Kommunikationsstandards in Wireless Sensor Networks

Lukas Tillmann
Betreuerin: Corinna Schmitt
Seminar Innovative Internet Technologien und Mobilkommunikation WS09/10
Lehrstuhl Netzarchitekturen und Netzdienste
Fakultät für Informatik, Technische Universität München
Email: tillmann@in.tum.de

ABSTRACT

Heutzutage findet immer mehr Funkkommunikation zwischen verschiedensten Geräten statt. Um die Probleme der drahtlosen Kommunikation in Hinsicht auf Energieverbrauch und Zuverlässigkeit zu lösen, wurden in den letzten Jahren viele Projekte zur Erstellung eines Kommunikationsstandards durch namenhafte Unternehmenszusammenschlüsse angestoßen. Die übergreifende Vernetzung verschiedenster Geräte zu einem Gesamtnetzwerk bei gleichzeitiger Effizienz soll durch diese Standards realisiert werden, da viele dieser Geräte keine feste Anbindung zu einem Stromnetz besitzen, und somit Möglichkeiten zur Energieeinsparung eine große Rolle spielen. Dieses Paper soll einen Einblick in drei verschiedene Standards bieten und diese kurz gegenüberstellen.

Keywords

Wireless Sensor Network, Sensornetz, Mobilkommunikation, Bluetooth, Zigbee, 6LoWPAN

1. EINLEITUNG

Dieses Paper beginnt mit einer Vorstellung des Grundkonzeptes der Wireless Sensor Networks (WSNs) in Kapitel 2. Die folgenden vier Kapitel erläutern die geläufigen Kommunikationsstandards in WSNs - Bluetooth, IEEE 802.15.4 und darauf aufbauend ZigBee und 6LoWPAN. Es folgt ein Vergleich der Standards im Hinblick auf Einsatzgebiet und Effizienz in Kapitel 7. Der Ausblick in Kapitel 8 stellt noch einige weitere Standards kurz vor, die sich im Anwendungsgebiet von den hier diskutierten stark unterscheiden.

2. WIRELESS SENSOR NETWORKS

Ein Wireless Sensor Network oder Sensornetz ist ein drahtloses Netzwerk zwischen verschiedenen Geräten, die gegebenenfalls auch Kleinstgeräte mit sehr begrenzten Funktionen darstellen 1. Ein Beispiel für so ein Kleinstgerät oder Sensor ist ein Thermometer, das mittels einer Antenne die Messergebnisse an eine Basisstation weitersendet. Neben solchen, oftmals stationären, Sensoren gibt es aber auch die Notwendigkeit nach ad-hoc Netzwerken, die sich selbstständig organisieren und einen Datenaustausch ermöglichen sollen. Hierfür gibt es verschiedenste Ansätze, die im Folgenden erläutert werden. werden.

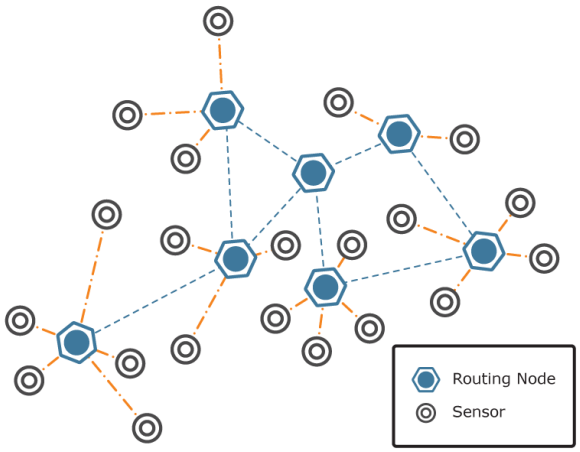

Figure 1: Beispielhafter Aufbau eines WSNs [7]

3. BLUETOOTH

Bluetooth ist der sicherlich bekannteste in diesem Paper diskutierte Kommunikationsstandard für Funknetze. Bluetooth wurde ursprünglich nicht für den Einsatz in WSNs entwickelt, kann aber durch seine Funktionsweise auch dafür eingesetzt werden. Hierbei wird das lizenzfreie ISM-Band (Industrial, Scientific and Medical Band) im Bereich von 2,402 bis 2,48 GHz verwendet, welches in 79 Kanäle mit einem Abstand von je 1 MHz eingeteilt ist [2].

Klasse	Sendeleistung	max. Reichweite
Klasse 1	100 mW	100 Meter
Klasse 2	2,5 mW	20 Meter
Klasse 3	1 mW	10 Meter

Table 1: Einteilung der Bluetooth Sendeleistung in Klassen [1]

Die Reichweite dieser Klassen aus Tabelle 1, die durch ihre Leistung definiert werden, hängt zudem von den Umweltbedingungen, wie etwa Störobjekte oder Interferenzen durch andere Geräte, ab. Die hier angegebenen Werte sind jeweils unter Optimalbedingungen erzielte Reichweiten, und sind

in der Realität oft nicht erreichbar. Heutzutage wird in der Regel eine Sendeleistung von 1 mW verwendet, was für die meisten Einsatzgebiete ausreichend ist [1].

3.1 Netztopologie

Generell gilt für alle Bluetooth Netzwerke, dass es pro Piconet einen Master gibt, und sich alle anderen Geräte diesem als Slave unterordnen [2]. Der Master übernimmt hierbei die Vergabe der Adressen und die Zuteilung der Sendezeiten mittels Zeitmultiplexingverfahren, also das Einräumen fester Zeitfenster für die Slaves. Grundsätzlich kann jedes Bluetooth Gerät die Rolle des Masters einnehmen. In einem Bluetooth-Netzwerk sind Unicast- und Multicast-Verbindungen möglich, welche aber alle über den Master verschickt werden müssen. Eine Kommunikation zwischen den einzelnen Slaves ist nicht möglich.
Es wird zwischen dem Piconet-Mono-Slave-Modus und dem Piconet-Multi-Slave-Modus unterschieden. Während der Mono-Slave-Modus lediglich zwei Knoten enthält, die miteinander kommunizieren, stellt der Multi-Slave-Modus ein Netzwerk von bis zu 8 aktiven Knoten dar. Diese werden durch ihre 3-Bit lange Active Member Address (AM_Address) angesprochen, welche vom Master vergeben werden. Zusätzlich können weitere Geräte mittels einer 8-Bit Park Member Address (PM_Address) gespeichert werden, die aber an der aktiven Kommunikation nicht teilnehmen können, bis sie eine AM_Address erhalten. Es können insgesamt 263 Knoten gleichzeitig in einem solchen Piconet vorhanden sein. Um diese Netzwerke noch zu erweitern, können mehrere Piconets in Sendereichweite ein so genanntes Scatternet bilden, in dem eine Kommunikation von mehr als 8 aktiven Knoten möglich wird. Hierbei darf es pro Netz nur einen Master geben, der gegebenenfalls die Rolle des Slaves in einem anderen, überlappenden Netzwerkes übernimmt. Slaves können mehreren Piconets angehören.

3.2 Verbindungsaufbau

Um eine ad-hoc Verbindung zwischen mehreren Bluetooth Geräten herzustellen wird der Inquiry-Modus gestartet. Dieser besteht dem Verschicken von so genannten ID-Paketen auf 32 festgelegten Frequenzen. Gleizeitig wird jeweils eine dieser Frequenzen in einem Inquiry_Scan nach ID-Paketen abgehört und jeweils nach 1,28 Sekunden in den nächsten Kanal gewechselt. Sobald ein Gerät ein ID-Paket auf diesem Frequenzkanal erhält schickt es eine Bestätigungsnachricht zurück und ordnet sich als Slave ein. Der Sender des ID-Paketes wird zum Master in diesem Netzwerk. Wenn ein solches Netzwerk etabliert wurde betritt der Master den Page-Modus und sendet in regelmäßigen Intervallen ID-Pakete auf 16 Frequenzen, in denen er die Slaves auffordert zu antworten. Hierbei erfolgt auch die Vergabe der Active Member Address (AM_Address), die zur Kommunikation zwischen Knoten in diesem Netzwerk erforderlich ist.

3.3 Verbindungstypen

Bei Bluetooth wird zwischen zwei Verbindungstypen unterschieden, je nach benötigtem Kommunikationstyp. Wie bereits erwähnt, wird bei der Kommunikation ein Zeitmultiplexing-Verfahren verwendet, also sind feste Sendezeiten für Slaves vorgesehen [2].

3.3.1 Asynchronous Connection Less (ACL)

ACL-Verbindungen sind für Multicast-Kommunikation zwischen dem Master und allen Slaves möglich, können aber nur in nicht für andere Kommunikation reservierten Zeitfenstern verwendet werden. Es kann maximal eine ACL Verbindung gleichzeitig existieren. Um die Datenintegrität sicherzustellen ist hierbei im Falle eines Paketverlusts eine Wiederholung der Datenübertragung vorgesehen .

3.3.2 Synchronous Connection Oriented (SCO)

Im Gegensatz zu ACL-Verbindungen stellt SCO eine Unicast-Verbindung zwischen einem Master und genau einem Slave her. Bis zu drei dieser Verbindungen zu einem oder unterschiedlichen Slaves sind gleichzeitig für einen Master möglich, ein Slave kann jedoch, falls er sich in einem Scatternet befindet, nur zu zwei Master-Knoten eine solche Verbindung unterhalten. Durch die Begrenzung mittels eines Zeitfensters ist keine wiederholte Datenübertragung vorgesehen, ein Paketverlust führt somit auch zum Datenverlust.

3.4 Datenübertragung

Um Interferenzen durch andere Geräte - und somit Datenverlust - bei der Übertragung gering zu halten setzt Bluetooth das Fast-Frequency-Hopping-Verfahren ein. Jedes Piconet hat hierfür eine durch die Bluetooth Device Address (BD_Address) festgelegte 79-stellige Frequency-Hopping-Sequenz, die den Ablauf des Frequenzwechsels bestimmt, und dem Sender sowie dem Empfänger bekannt ist. Die 79 zur Verfügung stehenden Kanäle werden hierbei abwechselnd gleichmäßig benutzt und alle 625 µs gewechselt. Falls also eine Datenkollision auf einer Frequenz stattfindet, ist es unwahrscheinlich, dass der Rest der Übertragung auch gestört wird, da von den Geräten unterschiedliche Hopping-Sequenzen verwendet werden, und sich die Kollision somit von alleine auflöst.

3.5 Fehlerkorrektur

Um eine zuverlässige Datenübertragung zu gewährleisten stellt Bluetooth drei Fehlerkorrekturverfahren zur Verfügung. Hierbei kann zwischen präventiver Fehlerkorrektur und dem nachträglichen Beheben von Fehlern durch einen erneuten Datenversand unterschieden werden [2].

3.5.1 Forward Error Correction (FEC)

Die Vorwärtsfehlerkorrektur wird als präventive Maßnahme zur vollständigen Datenübertragung eingesetzt. Eine Methode ist die 1/3 FEC, die durch Redundanz die Datenübertragung verbessern soll. Falls die Verbindungsqualität schlecht ist wird hierbei jedes Bit dreimal gesendet, wodurch einzelne Bitfehler unwahrscheinlicher werden. Dieses Verfahren wird für die Nutzlast dynamisch, je nach auftretender Fehlerrate eingesetzt, der Header wird aber in der Regel immer mittels 1/3 FEC gesichert. Eine weitere, neuere Methode ist die 2/3 FEC, bei der 10 Bits jeweils in 15 Bits konvertiert werden mittels des Hamming-Codes. Hierbei werden Paritätsbits gebildet, mit denen 1-Bit Fehler erkannt und korrigiert werden können.

3.5.2 Automatic Repeat Request (ARQ)

Bei dieser Art der Fehlerkorrektur wird durch den Sender Daten solange wiederholt verschickt, bis der Empfänger eine Empfangsbestätigung sendet. Der Empfänger ignoriert so

lange die einkommenden Nachrichten, bis der Cyclic Redundancy Check (CRC) mit der übertragenen Prüfsumme übereinstimmt.

3.6 Betriebsmodi

Da ein Slave gegebenenfalls nicht dauerhaft aktiv bleiben müssen und oft mit Batterien betrieben wird, implementiert Bluetooth einige Modi um die Aktivität eines Gerätes zu verringern und somit Energie zu sparen.

3.6.1 Active Modus

Im Active Modus ist das Gerät im Betrieb und kann ACL- und SCO-Verbindungen unterhalten. Es besitzt eine AM_Address und kann mittels dieser angesprochen werden.

3.6.2 Park Modus

Wenn ein Gerät den Park Modus besitzt, gibt es seine AM_Address auf und erhält eine 8-Bit PM_Address. Es findet lediglich der Austausch von ID-Paketen in regelmäßigen Abständen statt um eine zeitliche Synchronisation mit dem Master zu gewährleisten. Falls ein Datenaustausch wieder erforderlich wird, muss wieder eine Kommunkationsaufnahme mit dem Master stattfinden um eine AM_Address zu erhalten.

3.6.3 Sniff Modus

Der Sniff Modus wird von Geräten eingenommen, falls derzeit keine Kommunikation stattfinden muss und eine direkte Abmeldung aus dem aktiven Netzwerk nicht notwendig ist, aber trotzdem Energie eingespart werden soll. Hierbei wird den Slaves eine gewisse zeitliche Toleranz gegeben um auf ID-Pakete zu antworten. Es muss somit nicht mehr jedes ID-Paket beantwortet werden.

3.6.4 Hold Modus

Im Hold Modus werden nur noch SCO-Verbindungen aufrecht erhalten, ACL-Verbindungen finden nicht mehr statt. Dies reduziert die zu erhaltenden Pakete, da, abgesehen von regelmäßigen ID-Paketen, nur noch im zugewiesenen Zeitfenster Kommunikation stattfindet.

4. IEEE 802.15.4

Die im Folgenden vorgestellten Standards, Zigbee und 6LoWPAN, bauen beide auf dem IEEE 802.15.4 Standard auf, der hier erläutert werden soll. Es handelt sich hierbei um die Definition der Bitübertragungsschicht (Physical-Layer) und der Sicherungsschicht (MAC-Layer) für Funknetze [5].

4.1 Gerätetypen

Der IEEE 802.15.4 Standard sieht zwei grundlegende Arten von Gerätetypen vor: Die Full Function Devices (FFD) und die Reduced Function Devices (RFD). Während FFDs den vollen Funktionsumfang zur Verfügung stellen und eine Kommunikation zu allen anderen Geräten ermöglichen, sind RFDs für eher geringfügig aktive Sensoren einzusetzen. RFDs können ausschliesslich mit FFDs kommunizieren und können somit nur Endpunkte in einem Kommunikationsnetzwerk bilden. Zusätzlich gibt es einen Koordinator in einem Netzwerk, der ein Full Function Device sein muss, das Netzwerk mittels einer Personal Area Network Identifier eindeutig definiert und gegenüber anderen abgrenzt. Von diesem Knoten geht auch die zeitliche Synchronisation für verbundene Geräte aus und findet, in höheren Schichten des OSI-Modells, die aber durch IEEE 802.15.4 noch nicht definiert werden, das Routing statt.

4.2 Netztopologie

In IEEE 802.15.4 sind drei Arten von Netztopologien anhand des Typs der Verknüpfung der Knoten zu unterscheiden. Die Wahl der Topologie spielt durch die Möglichkeit RFDs einsetzen zu können eine große Rolle, da die Kommunikation nur zwischen FFD und RFD oder FFD und FFD stattfinden kann.

4.2.1 Star-Topologie

Bei einer Star-Topologie laufen alle Kommunikationswege an einer Basisstation (Koordinator) zusammen. Es findet keine direkte Kommunikation zwischen den Endgeräten statt, da jeder Weg über diesen zentralen Knoten läuft. Dementsprechend muss die Basisstation als ein Full Function Device implementiert sein, um die Kommunikation auch weiterführend in andere Netze zu ermöglichen. Die einzelnen Endknoten können sowohl FFDs als auch RFDs sein.

4.2.2 Peer-To-Peer-Topologie

Eine weitere Topologie ist die Peer-To-Peer Architektur, in der jeder Knoten einen "gleichberechtigten"Partner darstellt. Sofern eine Verbindung besteht können beliebige Knoten auch untereinander Daten austauschen, was dazu führt, dass jedes dieser Geräte ein FFD sein muss. Auch in dieser Topologie gibt es einen Koordinator, was aber hier ein beliebiges Gerät sein kann.

4.2.3 Cluster-Tree-Topologie

Die Cluster-Tree-Topologie stellt eine Baumstruktur dar, also einer kreisfreien und zusammenhängenden Struktur. Hierbei können die Blätter sowohl als FFDs als auch RFDs realisiert sein, die Wurzeln müssen jedoch alle FFDs sein, um die Kommunikation in dieser Architektur zu ermöglichen.

4.3 PHY-Layer

Die Bitübertragung bei IEEE 802.15.4 findet, wie bei Bluetooth auch, in der Regel im 2,4 GHz ISM-Band (2400 - 2483,5 MHz) statt und enthält eine Aufteilung in Kanäle. Zusätzlich gibt es noch lokal definierte Frequenzbänder, nämlich 868 - 868,6 MHz (Kanal 0) in Europa und 902 - 928 MHz (Kanale 1 bis 10) in den Vereinigten Staaten von Amerika. Im Gegenteil zu Bluetooth wird hier aber kein 1 MHz Sprung pro Kanal vorgenommen, sondern ein 2 MHz Sprung bei den Kanälen 1 bis 10, und ein 5 MHz Sprung im 2,4 GHz ISM Band, was in eine Aufteilung zu 16 Kanäle (11 bis 26) resultiert.

4.4 MAC-Layer

In der Sicherungsschicht wird die Art der Datenübertragung festgelegt. Grundsätzlich wird hierbei zwischen drei Arten des Datentransfers unterschieden. Daten können periodisch auftreten, das heiSSt eine Anwendung, die auf dem Sensor läuft aktiviert die Kommunikation wenn es nötig wird, also etwa wenn ein Test auf dem Sensor abgelaufen ist wird das Ergebnis übermittelt. Ein anderer Kommunikationstyp is der sporadische Datentransfer, der zeitlich unvorhersehbar eintritt. Hier kommen zum Beispiel Warnsysteme wie

Zigbee	Anwendungsschicht
Zigbee	Security
Zigbee	Vermittlungsschicht
IEEE 802.15.4 3	MAC-Layer
IEEE 802.15.4 3	PHY-Layer

Table 2: Protokollstapel der ZigBee Architektur [5]

Alarmanlagen oder Rauchdetektoren in Frage, bei denen lediglich eine Kommunikation notwendig ist, wenn ein abweichender Wert auftritt. Der dritte Typ ist der sich wiederholende Datentransfer, der zu festen Zeitpunkten stattfindet, wie etwa Temperaturfühler, die alle paar Sekunden Messdaten liefern sollen.
Um diesen Arten der Datenübertragung passend und energiesparend gerecht zu werden bietet der IEEE 802.15.4 Standard zwei verschiedene Modi, den Beacon und den Non-Beacon Modus.

4.4.1 Beacon-Modus

Im Beacon-Modus warten die Endgeräte auf den Beacon, also ein Signal des Koordinators, der periodisch versendet wird. Nach diesem Beacon ist ein Sendefenster für den Empfänger (Guaranteed-Time-Slot) vorgesehen, in dem Daten übermittelt werden können. Zudem wird hierbei der Zeitpunkt des nächsten Beacons definiert, wodurch das Endgerät Strom sparen kann, da es bis zu diesem Zeitpunkt auf keinerlei Kommunikation achten muss. Diese Methode findet vor allem bei periodischem und repetitivem Datenaustausch statt. Da der Koordinator auch nur Daten nach einem Beacon erwartet kann auch dieser in einen ßleepModus versetzt werden bis zum nächsten geplanten Empfang, was für Batteriebetriebene Koordinatorzellen die Lebenszeit erhöht.

4.4.2 Non-Beacon-Modus

Bei dem Einsatz eines Non-Beacon-Modus befindet sich der Koordinator immer in einem Empfangsbereiten Zustand, da der Datentransfer, der vorher inaktiven Endknoten sporadisch einsetzen kann. Die Endknoten bestätigen ledliglich in unregelmäßigen Abständen ihre Anwesenheit im Netzwerk und starten unmittelbar den Transfer zu dem Koordinator falls die auf dem Gerät laufende Applikation dies erfordert. Das kann dazu führen, dass - falls der Kanal derzeit durch einen anderen Sender belegt ist - Daten verloren gehen. Diese Technologie wird hauptsächlich bei sporadischem Datenaustausch eingesetzt. Dadurch, dass der Koordinator dauerhaft aktiv bleiben muss, ist ein Anschluss zu einem festen Stromnetz für diesen fast unabdingbar.

5. ZIGBEE

Auf dem IEEE 802.15.4 Standard aufbauend definiert ZigBee die Vermittlungs-, Security- und Anwendungsschicht (Tabelle 2).

5.1 Vermittlungsschicht und Security

In der Vermittlungsschicht (oder Network-Layer) wird der Aufbau eines Netzwerkes definiert [3]. Zu den Aufgaben zählt hierbei das Erstellen des Netzwerkes selbst, das Betreten und Verlassen eines Netwerkes, die Adressierung und das Routing.
Der Verbindungsaufbau kann auf mehreren Wegen erfolgen. Der Koordinator übernimmt hierbei die Rolle den Kanal, also den Frequenzbereich, zu wählen. Dazu wird ein Energy-Detect-Scan durchgeführt, der die Signalstärke des zu überprüfenden Kanals abhorcht, um Kollisionen mit anderen Netzes zu vermeiden. Ist ein freier Kanal gefunden, so kann der Koordinator einen Active-Scan durchführen, indem er auf diesem Kanal einen Beacon_Request sendet, bei dem jedes Gerät in Reichweite, das auf diesen Kanal hört, zu einer Antwort aufgefordert wird, um die Anwesenheit zu bestätigen. Soll sich ein FFD einem Netzwerk anschliessen, so wird nach Feststellung des Vorhandenseins eines Koordinators ein Active-Scan durchgeführt, bei dem das FFD den Koordinator auffordert eine Antwort zu senden und dieses Gerät in das Netzwerk aufzunehmen.
Eine weitere Möglichkeit ist ein sogenannter Passive-Scan, bei dem ein Gerät verschiedene Kanäle abhört, um einen Beacon zu empfangen und sich anschliessend mit dem Netzwerk zu verbinden.
Das Routing bei ZigBee findet mithilfe eines hierarchisch organisiertem Tabellesystem statt. Dafür wird eine Eltern-Kind Relation zwischen den einzelnen Knoten verwendet. Mittels dem so genannten Cskip-Verfahren [5], welches von ZigBee 1.0 und ZigBee 2006 implementiert wurde, werden hexadezimale Adressen vergeben, wobei 0x0000 sowohl die Adresse des Koordinators, als auch die Broadcast-Adresse darstellt. Alle an diesem Knoten angeschlossenen Geräte erhalten einen Adressblock zugewiesen, in dem diese wiederum an ihre Kindknoten Adressen vergeben können. Dieser Bereich wird durch den Koordinator mittels verschiedener Variablen eingeschränkt, wie etwa die maximal zulässige Anzahl von Kindknoten, der maximalen Anzahl an Routern und der größten zulässigen Tiefe.
Der Vorteil dieses Verfahrens ist eine sehr einfache Pfadwahl für das Routing, der Nachteil die Unflexibilität in adhoc Netzwerken. So kann es vorkommen, dass ein Knoten seinen Adressraum bereits ausgelastet hat, während ein anderer noch so gut wie keine Kindknoten besitzt. Dieses Problem soll mit ZigBee Pro behoben werden, indem anstatt Cskip ein zufallsbasierter Algorithmus eingesetzt wird.
Außerdem werden, falls es benötigt wird, Sicherheitsmaßnahmen ergriffen um Pakete zu schützen oder zu authentifizieren.

5.2 Anwendungsschicht

Die Anwendungsschicht definiert wie der Datenaustausch zwischen mehreren Geräten funktionieren soll. Hierbei kann eine logische Unterteilung in zwei verschiedene Subschichten vorgenommen werden. [3]

5.2.1 ZigBee Device Object

Das ZigBee Device Object bestimmt die Rolle des Gerätes in einem Netzwerk, also ob es einen Koordinator oder ein Endgerät darstellt. Die Verknüpfung von mehreren Geräten geht von diesen Punkten aus und die Art der Sicherung, z.B. mittels Public-/Private-Key, wird hier festgelegt.

5.2.2 Application Support Layer

Diese unterliegende Schicht dient der Erkennung der Kommunikation von Anwendungen zwischen mehreren Geräten. Weiterhin wird die logische Bindung der Geräte hier vorgenommen.

6. 6LOWPAN

Der dritte hier vorgestellte Standard ist 6LoWPAN, eine Abkürzung für ÏPv6 Low Power Wireless Premise Area Networks". Im Gegensatz zu ZigBee, das neue Schichten eigenständig definiert, und einen komplett neuen Ansatz darstellt, setzt 6LoWPAN auf die Weiterverwendung bestehender Standards, nämlich IPv6. Die Vorteile hierzu liegen auf der Hand:

- Es ist keine Konfiguration wie bei DHCP/NAT erforderlich, da IPv6 durch die Zero-Configuration und Neighbor Discovery diese Mechanismen bereits beinhaltet
- Tools und Know-How von IPv6 können auf diese Netze übertragen werden
- IP-basierte Protokolle wie ICMP, UDP und TCP können auch mit 6LoWPAN verwendet werden

6.1 Paketgröße

Eine große Herausforderung der Überführung von IPv6 in ein Format, das von der 127 Byte Maximum Transition Unit (MTU) des 802.15.4 Standards verwendet wird [6]. Die HeadergröSSe bei IPv6 Paketen beträgt bereits 40 Bytes, was vor allem bei intra-PAN Verbindungen nicht notwendig ist, aber einen groSSen Teil der MTU bereits verbraucht. Um diesen Header auf eine geringere Größe zu schrumpfen wird er hier anders generiert.[4]
Hierfür werden verschiedene, kombinierbare Möglichkeiten zur Verfügung gestellt. Der 1-Byte Dispatch Header zeigt hierbei um was für einen Typ sich das folgende Paket handelt. Die ersten 2 Bits zeigen hierbei an ob es sich um einen 6LoWPAN Frame handelt (01), oder nicht (00). Von den verbleibenden 6 Bits sind derzeit nur 5 Möglichkeiten definiert, die angeben ob das folgende Paket eine komprimierte IPv6 Adresse (HC1), oder eine vollständige enthält.
Der Mesh-Header stellt mit einer gröSSe von 4 Bytes die Quelle und Zieladresse auf dem Link Layer dar, sowie das Hop Limit. Die ersten zwei Bits geben hierbei an, ob die von IEEE 802.15.4 64 Bit Standardadresse, oder die verkürzte 16-Bit ßhort address"verwendet wird. In den nächsten 4 Bits werden die verbleibenden Hops definiert, wobei der 1111 hier nicht für 15, sondern für 255 steht, um auch in größeren Netzen routen zu können.
Im Fragmentation Header wird das Zerlegen und Rekonstruieren der Framepakete von 802.15.4, die eine Payload von bis zu 102 Bytes unterstützt. Da in IPv6 aber eine MTU von 1280 möglich ist, müssen diese dementsprechend verkürzt werden. Ein Tag-field gibt hierbei die Zugehörigkeit zu einem Fragment an.
Aus der Kombination dieser Methoden ergeben sich flexible Pakete, die je nach benötigter Information zusammengesetzt werden können. Das kleinste Beispiel für ein Paket wäre die direkte Kommunikation zwischen zwei Sensoren, wobei nur der Dispatch Header und die komprimierte HC1 Adresse, was zu einer Gesamtgröße von nur 2 Bytes für den Header resultieren würde.

7. VERGLEICH DER STANDARDS

Die hier vorgestellten Standards unterscheiden sich in nicht nur in ihrer Funktionsweise, sondern auch in anderen Punkten. Während ZigBee und 6LoWPAN die gleichen Ziele und Motivationen verfolgen stellt Bluetooth einen komplett verschiedenen Ansatz dar.

	ZigBee	6LoWPAN
RAM Req.	8 K	4 K
Netzgroße	65 K	2^{64}
Transport	Keine	TCP/UDP Unterstützung
Anbindung	nur ZigBee Gateway	Bridge oder Router

Table 3: Anforderungen von ZigBee und 6LoWPAN im Vergleich [4]

Es gibt auch bei den Anwendungsgebieten der Technologien sehr deutliche Unterschiede. Während ZigBee und 6LoWPAN hauptsächlich für Kleinstsensoren in groSSen Netzen entwickelt wurden, wie etwa Temperaturfühler, Alarmanlagen oder Feuermelder, und eher geringe Datenmengen zu übertragen haben, ist Bluetooth eher für kleine Netzwerke geeignet, in denen groSSe Datenmengen, wie etwa Sprachübertragungen, versendet werden. Oftmals finden Bluetooth-Verbindungen nur kurzfristig statt, und sollen ein schnelles Ad-Hoc Netzwerk aufbauen, das für einen zeitlich begrenzten Datenaustausch aufrecht erhalten werden soll. Bei ZigBee und 6LoWPAN finden sich im Gegenteil dazu oft statische Netze vor. Auch die Energieeinsparungen wurden bei der Entwicklung von ZigBee und 6LoWPAN eher berücksichtigt als bei Bluetooth, da Erstere eher für kleine Sensoren mit langer Lebenszeit entwickelt wurden, während bei Bluetooth hauptsächlich die kabellose Datenübertragung im Vordergrund stand, wie zum Beispiel bei einem Handy.
Gegenüber ZigBee gibt Mulligan [4] ausser dem Weiterverwenden des bestehenden Standards IPv6 die geringeren Systemanforderungen als Vorteile an (siehe Tabelle 3).

Zudem gibt es bereits Open-Source Lösungen für 6LoWPAN. Die folgenden Jahre werden zeigen welcher Standard sich durchsetzen wird.

8. AUSBLICK

Ausser den hier vorgestellten Standards gibt es noch einige weitere, die sich in Implementierungsweise und Anwendungsgebiet zu diesen unterscheiden.
So entwickelt die Bluetooth Special Interest Group (SIG) derzeit mit Bluetooth low energy (vormals: Wibree) eine energieeffizientere Alternative zu Bluetooth. Hierbei wird die Datenrate von 2,1 Mbit/s auf 1 Mbit/s reduziert, wobei die Kompatibilität zur originalen Bluetooth Technologie erhalten bleiben soll.
Ein weiteres Konzept ist die Einführung von Ultra Wide Band, einer Technologie, die im Frequenzbereich von 3,1 GHz bis 10,6 GHz Daten groSSe Daten über eine kurze Strecke übermitteln soll. Um einen Datendurchsatz von bis zu 1320 Mbit/s zu erhalten wird hierbei eine Bandbreite von mindestens 500 MHz verwendet. Anwendungsgebiete für diese Technologie sollen unter anderem Funkübertragung zu Monitoren werden.
Obwohl viele dieser Kommunikationsstandards konkurrierend sind, besteht dennoch ein Zwang mehrere Standards weiter zu entwickeln, um den stark variierenden Anforderungen unterschiedlicher Netzwerke gerecht zu werden.

9. REFERENCES

[1] "How Bluetooth Technology Works", Bluetooth SIG. , http://www.bluetooth.com/Bluetooth/Technology/Works/

[2] L. Hein. "Bluetooth - Die Grundlagen", All About Security, October 2006. http://www.all-about-security.de/security-artikel/endpoint-sicherheit/mobile-computing-und-pdas/artikel/282-bluetooth-die-grundlagen/

[3] P. Kinney. SZigBee Technology: Wireless Control that simply works.", Communications Design Conference, Kinney Consulting LLC, October 2003. http://www.zigbee.org/en/press_kits/2009_12_16/documents/white_papers/wp_zigbeetechwireless_final.pdf

[4] G. Mulligan. "The 6LoWPAN Architecture", 6LoWPAN Working Group, Internet Engineering Task Force, 2007.

[5] G. Kupris, A. Sikora. SZIGBEE - Datenfunk mit IEEE 802.15.4 und ZIGBEE", ISBN 3772341594, Franzis Verlag, 2008.

[6] Z. Shelby, C. Bormann "6LoWPAN - The wireless embedded internet", ISBN 0470747994, John Wiley and Sons, November 2009.

[7] I. Vazquez "Project SmartMotes", Tecnológico Fundación Deusto, 2007. http://www.tecnologico.deusto.es/projects/smartmotes/