einfach eine Mehrheitsentscheidung tref-

11 Die Bit-Reihenfolge in Bluetooth-Paketen und -Nachrichten folgt dem so genannten Little Endian Format. Das heißt, dass das erste empfangene Bit, das von einer höheren Schicht stammt, als b_0 interpretiert wird. Pakete und Nachrichten werden dem Least Significant Bit, LSB (geringst signifikante Bit), beginnend übertragen. Der 3-Bit-Parameter X=3 wird so als $b_0b_1b_2 = 110$ übertragen. Analog zum geringst signifikanten Bit wird das höchst- oder meistsignifikante Bit als MSB (Most Significant Bit) bezeichnet. In dem Beispiel entspricht $b_2 = 0$ dem MSB.

12 Genauere Angaben hierzu stehen in [Bluetooth2003c].

13 Grundprinzip der Vorwärtsfehlerkorrektur ist, redundante Informationen zum ursprünglichen Paketstrom zuzufügen. Für die Kosten einer nur relativ geringen Erhöhung der Übertragungsrate des Bitstroms können die redundanten Daten genutzt werden, um Fehler bei der Übertragung zu kompensieren oder verlorene Pakete zu rekonstruieren. Der Aufwand einer Neuübertragung eines Paketes wird so gespart.

fen; jedes Tripel eines Bits wird auf den Wert abgebildet, der die Mehrheit im Tripel hat. Die Übertragung des Headers mit einer 1/3 FEC führt zur Headerlänge von 54 Bit: 3 x 18 Bit.
Im ersten Feld des Paketkopfes wird die *Active Member Address (AM_ADDR* oder *AMA)* übertragen. Die AM_ADDR wird einem Gerät innerhalb eines Pikonetzes vorübergehend durch den Master des Pikonetzes zugewiesen. Die Adresse 0 ist die Master-Adresse, diese Adresse wird für *Broadcasts (Gruppenübertragung)* verwendet. Sobald ein Master Daten an einen Slave sendet, wird die AM_ADDR als Empfängeradresse interpretiert. Will ein Slave Daten an den Master senden, so entspricht die AM_ADDR der des Absenders. Dieses einfache Schema ist hinreichend, da nur ein Master mit einem Slave kommunizieren darf. Aufgrund der Größe des Feldes von 3 Bit können 7 mögliche aktive Bluetooth-Geräte in einem Pikonetz eindeutig adressiert werden.
Das *Typfeld (Type)* dient zur Kodierung des Pakettyps; 16 verschiedene Pakettypen sind möglich (siehe auch Tabelle 2 auf Seite16). Pakete können entweder der Steuerung dienen oder zur Übertragung von synchronen oder asynchronen Datenpaketen. Durch den Pakettyp ist auch die Slot-Länge eines Paketes kodiert, so dass der Empfänger sich auf die möglichen 1, 3 oder 5 Slot großen Pakete einstellen kann. Eine Darstellung der verschiedenen Pakettypen folgt nach der Vorstellung der Physikalischen Verbindungen.
Die folgenden drei Felder im Paketkopf dienen der Steuerung des Datenflusses. Mit Hilfe des *Flow-Bits (F, Flusskontrolle)*, kann eine einfache Flusssteuerung für den asynchronen Datenverkehr umgesetzt werden. Sobald ein Paket mit gelöschtem F-Bit empfangen wird, muss die Datenübertragung angehalten werden. Ein Gerät signalisiert damit, dass der Empfangspuffer voll ist. Die Übertragung wird dann wieder aufgenommen, wenn das F-Bit auf 1 gesetzt ist.
Zur Empfangsbestätigung von Paketen dient die *Acknowledgement Number (ARQN, Bestätigungsnummer)*. Eine positive Empfangsbestätigung wird durch ARQN=1 *(positive acknowledgement)* signalisiert, andernfalls wird eine negative Quittung (ARQN=0, *negative acknowledgement – NAK*) gesendet. NAK ist die voreingestellte Antwort.
Mit dem Bit für die Folgenummer (SEQN, Sequence Number) kann ein Paketverlust erkannt werden. Bei jedem Paket, das einen CRC-Wert[14] enthält (dies sind nur bestimmte Pakete; siehe dazu auch den Abschnitt zu den Pakettypen), wird das Bit für die Folgenummer invertiert. Wurde aufgrund einer negativen Empfangsbestätigung ein Paket erneut gesendet, so kann der Empfänger dies anhand der Sequenznummer erkennen.
Dieses einfache Schema zur Empfangsbestätigung reicht aus, da Bluetooth TDD, also eine Richtungstrennung in der Zeit verwendet. Wird eine Empfangsbestätigung für ein Paket gefordert, so sendet Bluetooth diese Bestätigung in dem den Daten nachfolgenden Zeitschlitz.
Die 10 Bits des Paketkopfes werden durch eine 8-bittige Prüfsumme (HEC, Header Error Correction) geschützt. Anhand der Prüfsumme kann festgestellt werden, ob Daten im Header verfälscht wurden. Zur genauen Berechnung der HEC sei an dieser Stelle auf die Bluetooth-Spezifikation verwiesen: [Bluetooth2003c].

Die *Nutzdaten (Payload)* eines Bluetooth-Pakets können bis zu 343 Bytes groß sein. Das Datenfeld der Nutzdaten besteht aus bis zu drei Segmenten: dem Nutzdaten-Header, den eigentlichen Nutzdaten und unter Umständen einer Prüfsumme (siehe Abbildung 11 und 12). Die Struktur der Nutzdaten hängt vom Verbindungstyp ab und wird im Abschnitt zu den Bluetooth-Paketen genauer beschrieben.

Physikalische Verbindungen

Bluetooth bietet zwei verschiedene Kommunikationstypen bzw. der physikalischen Verbindungen zwischen Geräten an:

14 Cyclic Redundancy Check

- die leitungsvermittelte synchrone Kommunikation und
- die paketvermittelte asynchrone Kommunikation.

Leitungs- und Paketvermittlung (Circuit Switching, Packet Switching)[15] unterscheiden sich in vielerlei Hinsicht. Der erste wichtige Unterschied ist, dass bei der Leitungsvermittlung vor Beginn der Kommunikation eine (virtuelle) Leitung vom Sender zum Empfänger eingerichtet werden muss. Bei der Paketvermittlung ist keine vorherige Einrichtung erforderlich. Leitungsvermittelte Dienste sind aus der Telefonie bekannt. Möchte ein Teilnehmer mit einem anderen Teilnehmer kommunizieren ist zuerst eine Verbindung aufzubauen. Erst nachdem die Verbindung aufgebaut worden ist, kann die Kommunikation zwischen den Teilnehmern stattfinden. Der Vorteil der Leitungsvermittlung ist, dass die für die Verbindung eine feste Bandbreite definiert ist und diese für die Dauer der Verbindung reserviert ist. Leitungsvermittelte Dienste werden im allgemeinen für die Übertragung von Sprache genutzt, da sich hier Variationen in der Übermittlung direkt hörbar (Hall, Echo etc.) auswirken. Bei der Paketvermittlung gibt es keinen festen Weg, so dass Pakete auch auf verschiedenen Wegen gesendet werden können, um den Adressaten zu erreichen. Pakete können auch in einer anderen Reihenfolge als der Sendereihenfolge ankommen. Aus diesem Grund setzt die Paketvermittlung ein speicherndes Verhalten der Übertragungsgeräte voraus. Das Internet ist das wohl bekannteste Beispiele für ein paketvermitteltes Netz. Datenpakete werden mit einer Zieladresse versehen und dann auf den Weg gegeben. Auf dem Weg kann das Paket mehrere Stationen (Router) passieren, die das Paket jeweils weiter versenden, bis es beim Empfänger angelangt ist. Dieser sendet bei Empfang eine Bestätigung an den Sender des Paketes.[16]

Die zwei unterschiedlichen Vermittlungstechniken werden bei Bluetooth mit unterschiedlichen *physikalischen Verbindungen (Physical Links)* realisiert. Für die leitungsvermittelte synchrone Kommunikation bietet Bluetooth die *SCO-Verbindung (Synchronous Connection Oriented Link)*; die paketvermittelte asynchrone Kommunikation wird durch eine *ACL-Verbindung (Asynchronous Connection-Less Link)* realisiert.

- Die synchrone verbindungsorientierte Kommunikation (*SCO-Link*) realisierte eine symmetrische, leitungsvermittelte Punkt-zu-Punkt-Verbindung zwischen einem Master und einem Slave. Der Master reserviert in regelmäßigen Abständein Zeitschlitze für die Übertragung; der Master kann in einem festgelegten Zeitschlitz (den so genannten SCO Intervallen, T_{SCO}) Daten an den Slave senden, der Slave kann in dem darauf folgenden Zeitschlitz seine Daten senden. Ein Master kann bis zu drei SCO-Verbindungen zu einem oder mehreren Slaves unterstützen. Ein Slave kann bis zu drei SCO-Verbindungen zu einem Master oder maximal zwei SCO-Verbindungen zu unterschiedlichen Mastern unterhalten. SCO-Verbindungen sind darauf ausgerichtet, eine effiziente Sprachübertragung zu gewährleisten. Jede SCO-Verbindung kann kann Sprache mit 64 kbit/s übertragen (siehe auch Tabelle 3 auf Seite 18) [Bluetooth2003c]. Bei SCO-Verbindungen findet keine Überprüfung der Datenintegrität statt. In dem Fall, dass Daten bei der Übermittlung verloren gehen, findet keine erneute Übertragung statt, da dies für die nachfolgenden Datenpakete eine Verzögerung bedeuten würde. Zur Kodierung der Sprachdaten wird ein robustes Verfahren, die so genannte Continuous Variable Slope Delta (CVSD) Modulation, eingesetzt [Haartsen1998]. Ein weiterer *CODEC (Codierer/Decodierer)* greift auf die *logarithmische Puls Code Modulation (PCM)* mit zwei Charakteristika (*A-law* und *μ-law*) zurück [Bluetooth2003c]. Dieses Verfahren wird z.B. auch bei der ISDN-Telefonie eingesetzt. Bei der Übertragung der

15 Für eine ausführliche Darstellung der Paket- und Leitungsvermittlung siehe u.a. [Tanenbaum2003].

16 Dies ist eine äußerst stark vereinfachte Darstellung der Übertragung von Daten im Internet (bzw. einem paketvermittelten Netz). Für eine ausführlichere Darstellung siehe u.a. [Tanenbaum2003] oder [Holtkamp2003].

Sprachdaten kann entweder keine Vorwärtsfehlerkorrektur (HV3-Paket), 1/3 FEC (HV1) oder 2/3 FEC (HV2) eingesetzt werden. Die FEC 1/3 entspricht der Fehlerkorrektur für den Paketkopf und verdreifacht das Datenvolumen. Die FEC verursacht stets einen Mehraufwand, kann jedoch eine Wiederholung der Übermittlung vermeiden, wenn eine zuverlässige Übertragung gefordert ist. Für weitere Details zur Fehlerkorrektur in den Bluetooth-Paketen sei an dieser Stelle auf die Bluetooth-Kernspezifikation [Bluetooth2003c] verwiesen.

- Die asynchrone verbindungslose Kommunikation (*ACL-Link*) stellt einen verbindungslosen, paketvermittelnden Dienst zur Verfügung. Ein ACL-Link kann immer dann genutzt werden, wenn der Kanal nicht für einen SCO-Link reserviert ist. Zwischen einem Master und einem Slave kann zu einem Zeitpunkt immer nur eine ACL-Verbindung aufgebaut sein. Im Rahmen einer ACL-Verbindung kann ein Master auch Pakete an alle Slaves in seinem Pikonetz versenden (Rundruf). In diesem Fall gibt der Master im Paketkopf keine Zieladresse an (s.o.). ACL-Verbindungen sind für eine effiziente Datenübertragung ausgelegt. Bei der Übermittlung spielt dabei die Verzögerung meist eine untergeordnete Rolle, während die Datenintegrität sehr wichtig ist. Für die Datenübertragung können Pakete für einen, drei oder fünf Zeitschlitze genutzt werden. Die Nutzlast wird außer bei AUX1-Paketen (siehe den nachfolgenden Abschnitt) stets über eine Prüfsumme abgesichert. Zusätzlich können die Nutzdaten mit einer 2/3 FEC geschützt werden. Es kann jedoch sein, dass der Mehraufwand für die Übertragung mittels der 2/3 FEC zu hoch ist. Deshalb stellt Bluetooth neben den zwei Verfahren zur Vorwärtsfehlerkorrektur ein Verfahren zur *automatischen Übertragungswiederholung (Automatic Repeat Request – ARQ)* zur Verfügung, um damit eine zuverlässige Datenübertragung anzubieten. Für Details zum ARQ-Schema sei auf die Bluetooth-Kernspezifikation verwiesen: [Bluetooth2003c]. Während ein SCO-Link immer *symmetrisch* ist, das heißt, dass Hin- und Rückkanal die gleiche Bandbreite haben, kann ein ACL-Link sowohl symmetrisch als auch *unsymmetrisch* betrieben werden. Bei der symmetrischen Verbindung werden sowohl Hin- als auch der Rückkanal mit der gleichen Bandbreite genutzt. Bei einer asymmetrischen Verbindung kann gewählt werden, welche Bandbreite im Hin- und Rückkanal genutzt werden soll. Eine Übersicht gibt Tabelle 3.

Pakete

In jeder der Verbindungsarten stehen verschiedene Pakettypen, die versendet werden dürfen, zur Verfügung. Die Pakettypen sind in den Tabellen 2 und 3 aufgeführt. Die verschiedenen Pakettypen werden im Feld *Type* des Bluetooth-Paketkopfs (siehe Abbildung 9) mit 4 Bits kodiert. Neben Paketen für ACL-Links und SCO-Links gibt es auch gemeinsame Pakete zur Abfrage von Slaves, zur Synchronisation der Sprungfolge und zur Bestätigung von Datenübertragungen.

- Sowohl für SCO- als auch ACL-Links stehen fünf *gemeinsame Pakettypen zur Verbindungssteuerung (Link-Pakete)* zur Verfügung:
  - Das *ID-Paket* besteht lediglich aus dem Zugriffscode (DAC oder IAC) und hat damit eine Länge von 68 Bit. ID-Pakete werden für Paging, Inquiry oder Antworten verwendet.
  - *NULL-Pakete* enthalten neben dem Zugriffscode (CAC) nur den Paketkopf und haben eine Länge von 126 Bit. NULL-Pakete enthalten Informationen über den Empfänger und den Datenfluss. Sie dienen der Antwort von Datenpaketen und werten das Bit zur Flusssteuerung (Flow) und zur Bestätigung (ARQN) aus. NULL-Pakete werden nicht beantwortet.

- Das *POLL-Paket* ist dem NULL-Paket sehr ähnlich. Es wertet aber nicht die Bits zur Flussteuerung und Bestätigung aus, muss aber vom Empfänger bestätigt werden. Das Paket wird dazu verwendet, um zu prüfen ob noch (gültige) Slaves im Pikonet vorhanden sind (*polling*).
- Das wichtigste Paket zur Verbindungssteuerung ist das *FHS-Paket (Frequency Hopping Synchronisation)*. Mit dem FHS-Paket werden die Stationsadresse, die Uhr (Clock), die aktive Mitgliedsadresse und weitere Informationen des Senders übermittelt. Die Informationen über die Uhr werden bei jedem Aufbau eines Pikonetzes zwischen dem Master und den Slaves ausgetauscht, so dass sich die Slaves mit dem Netz synchronisieren können. FHS-Pakete werden ebenfalls übertragen, falls bei der Kommunikation im Pikonetz Synchronisationsinformationen notwendig sind. Die Nutzdaten eines FHS-Pakets betragen 144 Bit plus 16 Bit CRC. Die Nutzdaten sind mit einer 2/3 FEC versehen, was die übertragenen Nutzdaten auf 240 Bit erhöht [Bluetooth2003c]. Abbildung 16 stellt das Format eines FHS-Paketes dar[17].

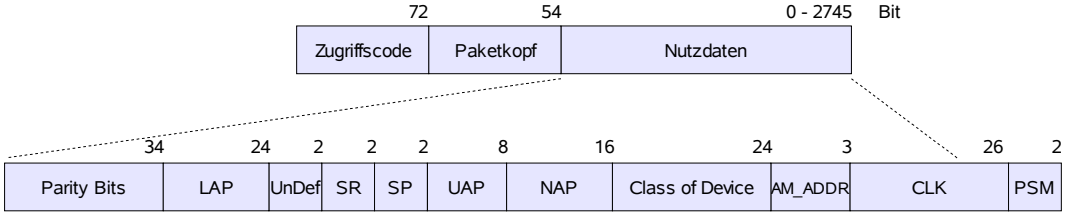

Abbildung 10: Format des FHS-Pakets (nach [Bluetooth2003c])

- Das *DM1-Paket (Data Medium Rate 1 Slot)* dient zur Übertragung von Steuerinformationen. Das Paket kann auch zur Übertragung normaler Nutzdaten dienen (deshalb wird es bei der Beschreibung der Pakete für ACL-Links erneut angesprochen). Ein DM1-Paket kann die synchrone Kommunikation stoppen (da es ein gemeinsames Paket ist, wird es auch bei einem SCO-Link erkannt), um Kontrollinformationen zu senden.

Segment	Paketcode	Slots	SCO-Link	ACL-Link
1	0000	1	NULL	NULL
	0001	1	POLL	POLL
	0010	1	FHS	FHS
	0011	1	DM1	DM1
2	0100	1	-	DH1
	0101	1	HV1	-
	0110	1	HV2	-
	0111	1	HV3	-
	1000	1	DV	-
	1001	1	-	AUX1
3	1010	3	-	DM3
	1011	3	-	DH3
	1100	3	-	-
	1101	3	-	-
4	1110	5	-	DM5
	1111	5	-	DH5

Tabelle 2: Bluetooth-Pakettypen [Bluetooth2003c]

17 Für eine übersichtliche und detaillierte Darstellung des FHS-Pakets sein auf [Wollert2002} verwiesen.

- Für *SCO-Links* werden ausschließlich 1-Slot große Pakete verwendet. In derzeitigen Bluetooth-Spezifikation [Bluetooth2003c] sind im wesentlichen drei so genannte *HV-Pakete (Highquality Voice)* definiert. Die Pakete sind generell 240 Bit groß und unterscheiden sich in der Kodierung ihrer Nutzdaten (siehe auch Tabelle 3). HV-Pakete werden niemals erneut versendet und sind nicht über eine Prüfsumme geschützt. Neben den Paketen zur reinen Sprachübertragung steht noch ein 240 Bit großes Paket zur gemeinsamen Übertragung von *Sprache und Nutzdaten* zur Verfügung (*Data Voice Packet – DV*).

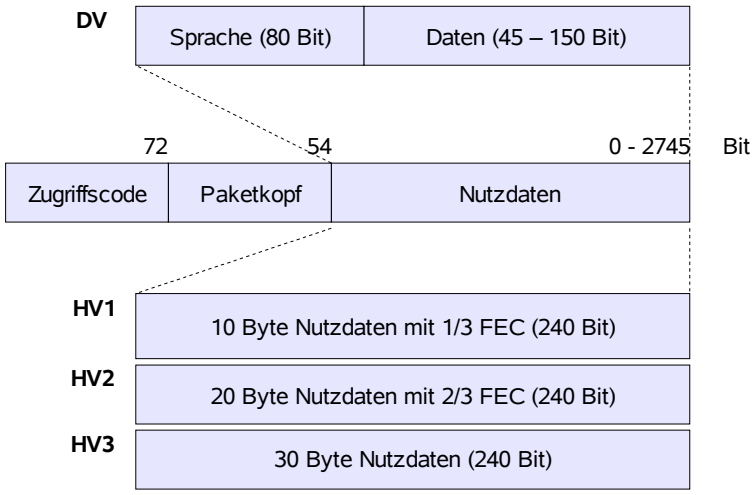

Abbildung 11: SCO-Link Pakete (nach [Bluetooth2003c] und [Wollert2002])

  - HV1-Pakete (Highquality Voice FEC 1/3) transportieren 10 Bytes Nutzdaten, dies entspricht 1,25 ms Sprache bei einer Datenrate von 64kbit/s [Bluetooth2003c]. HV1-Pakete sind mit einer 1/3 FEC versehen. Die Pakete werden in jedem zweiten Slot versendet (T_{SCO}=2).
  - Ein HV2-Paket kann 20 Bytes Nutzdaten versenden, was 2,5 ms Sprache entspricht. Die Pakete werden mit einer 2/3 FEC geschützt. HV2-Pakete müssen in jedem vierten Slot gesendet werden (T_{SCO}=4).
  - HV3-Pakete werden vollkommen ohne Redundanz übertragen. Die 240 Bit des HV-Pakets dienen zur reinen Übertragung der Nutzdaten, die demzufolge 30 Byte groß sein können. Mit einem HV3-Paket lassen sich 3,75 ms Sprache übertragen. Die Pakete müssen in jedem sechsten Slot gesendet werden (T_{SCO}=6).
  - Das DV-Paket ist ein kombiniertes Sprach- und Datenpaket mit dem 10 Byte Sprache und bis zu 150 Bit Daten übertragen werden können. Das Sprachfeld ist ohne eine FEC versehen, während das Datenfeld durch eine 2/3 FEC geschützt ist.
- Für *ACL-Links* können Pakete mit einer Länge von 1 Slot, 3 Slots oder 5 Slots genutzt werden. Im wesentlichen werden drei Pakettypen unterschieden: DM*x*-Pakete, DH*x*-Pakete (das *x* steht für die Anzahl Slots, die ein Paket nutzen kann) und das AUX1-Paket.
  - DM*x*-Pakete haben eine mittlere Datenrate, daher die Bezeichnung *DM* für *Data Medium Rate.* DM-Pakete werden mit einer 2/3 FEC übertragen. DMx-Pakete können 18 Byte (DM1), 123 Byte (DM3) oder 226 Byte (DM5) an Nutzdaten übertragen werden. Die Nutzdaten werden mit einer Prüfsumme (CRC) versehen. Der Aufbau und die Grö-

ße des Nutzdaten-Headers ist abhängig von der Länge des Pakets. 1-Slot-Pakete haben einen 1 Byte großen Nutzdaten-Header, Multi-Slot-Pakete haben einen 2 Byte großen Header (siehe Abbildung 12).

- DH*x*-Pakete werden ohne FEC übertragen, weshalb sie die Bezeichnung *DH* für *Data High Rate* tragen. Da DHx-Pakete ohne FEC übertragen werden können bei den einzelnen Paketen mehr Nutzdaten übertragen werden: 28 Byte bei DH1-Paketen, 185 Byte bei DH3-Paketen und bis zu 341 Byte bei DH5-Paketen. Die Nutzdaten werden wie bei den DM-Paketen mit einer Prüfsumme (CRC) versehen. Wie bei den DMx-Paketen ist der Aufbau und die Größe des Nutzdaten-Headers ebenfalls von der Länge des Paketes abhängig.
- AUX1-Pakete entsprechen einem DH1-Paket, wobei auf eine Prüfsumme (CRC) verzichtet wird. Ein AUX1-Paket kann bis zu 30 Byte Nutzdaten in einem Slot übertragen

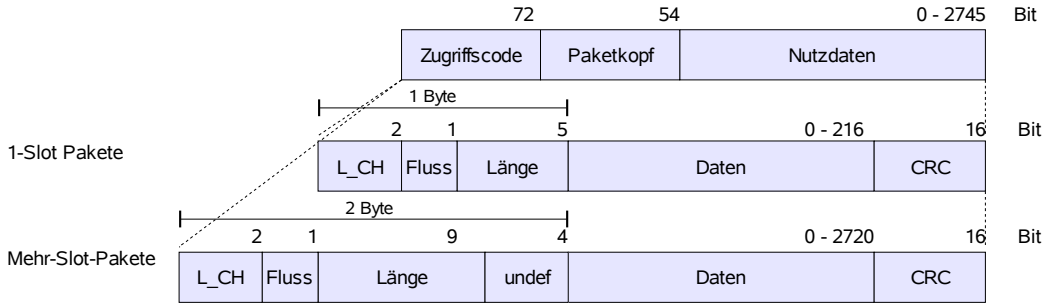

Abbildung 12: Nutzdaten-Header für ACL-Link 1-Slot- und Mehr-Slot-Pakete

(nach [Bluetooth2003c] und [Wollert2002])

							Asymm. Datenrate [kbps]	
	Type	**Header [Byte]**	**Nutzlast [Byte]**	**FEC**	**CRC**	**Symm. Datenrate [kbps]**	**Hinkanal**	**Rückkanal**
Link-Pakete	ID	-	-	-.	-	-	-	-
	NULL	-	-	-	-	-	-	-
	POLL	-	-	-	-	-	-	-
	FHS	-	18	2/3	ja	-	-	-
ACL-Pakete	DM1	1	0-17	2/3	ja	108,8	108,8	108,8
	DH1	1	0-27	nein	ja	172,8	172,8	172,8
	DM3	2	0-121	2/3	ja	258,1	387,2	54,4
	DH3	2	0-183	nein	ja	390,4	585,6	86,4
	DM5	2	0-224	2/3	ja	286,7	477,8	36,3
	DH5	2	0-339	nein	ja	433,9	723,2	57,6
	AUX1	1	0-29	nein	nein	185,6	185,6	185,6
SCO-Pakete	HV1	-	10	1/3	nein	64,0	-	-
	HV2	-	20	2/3	nein	64,0	-	-
	HV3	-	30	nein	nein	64,0	-	-
	DV	1	10+(0-9) D	2/3 D	ja D	64,0+57,6 D	-	-

Tabelle 3: Bluetooth-Datenpakete (nach [Bluetooth2003c])

Details zu den einzelnen Pakettypen, insbesondere auch zum Aufbau der ACL-Link Nutzlasttypen werden in der Bluetooth-Kernspezifikation [Bluetooth2003c] und in einer sehr übersichtlichen Form in [Wollert2002] beschrieben.

Logische Kanäle

Logische Kanäle beziehen sich auf verschiedene Typen von Kanälen, die über eine physikalische Verbindung laufen. Die Daten, die über einen physikalischen Kanal übertragen werden, haben unterschiedliche logische Bedeutungen. Bluetooth definiert fünf logische Kanäle für Steuer- und Benutzerinformationen:

- Link Control (LC)
- Link Manager (LM)
- User Asynchronous (UA)
- User Isochronous (UI)
- User Synchronous (US)

Der *Link Control Channel (LCC)* und der *Link Manager Channel (LMC)* stellen Steuerkanäle dar, die auf der Ebene der Verbindungssteuerung bzw. des Link-Managers verwendet werden. Die Nutzerkanäle dienen der Übertragung von asynchronen, isochronen und synchronen Daten, die aus der Benutzerebene, den Applikationen kommen. Unterschieden wird zwischen zwei logischen Kanälen auf der Basis der physikalischen ACL-Links, dem *User Asynchronous Channel (UAC)* und dem *User Isochronous Channel (UIC)*, und einem synchronen logischen Kanal auf Basis des SCO-Links, der *User Synchronous Channel (USC)*. Weitere Details zu den logischen Kanälen sind in [Bluetooth2003c] und [Wollert2002] beschrieben.

6 Bluetooth-Verbindungsverwaltung

Das Link Manager Protokoll (LMP) für die Verbindungsverwaltung stellt alle wesentlichen Funktionen zur Verfügung, um den höheren Schichten ein einfaches Kommunikationsmodell anzubieten. Das LMP erweitert die Funktionalität der Basisbandschicht, jedoch können höhere Schichten auch direkt auf das Basisband zugreifen (z.B. Audio und Control; vgl. Abbildung 3). Alle Aktivitäten zum Verbindungsaufbau und zur Verwaltung eines Pikonetzes werden über den Link Manager (der sich zumeist auf dem Bluetooth-Controller befindet und in Hardware bzw. in Form des LMP als Firmware implementiert ist; vgl. [Wollert2002]). geregelt und bleiben für die höheren Schichten transparent. Die folgenden Gruppen von Funktionen werden durch das LMP abgedeckt[18]:

- *Authentifikation, Paarbildung und Verschlüsselung:*
  Um Geräte authentifizieren zu können, verwendet Bluetooth ein Abfrage-Antwort-Schema (Challenge-Response Scheme) [Bluetooth2003c]. Sowohl der Master als auch der Slave können eine Überprüfung vornehmen. Das Gerät, dass eine Authentifikation durchführen möchte, sendet ein bestimmtes LMP-Paket der abgefragten Station eine Zufallszahl. Die abgefragte Station generiert mit der Zufallszahl, ihrer eigenen Bluetooth-Adresse und einem geheimen Schlüssel eine Antwort. Der Dienst zur Paarbildung (*Pairing*)von zwei Bluetooth-Geräten dient zum Aufbau eines Vertrauensverhältnis zwischen den Geräten.

18 Die Bluetooth-Spezifikation [Bluetooth2003c] nennt detailliert die folgenden Prozeduren und Regeln des Link Manager Protokolls: Generelle Antwort-Nachrichten (Generell Response Messages), Authentifikation, Paarbildung (Pairing), Wechseln des Verbindungsschlüssels (Change Link Key), Wechseln des aktuellen Links (Change Current Link Key), Verschlüsselung (Encryption), Anfrage des Uhr-Offset (Clock Offset Request), Slot-Offset Informationen (Slot Offset Information), Timing Informationen (Timing Accuracy Information Request), LMP-Version, Unterstütze Eigenschaften (Supported Features), Wechsel der Master-Slave Rolle (Switch of Master-Slave Role), Namen-Anfrage (Name Request), Trennen (Detach), Halten-Modus (Hold Mode), Schnüffelmodus (Sniff Mode), Parkmodus (Park Mode), Leistungskontrolle (Power Control), Dienstgüte (Quality of Service), SCO-Links, Kontrolle von Multi-Slot Paketen (Controll of Multi-slot Packets), Paging-Schema (Paging Scheme), Link-Überwachung (Link Supervision).

Die Geräte müssen, um eine Verbindung eingehen zu können, einen gemeinsamen *Verbindungsschlüssel (Link Key)* besitzen. Dieser Schlüssel kann dann weiter verwaltet (wenn Geräte einen gemeinsamen Verbindungsschlüssel besitzen kann dieser auch geändert werden), akzeptiert oder zurückgewiesen werden. Bei der Authentifikation und bei der Paarbildung werden Dienste des Basisbandes verwendet [Wollert2002]. Eine Verschlüsselung der übertragenen Daten kann optional verwendet werden, nicht alle Bluetooth-Geräte müssen eine Verschlüsselung unterstützen [Wollert2002]. Der Master stellt beim Slave eine Anfrage auf Verschlüsselung. Hierbei wird unterschieden, ob eine Punkt-zu-Punkt-Verbindung oder ein Broadcast verschlüsselt wird.

- *Synchronisierung:*
  Wie bereits beschrieben ist eine genaue Synchronisierung zwischen den Geräten sehr wichtig. Aus diesem Grund wird die interne Uhr eines Bluetooth-Gerätes nach jedem Empfang eines Datenpaketes justiert. Es können spezielle Pakete gesendet und empfangen werden, welche die Synchronisierung unterstützen.

- *Geräteeigenschaften:*
  Nicht nur Informationen über die Version des eingesetzten LMP, sondern auch Informationen über die von einem Gerät unterstützten Leistungsmerkmale können ausgetauscht werden. Die Bereitstellung von Informationen über den Link Manager sind notwendig, da mittlerweile bereits zwei verschiedene Versionen des Link Mangers vorhanden sind (Version 1.0 und Version 1.1) und weitere Versionen zu erwarten sind. Nicht alle Bluetooth-Geräte müssen alle Fähigkeiten unterstützen, die im Standard beschrieben sind. Aus diesem Grund können Geräte sich z.B. darüber verständigen, ob Multi-Slot-Pakete, Verschlüsselung, SCO-Verbindungen, verschiedene Pakete etc. gemeinsam genutzt werden können.

- *Leistungssteuerung, Dienstgüte:*
  Es sind verschiedene Parameter vorhanden, welche die Dienstgüte eines Bluetooth-Gerätes auf den unteren Schichten steuern. So kann z.B. die Latenz und Bandbreite einer Verbindung über das Abfrageintervall, die maximale Zeit, die zwischen zwei Übertragungen zwischen Master und Slave verstreicht, gesteuert werden. Ein Bluetooth-Gerät kann jederzeit die Stärke des empfangenen Signals messen. In Abhängigkeit von dieser Signalstärke kann das Gerät den Sender anweisen, die Sendeleistung zu erhöhen oder zu verringern. Ebenso kann anhand der Signalqualität gesteuert werden, ob DMx- oder DHx-Pakete eingesetzt werden (also 2/3 FEC zum Schutz oder kein Schutz).

- Verbindungssteuerung, Zustands- und Übertragungsüberwachung:
  Das LMP steuert die Aktivitäten einer Verbindung. Jedes Bluetooth-Gerät kann unabhängig von der Master- oder Slave-Rolle seine Verbindung zu jedem Zeitpunkt beenden oder ihren Betriebszustand wechseln. Die möglichen Betriebszustände sowie die Verbindungssteuerung werden im Folgenden näher erläutert. Abbildung 13 gibt dazu eine kompakte Darstellung.

Bei einer möglichen Sendeleistung von bis zu 100 mW können Bluetooth-Geräte Distanzen von bis zu 150m überbrücken (vgl. Tabelle 1 auf Seite 10). Soll diese Sendeleistung genutzt werden und dennoch ein stromsparender Betrieb möglich sein, kann ein Bluetooth-Gerät nicht permanent im aktiven Zustand sein. Bluetooth definiert aus diesem Grund verschiedene Modi zum Energiesparen. Abbildung 13 stellt die verschiedenen Betriebsmodi sowie die Übergänge zwischen den verschiedenen Zuständen dar.

Jedes Bluetooth-Gerät, das nicht an einem Pikonetz teilnimmt und nicht abgeschaltet ist, befindet sich im *Bereitschaftszustand (standby)*, das ist der Normalzustand. Dieser Zustand zeichnet sich durch eine sehr geringe Stromaufnahme aus, da lediglich die interne Uhr des Ge-

rätes weiter betrieben wird. vom Bereitschaftszustand kann auf zwei Arten in den Erkundigungszustand (inquiry) übergegangen werden: Entweder will das Gerät selbst ein Pikonetz aufbauen oder erkundet, ob bereits eine Kommunikation im Gange ist.

- *Ein Gerät möchte selbst ein Pikonetz gründen:* Das Gerät (oder der Nutzer des Gerätes) möchte nach anderen Geräten in Funkreichweite suchen. Zu diesem Zweck sendet das Gerät einen Inquiry Access Code (IAC) aus, der für alle Bluetooth-Geräte gleich ist. Dieser IAC wird über 32 standardisierte Trägerfrequenzen ausgesendet (vgl. [Wollert2002]).
- *Ein Gerät hört periodisch in das Medium hinein:* Geräte im Bereitschaftszustand können periodisch in den Erkundigungszustand wechseln, um nach IAC-Nachrichten auf den Trägerfrequenzen zu lauschen. Sobald ein Gerät eine solche Nachricht erkennt, sendet es selbst ein Paket aus, das die Geräteadresse und Zeitinformationen enthält, die vom Master für den Verbindungsaufbau benötigt werden. Das Gerät arbeitet von diesem Moment an als Slave.

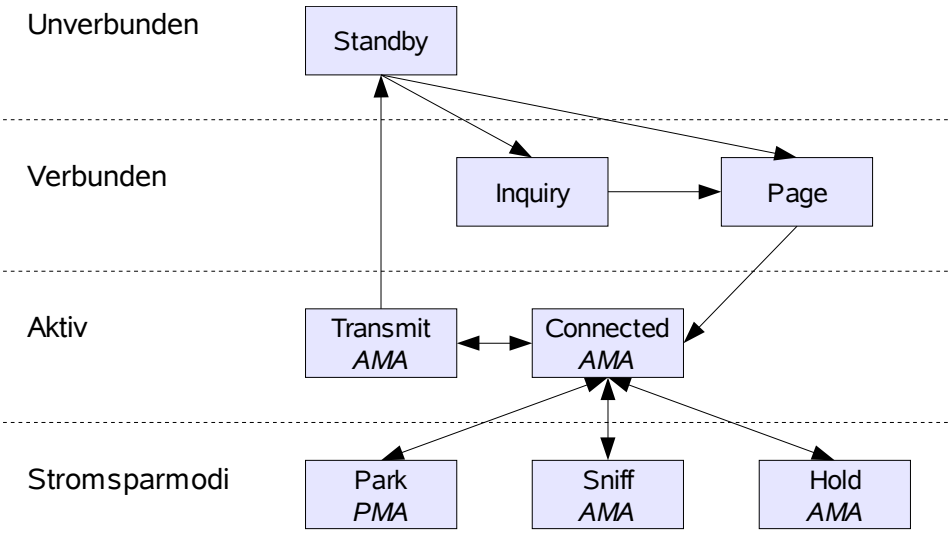

Abbildung 13: Betriebsmodi von Bluetooth-Geräten (nach [Wollert2002])

Falls die Erkundigung nach anderen Geräten erfolgreich war, wechselt ein Bluetooth-Gerät in den *Ausrufen-Zustand (paging)*[19]. Auch im Zustand *page* sind wiederum verschiedene Rollen festgelegt. Nachdem ein Gerät oder Geräte gefunden wurden kann ein Master damit beginnen Verbindungen zu den einzelnen Geräten herzustellen, also ein Pikonetz aufbauen. In Abhängigkeit von der Geräteadresse, welche der Master während der Erkundigung von den einzelnen Geräten empfangen hat, berechnet der Master eine spezielle Sprungfolge, damit alle Geräte individuell aufgerufen werden können. Die Slaves antworten auf diesen individuellen Aufruf und synchronisieren sich mit der Uhr des Masters. Der Master kann nun fortfahren, weitere Geräte aufzurufen und sie so dem Pikonetz hinzuzufügen. Sobald ein Gerät seine Sprungfolge der des Pikonetzes angepasst hat, wechselt es in den Zustand *verbunden (connected)*.

Der Verbunden-Zustand umfasst den *aktiven Zustand (active)*, sowie die drei Energiesparzustände *parken (park)*, *schnüffeln (sniff)* und *halten (hold)*.

- Im *aktiven Zustand* kann eine Folgestation am Pikonetz durch Zuhören, Übertragen oder Empfangen teilnehmen. Ein Master synchronisiert periodisch alle Folgestationen. Jedes aktive Gerät besitz eine 3 Bit große Adresse, die *Active Member Address (AMA)*. Ein Gerät

19 Es kann allerdings ein Weile dauern, bis die Erkundigung nach anderen Geräten erfolgreich ist, da Kollisionen auftreten können, insbesondere bei einer großen Anzahl an suchenden Geräten.

im aktiven Zustand kann durch eine *Abtrennungsprozedur (detach)* wieder in den Bereitschaftsmodus wechseln. Im aktiven Modus nutzen Master und Slave den Übertragungskanal gemeinsam. Aktive Slaves erwarten in den geradzahligen Slots Pakete vom Master.

Ist einmal eine Verbindung hergestellt kann ein Bluetooth-Gerät einen der drei Energiesparzustände einnehmen bzw. vom Master in einen der Zustände gesetzt werden:

- *Schnüffeln (sniff):* In diesem Zustand wird die Häufigkeit reduziert, mit der der Slave bereit ist, Pakete von einem Master zu empfangen. Normalerweise horcht eine Station in jedem Masterzyklus auf einen Aufruf vom Master. Im Sniff-Modus kann eingestellt werden, in welchen Intervallen der Masteraufruf ausgewertet wird. Das Zeitintervall nach dem der Slave periodisch zum Empfang bereit ist, wird zwischen Master und Slave „abgesprochen". Von den drei Energiesparmodi hat der Sniff-Mode noch die höchste Leistungsaufnahme.
- *Halten (hold):* In diesem Modus nimmt der Slave nicht mehr an ACL-Übertragungen teil, kann aber immer noch SCO-Pakete übertragen bzw. empfangen. Der Slave behält in diesem Modus seine AMA und die Synchronisationsinformationen. Der Halten-Modus wird durch den Master gesetzt.
- *Parken (park):* Im Parkmodus nimmt der Slave überhaupt nicht mehr aktiv an der Kommunikation im Pikonetz teil. Der Slave verliert in diesem Modus seine AMA, erhält aber eine neue Adresse, mit denen er vom Master weiterhin angesprochen werden kann, die Parked Member Adress (PM_ADDR), die 8 Bit groß ist. Bis zu 255 Slaves können somit durch einen Master geparkt werden. Der Parkzustand erlaubt sowohl die Reduzierung der Energieaufnahme beim Slave als auch die Aufnahme von mehr als sieben Slaves in ein Pikonetz. Geparkte Geräte sind immer noch über die Sprungsequenz im Pikonetz synchronisiert und wachen zu bestimmten Zeitpunkten auf, um sich erneut zu synchronisieren.

Weitere Details zur Verbindungsverwaltung und zum Link Manager finden sich in [Bluetooth2003c] und [Wollert2002].

7 Bluetooth-L2CAP

Das *Logical Link Control and Adaptation Layer Protocol (L2CAP)* ist das zweite Link Manager Protokoll oberhalb des Basisbandes. Das L2CAP stellt eine hostseitige Schnittstelle für die oberen Protokollschichten zur Verfügung, während das im vorhergehenden Abschnitt dargestellte Link Manager Protokoll (LMP) die Kommunikation auf Controller-Ebene abwickelt [Wollert2002]. L2CAP gehört zu den sogenannten *Host Layer Protokollen*, zu denen u.a. auch

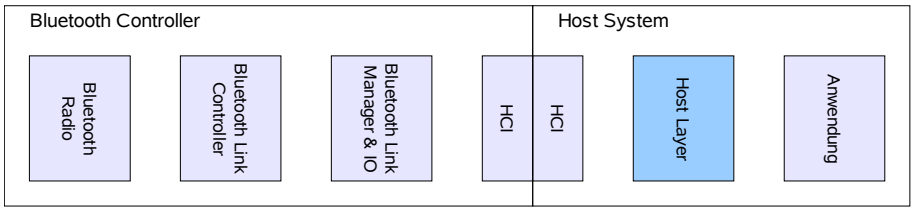

Abbildung 14: Beziehung des hardwarenahen Teils eines Bluetooth-Systems und der Host-Schnittstelle (nach [Wollert2002])

das Service Discovery Protokoll (SDP), RFCOMM und Telephony gehören. Abbildung 14 zeigt die Position des LMP und des L2CAP innerhalb eines Bluetooth-Systems dar. Das *Host*

Controller Interface (HCI) definiert dabei den Übergang zwischen den Controller-seitigen Protokollen und den Host-seitigen Protokollen[20].

Das L2CAP ist nur für ACL-Verbindungen verfügbar; SCO-Verbindungen werden in der aktuellen Spezifikation nicht durch diese Schicht unterstützt. Zur Kommunikation in der L2CAP-Schicht wird das Konzept der Kanäle verwendet. Es handelt sich hierbei um virtuelle (logische) Kanäle, die durch eine *Kanal-ID (Channel Identifier – CID)* identifiziert werden. L2CAP bietet drei verschiedene Typen virtueller Kanäle:

- *Signalisierung:* Diese Art von Kanal dient zum Austausch von Signalisierungsnachrichten zwischen L2CAP-Instanzen. Der Kanal wird zur Steuerung der Kommunikation verwendet und ist damit bevorzugt. Signalisierungskanäle werden immer durch die CID 1 gekennzeichnet.
- *Verbindungslos:* Verbindungslose Kanäle stellen eine gerichtete Punkt-zu-Multipunkt-Verbindung dar, die typischerweise für Rundrufe (Broadcasts) verwendet wird (z.B. von einem Master für Rundrufe zu den Slaves). Für die verbindungslose Kommunikaton verwendet das sendende Geräte einen Kanal mit einer dynamisch vergebenen CID ein, während alle empfangenden Geräte einen Kanal mit der CID 2 verwenden.
- Verbindungsorientiert: Verbindungsorientierte Kanäle dienen für Punkt-zu-Punkt-Verbindungen. Diese Verbindungen werden durch dynamisch zugewiesene CIDs realisiert. An jedem Kanalende wird eine eindeutige CID zugewiesen, um die Verbindung so eindeutig zu kennzeichnen. Verbindungsorientierte Kanäle verwenden eine CID größer oder gleich 64. Die CIDs 3 bis 63 sind reserviert [Bluetooth2003c].

Abbildung 15 zeigt schematisch die Verwendung von logischen Kanälen zwischen einem Master und zwei Slaves. Der Master hat einen Signalisierungskanal zu jedem der beiden Slaves, die CID ist an jedem Ende des Kanals 1. Daneben hat der Master zu den beiden Slaves einen verbindungslosen Kanal geöffnet. Die CID am Anfang des Kanals (also beim Master) wird dynamisch zugewiesen (d_{M0}), während die CID bei den Slaves 2 ist. Weiterhin hat der Master zu den Slaves jeweils einen verbindungsorientierten Kanal geöffnet. Die CIDs für diese Kanäle werden dynamisch zugewiesen (d_{M1},d_{M2}, d_{S1}, d_{S2}).

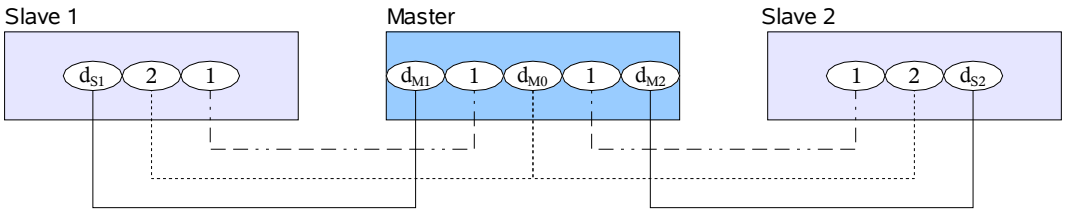

Abbildung 15: Virtuelle Kanäle zwischen einem Master und zwei Slaves

Die L2CAP-Schicht hat im wesentlichen drei Hauptfunktionen:

- Die L2CAP-Schicht kann Pakete mit einer Länge von bis zu 64 kByte von den oberen Schichten empfangen und zerlegt sie ggf. zur Verarbeitung in den unteren Schichten in kleinere Datenpakete (Segmente). Am anderen Ende werden die Segmente wieder zu Paketen zusammengestellt.

20 Die Definition der HCI-Funktionen nimmt einen großen Teil der Bluetooth-Spezifikation [Bluetooth2003c] ein, wird an dieser Stelle jedoch nicht weiter betrachtet. [Wollert2002] gibt eine sehr gute, kompakte Übersicht über die Funktionalität des HCI und der HCI-Ausprägungen.

- Sie verwaltet das Multiplexen und Demultiplexen von mehreren Paketquellen. Wird ein Paket wieder zusammengesetzt, bestimmt die L2CAP-Schicht an welches Protokoll der höheren Schichten das Paket weitergeleitet wird.
- L2CAP bietet Funktionen zur Aushandlung von Dienstgüte (Quality of Service) und Konfigurationsparametern. So kann u.a. die maximale Größe der Nutzdaten ausgehandelt werden, damit ein Gerät mit eingeschränkten Ressourcen nicht von zu großen Paketen überschwemmt wird. Mit dem Konfigurationsparameter für die Dienstgüte können die Eigenschaften der Übertragung festgelegt werden: best-effort (nach bestem bemühen, es werden keinerlei Garantien für die Datenübertragung ausgesprochen) oder garantiert.

8 Bluetooth-SDP

Das *Service Discovery Protokoll (SDP)* dient zur Erkennung von Dienstleistungen auf Geräten. Bluetooth-Geräte sollen in der Lage sein, in verschiedenen Umgebungen spontan mit anderen Geräten zusammenzuarbeiten. Daher ist es notwendig, zu wissen, welche Dienste von welchen Geräten in Funkreichweite zur Verfügung gestellt werden. Zur Erkennung von Diensten wurde in Bluetooth das SDP spezifiziert. Alle Geräte, die Dienste anbieten möchten, müssen einen *SDP-Server* verwenden, für alle anderen Geräte reicht ein *SDP-Client* aus.

SDP stellt in einer Service-Datenbank die auf einem Gerät zu verfügbaren Dienste bereit. Die Informationen über den Dienst, die der SDP-Server besitzt, werden in einem *Service Record* abgelegt. Der Service Record besteht aus einer Liste mit *Dienstattributen*, welche die Eigenschaften des Dienstes genauer beschreiben, und wird durch eine eindeutige ID identifiziert (32-bit Service Record Handle).

Eine Dienstanfrage läuft nach einem definierten Schema ab. Das Gerät, das einen Dienst finden möchte, nutzt den SDP-Client, um dem SDP-Server auf einem entfernten Gerät eine Anfrage zu stellen, welche Dienste das Gerät zur Verfügung stellt. Der SDP-Server des entfernten Gerätes hat in einer Datenbank die Service Records hinterlegt, die hierarchisch organisiert sind. Die Informationen aus den Service Records können dem anfragenden Gerät nun übermittelt werden. Mit den Informationen aus den übermittelten Service Records ist das anfragende Gerät nun in der Lage zu ermitteln, welche Dienste es benutzen kann und benutzten darf. Das Gerät kann nun eine Verbindung zu einem bestimmten Dienst aufbauen und muss hierzu nicht noch einmal das SDP verwenden. Einmal bereits gefundene Dienste können in einem Gerät auch zwischengespeichert werden.

Eine ausführlich Darstellung des SDP ist in [Wollert2002] und [Bluetooth2003c] zu finden.

9 Bluetooth-Profile

In den vorangegangenen Abschnitten wurden die grundlegenden Bluetooth-Protokolle kurz vorgestellt. Abbildung 3 auf Seite 6 gibt einen Überblick über die verschiedenen Bluetooth-Protokolle und den Bluetooth-Protokollstapel. Wie die Grafik zeigt, umfasst die Bluetooth-Kernspezifikation in ihrer derzeitigen Version 1.1 [Bluetooth2003c] bereits sehr viele Protokolle. Nicht alle Protokolle müssen aber zwingend in einem Bluetooth-Gerät implementiert sein. Bluetooth-Geräte sollen kostengünstig sein und müssen daher nicht jeden einzelnen Anwendungsfall abdecken können. Aus diesem Grund benennt die Bluetooth-Spezifikation in der Version 1.1 [Bluetooth2003d] 13 verschiedene *Anwendungsprofile*, die zusätzlich zu den Protokollen spezifiziert wurden. Jedes dieser Anwendungsprofile stellt im Prinzip für jeden Anwendungsfall einen anderen Protokollstapel bereit. Dies führt auf der einen Seite zu einer hohen Komplexität, ermöglicht auf der anderen Seite aber, dass trotz der hohen Anzahl an

verfügbaren Parametern in Bluetooth die Interoperabilität zwischen Geräten verschiedener Hersteller gegeben ist.

Profile stellen Standardlösungen für eine bestimmte Nutzungsart dar. Ein Profil beschreibt, welche Protokolle und welche Parameter gesetzt werden müssen, damit ein Bluetooth-Gerät ein bestimmtes Nutzungsmodell erfüllt. Auf Seite 8 wurde bereits kurz der Zusammenhand zwischen Profilen und Protokollen dargestellt. Dies sei an dieser Stelle wiederholt.

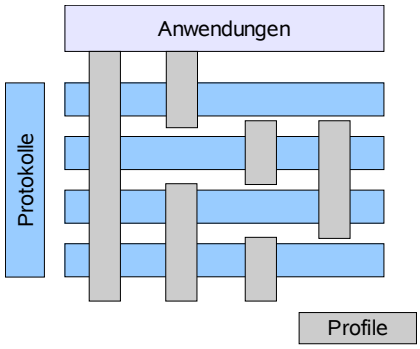

Abbildung 16: Bluetooth-Profile

Profile beschreiben vertikale Schnitte durch den Bluetooth-Protokollstapel, die Protokolle stellen horizontale Schichten dar. Innerhalb der Profile werden die notwendigen und optionalen Funktionen der Schichten definiert. Mit diesen standardisierten Profilen kann die Interoperabilität zwischen verschiedenen Geräten sicher gestellt werden.

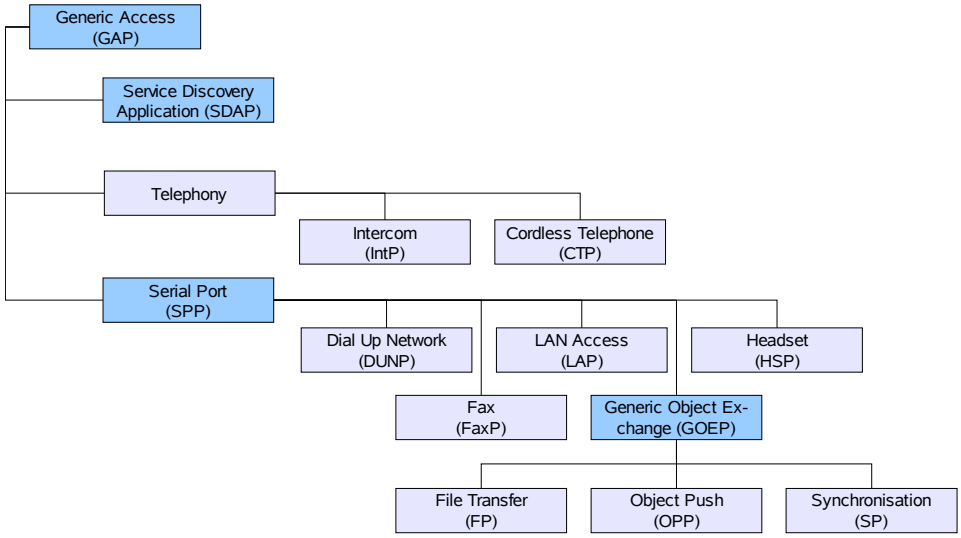

Abbildung 17:Hierachie der Bluetooth-Profile (nach [Wollert2002])

Die Bluetooth-Spezifikation [Bluetooth2003d] beschreibt zur Zeit 13 verschiedene (Basis) Profile – Generic Access, Service Discovery, Serial Port, Generic Object Exchange, Cordless Telephony, Intercom, Headset, Dial-Up Networking, Fax, LAN Access, Object Push, File Transfer, Synchronisation - von denen eine Auswahl in diesem Kapitel kurz beschrieben werden soll[21]. Neben diesen bereits definierten Protokollen arbeitet die Bluetooth-SIG bereits an weiteren Profilen und es ist zu erwarten, dass die Anzahl an definierten Profilen in den nächsten Versionen der Bluetooth-Spezifikation wächst.

21 Die Version 1.1 der Bluetooth-Spezifikation [Bluetooth2003d] beschreibt die 13 bisher definierten Profile auf mehr als 450 Seiten!

Die vier grundlegenden Profile der Bluetooth-Spezifikation sind

- das Generic Access Profile,
- das Service Discovery Application Profile,
- das Serial Port Profile und
- das Generic Object Exchange Profile.

Die Abbildungen 17 und 18 zeigen die Hierarchie bzw. die Abhängigkeiten der bisher definierten Profile.

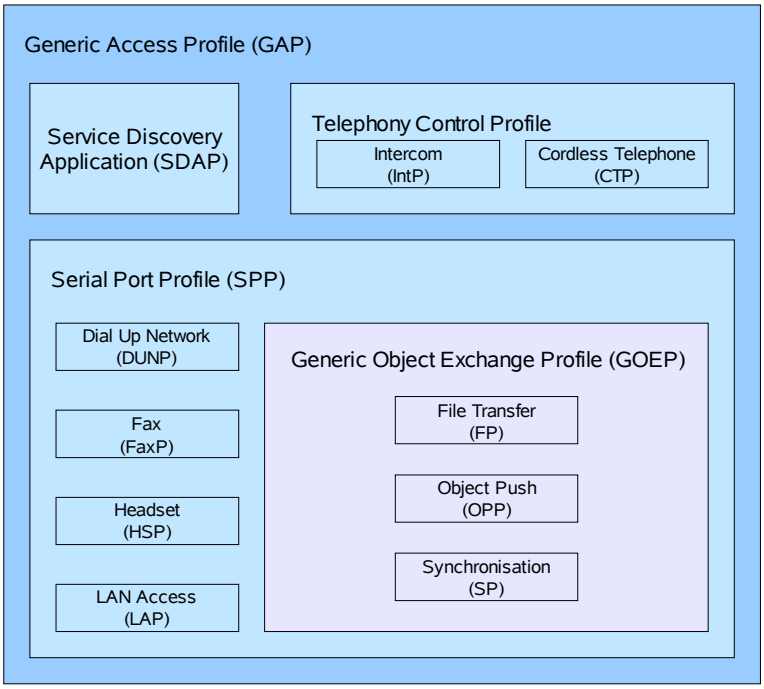

Abbildung 18: Abhängigkeiten der Bluetooth-Profile (nach [Wollert2002])

Grundlegend ist das *Generic Access Profile (GAP) – Profil für den allgemeinen Zugriff*. Das Profil ist keine wirkliche Anwendung, sondern bildet die Basis, auf der die eigentlichen Anwendungen erstellt werden können. Es beschreibt alle wesentlichen Funktionen, die ein Bluetooth-Gerät auf der untersten Ebene erfüllen muss. Dazu gehören z.B. die Funktionen zum Verbindungsaufbau und zur Verbindungsverwaltung (Link Establishment, Channel Establishment, Connection Establishment), die unterstützten Betriebsmodi (Discoverable Mode, Connectability Mode, Pairing Mode) und die Sicherheit einer Verbindung (Authentification, Security Mode).

Ein weiteres wichtiges Profil, das ebenfalls sehr allgemein ist, ist das *Service Discovery Application Profile (SDAP) – Diensterkennungsprofil*. Das SDAP definiert die Zugriffsschnittstelle für das Service Discovery Protocol (SDP), mit dem Geräte die von anderen Geräten angebotenen Dienste erkennen bzw. abfragen können. SDP baut auf dem GAP auf.

Alle Bluetooth-Geräte müssen diese beiden Profile implementieren [Tanenbaum2003].

Das *Serial Port Profile (SPP) – Profil für die serielle Schnittstelle* – wird von den meisten anderen Bluetooth-Profilen genutzt. Eine Ausnahme bilden die Telephony Control Profile. Es kommt quasi immer dann zum Einsatz, wenn Bluetooth als Kabelersatz verwendet wird bzw.

wenn eine serielle Datenverbindung genutzt werden soll. Das SPP baut auf dem GAP auf und nutzt das RFCOMM-Protokoll[22].

Das *Generic Object Exchange Profile (GOEP) – Allgemeines Objektaustauschprofil* – definiert die Grundlegenden Funktionen, die zum „Austausch komplexer Objekte“ [Wollert2002] notwendig sind. Es definiert eine Client-Server-Beziehung zum Austausch von Daten. Das GOEP bildet wie das SPP die Grundlage für weitere Profile, wie z.B. dem File Transfer Profile (FP), dem Object Push Profile (OPP) und dem Synchronisation Profile (SP).

Tabelle 4 fasst die Profile der Bluetooth-Spezifikation in der Version 1.1 mit einer kurzen Beschreibung zusammen.

Profil-Name	**Beschreibung**	**Beispielgeräte (Auswahl)**
Generic Access Profile (GAP)	Verbindungsaufbau und -steuerung	-
Service Discovery Application Profile (SDAP)	Anbieten und Erkennen von Diensten	-
Serial Port Profile (SPP)	Ersatz serieller Datenverbindungen	PC, Notebook, PDA, Handy, Drucker, Modem
Generic Object Exchange Profile (GOEP)	Allgemeiner Objektaustausch, definiert eine Client/Server-Beziehung	-
Dial Up Network Profile (DUNP)	Einwahlzugang über Modem oder Handy	Analog-Modem, ISDN-Modem, PC, Notebook, PDA, Handy
Fax Profile (FaxP)	Steuerung von Faxdiensten zwischen Geräten	Analog-Modem, ISDN-Modem, PC, Faxgerät
Headet Profile (HSP)	Steuerung von Headsets/Freisprecheinrichtungen	Headset, Handy, PC, PDA
LAN Access Profile (LAP)	LAN-Zugriff üder das Point-to-Point-Protokoll (PPP)	ISDN-,DSL-,LAN-Access-Point
Cordless Telephony Profile (CTP)	Unterstützung für schnurlose Telefondienste (Funktionalität einer DECT[23]-Umgebung)	Handy, Basisstation
Intercom Profile (IntP)	Kommunikation zwischen Handsets (Terminals, eine Art Walkie-Talkie	Handy, PDA, Gegensprechanlage
File Transfer Profile (FP)	Übertragung von Dateien zwischen Bluetooth-Geräten	Handy, PC, PDA, Notebook
Object Push Profile (OPP)	Übertragung von Datenobjekten (z.B. Austausch von Visitenkarten – vCard – und Terminen – vCal)	PC, PDA. Notebook, Handy, Drucker, Scanner, Faxgerät
Synchronisation (SP)	Synchronisation von Geräten auf Basis von typischen PIM-Daten (Personal Information Manager)	PC, Notebook, Handy, PDA

Tabelle 4: Bluetooth-Profile (nach [Tanenbaum2003] und [Zivadinovic2003a])

22 RFCOMM ist ein Kabelersatzprotokoll (oft auch als RFCOMM Cable Replacement Protocol bezeichnet), das nicht näher in diesem Text beschrieben wird. Der Leser findet eine Beschreibung u.a. in [Wollert2002] und [Bluetooth2003b].

23 DECT- Digital Enhanced Cordless Communications. Ein von der ETSI (European Telecommunications Standards Institute) standardisiertes digitales Mobilfunknetz, das als Ersatz für analoge schnurlose Telefonsysteme dient.

10 Quellenverzeichnis

[Ahlers2001] Ahlers, E.: Leinen los! Bluetooth kommt - langsam, aber sicher, c't 9/2001, S. 100 ff.

[Bluetooth2003a] The Official Bluetooth Wireless Info Site, http://www.bluetooth.com (zuletzt eingesehen am 04.11.2003)

[Bluetooth2003b] The Official Bluetooth Membership Site, http://www.bluetooth.com (zuletzt eingesehen am 04.11.2003)

[Bluetooth2003c] Specification of the Bluetooth System, Band 1, Core, Version 1.1, Bluetooth Special Interest Group, http://www.bluetooth.org

[Bluetooth2003d] Specification of the Bluetooth System, Band 2, Profiles, Version 1.1, Bluetooth Special Interest Group, http://www.bluetooth.org

[ct2003a] c't – Magazin für Computertechnik, Bluetooth-Portal,Heise Verlag, http://www.heisemobil.de/bluetooth (zuletzt eingesehen am 04.11.2003)

[ct2003b] c't – Magazin für Computertechnik, Bluetooth-Datenbank, Heise Verlag, http://www.bluetooth-db.de (zuletzt eingesehen am 04.11.2003)

[Haartsen1998] Haartsen, J.: Bluetooth – the universal radio interface for ad hoc, wireless connectivity, Ericsson Review, Nr. 3, 1998, http://www.ericsson.com/about/publications/review/1998_03/14.shtml (zuletzt eingesehen am 04.11.2003)

[Haartsen2000] Haartsen, J.: The Bluetooth Radio System, IEEE Personal Commun. Magazine, Bd. 7, S. 28-36, Feb. 2000. Auch online verfügbar unter http://ieeexplore.ieee.org/xpl/abs_free.jsp?arNumber=824570 (kostenpflichtig!, zuletzt eingesehen am 04.11.2003)

[Hole2003] Kjell Jørgen Hole: Wireless Communication Courses: WiFi and Bluetooth Course, http://www.kjhole.com/Standards/Intro.html (zuletzt eingesehen am 04.11.2003)

[Holtkamp2001] Holtkamp, H.: Einführung in TCP/IP, http://www.rvs.uni-bielefeld.de/~heiko (zuletzt eingesehen am 04.11.2003)

[IEEE2002] IEEE P802.15. The Working Group for WPAN, Institute of Electrical and Electronics Engineers, http://www.ieee802.org/15

[Microsoft2001] Microsoft: Microsoft Encarta Enzyklopädie 2001

[Palo2003] Palo Wireless – Bluetooth Resource Center. http://www.palowireless.com/bluetooth (zuletzt eingesehen am 04.11.2003)

[Siep2001] Siep, T.: Examining the Changes with Bluetooth Core Specification Version 1.1, Texas Instruments, http://grouper.ieee.org/groups/802/15/pub/2001/Mar01/Misc/Siep_Changes_to_Bluetooth_r1.PDF (zuletzt eingesehen am 04.11.2003)

[Tanenbaum2003] Tanenbaum, A.S.: Computer Networks, 4th. ed., Prentice Hall, Pearson Education, 2003

[ThiBeu2000] Thiele, L., Beutel, J.: Eingebettete Systeme – Bluetooth, Computer Engineering and Networks Laboratory, Swiss Federal Institute of Technology Zurich, 2000, http://www.tik.ee.ethz.ch/tik/education/lectures/ES/WS00/Bluetooth.pdf (zuletzt eingesehen am 04.11.2003)

[Wollert2002] Wollert, J.F.: Das Bluetooth-Handbuch, Franzis, 2002

[Zivadinovic2001] Zivadinovic, D.: Bluetooth löst Kabelsalat ab – Erste Geräte im Praxiseinsatz, c't 9/2001, Seite 102ff.

[Zivadinovic2003a] Zivadinovic, D.: Firstclass Luftverkehr – Bluetooth setzt zum Boom an, c't 23/2003, S. 142ff.

[Zivadinovic2003b] Zivadinovic, D.: Privatfunk – Bluetooth als Netzwerk- und Schnittstellenmodul, Heise Mobil, http://www.heise.de/mobil/artikel/2003/02/26/privatfunk (zuletzt eingesehen am 04.11.2003)

[Zivadinovic2003b] Zivadinovic, D.: Funk-Schwarm – Bluetooth-USB-Adapter im Test, c't 23/2003, S. 146ff.

[ZivSap2002] Zivadinovic, D., Sappok, S.: Profile in Blau - Die Protokolle des Kurzstrecken-Funks Bluetooth, c't 18/2002, Seite 148 ff.