

Bluetooth

Ein Standard für die drahtlose Kommunikation im Nahbereich

Januar 2007

Bluetooth ist eine Technologie für die Verbindung von Mobiltelefonen, Headsets, Freisprecheinrichtungen im Fahrzeug, PDA, GPS-Empfängern, Datenerfassungsgeräten, Access Points, Telefonanlagen miteinander oder mit dem PC. Ein wesentliches Ziel von Bluetooth ist eine automatische Verbindungsaufnahme zwischen verschiedenen Geräten bzw. Anwendungen. Damit ergeben sich neue Merkmale wie z.B. das Starten von Anwendungen wenn die entsprechenden Geräte in Reichweite. Beispiel ist die automatische Verbindung eines Mobiltelefons mit der Freisprecheinrichtung im Fahrzeug.

Geschichte und Entwicklung

Bluetooth geht auf Entwicklungen bei der Firma Ericsson zurück. Anfang 1998 wurde die Bluetooth Special Interest Group (SIG) von Ericsson, IBM, Intel, Nokia und Toshiba gegründet. Im Januar 2007 hat die Bluetooth SIG über 7000 Mitglieder.

Im Februar 2001 wurde die Spezifikation 1.1 vorgestellt. Im November 2003 die Version 1.2, im Oktober 2004 die Version 2.0+EDR (Enhanced Data Rate) und im Oktober 2006 die Version 2.1+EDR.

Architektur eines Bluetooth Systems mit Stack und Anwendungsprofilen

Die Entwicklung von Bluetooth erfolgte aus der Perspektive des Anwenders. Bluetooth als Kommunikationsprotokoll definiert für bestimmte Use Cases (Anwenderszenarios) Prozeduren und einen minimalen Leistungsumfang für diese Anwendungen. Diese Definition wird bei Bluetooth als Profil bezeichnet. Beispiele für Profile sind u.a. Headset, Hands-Free, Dial-Up Network, Fax, Object Push, File Transfer, Phone Book Access, Basic Printing, Basic Imaging, Audio/Video und Synchronisation. Profile definieren die Regeln für die Verbindungsaufnahme, die Art der Datenübertragung, Sicherheitsmechanismen, die beteiligten Protokollayer usw. Profile sind einer der Schlüssel für die Interoperabilität bei Bluetooth.

Bei Bluetooth unterscheidet man zwischen dem Bluetooth Host und dem Bluetooth Radio. Bluetooth Host ist die CPU bzw. der DSP, auf dem der Upper Layer Protokollstack (s. u.) läuft. Der Bluetooth Host kommuniziert über das HCI (Host Controller Interface) mit dem Bluetooth Radio. Das Bluetooth Radio beinhaltet den Lower Layer Stack (Link Manager usw.) sowie die eigentliche Hardware für das Senden und Empfangen.

Der Bluetooth Host ist die CPU, die auch die eigentliche Applikation (Freisprecheinrichtung, GPS Empfänger, Access Point) abarbeitet. Von der Host CPU werden alle Daten über das HCI an das Bluetooth Radio übertragen.

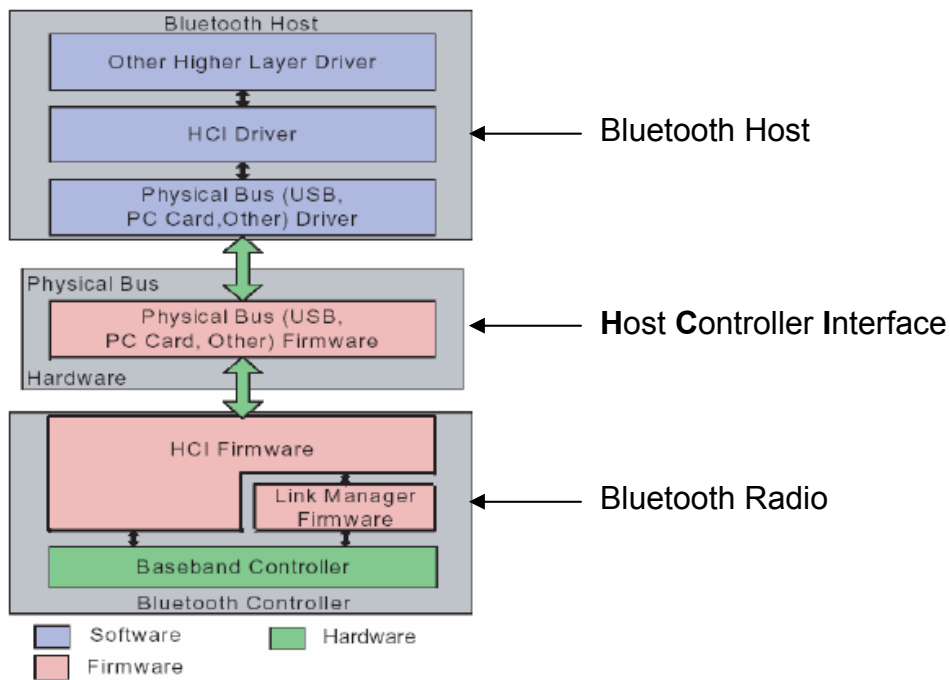
Dem HCI kommt eine besondere Bedeutung zu. Aktuelle sind folgende HCI gebräuchlich:

- H2 USB
- H4 UART (Rx/Tx, CTS, RTS, GND)
- H5 Three-Wire URT (Rx, Tx, GND)

H2 wird fast ausschließlich für USB Adapter eingesetzt. H4 und H5 finden in praktisch allen anderen Geräten inkl. PC und PDA Anwendung. Über das HCI werden alle Daten zum Bluetooth Radio übertragen. Sprache wird PCM kodiert und i. D. R. direkt auf das Basisband des Radios übertragen.

Das Bluetooth Radio wird über das HCI mit Kommandos eingestellt bzw. erhält über das HCI die Daten. Das Bluetooth Radio bestätigt Kommandos mit HCI Events.

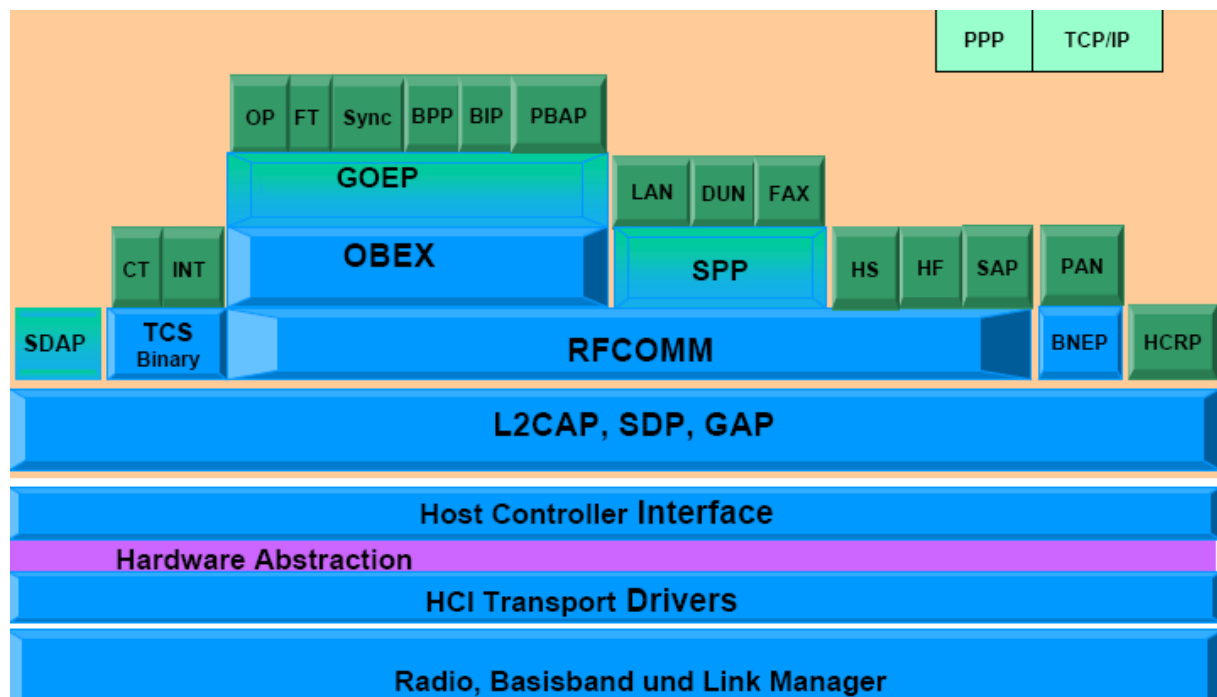
Bild 1 Architektur eines Bluetooth Systems



Der Architektur von Bild 1 entsprechen die überwiegende Anzahl aller Bluetooth Designs. Vereinzelt gibt es auch Anwendungen bei denen der Stack auf der Basisband CPU des Bluetooth Radio läuft. Dabei ist eine leichte Erweiterbarkeit der Anwendung bzw. die Erweiterung um neue Profile nicht gegeben.

Der Bluetooth Stack beinhaltet die für die Übertragung notwendigen Protokolle sowie die Profile. Die Profile sind die Schnittstelle zur Anwendung.

Bild 2 Stack mit den Anwendungsprofilen (ohne Audio/Video) und HCI



Beschreibung der einzelnen Funktionsblöcke

L2CAP	Logical Link Control and Adaptation Protocol für Client Anwendungen
SDP	Service Discovery Protocol mit einem Verzeichnisdienst.
TCS Binary	Telephony Control & Signaling
RFCOMM	Emulation einer seriellen Schnittstelle über Bluetooth
BNEP	Bluetooth Network Encapsulation Profile
OBEX	Object Exchange , ein Filetransfer-Protokoll
GAP	Generic Access Profile , definiert grundlegende Bluetooth Merkmale
SDAP	Service Discovery Application Profile
GOEP	Generic Object Exchange Profile
SPP	Serial Port Profile (für serielle Bluetooth Datenübertragung)
Management Entity	Bietet Dienste und Services u. a. für: <ul style="list-style-type: none">- Suchen von anderen Geräten und den Verbindungsaufbau- Link Key und Security Management

Das **Basisband** ist für das Timing, die Struktur der Pakete (Frames), sowie die Steuerung der einzelnen Links zuständig. Das Basisband Protokoll (Link Manager Protokoll, LMP) unterstützt Circuit- und Packet-Switching.

Der Verbindungsaufbau wird vom **LMP** übernommen. Das LMP erkennt und kommuniziert mit anderen Geräten im Empfangsbereich und verwaltet die Zustände der einzelnen Verbindungen. Der Link Manager löst auch Konflikte zwischen den Slaves auf und verwaltet die Power States (Park, Sniff und Hold). In diesem Layer werden auch Adressanfragen beantwortet, die Art der Datenübertragung (Daten, Sprache usw.) ermittelt bzw. festgelegt. Die Verwaltung der einzelnen Devices ist über einen Identifier (Namen des Bluetooth Gerätes) möglich.

Das **Logical Link Control Adaption Protocol (L2CAP)** adaptiert die oberen Protokolle an das Basisband bzw. Link Manager. Dabei werden den höheren Protokollen verbindungsorientierte und verbindungslose Services zur Verfügung gestellt. Segmentation, Re-assembly, Multiplexing und Zusammenfassung der Teilnehmer in Gruppen wird unterstützt.

RFCOMM ist die Umsetzung einer V.24 Schnittstelle über eine Funkstrecke gemäß ETSI GSM 7.10. Damit können serielle Daten vorhandener Anwendungen wie z.B. Modems übertragen werden. Die V.24 wird dabei vollständig (inkl. Flusskontrolle) emuliert.

Für IP basierende Übertragungen wird das **Bluetooth Network Encapsulation Protocol (BNEP)** bzw. das **Personal Area Network (PAN)** Profil eingesetzt.

Eine Anwendung oberhalb RFCOMM ist das bereits bei IrDA verwendete **OBEX (Object Exchange)** mit den Profilen Object Push, File Transfer und Synchronisation mittels **Infrared Mobile Communication (IrMC)**, einem ursprünglich bereits bei IrDA definierten Synchronisationsstandard. Für Object Push wird als Anwendung sehr häufig **vCard/vCalender** eingesetzt. Weitere Profile die OBEX als Grundlage haben sind Basic Printing, Basic Imaging und das Phone Book Access Profil. OBEX eignet sich sehr gut für die Übertragung von Files.

Die **Sprache** (z. B. im Headset und Mobiltelefon) wird direkt an das Basisband übergeben.

Sicherheit bei Bluetooth

Bluetooth unterstützt eine sichere Kommunikation mit verschiedenen Verfahren.

Authentication ist der Vorgang zur Ermittlung des anderen Gerätes (nicht Nutzers!) durch Abfrage der Bluetooth Adresse. Authentication wird durch gespeicherten Link Key bzw. durch die Eingabe einer PIN mit 4 Ziffern erreicht. Bluetooth 2.1 verwendet eine PIN mit 16 Stellen (Ziffern).

Authorization ist die Abfrage ob ein Device Zugriff auf Services eines anderen Devices hat. Mit Encryption werden die zu sendenden Daten mit einem Schlüssel von 56 Bit bis 128 Bit encrypted. Jeder Link im Piconet verwendet einen anderen Schlüssel (wird nie veröffentlicht).

Die Verschlüsselung und das verwendete Frequenzsprungverfahren gewährleisten eine sehr sichere Übertragung bei Bluetooth. Die bekannten Maßnahmen um eine Bluetooth Kommunikation aufzuzeichnen basieren auf einer bekannten PIN, bekannten MAC Adressen, dem Zeitpunkt zu dem ein Inquiry gestartet wurde bzw. verwendeten ein entsprechenden Messmittel- und Rechneraufwand.

Um die Sicherheit zu erhöhen, wird bei sensiblen Anwendungen ein Wechsel der PIN (und damit des Link Key) empfohlen. Zukünftige Bluetooth Versionen vereinfachen das.

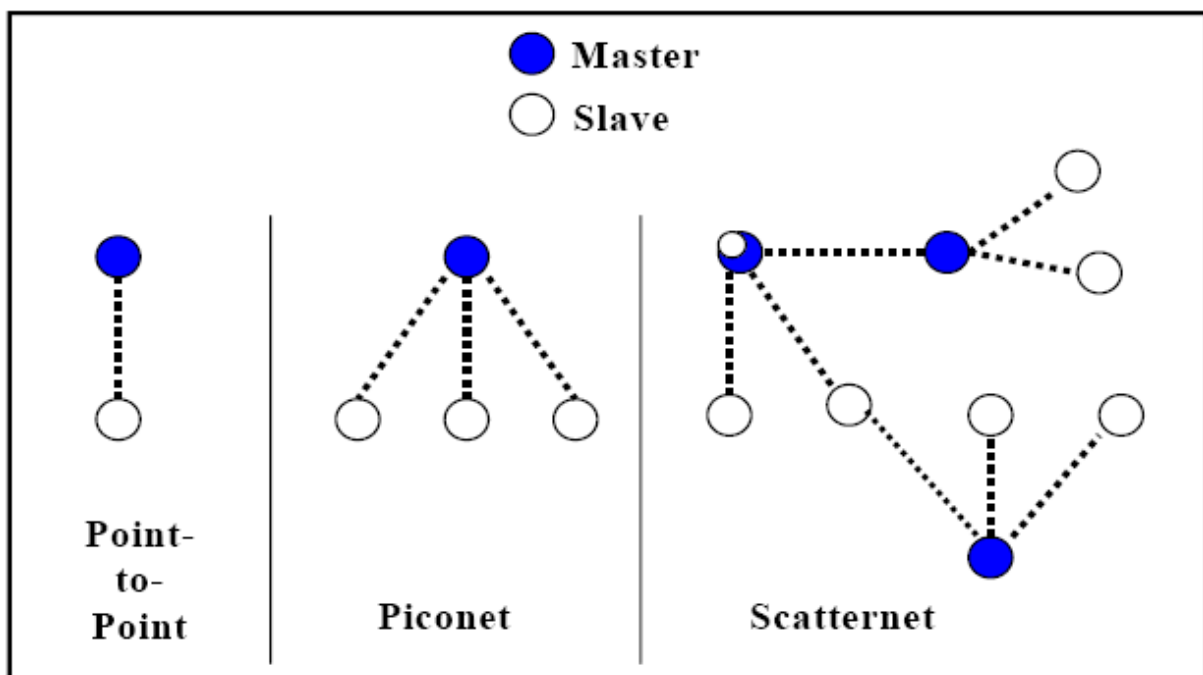
Die in **Bild 2** grün markierten Blöcke zeigen die Anwendungsprofile. Weitere Profile (u.a. Audio/Video) sind bei der Bluetooth SIG in Entwicklung.

HCRP	H ardcopy C able R eplacement P rofile für das Drucken.
PAN	P ersonal A rea N etwork Profile für IP basierende Anwendungen.
SAP	S IM A ccess P rofile für das telefonieren mit den Daten der SIM Karte.
HF	H ands-Free Profile für das Freisprechen im Fahrzeug.
HS	H eadset Profile für Headsets.
FAX	T elefax Profile für die Telefaxübertragung.
DUN	D ial-Up N etwork Profile für Modemverbindungen (z. B. zum Mobiltelefon).
LAN	L AN Access Profile (Nicht mehr empfohlen. Wurde abgekündigt)
PBAP	P hone B ook A ccess P rofile für Zugriff auf Telefonbuchdaten im Mobiltelefon.
BIP	B asic I maging P rofile für die Übertragung von Bildern auf andere Geräte.
BPP	B asic P rinting P rofile für das Drucken.
SYNC	S ynchronisation für die Synchronisation zwischen Mobiltelefon und PC.
FT	F ile T ransfer für eine Filetransferanwendung (z. B. Mobiltelefon und PC).
OP	O bject P ush für die Datenübertragung mittels PUSH Funktion.
INT	I ntercom Profile für eine direkte Sprachverbindung (nicht verbreitet).
CT	C ordless T elephony Profile für drahtlose Telefone (nicht verbreitet).

Netzwerk, Verbindungsaufbau und Datenübertragung

Es sind Punkt-zu-Punkt und Punkt-zu-Multipunkt Verbindungen möglich. Dabei bilden bis zu maximal 8 Bluetooth Kommunikationsgeräte ein sog. **Piconet**. Wenn mehrere Piconets untereinander kommunizieren bilden diese ein **Scatternet** (praktisch von keinem kommerziell erhältlichen Gerät unterstützt).

Bild 3 Mögliche Bluetooth Netze (vereinfacht).



Der Initiator der ersten Verbindung übernimmt dabei die Rolle eines Masters für die Kommunikation, d.h. er verwaltet die Adressen (48 Bit, ähnlich IEEE 802) und versendet Signale für die Synchronisation. Die anderen Geräte arbeiten als Slave. Slaves können nicht direkt miteinander kommunizieren.

Wenn die Geräte eingeschaltet sind, befinden diese sich im **Standby** Mode. Stationen in diesem State sind nicht im Piconet verbunden.

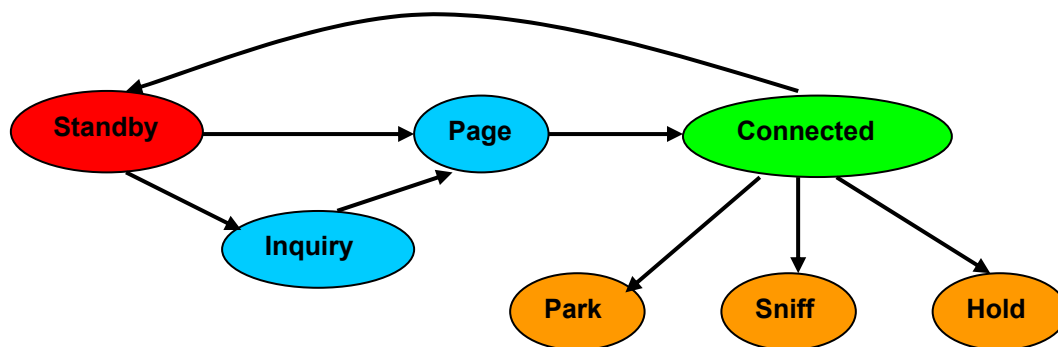
Die Station, welche als erste eine Verbindung aufbaut, ist der **Master**. Die andere/n Station/en ist/ sind **Slave/s**. Der Verbindungsaufbau erfolgt durch das Senden einer **Page** (MAC Adresse ist bekannt, es bestand bereits eine Verbindung mit diesem Gerät) Nachricht oder einer **Inquiry** (MAC Adresse ist nicht bekannt und wird erst ermittelt) Nachricht an alle Stationen. Die Verzögerung bevor der Master einen Slave erreicht beträgt ca. 2-5 Sekunden. Wenn sich Geräte gefunden haben kann eine Verbindung aufgebaut werden.

Danach befindet sich die Station im State **Connected**.

Nach Abschluss der Datenübertragung können der Empfänger bzw. beide Geräte in den **Standby** Mode gehen.

Vor dem Aufbau einer Verbindung wird eine PIN abgefragt. Aus dieser PIN, einer Zufallszahl und der MAC Adresse wird ein Link Key berechnet. Dieser Link Key wird in beiden Geräten abgespeichert (Pairing). Dieser Key wird dann für jede weitere Verbindung zwischen diesen beiden Geräten verwendet. Eine PIN Eingabe ist dafür nicht mehr notwendig.

Bild 4 Übergänge zwischen den einzelnen Bluetooth States.



Um Strom zu sparen, können Stationen die nicht senden oder empfangen, in verschiedene Zustände (Power Safe Modes) wechseln. Diese Zustände sind in der Reihenfolge des Energieverbrauchs (abfallend):

Sniff: Der Empfänger wird in einstellbaren Intervallen aktiv gesetzt.

Hold: Slaves können in diesen Zustand - vom Master oder sich selbst initiiert - wechseln.

Park: Die Stationen nehmen nicht an der Kommunikation teil. Synchronisationsnachrichten, und Broadcast Nachrichten werden aber erkannt und verarbeitet.

Die Power Safe Modes sind nicht in jedem Bluetooth Radio mit allen Optionen integriert.

Ein Bluetooth Gerät unterstützt einen asynchronen Link (ACL, Asynchronous Connection Less Link) und bis zu 3 synchrone Links (SCO Link, Synchronous Connection Oriented Link). Die Datenübertragung verwendet Time Division Duplex (TDD) für den Full-Duplex Betrieb. Die Übertragung der Daten auf den Links erfolgt in Paketen. Sprache hatte immer Priorität, d. h. wenn in der Anwendung Sprache unterstützt werden soll, wird Bluetooth Bandbreite „a priori“ für die Sprachübertragung vergeben. Das geht u. U. zu Lasten der Bandbreite für ACL Daten.

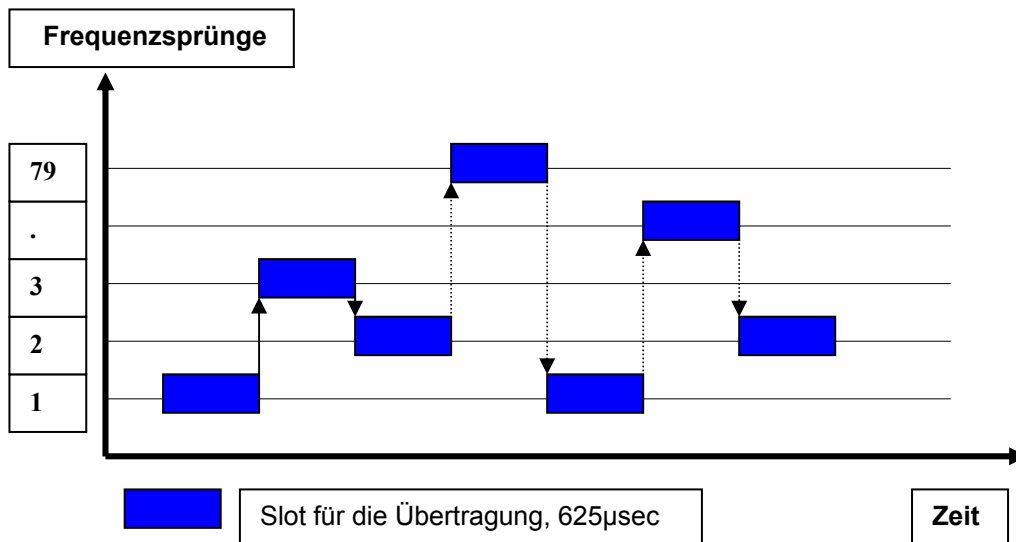
ACL (Daten) Übertragung: Dabei stehen bis zu 7 ACL Links pro Piconet zur Verfügung. Die Daten werden bei Bluetooth 1.2 mit maximal 432,6 KBit/s in beide Richtungen, oder mit 721 KBit/s (57,6 Kbit/s im Rückkanal) in eine Richtung übertragen. Die Übertragung ist symmetrisch oder asymmetrisch. Die genannten Werte sind die theoretischen Bestwerte.

SCO (Sprach) Übertragung: Hier stehen 3 Kanäle pro Piconet zur Verfügung. Die Übertragung der PCM kodierten Sprache erfolgt mit Continously Variable Slop Delta (CVSD) Modulation.

Physical Layer und Hardware

Der Bluetooth Sender arbeitet bei **2,402 bis 2,48 GHz** mit einem **Frequenzsprungverfahren** (Frequency Hopping) und verwendet maximal **1600 Frequenzsprünge** (Hops) in der Sekunde. Dabei wird zwischen **79 Frequenzstufen im 1 MHz Abstand** hin- und her gesprungen. Mit diesem Verfahren werden Interferenzen und Störungen (Mikrowellengeräte arbeiten bei ca. 2,45 GHz) vermieden. Bei jedem Sprung wird ein Datenpaket übertragen. Für den Empfang von Daten muss der Empfänger mit dem Sender synchronisiert sein, d.h. er muss die gleiche Sprungfolge verwenden.

Bild 5 Frequenzsprungverfahren bei Bluetooth.



Ein Datenpaket ist mindestens ein Slot, drei oder maximal 5 Slots lang. Um die Übertragung auf der Funkstrecke unempfindlicher gegen Störungen zu machen, kann Forward Error Correction (FEC) verwendet werden. Pakete werden beim Empfang mit ACK quittiert. Im Fehlerfall werden Pakete wiederholt.

Reichweite/Sendeleistung

Bluetooth Radios werden nach deren Sendeleistung eingeteilt (s. Tabelle 1).

Klasse	maximale Ausgangsleistung	minimale Ausgangsleistung	Bereich der Ausgangsleistung	Reichweite
1	100 mW (20 dBm)	1 mW (0 dBm)	4 – 20 dBm opt. -30 – 0 dBm	max. 100 Meter
2	2,5 mW (4 dBm)	0,25 mW	opt. -30 – 0 dBm	max. 20 Meter
3	1 mW (0 dBm)	---	opt. -30 – 0 dBm	max. 10 Meter

Tabelle 1 Sendeleistung und ungefähre Reichweite

Bluetooth und Störungen

Bluetooth ab Version 1.2 ist sehr robust gegen Störungen. Ab 1.2 wird Adaptive Frequency Hopping verwendet. Damit werden belegte bzw. gestörte Funkkanäle vom Frequenzsprungverfahren ausgeschlossen.

Datenübertragungsrate

Die erzielten Übertragungsraten sind sehr stark von den verwendeten Paketen und der Qualität der Übertragungsstrecke abhängig!

Die Übertragung von Sprache erfolgt mit 64 Kbit/s. Dabei können unterschiedliche Pakete verwendet werden. Ab Bluetooth 1.2 können Sprachpakete auch wiederholt werden. Ab Bluetooth 2.0+EDR können Sprachpakete mit höherer Bandbreite übertragen werden.

Tabelle 2 ACL-Pakete, Nutzdaten und maximale Übertragungsraten in KBit/s

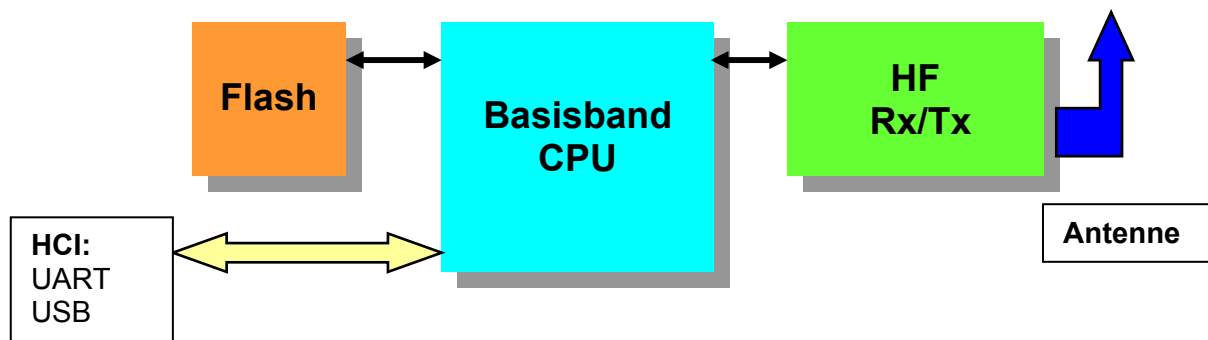
Type	Payload Header (bytes)	User Payload (bytes)	FEC	CRC	Symmetric Max. Rate (kb/s)	Asymmetric Max. Rate (kb/s)		
						Forward	Reverse	
DM1	1	0-17	2/3	yes	108.8	108.8	108.8	
DH1	1	0-27	no	yes	172.8	172.8	172.8	
DM3	2	0-121	2/3	yes	258.1	387.2	54.4	
DH3	2	0-183	no	yes	390.4	585.6	86.4	
DM5	2	0-224	2/3	yes	286.7	477.8	36.3	
DH5	2	0-339	no	yes	433.9	723.2	57.6	
AUX1	1	0-29	no	no	185.6	185.6	185.6	
2-DH1	2	0-54	no	yes	345.6	345.6	345.6	EDR
2-DH3	2	0-367	no	yes	782.9	1174.4	172.8	EDR
2-DH5	2	0-679	no	yes	869.7	1448.5	115.2	EDR
3-DH1	2	0-83	no	yes	531.2	531.2	531.2	EDR
3-DH3	2	0-552	no	yes	1177.6	1766.4	235.6	EDR
3-DH5	2	0-1021	no	yes	1306.9	2178.1	177.1	EDR

Anm.: Mit EDR gekennzeichnete Pakete werden nur bei Bluetooth 2.0+EDR verwendet.

Hardware-Realisierung eines Bluetooth Radios

Die Funktionen eines Bluetooth Radios werden u. a. durch Baseband CPU und HF Empfänger/Sender realisiert. Diese Blöcke können in zwei Chips oder auch nur einem Chip realisiert werden. Der Baseband Prozessor ist häufig eine 32 Bit bzw. 16 Bit CPU.

Bild 6 Vereinfachter Aufbau eines Bluetooth Radios.



Zwischen den einzelnen Bluetooth Radios bestehen Unterschiede hinsichtlich Funktionsumfang und Leistungsverbrauch. Bei der Auswahl einer Bluetooth Hardware empfiehlt sich immer die Prüfung der implementierten HCI-Kommandos.

Das Bluetooth Radio sendet und empfängt die Daten. Das Radio kann als Single oder Dual-Chip bzw. Modul (beinhaltet alle externen passiven und aktiven Komponenten sowie ggf. die Antenne) aufgebaut werden. Ein Bluetooth 1.2 Radio unterstützt nicht Bluetooth 2.0. Ein Bluetooth 2.0+EDR Radio unterstützt aber alle vorhergehenden (2.0, 1.2 und 1.1) Bluetooth Versionen.



ARS Software GmbH
Software für embedded Systeme
Starnberger Str. 22
D-82131 GAUTING/München
Telefon: 089-893 41 30 Fax: 089-893 41 310
Email: info@ars2000.com
www.ars2000.com