

## First Contact:

# „Bluetooth™ Wireless Technology meets CAN Fieldbus“

Dr.-Ing. Jürgen Brehm, brehm@irb.uni-hannover.de

Institut für Technische Informatik – Rechnerstrukturen und Betriebssysteme, Universität Hannover

Dipl.-Ing. Matthias Fuchs, matthias.fuchs@esd-electronics.com

esd electronic system design gmbh, Hannover

## Abstrakt

Für die Analyse und den Zugriff auf industrielle Feldbusse in autonomen oder schwer zugänglichen Systemen bietet sich die Datenübertragung über eine Funkstrecke an. Ein neuer und für die drahtlose Kommunikation entwickelter Standard ist die *Bluetooth™* Funktechnologie. Ein Bluetooth Interface wird in naher Zukunft für alle portablen Kleingeräte wie Laptop, PDA und Handy verfügbar sein.

Dieser Beitrag zeigt Realisierungsmöglichkeiten zur Übertragung von Nachrichten des CAN-Feldbusses über die Bluetooth Funkschnittstelle. Es eine erste Implementierung wird vorgestellt.

## 1 Einleitung

In der industriellen Automation werden dezentrale Peripherie und Steuerrechner häufig durch Feldbus-Systeme miteinander vernetzt. Unter den zahlreichen verfügbaren Feldbus-Systemen hat sich das *Controller Area Network* (CAN), welches ursprünglich von der Robert Bosch GmbH für den Einsatz in Kraftzeugen entwickelt wurde, in vielen Bereichen durchgesetzt.

Für die Analyse und den Zugriff auf industrielle Feldbusse in autonomen oder schwer zugänglichen Systemen bietet sich häufig die Datenübertragung über eine Funkstrecke an. Bisherige Ansätze zur drahtlosen Übertragung von Feldbus-Daten basierten meist auf proprietären Funksystemen, da es keinen einheitlichen Standard für diese Anwendungen gab. Eine Kommunikation zwischen Geräten unterschiedlicher Anbieter war deshalb ausgeschlossen.

Von der Firma Ericsson wurde 1998 ein neuer Standard vorgeschlagen: die *Bluetooth Funkschnittstelle*. Sie soll der drahtlosen Kommunikation zwischen mobilen Kleingeräten, wie Handy, PDA, Laptop und drahtlosem Headset, dienen. Eine Bluetooth Schnittstelle wird in naher Zukunft in vielen portablen Geräten wie Notebooks und PDAs integriert sein. Durch den erwarteten hohen Verbreitungsgrad der neuen Funktechnologie wird der Einsatz der Bluetooth Technik auch für industrielle

Applikationen von Bedeutung sein. Im Rahmen einer Diplomarbeit [3] wurde am Institut für Technische Informatik – Rechnerstrukturen und Betriebssysteme der Universität Hannover die Eignung der Bluetooth Funkschnittstelle für die Übertragung von Nachrichten des CAN-Feldbusses untersucht. Das Ergebnis dieser Arbeit ist ein eigenständiges Gerät, das die Kopplung der beiden Technologien, Bluetooth Funkschnittstelle und CAN Feldbus, realisiert.

## 2 Die Technologien

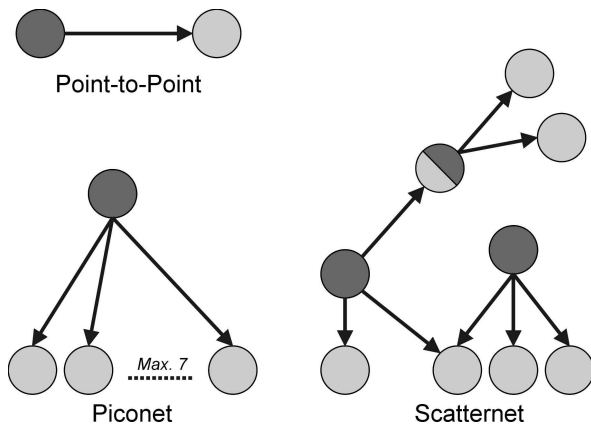
### 2.1 Die Bluetooth Funkschnittstelle

Bluetooth ist der Name einer neuen Technologie zur drahtlosen Übertragung von Sprache und Daten über kurze Distanzen. Der Bluetooth Standard [1] wurde von den Firmen Ericsson, Nokia, Toshiba, IBM und Intel ins Leben gerufen. Zusammen mit über 2000 weiteren Unternehmen bilden sie die *Bluetooth Special Interest Group* (SIG)

Die Motivation zur Entwicklung der Bluetooth Technologie war es, die große Vielfalt an Peripherieschnittstellen in der Computer und Multimediaetechnik, durch eine einzige zu ersetzen. Es war das Ziel, eine universelle und kostengünstige Schnittstellentechnologie zu entwickeln, die sowohl

---

<sup>1</sup> The BLUETOOTH trademarks are owned by Telefonaktiebolaget LM Ericsson, Sweden



konventionelle Kabel ersetzen kann, als auch einen hohen Grad an Interoperabilität bietet.

Der Anwender sollte dabei nur geringfügig mit Konfigurationsaufgaben belastet werden, daß heißt, es wurde eine weitgehend automatische Konfiguration der Kommunikationsteilnehmer angestrebt. Die Mobilität des Nutzers sollte in der Hinsicht unterstützt werden, indem er mobile Endgeräte (z.B. Laptop, PDA, Handy) in seiner Umgebung ad hoc untereinander vernetzen kann.

Bluetooth konkurriert in einigen Anwendungen, wie beispielsweise dem mobilen Zugang zu LANs, mit der WLAN-Technologie nach IEEE 802.11, die heute schon in vielen Bereichen Anwendung findet. Jedoch liegen die Ziele der Bluetooth Technologie im Wesentlichen in der Nahbereichsvernetzung von mobilen Kleingeräten. Bei diesen Geräten ist ein geringer Strombedarf des Bluetooth Interfaces („Low-Power“) aufgrund der stark begrenzten Energie-reserven besonders wichtig.

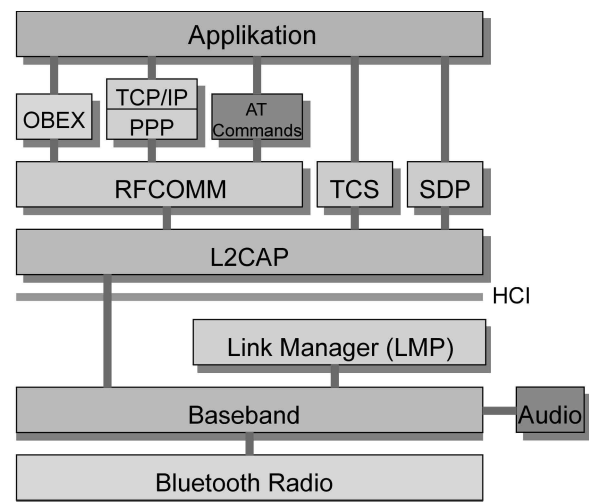
Bluetooth nutzt das in vielen Ländern nahezu einheitlich und lizenzfrei verfügbare ISM<sup>2</sup>-Frequenzband bei 2,4 GHz zur Übertragung von Sprache und Daten. Um die Übertragung gegenüber Störeinflüssen unanfälliger zu machen, wird Frequency Hopping Spread Spectrum (FHSS) als Übertragungsverfahren eingesetzt. Die Übertragungsfrequenz wird bis zu 1600 mal pro Sekunde über 79 Kanäle mit einem Kanalabstand von 1MHz gewechselt, so daß das Frequenzspektrum aufgespreizt wird. Dieses Verfahren gewährleistet eine hohe Störsicherheit gegenüber den Störquellen in diesem Frequenzband (z.B. WLANs und Mikrowellen).

Die maximale Reichweite einer Bluetooth Funkverbindung beträgt üblicherweise 10m bei 1mW

Sendeleistung. Reichweiten bis zu 100m sind von der Bluetooth Spezifikation ebenfalls vorgesehen.

Bei einer Bluetooth Kommunikation übernimmt eine der beteiligten Stationen die Rolle des Masters und die anderen werden als Slave bezeichnet. Neben der klassischen Point-to-Point-Verbindung (siehe Bild 1) sind zwei weitere Netztopologien möglich. Ein Master kann eine Point-to-Multi-Point Kommunikation mit bis zu sieben Slaves durchführen. Dieses Szenario wird als *Piconet* bezeichnet (siehe Bild 1). Weiterhin ist eine Überlappung mehrerer Piconets möglich. Einzelne Bluetooth Geräte sind dabei Slaves in einem Piconet und gleichzeitig Master eines anderen Piconets. Diese Topologie wird in der Bluetooth Funktechnologie *Scatternet* genannt.

In Bild 2 ist der Bluetooth Protokollstack abgebildet. In Abhängigkeit von den Funktionen, die ein Bluetooth Gerät zur Verfügung stellt, entfallen einige



Protokolle, so daß sich der Stack vereinfacht. Der Bluetooth Standard definiert hierzu *Profiles*, die die notwendigen Protokolle und Funktionen für bestimmte Geräteklassen festlegen.

Man kann die Protokollschichten in vier Kategorien unterteilen:

1. *Core Protocols* - Baseband, LMP<sup>3</sup>, L2CAP<sup>4</sup> und SDP<sup>5</sup> werden als Bluetooth Kernprotokolle bezeichnet. Sie bilden die kleinste Menge an Protokollen, die in einem Bluetooth Gerät implementiert sein müssen.

<sup>3</sup> Link Manager Protocol

<sup>4</sup> Logical Link Control and Adaptation Protocol

<sup>5</sup> Service Discovery Protocol

<sup>2</sup> Industrial Scientific Medical

2. *Cable Replacement Protocols* - Zu dieser Kategorie gehört das RFCOMM<sup>6</sup>-Protokoll. Es stellt höheren Protokollschichten virtuelle, codetransparente serielle Schnittstellen zur Verfügung.
3. *Telephony Control Protocols* - Die Protokolle TCS-BIN und AT-Commands steuern den Auf- und Abbau von Sprachverbindungen und dienen der Modem-Emulation.
4. *Adopted Protocols* - Einige bewährte Protokolle der Datenkommunikation sind in den Bluetooth Standard übernommen worden. Dies ermöglicht zum einen eine einfachere Integration dieser Protokolle, weil bereits erprobte Implementierungen existieren. Zum anderen wird durch den Einsatz bewährter Protokolle eine höhere Kompatibilität erreicht.

Der Bluetooth Baseband-Layer ist für das Senden und Empfangen von Datenpaketen über die physikalischen Funkverbindungen verantwortlich. Es wird hauptsächlich zwischen zwei physikalischen Verbindungstypen unterschieden:

*SCO<sup>7</sup>-Verbindungen* dienen der Übertragung von synchronen Sprachdaten mit einer konstanten Datenrate von 64kbit/s. Innerhalb eines Piconets können bis zu drei Sprachverbindungen gleichzeitig bestehen. Es wird keine automatische Sendewiederholung im Falle eines Übertragungsfehlers durchgeführt. Es ist jedoch möglich, den Sprachdaten Redundanz für eine Forward-Error-Correction (FEC) zuzufügen.

*ACL<sup>8</sup>-Verbindungen* dienen dem verbindungslosen Übertragen von Datenpaketen. ACL-Pakete besitzen eine Prüfsumme (CRC), so daß Übertragungsfehler erkannt werden können und eine automatische Sendewiederholung (ARQ) stattfinden kann. Die maximale Datenrate, die von ACL-Paketen übertragen werden kann, beträgt in einem asymmetrischen Fall 723kbit/s in einer Richtung und 56kbit/s in der Rückrichtung oder 444kbit/s in beide Richtungen bei einer symmetrischen Verbindung. Die Angaben verstehen sich als maximale Datenrate innerhalb eines Piconets, daß heißt zwischen einem Master und maximal sieben Slaves.

## 2.2 Der CAN-Feldbus

Mit seinem Ursprung in der Automobil-Industrie hat sich der industrielle CAN<sup>9</sup>-Feldbus mittlerweile in vielen Bereichen der Automation verbreitet. Auch beim CAN-Bus gehörte Kabelersatz zu den Einsatzziele. Dazu kam der Wunsch nach einer zuverlässigen und echtzeittauglichen Vernetzungsmöglichkeit zwischen dezentraler Peripherie und Steuerrechnern.

CAN [2] verwendet einen seriellen Bus zur Kommunikation zwischen den Busteilnehmern. Die Bitrate auf dem Bus kann bis zu 1Mbit/s betragen mit einer maximalen Buslänge von ca. 40m. Die Datenübertragung im CAN ist nachrichtenorientiert, d.h. es wird nicht das Ziel der Daten adressiert, sondern die Daten werden durch einen Identifier gekennzeichnet. Jeder Busteilnehmer empfängt die Nachrichten auf dem Bus auf diese Art zeitgleich (Broadcast), so daß das Controller Area Network zur Synchronisation von verteilten Steuerungen genutzt werden kann.

Der CAN-Bus ist ein Bus ohne eine dedizierte Verwaltungsinstanz (Master), die die Arbitrierung bzw. den Zugriff auf den Bus organisiert. Master sind häufig Schwachstellen, da im Falle eines Ausfalls dieser Station zunächst ein neuer Master bestimmt werden muß. Das Zeitverhalten solcher Systeme ist nur schwer vorherzusagen.

Der Buszugriff beim CAN geschieht dezentral (Multi-Master) wie z.B. beim Ethernet - aber dennoch kollisionsfrei. Es wird ein priorisiertes Zugriffsverfahren eingesetzt, bei dem die Nachrichten unterschiedliche Prioritäten besitzen. Hoch priorisierte Nachrichten haben Vorrang gegenüber niedriger priorisierten. Die Priorität wird durch den Identifier einer Nachricht festgelegt.

Eine Nachricht auf dem CAN-Bus kann bis zu acht Datenbytes transportieren. Neben den Nutzdaten enthält eine CAN-Nachricht einen vergleichsweise großen Overhead<sup>10</sup>, so daß über einen CAN-Bus eine maximale Netto-Datenrate von ca. 577kbit/s übertragen werden kann.

<sup>6</sup> Radio Frequency Communication Protocol

<sup>7</sup> Synchronous Connection-Oriented

<sup>8</sup> Asynchronous ConnectionLess

<sup>9</sup> Controller Area Network

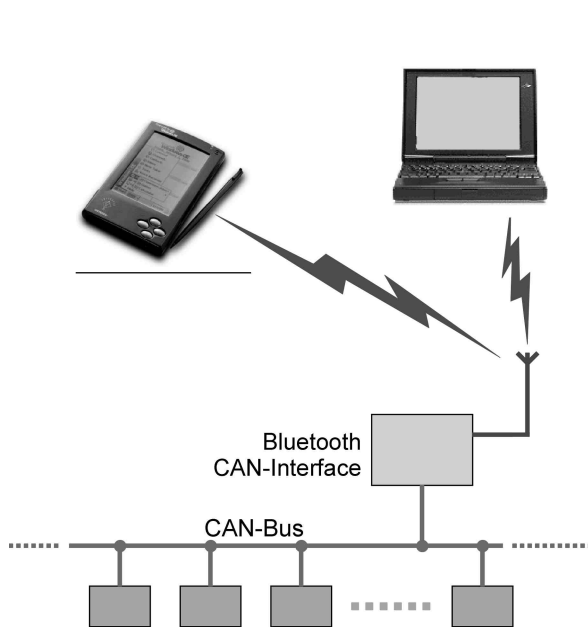
<sup>10</sup> Die Nachrichtenlänge beträgt bei einem 11Bit langem Identifier und 8 Bytes Payload mindestens 111 Bits.

### 3 First Contact

Bei der Kopplung der Bluetooth Funktechnologie mit einem CAN-Feldbus treffen zwei sehr unterschiedliche Technologien aufeinander. So hat der CAN-Feldbus seinen Ursprung in der Industrie, wo er für Echtzeit-Steuerungsaufgaben eingesetzt wird. Bluetooth ist für äußerst flexible, drahtlose Kommunikation zwischen tragbaren Kleingeräten konzipiert. Tabelle 1 zeigt einen Vergleich der beiden Technologien.

#### 3.1 Anwendungsszenarien

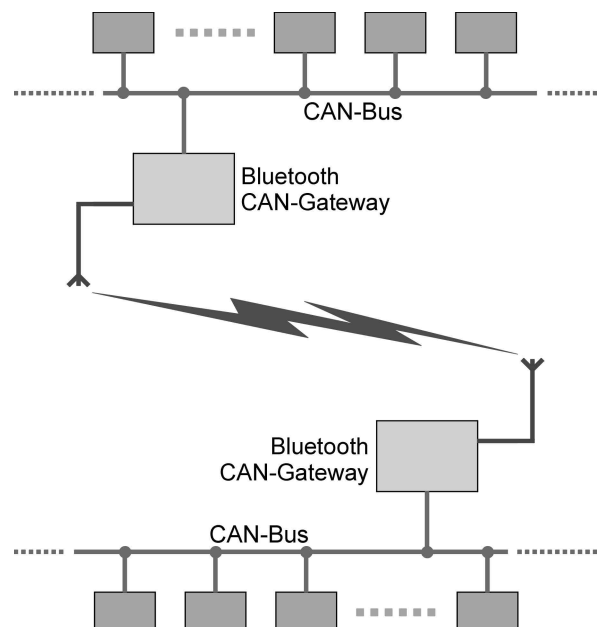
Für Übertragung von CAN-Nachrichten über eine Bluetooth Funkverbindung ergeben sich im Wesentlichen zwei Szenarien:



**Bild 3** - Bluetooth CAN-Interface

#### 3.1.1 Das Bluetooth CAN-Interface

Das Bluetooth CAN-Interface dient dem drahtlosen Zugriff auf einen CAN-Bus von herkömmlichen Bluetooth Geräten wie einem Laptop oder einem PDA (siehe Bild 3). Das Bluetooth CAN-Interface ist ein herkömmlicher Bus-Teilnehmer aus der Sicht des CAN. Als Anwendung sind Servicefälle denkbar, bei denen das Bluetooth CAN-Interface die Nachrichtepakete des CAN an eine Analysesoftware auf einem Laptop überträgt oder sie von diesem empfängt. Die drahtlose Übertragung ist hier besonders nützlich wenn sich der CAN-Bus auf einer mobilen Plattform, wie z.B. ein Service-Roboter oder KFZ, befindet.



**Bild 4** - Bluetooth CAN-Gateway

	<i>Bluetooth Technologie</i>	<i>CAN-Feldbus</i>
Brutto-Bitrate	1Mbit/s	10kbit/s – 1Mbit/s
max. Netto-Bitrate	723kbit/s ? 56kbit/s (asymmetrisch) 444kbit/s (symmetrisch)	577kbit/s
Übertragungs-verfahren	Verbindungsorientiert/-los	Nachrichtenorientiert
Adressierung	Bluetooth Adresse (48Bit)	Nachrichten-Identifizier (11Bit oder 29 Bit)
Medienzugriff	Time Division Duplex (TDD), Zuteilung durch Piconet-Master	Priorisiertes CSMA/CA
Topologie	Stern / Baum (siehe Bild 1)	Bus
Netzdurchmesser	10m (bis 100m)	>1km
Latenzzeit	Nicht garantiert für Daten- Verbindungen; Sprachkanäle bieten konstante Latenzzeit	Vorhersagbar, unter 1 ms möglich

**Tabelle 1** - Technologievergleich

### 3.1.2 Das Bluetooth CAN-Gateway

Zwei oder mehrere CAN-Busse können drahtlos durch Bluetooth CAN-Gateways verbunden werden (siehe Bild 4). Bei einer solchen Gateway-Anwendung können die Bluetooth CAN-Gateways eine Filterung der CAN-Nachrichten vornehmen und auf diese Art nur bestimmte Nachrichten von einem auf einen anderen CAN-Bus drahtlos übertragen. Man kann dieses Szenario auch als CAN-Bridge bezeichnen.

## 3.2 Realisierungsalternativen

Es sind mehrere Realisierungsansätze für die Übertragung von CAN-Frames über eine Bluetooth Verbindung denkbar. Bedingt durch die unterschiedlichen Ziele und Eigenschaften der Bluetooth Funktechnologie und des CAN, beeinflussen die Ansätze die Vorteile der jeweiligen Technologie. Für einen praktischen Einsatz einer Bluetooth Verbindung als Transportmedium für CAN-Frames muß im Einzelfall entschieden werden, welcher Ansatz am Besten geeignet ist.

Im folgenden werden verschiedene Ansätze diskutiert, die sich in dem resultierenden Protokoll-stack unterscheiden. Bei allen Ansätzen ist auf jeden Fall die Datenrate der Bluetooth Übertragung problematisch. Da die in Tabelle 1 genannte maximale Datenrate für die gesamte Datenübertragung innerhalb eines Piconets gilt, kann sie nur bei einer Point-to-Point-Verbindung voll genutzt werden. Weiterhin wird die erzielbare Datenrate durch äußere Störungen negativ beeinflusst.

Bei den Ansätzen steht im Vordergrund, daß die hohe Interoperabilität, die die Bluetooth Funktechnologie ermöglichen soll, erhalten bleibt. Der Transport von CAN-Frames über eine Bluetooth Funkverbindung soll durch die aktuelle Bluetooth Spezifikation abgedeckt werden. Alternative Ansätze, die beispielsweise Manipulationen am Baseband-Layer oder am Zugriff auf die Luftschnittstelle vornehmen, sind nichtberücksichtigt, da sie eine Änderung der Bluetooth Spezifikation bedeuten würden.

Die folgenden Abschnitte geben eine Übersicht der möglichen Ansätze unter Angabe der Vorteile(+) und Nachteile(-).

### 3.2.1 Übertragung von CAN-Nachrichten über SCO-Verbindungen

SCO-Pakete dienen normalerweise der synchronen Übertragung von Audio- bzw. Sprachdaten. Physikalische SCO-Verbindungen verwenden keinen Mechanismus zur Sendewiederholung bei Auftreten eines Übertragungsfehlers. SCO-Pakete enthalten deshalb keine Prüfsumme. Jedoch können den Paketen

FEC<sup>11</sup>-Daten zugefügt werden, so daß einige Übertragungsfehler erkannt und korrigiert werden können. Es ergeben sich folgende Eigenschaften:

- + Garantierte und konstante Latenzzeit durch synchrone Übertragung.
- + Für die CAN-Datenübertragung sind nur die untersten Bluetooth Protokollschichten nötig. Der Speicherbedarf für die Bluetooth Software auf einer Hardwareplattform ist dadurch gering.
- Fehlererkennung und deren Korrektur ist nur bedingt anhand der FEC-Daten möglich.
- Die Übertragungskapazität ist gering. Bei Verwendung von SCO-Paketen ohne FEC steht eine Übertragungskapazität von 192kbit/s (duplex) zur Verfügung. Je nach verwendetem SCO-Pakettyp sind auch 128 oder 64 kbit/s möglich, mit dafür steigendem Anteil an FEC-Daten. Es kann in allen Fällen kein hoch belasteter CAN-Bus transparent übertragen werden.
- Eine SCO-Verbindung mit 192KBit/s nutzt alle verfügbaren Timeslots, so daß Master und Slave keine weiteren Bluetooth Aktivitäten ausführen können.

Dieser Ansatz ist für Anwendungen geeignet, bei denen keine hohen Daten- bzw. Nachrichtenraten übertragen werden müssen und eine maximale Latenz garantiert werden muß.

Der Ansatz erfordert jedoch ein neues spezielles Bluetooth Profile, daß die Echtzeitdatenübertragung von CAN-Nachrichten über SCO-Verbindungen spezifiziert.

Bedingt durch die geringe Übertragungskapazität der SCO-Verbindungen kann es für bestimmte Anwendungen sinnvoll sein, hochpriorisierte CAN-Nachrichten bevorzugt zu übertragen und niederpriorisierte Nachrichten zunächst im Bluetooth CAN-Interface/Gateway zwischen-zuspeichern.

### 3.2.2 Übertragung von CAN-Frames über einen L2CAP-Kanal

Die L2CAP-Protokollschicht ist in allen Bluetooth Geräten implementiert. Die Übertragung von CAN-Nachrichten würde bei dieser Implementierung ein neues Protokoll oberhalb des L2CAP-Layers erfordern.

- + Einfache Implementierung, da L2CAP immer vorhanden ist.
- + L2CAP verwendet ACL Pakete für die Kommunikation, so daß die maximale Bluetooth Datenrate nahezu ausgeschöpft

---

<sup>11</sup> Forward Error Correction

- werden kann. Ein zusätzlicher Overhead durch das L2CAP-Protokoll ist sehr gering.
- Kein deterministisches Zeitverhalten. Durch das ARQ-Verfahren, welches bei ACL-Paketen angewendet wird, ist das Zeitverhalten einer Übertragung schwer bestimmbar.
- In der Bluetooth Spezifikation existiert noch kein Protokoll, welches für die Übertragung von CAN-Nachrichten oberhalb des L2CAP-Layers verwendet werden kann.

Genau wie in Abschnitt 3.2.1 erfordert dieser Ansatz eine Erweiterung der Bluetooth Spezifikation. Neben der Einführung eines neuen Protokolls oberhalb der L2CAP-Schicht ist auch ein Bluetooth Profile erforderlich, welches die Nutzung des neuen Protokoll festlegt.

### 3.2.3 Übertragung von CAN-Frames über eine RFCOMM-Verbindung

Das RFCOMM-Protokoll wird von sehr vielen potentiellen Kommunikationspartnern eines Bluetooth CAN-Interfaces (Laptops, PDAs etc) unterstützt werden, da es von nahezu allen bisher spezifizierten Datenkommunikations-Profiles genutzt wird.

- + Hohe Kompatibilität mit anderen Bluetooth Geräten, da RFCOMM auf Standardgeräten fast immer implementiert ist.
- + Applikation oder CAN-Treiber kann auf einer von RFCOMM bereitgestellten virtuellen seriellen Schnittstelle aufsetzen. Die CAN-Software kann auf diese Art portabel für andere Plattformen gehalten werden.
- Kein deterministisches Zeitverhalten. Durch das ARQ-Verfahren, welches bei ACL-Paketen angewendet wird, ist das Zeitverhalten einer Übertragung schwer bestimmbar.
- Hohe Latenzzeiten, da viele tiefere Protokollschichten durchlaufen werden müssen.

### 3.2.4 Übertragung von CAN-Frames über das TCP/IP und HTTP-Protokoll mittels eines Web-Browsers

Für diesen Ansatz enthält das Bluetooth CAN-Interface einen eingebetteten Web-Server, der ein Web-Frontend für den primitiven Zugriff auf den CAN-Bus bereitstellt. Dadurch wird ein einfacher Zugriff auf einen CAN-Bus durch jeden

handelsüblichen Bluetooth PDA möglich. Es wird nur ein Web-Browser auf dem PDA benötigt.

- + Dieser Ansatz wird voll von der Bluetooth Spezifikation durch das *LAN Access Profile* berücksichtigt.
- Nur geeignet für Anwendungen mit sehr geringer Nachrichtenrate auf dem CAN-Bus.
- Versenden von CAN-Frames über ein Web-Interface ist unkomfortabel.

Dieser Ansatz eignet sich nur für einfache Demonstrationen, Service- und Wartungsarbeiten an einem CAN-System.

## 3.3 Implementierung

Für die erste Implementierung eines Bluetooth CAN-Interfaces ist zunächst eine Hardware-Plattform entwickelt worden. Im zweiten Schritt wurde die Software für den Transport von CAN-Nachrichten oberhalb der RFCOMM-Protokollschicht implementiert (siehe Kapitel 3.2.3). Zusätzlich ist das einfache Senden und Empfangen von CAN-Nachrichten über ein Web-Interface implementiert worden (siehe Kapitel 3.2.4).

### 3.3.1 Hardware-Plattform

Die Hardware besteht aus einer Embedded-PowerPC Plattform auf der der IBM PowerPC PPC405CR zum Einsatz kommt. Diese CPU enthält bereits zahlreiche Interfaces (SDRAM-Controller, UARTs), so daß nur wenig zusätzliche Komponenten nötig sind, um eine leistungsfähige Entwicklungsplattform zu erhalten. Für die CAN-Schnittstelle wird der CAN-Controller SJA1000 von Philips eingesetzt. Die Bluetooth Anbindung wird durch das Bluetooth Modul ROK101007 der Firma Ericsson realisiert. Das ROK101007 ist eine der ersten integrierten Bluetooth Lösungen die überhaupt angeboten worden sind. Aufgrund der anfangs schlechten Verfügbarkeit dieses Bauteils wurde der Prototyp auf Basis des ROK101008 aufgebaut. Dieses Modul unterstützt keine Pico- oder Scatternet Topologien, sondern nur Point-To-Point-Verbindungen mit einer Gegenstation. Ein detailliertes Blockschaltbild der Hardware kann [5] entnommen werden.

### 3.3.2 Software

Zunächst wurde Linux als Betriebssystem für das Bluetooth CAN-Interface ausgewählt. Der wesentliche Grund für diese Entscheidung ist die Verfügbarkeit eines freien Bluetooth Protokollstacks [4], der von Axis Communications AB unter einer OpenSource Lizenz (LGPL) entwickelt wird. Dieser Protokollstack implementiert die Bluetooth Protokolle L2CAP, RFCOMM und SDP. Für den Datenaustausch mit dem Bluetooth Modul wird das HCI<sup>12</sup>-Protokoll verwendet.

Neben der nötigen Software für das Bluetooth Verbindungsmanagement war es erforderlich ein Transportprotokoll zu entwickeln, daß die Übertragung von CAN-Frames über eine RFCOMM-Verbindung abwickelt. Dieses Protokoll wurde *CANBT* getauft.

Das CANBT-Protokoll definiert einen Protokollrahmen gemäß Bild 5: Einer Payload von maximal 31 Byte wird ein Protokoll-Header vorangestellt. Die Payload enthält die eigentliche CAN-Nachricht, bestehend aus dem Identifier und den bis zu acht Bytes CAN-Nutzdaten.

Vereinfacht enthält der Header<sup>13</sup> das *COMMAND*-Feld, welches den Aufbau der Payload festlegt und das *LEN*-Feld, das die tatsächliche Länge der Payload angibt.

Durch Verwendung dieses Protokollrahmens ist es den Instanzen dieses Protokoll in den beteiligten Bluetooth Geräten möglich, den Datenstrom, der über die RFCOMM-Verbindung übertragen wird, auszuwerten. Mittels des *COMMAND*-Feld ist es möglich zwischen verschiedenen Pakettypen zu unterscheiden und bei Bedarf verschiedene Zusatzinformationen in die Payload aufzunehmen.

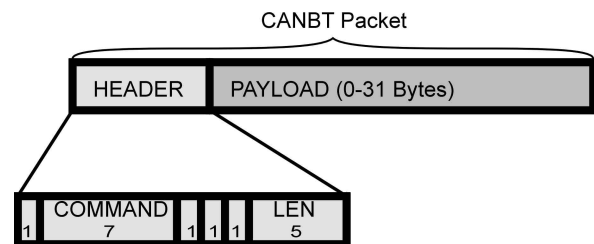
Um die relative zeitliche Beziehung zwischen CAN-Nachrichten nach der Bluetooth Übertragung zu erhalten kann ein CANBT-Paket einen Timestamp enthalten. Diese Zusatzinformation gibt Auskunft über den Zeitpunkt des erfolgreichen Versendens bzw. des Empfangs einer CAN-Nachricht.

Weitere Pakettypen zwingen die Gegenstation eine Quittung zu senden oder die lassen es zu, daß auf eine Bestätigung verzichtet werden kann.

Durch die verschiedenen Pakettypen ist versucht worden einige der CAN-Eigenschaften zu erhalten. Durch Verzicht auf Bestätigungen und Timestamps wird die Datenrate über die RFCOMM-Verbindung gering gehalten, so daß eine hohe CAN-Nachrichtenrate übertragen werden kann. Werden Informationen über den Zeitpunkt einer erfolgreichen

Nachrichtensendung benötigt, können diese Informationen (Timestamp) zusätzlich übertragen werden, jedoch mit negativem Einfluß auf die Nachrichtenrate.

Neben der eigentlichen Übertragung von CAN-Nachrichten sind einige Pakettypen definiert worden, die der Konfiguration eines Nachrichtenfilters dienen. Durch einen Nachrichtenfilter ist es möglich, nur genau die Nachrichten zu übertragen, die von der Gegenstation benötigt werden.



## 4 Erste Ergebnisse

Während der ersten Tests der beschriebenen Implementierung diente ein Laptop, ausgestattet mit einer Bluetooth PCMCIA Karte, als Gegenstation. Eine weiterer „Gegner“ während der Tests war ein zweites Bluetooth CAN-Interface.

Bei beiden Gegenstationen betrug der maximal erreichte Datendurchsatz bei der Datenübertragung mittels des RFCOMM-Protokolls weniger als 20kByte/s. Dieser Datendurchsatz wird durch die (HCI-) Schnittstelle zwischen Bluetooth Modul und UART begrenzt. Das ausgewählte Bluetooth Modul von Ericsson läßt eine maximale Baudrate von 460800 Baud zu, wodurch die maximale, asymmetrische Datenrate nicht erreicht werden kann. Eine weitere Reduktion des erreichbaren Datendurchsatzes ist in dem frühen Status des Bluetooth Protokollstacks begründet.

Die Latenzzeiten einzelner CAN-Nachrichten von bis zu 10ms beim Transport mittels des CANBT-Protokolls über eine Bluetooth Verbindung ist für viele potentielle Zielanwendungen zu hoch. Die Übertragungszeiten werden zusätzlich von einem Jitter überlagert.

Dieser zweite Effekt ist im Einsatz von Linux ohne Echtzeiterweiterungen auf der Testplattform begründet. Es wird erwartet, daß sich die Latenzzeiten für einzelne CAN-Nachrichten mit der Weiterentwicklung der Bluetooth Module und des Protokollstacks auf einen erträglichen Wert verbessern.

<sup>12</sup> Host Controller Interface

<sup>13</sup> Einige Bits des Protokoll-Headers sind reserviert für zukünftige Erweiterungen.

Die Übertragung von CAN-Nachrichten über eine Bluetooth-Verbindung ist im Rahmen dieser Studie erfolgreich durchgeführt worden. Dabei haben sich einige theoretische Probleme ergeben:

Obwohl die Bluetooth Funktechnologie Übertragungsraten bis zu 723kbit/s ermöglicht, läßt sich ein CAN-Bus mit einer maximalen Nutzdatenrate von 577kbit/s nicht transparent übertragen. Um den Nachrichtencharakter von CAN-Frames zu erhalten, muß neben den Nutzdaten einer CAN-Nachricht auch deren Längeninformation und der Identifier übertragen werden. Für den vollbelasteten CAN-Bus ergibt sich somit eine Forderung von mindestens 712kbit/s<sup>14</sup> an die Übertragungsrate einer Bluetooth Verbindung. Diese erforderliche Datenrate weiterhin durch den Overhead der Zusatzinformationen (z.B. Timestamps, s.o.) vergrößert. Eine Lösung bringt die Nachrichtenfilterung, welche nur die relevanten CAN-Nachrichten über die Luftstrecke überträgt und so Übertragungskapazität einspart.

Besonders im Einsatz als Diagnosewerkzeug kann es vertretbar sein, ein starkes Datenaufkommen auf dem CAN-Bus, zunächst im Bluetooth CAN-Interface zwischenzuspeichern.

## 6 Zusammenfassung

Die Nutzung der Bluetooth Funktechnologie zur Übertragung von CAN-Nachrichten ist eine interessante Alternative zur Funkübertragung über andere, proprietäre Funkstandards. Gerade durch den zu erwartenden hohen Verbreitungsgrad von Bluetooth Geräten, wird der CAN-Zugang mittels handelsüblicher Bluetooth Geräte, besonders bei Serviceanwendungen sehr interessant.

Es hat sich herausgestellt, daß die Bluetooth Übertragungskapazität nur bei weniger ausgelasteten CAN-Systemen ausreichend ist.

Die diskutierten möglichen Ansätze zur Übertragung von CAN-Nachrichten, insbesondere unter Echtzeitanforderungen, lassen sich mit der aktuellen Bluetooth Spezifikation nur schwer umsetzen. Eine Erweiterung der Spezifikation um ein Echtzeit-datenübertragungs-Profil ist wünschenswert.

In zukünftigen Weiterentwicklungen muß auf jeden Fall das Zeitverhalten bei der CAN-CAN-Kopplung mittels zwei Bluetooth CAN-Interfaces verbessert werden, so daß die drahtlose CAN-Bridge attraktiver wird.



Ein Bluetooth CAN-Interface (Bild 5) einschließlich Applikationssoftware ist mittlerweile bei der esd GmbH erhältlich. Das Gerät [5] bietet neben der Funktionalität eines drahtloses CAN-Interfaces die Möglichkeit einer Analogaufnahme der CAN-Leitungen („Speicheroszilloskop“) sowie eine automatische Baudratenerkennung auf dem CAN-Bus und kann somit als mobiles oder stationäres Diagnosewerkzeug eingesetzt werden. Mit zwei dieser Geräte läßt sich eine drahtlose Kopplung von zwei CAN-Systemen realisieren.

## Literatur

- [1] Bluetooth Special Interest Group: Bluetooth specification 1.1, [www.bluetooth.com/developer/specification/Bluetooth\\_11\\_Specifications\\_Book.pdf](http://www.bluetooth.com/developer/specification/Bluetooth_11_Specifications_Book.pdf)
- [2] CAN in Automation: CAN specification 2.0, [www.can-cia.de/CAN20A.pdf](http://www.can-cia.de/CAN20A.pdf), [www.can-cia.de/CAN20B.pdf](http://www.can-cia.de/CAN20B.pdf)
- [3] „Entwicklung von Hard- und Software zur Kopplung des CAN-Feldbusses mit der Bluetooth Funkschnittstelle“, Diplomarbeit von Matthias Fuchs, Hannover, 2001
- [4] Axis Communications AB: OpenBT opensource Bluetooth protocol stack, [developer.axis.com/software/bluetooth/](http://developer.axis.com/software/bluetooth/)
- [5] esd electronic system design gmbh: Datenblatt zum Bluetooth CAN-Interface, [www.esd-lectronics.com/english/products/CAN/can-bluetooth.htm](http://www.esd-lectronics.com/english/products/CAN/can-bluetooth.htm)

<sup>14</sup> Bei Übertragung von CAN-Nachrichten mit ausschließlich 11-Bit Identifierlänge